

Active Learning in Incomplete Label Multiple Instance Multiple Label Learning

Tam Nguyen, *Member, IEEE*, and Raviv Raich, *Senior Member, IEEE*

Abstract—To alleviate labeling complexity, in multi-instance multi-label learning, each sample/bag consists of multiple instances and is associated with a set of bag-level labels leaving instances therein unlabeled. This setting is more convenient and natural for representing complicated objects with multiple semantic meanings. Compared to single-instance labeling, this approach allows for labeling larger datasets at an equivalent labeling cost. However, for sufficiently large datasets, labeling all bags may become prohibitively costly. Active learning (AL) uses an iterative labeling and retraining approach to provide reasonable classification performance using a small number of labeled samples. To our knowledge, only two approaches have been previously proposed for AL in the MIML setting. These approaches either require labeling all classes in a selected bag or involve partial instance-level labeling. To further reduce labeling costs, we propose a novel bag-class pair-based approach for AL in the MIML setting. Due to the partial availability of bag-level labels, we focus on AL in the incomplete-label MIML setting. For the query process, we adapt AL criteria to the novel bag-class pair selection strategy. Additionally, we introduce an online approach for learning a discriminative graphical model based classifier. Numerical experiments on benchmark datasets demonstrate the effectiveness of the proposed approach.

Index Terms—Active learning, multiple instance multiple label learning, expected gradient length, uncertainty sampling, incomplete-label learning, bag-class pair.

1 INTRODUCTION

IN many real world applications, there are plentiful unlabeled data but limited labeled data, and the acquisition of class labels is usually costly and difficult. By actively and iteratively selecting the most valuable data to query their supervised information, active learning tries to train an effective model with least labeling cost. Under the traditional single-instance single-label (SISL) learning, where each example is labeled by a single label, active learning methods select the most valuable instances and then query their labels from the annotator (oracle) [1], [2], [3], [4], [5], [6], [7], [8], [9], [10], [11], [12], [13], [14], [15], [16], [17], [18], [19], [20], [21], [22], [23]. The key task is to design a criterion for instance selection. In multi-instance learning (MIL), instances are grouped into bags which may contain any number of instances. A bag is labeled negative if and only if it contains all negative instances. A bag is labeled positive, however, if at least one of its instances is positive. There are two scenarios in active learning for MIL. The first is to simply allow the learner to query for the label of unlabeled bags [24], [25]. A second scenario is one in which all bags in the training set are labeled and the learner is allowed to query for the labels of instances selected from positive bags [26]. In multi-label learning (MLL), where each example is labeled by a label set, there are several query approaches for active learning: (i) An instance is selected and the label for each class is obtained in a single query [27], [28], [29], [30]. (ii) An instance-class pair is selected to be

labeled at each query [31], [32]. (iii) An instance is selected first and then a class for the selected instance is labeled in each query [33]. In MIML setting, every example is represented by a bag of multiple instances and is annotated with multiple class labels to express the presence or absence of each class in the bag. The MIML setting provides an appropriate framework for learning with complex objects. However, when the instance feature vector dimension and the number of classes increase, training an effective model requires more data. Moreover, since there are a large number of candidate labels in MIML, it becomes much more costly to annotated an example comparing to MIL. Hence, active learning for MIML is highly desired to reduce the labeling cost. Similar to active learning in MLL, the query process for active learning in MIML learning presents several options. To the best of our knowledge, the only two approaches of querying in MIML, which are available in the literature are: (i) A bag is selected and all classes are labeled at each query [34]. This approach may lead to redundant labeling of classes, which do not help to increase the performance of the model. (ii) A bag is selected first and then a single class for that bag is selected to be labeled at each query [35], [36]. In [35], [36], after selecting the bag and the class to be labeled, the oracle decides whether they are relevant. If the oracle returns a negative label the query process is complete and all instances are assumed negative for the selected class. If the oracle returns a positive label, then the oracle is queried for an instance in the bag that is positive for the selected class. In either case, the information provided in each query can be directly mapped to instance labels. In turn, an instance level model is updated based on the specific instance information. The queried bags are moved from the unlabeled set to the labeled set once all the labels of these bags are available. Beyond the standard cost for labeling a bag-class label pair,

- T. Nguyen was with the School of Electrical Engineering and Computer Science, Oregon State University, Corvallis, OR, 97330-5501.
E-mail: nguyeta4@oregonstate.edu
- R. Raich is with the School of Electrical Engineering and Computer Science, Oregon State University, Corvallis, OR, 97330-5501.
E-mail: raich@eecs.oregonstate.edu

Manuscript received xxx; revised xxx.

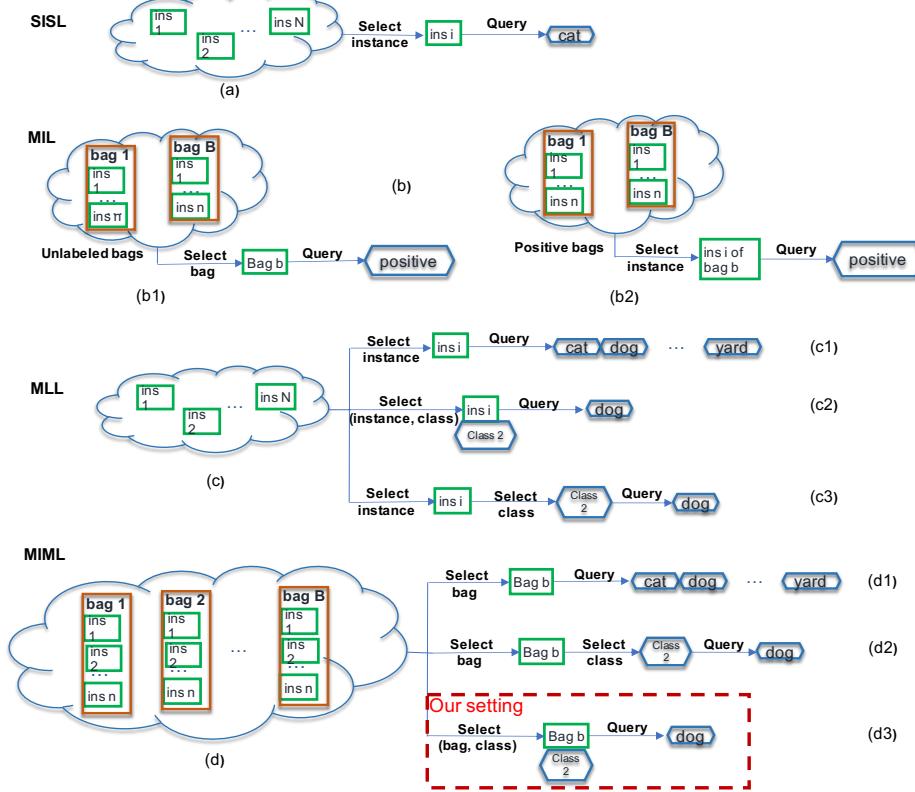


Fig. 1: Active learning different settings. (a) Active learning in SISL setting. Each query selects the most informative instances from the unlabeled data to ask for labeling. (b) Active learning in MIL setting. (b1) Each query selects a bag of instances to label. (b2) Each query selects an instance from a positive bag to label. (c) Active learning in MLL setting. (c1) Each query selects an instance to label all classes. (c2) Each query selects pair of instance and class to label. (c3) Each query selects one instance, and then select which class to label for selected instance. (d) Active learning in MIML setting. (d1) Each query selects a bag to label all classes. (d2) Each query selects one bag, and then select which class to label for selected bag.(d3) **(Our approach)** each query selects pair of bag and class to label.

an additional cost is involved in the process of querying the labeler for the key instance associated with a positively-labeled class. To the best of our knowledge, there are no active learning methods for a single bag-class label pair query. This approach can be faster and/or less costly to label because in each query only one bag-class pair is presented to the labeler to label. The increase in labeling all classes in a given bag (e.g., the number of times the labeler reviews the bag-of-instances) compared to labeling a single class in the bag (i.e., a bag-class pair) may be small when the number of classes is small. If the number of classes considered is large, the cost of labeling all classes in a bag compared to a single class may be significant, e.g., proportional to the number of classes considered. For example, in bird species recognition [37], a labeler can listen to an audio recording one time and determine the presence or absence of a single target species or even a small set of target species. However, if the number of target species is large, the labeler may need to repeatedly listen to the audio recording as they carefully review each class in their provided list of target species. Though labeling a single bag-class pair can reduce the labeling cost, this approach introduces the following challenge. After a bag is labeled for one of its classes, the available label set for the bag is incomplete. Incomplete-label MIML is used to describe the setting, in which each training bags

are provided with a subset of the correct label set. Since up until recently no incomplete-label MIML methods were available, training based on data from the aforementioned active learning method was a challenge. The recent work in [38] introduces a method to train data with sub-set of labels available. With the availability of training methods for incomplete MIML, we are now in position to evaluate the bag-class label pair query paradigm for active learning in MIML. Fig. 1 presents active learning query types for different learning settings including SISL, MIL, MLL, and MIML. The active learning strategy proposed in this paper is highlighted.

In summary, in this paper, we propose a novel framework for active learning under the MIML-ILL setting. Our MIML-ILL classifier bases on the discriminative graphical model with exact inference. We develop an online version of the model update to maximize the marginal log-likelihood to reduce the computational complexity of our framework and make it scalable. In the query process, we propose a novel approach to select bag-class pair based on EGL and uncertainty sampling that rely on the bag-class probability determined by our model. Finally, we build up a comprehensive comparisons to show the effectiveness of the proposed approach.

2 RELATED WORK

Active learning selectively queries the most valuable information from the oracle and aims to train an effective model with least queries. The key task in active learning is to design a proper strategy such that the queried information is most helpful for improving the learning model. There have been many active learning methods proposed under traditional single instance single label (SISL) setting [39] with three main settings. (i) Membership query synthesis: the learner may request labels for any unlabeled instance in the input space, including (and typically assuming) queries that the learner generates de novo, rather than those sampled from some underlying natural distribution [1], [2], [3], [4]. (ii) Stream-based selective sampling: the key assumption is that obtaining an unlabeled instance is free (or inexpensive), so it can first be sampled from the actual distribution, and then the learner can decide whether or not to request its label [5], [6], [6], [7], [8], [9], [10], [11], [12], [13], [14]. (iii) Pool-based sampling: there is a small set of labeled data and a large pool of unlabeled data available. Queries are selectively drawn from the pool, which is usually assumed to be closed [15], [16], [17], [18], [19], [20], [21], [22], [23].

While most active learning research focuses on traditional setting, there are a few works that extend the ideas to multi-instance learning [24], [25], [26] or multi-label learning [27], [28], [29], [30], [31], [33]. In [26], the authors introduce two active query selection strategies in multi-instance (MI) active learning: MI uncertainty sampling and expected gradient length. They explore the case where a MI learner may query unlabeled instances from positively labeled bags in order to reduce the inherent ambiguity of the MI representation. In [24], the authors describe a multiple-instance active learning algorithm for such incremental learning in the context of building models of relevant domain objects. Each bag of instance is selected to be labeled after each query. They introduce the concept of bag uncertainty sampling, enabling robots to identify the need for feedback, and to incrementally revise learned object models by associating visual cues extracted from images with verbal cues extracted from limited high-level human feedback. Two general multiple-instance active learning (MIAL) methods are introduced in [25], multiple-instance active learning with a simple margin strategy (S-MIAL) and multiple-instance active learning with Fisher information (F-MIAL). These two approaches are applied to the active learning in localized content based image retrieval. S-MIAL considers the most ambiguous picture as the most valuable one, while F-MIAL utilizes the Fisher information and analyzes the value of the unlabeled pictures by assigning different labels to them. Both methods select a bag to label for each query. For active learning in the MLL setting, the approaches in [27], [28], [29], [30] select a single instance to label all classes after each query. In [27], the authors first propose two novel multi-label active learning strategies, a max-margin prediction uncertainty sampling strategy and a label cardinality inconsistency strategy, and then integrate them into an adaptive framework of multi-label active learning. In [29], the authors propose a multi-label SVM active learning method. They provide two selection strategies: max loss strategy and mean max loss strategy. Auxiliary learner is introduced in [30]. They

extend maximum loss reduction with maximum confidence (MMC) to a more general framework that removes the heavy dependence and clarifies the roles of each component in MMC. In particular, the framework is characterized by a major learner for making predictions, an auxiliary learner for helping with query decisions and a query criterion based on the disagreement between the two learners. In [28], the authors propose a semantic-gap-oriented active learning method, which incorporates the semantic gap measure into the information-minimization-based sample selection strategy. The basic learning model used in the active learning frame-work is an extended multi-label version of the sparse-graph-based semi-supervised learning method that incorporates the semantic correlation. The different strategy of active learning in MLL is introduced in [31], [32]. In these approaches, a pair of instance-label is selected simultaneously to label after each query. Specifically, in [31], the authors propose a novel example-label based multi-label active learning method. They consider how to select the most informative example-label pairs by computing the uncertainty of each example-label pair with the boundary, but they did not take the label correlation of an example into consideration. In [32], the authors propose to select sample-label pairs to minimize a multi-label Bayesian classification error bound. This active learning strategy not only considers the sample dimension but also the label dimension and is termed Two-Dimensional Active Learning (2DAL). In [33], the authors propose an approach to select a single instance first and then a class for the selected instance is labeled after each query. In this approach, the selected instance is the one that maximizes the label cardinality inconsistency (LCI). LCI measures the inconsistency between the number of predicted positive labels of an instance and the average label cardinality (the average of the number of positive labels) on the fully labeled data. And then a class is selected based on the distance between its and the dummy label. All the above studies are focusing on either multi-instance or multi-label learning, and cannot be directly applied to MIML setting. There are several studies which are developed for active learning in MIML setting. The method in [34] is specifically designed based on MIMLSVM. It firstly degenerates the bags to single-instance representation and then directly employ traditional active learning method for label querying (select each bag to label all classes for each query), which does not truly exploit the characteristics of MIML tasks. The authors in [35] propose an approach for active learning in MIML setting based on the work in [33]. For each query, the bag is selected first based on the uncertainty (the gap between the predicted number of positive labels of the bag and the average number of positive labels of the training data) and diversity of the bag (how many labels of the bag was queried before). Then a class is pointed out to be labeled for the selected bag based on the distance from the label to the thresholding dummy label. In their methods, not only one selected label is queried, but also the key instance which is most relevant to queried label is asked. In [36], the authors extend work in [35] with a modification in bag label prediction achieving from instance-level predictions to predicting protein functions of bacteria genomes. All these approaches select a bag to label all classes or a bag and then select a class to label. To the best of our knowledge, there

are no studies that focus on selecting a bag-class pair in the MIML setting.

3 THE PROPOSED APPROACH

In this section, we introduce the problem of active learning for MIML data with missing labels. We present a novel instance selection approach, in which the presence or absence of a specific class in a given bag is obtained with each query. We demonstrate how criteria such as expected gradient length (EGL) and uncertainty sampling, commonly developed for querying a multi-class label for the single instance case, can be modified for the selection of the bag-class pair as an instance. Finally, to facilitate an efficient model update, we develop an online SGD approach for learning the model parameters.

3.1 Problem formulation

Our main goal is to develop a model to learn an effective classifier that can label a newly unseen bag/instance under the setting of MIML learning with missing labels with as smallest as possible number of training data using active learning. We begin with a description of the data and related notation. We then continue with the probabilistic model for this setting and the associated inference approaches.

Data description: We consider an entire dataset consisting of a collection of B bags and their associated label sets $\{(\mathbf{X}_b, \mathbf{Y}_b)\}_{b=1}^B$, respectively. Each bag \mathbf{X}_b is a set of instance feature vectors, $\mathbf{X}_b = \{\mathbf{x}_{b1}, \mathbf{x}_{b2}, \dots, \mathbf{x}_{bn_b}\}$, where $\mathbf{x}_{bi} \in \mathcal{X} \subseteq \mathbb{R}^d$ is the feature vector for the i th instance in the b th bag and n_b denotes the number of instances in the b th bag. Bag b is labeled by a label vector $\mathbf{Y}_b \in \{-1, 0, 1\}^C$, where C is the number of classes and for each class the c th-entry $Y_{bc} \in \{-1, 0, 1\}$ indicates a positive label 1, negative label 0, and the absence of the label -1 . The set of available labels in bag b is denoted by S_b , which is defined as:

$$S_b = \{c | Y_{bc} \neq -1, c = 1, 2, \dots, C\}. \quad (1)$$

Additionally, we introduce the set of positively labeled classes S_b^+ , negatively labeled classes S_b^- , and unlabeled classes \bar{S}_b in bag b :

$$\begin{aligned} S_b^+ &= \{c | Y_{bc} = 1, c = 1, 2, \dots, C\} \\ S_b^- &= \{c | Y_{bc} = 0, c = 1, 2, \dots, C\}. \\ \bar{S}_b &= \{c | Y_{bc} = -1, c = 1, 2, \dots, C\}. \end{aligned} \quad (2)$$

Note that $S_b = S_b^+ \cup S_b^-$ and \bar{S}_b is the complement of S_b . For example, let $C = 6$ and $Y_b = [0, -1, 1, 1, 0, -1]^T$. Hence, we only observe a label for Y_{b1}, Y_{b3}, Y_{b4} and Y_{b5} . Therefore, $S_b = \{1, 3, 4, 5\}$, $S_b^+ = \{3, 4\}$, $S_b^- = \{1, 5\}$, and $\bar{S}_b = \{2, 6\}$. Moreover, we can define the set \mathcal{L} , the available label index set, as the set of indices (b, c) for which $Y_{bc} \neq -1$, i.e., $\mathcal{L} = \{(b, c) | Y_{bc} \neq -1\}$. Similarly, we can define the unavailable label index set, as the set of indices (b, c) for which $Y_{bc} = -1$, $\mathcal{U} = \{(b, c) | Y_{bc} = -1\}$. We consider two steps in active learning: (1) instance selection and (2) model update. An instance selection criterion is applied to obtain bag-class pair $(b^*, c^*) \in \mathcal{U}$ for which a label will be provided. After a label is provided for such an instance, the corresponding $Y_{b^*c^*}$ will no longer be -1 and consequently

$\mathcal{L} \leftarrow \mathcal{L} \cup \{(b^*, c^*)\}$ and $\mathcal{U} \leftarrow \mathcal{U} \setminus \{(b^*, c^*)\}$. For the model update, it is common to retrain the model after obtaining an additional label. It is important to note that using our notations the training set remains $\{(\mathbf{X}_b, \mathbf{Y}_b)\}$ regardless of how many unavailable labels. Their availability is directly encoded in the \mathbf{Y}_b vector. With more bag-class pairs becoming known, some of the entries in Y_{bc} change from -1 to either 1 or 0. Though bags for which \mathbf{Y}_b is the all -1 vector are also included in the training set, they play no role in the update step since the log-likelihood will only be computed based on the available labels.

Given this setting, we develop a novel framework of active learning in MIML-ILL. In which, (i) we adopt the MIML-ILL model from [38] (as reviewed in this section) to learn the base classifier; (ii) we propose a novel instance selection, called bag-class pair selection based on EGL and uncertainty sampling for the MIML-ILL setting to select the most informative bag-class pair from the unlabeled data to update the model; (iii) we present an online version of optimization problem to maximize the marginal log-likelihood of the model in [38]. The complete algorithm of our proposed approach is summarized in Algorithm 1.

Algorithm 1 The MIMLILL-AL algorithm

```

1: Input:
     $\mathcal{B} = \{(\mathbf{X}_b, \mathbf{Y}_b)\}_{b=1}^B$ : entire dataset (may include incomplete label vectors).
     $\mathcal{U} = \{(b, c) | Y_{bc} = -1\}$ : the unavailable bag-class pairs;
     $Q$ : number of queries; (no more than  $|\mathcal{U}|$ )
2: Initialize:
     $q = 1$ 
     $\mathbf{w}_0 = 0$ 
     $\mathbf{w} = Model\_Update(\mathcal{B}, \mathbf{w}_0)$  (as in Section 3.2)
3: while  $q \leq Q$  do
4:    $(b^*, c^*) = Instance\_Selection(\mathcal{B}, \mathcal{U}, \mathbf{w})$  (as in Section 3.3)
5:   Query  $Y_{b^*c^*}$ 
6:   Update training data:
       $\mathcal{U} \leftarrow \mathcal{U} \setminus \{(b^*, c^*)\}$ .
7:    $\mathbf{w} = Model\_Update(\mathcal{B}, \mathbf{w})$  (as in Section 3.4)
8:    $q = q + 1$ 
9: end while

```

3.2 Background

Our proposed model is adopted from [38] paper (presented in Fig. 2). Specifically, we assume that the instance feature vectors and the bag labels in each bag are independent across bags, i.e., observations $(\mathbf{X}_b, \mathbf{Y}_b)$ are independent for $b = 1, 2, \dots, B$. To define the model for a single bag, we assume that the latent multi-class instance labels y_{bi} for $i = 1, 2, \dots, n_b$ are independent conditioned on \mathbf{X}_b and the probability for $y_{bi} \in \{1, 2, \dots, C\}$ given \mathbf{x}_{bi} follows the multinomial logistic regression model

$$P(y_{bi} = c | \mathbf{x}_{bi}, \mathbf{w}) = \frac{e^{\mathbf{w}_c^T \mathbf{x}_{bi}}}{\sum_{k=1}^C e^{\mathbf{w}_k^T \mathbf{x}_{bi}}}, \quad (3)$$

where $\mathbf{w} = [\mathbf{w}_1^T, \dots, \mathbf{w}_C^T]^T$ is the model parameter column vector and \mathbf{w}_c for $c = 1, 2, \dots, C$ is a d -dimensional column vector. As discussed in the introduction, due to various reasons (e.g., labeling cost and/or labeling strategy), only a subset of the classes $\{1, 2, \dots, C\}$ may be labeled. We model the labeled (i.e., observed) entries of \mathbf{Y}_b conditioned on $\mathbf{y}_b = [y_{b1}, \dots, y_{bn_b}]$ as independent and consequently $P(\mathbf{Y}_b | \mathbf{y}_b) = \prod_{c \in S_b} P(Y_{bc} | \mathbf{y}_b)$. To model Y_{bc} for $c \in S_b$ given \mathbf{y}_b , we follow the OR rule assumption, i.e., $Y_{bc} = 1$ if at least

one of the y_{bi} 's equals to c and 0 otherwise. Hence, we define the probability of Y_{bc} for $c \in S_b$ given \mathbf{y}_b as:

$$P(Y_{bc} = Y | \mathbf{y}_b) = Y \left(1 - \prod_{i=1}^{n_b} I(y_{bi} \neq c) \right) + (1 - Y) \prod_{i=1}^{n_b} I(y_{bi} \neq c), \quad (4)$$

where $Y \in \{0, 1\}$. Based on the aforementioned modeling assumptions, the single bag log-likelihood for the b th bag is given by

$$\log P(\mathbf{Y}_b, \mathbf{X}_b) = \log \sum_{\mathbf{y}_b} \prod_{c \in S_b} P(Y_{bc} | \mathbf{y}_b) \prod_{j=1}^{n_b} P(y_{bj} | \mathbf{x}_{bj}, \mathbf{w}) + \log P(\mathbf{X}_b), \quad (5)$$

where $P(y_{bj} | \mathbf{x}_{bj}, \mathbf{w})$ and $P(Y_{bc} | \mathbf{y}_b)$ are given by (3) and (4), respectively. Note that term $\log P(\mathbf{X}_b)$ is not a function of the parameter vector \mathbf{w} and hence is treated as a constant. According to [38], the computational complexity associated with the E-step of EM algorithm used to maximize the log-likelihood function is the exponential degree of the number of positive labels per bag. To reduce the computational complexity associated with the E-step when the number of positive labels per bag is large, we adopted the marginal maximum likelihood (MML) approach to maximum likelihood. We consider an objective that is the sum of the log-likelihoods associated with each class label. The contribution of each class can be handled separately. Specifically, the use of a single label at a time allows us to compute the objective function in closed-form and implement its gradient efficiently, thereby reducing the complexity to linear in the number of classes at the potential expense of breaking down some of the label dependence relations.

Based on the model, the sum of marginal log-likelihoods is:

$$L_{MML}(\mathbf{w}) = \sum_{b=1}^B \sum_{t \in S_b} \log P(Y_{bt} | \mathbf{X}_b, \mathbf{w}). \quad (6)$$

We can further derive the criterion by marginalizing $P(Y_{bt}, \mathbf{y}_b | \mathbf{X}_b, \mathbf{w}) = P(Y_{bt} | \mathbf{y}_b) \prod_{i=1}^{n_b} P(y_{bi} | \mathbf{x}_{bi}, \mathbf{w})$ over \mathbf{y}_b while substituting $P(Y_{bt} | \mathbf{y}_b)$ from (4) [38]. The aforementioned step along with some simplifications yields the follow expression for the marginal log-likelihood objective:

$$L_{MML}(\mathbf{w}) = \sum_{b=1}^B \sum_{t \in S_b} \left[I(Y_{bt} = 0) \sum_{i=1}^{n_b} \log P(y_{bi} \neq t) + I(Y_{bt} = 1) \log \left(1 - e^{\sum_{i=1}^{n_b} \log P(y_{bi} \neq t)} \right) \right]. \quad (7)$$

For $c = 1, \dots, C$, the update rule of \mathbf{w}_c is given by

$$\mathbf{w}_c^{k+1} = \mathbf{w}_c^k + \eta_k \frac{\partial L_{MML}(\mathbf{w})}{\partial \mathbf{w}_c} \Big|_{\mathbf{w}=\mathbf{w}^k}, \quad (8)$$

where $\frac{\partial L_{MML}(\mathbf{w})}{\partial \mathbf{w}_c}$ is computed as

$$\begin{aligned} & \sum_{b=1}^B \sum_{i=1}^{n_b} \sum_{t \in S_b} \left(I(Y_{bt} = 1) P(Y_{bt} = 0) \frac{P(y_{bi} = t)}{P(Y_{bt} = 1)} - I(Y_{bt} = 0) P(y_{bi} = t) \right) \\ & \quad \left(I(c = t) - I(c \neq t) \frac{P(y_{bi} = c)}{1 - P(y_{bi} = t)} \right) \mathbf{x}_{bi} \end{aligned} \quad (9)$$

and

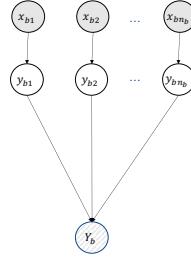


Fig. 2: Incomplete label learning model. Shaded nodes are observed (i.e., $\mathbf{x}_{b1}, \dots, \mathbf{x}_{bn_b}$) and striped nodes are partially-observed (i.e., \mathbf{Y}_b).

$$\begin{aligned} P(Y_{bt} = 0) &= \prod_{i=1}^{n_b} p(y_{bi} \neq t), \\ P(Y_{bt} = 1) &= 1 - P(Y_{bt} = 0). \end{aligned} \quad (10)$$

The step size η_k is determined using the backtracking line search algorithm.

We develop our framework for active learning under the MIML-ILL setting based on this model. The aforementioned inference (8)-(10) was proposed in [38] as means of training the model-based MIML-ILL classifier. In active learning, the model is typically updated with every query to ensure that instance selection is taking into account the most current and accurate model. This task is computationally intensive with a computational complexity that grows linearly with the the size of the labeled data. To reduce the computational complexity of our framework and make it scalable, we develop an online version of the model update to maximize the marginal log-likelihood (the detail is provided in Section 3.4). In the query process, we propose a novel approach to select bag-class pair based on EGL and uncertainty sampling that rely on the bag-class probability determined by this model. Details are provided in Section 3.3.

3.3 Instance Selection

Recall an unavailable label index set, as the set of indices (b, c) for which $Y_{bc} = -1$, $\mathcal{U} = \{(b, c) | Y_{bc} = -1\}$, where $Y_{bc} \in \{-1, 0, 1\}$ and $c = 1, \dots, C$. Let $l \in \{0, 1\}$. In our active learning scenario, each query selects a pair of bag-class (b, c) in \mathcal{U} and queries for $Y_{bc} = l$. We introduce two criterion to select bag-class pair from the unlabeled pool: (i) EGL and (ii) uncertainty sampling.

3.3.1 EGL for MIML

Expected gradient length is an approach used for discriminative probabilistic model classes. This approach selects the instance that would impart the greatest change to the current model if we knew its label. Specifically, the learner should query instance \mathbf{x} which, if labeled and added to training data \mathcal{L} , would result in the new training gradient of the largest magnitude. Let $f_\theta(\mathcal{L})$ be the objective function used in training a classifier (e.g., log-likelihood or regularized loss), and $\nabla f_\theta(\mathcal{L})$ be the gradient of the objective function f w.r.t θ . Now let $\nabla f_\theta(\mathcal{L} \cup \{(\mathbf{x}, y)\})$ be the new gradient obtained by adding the training tuple (\mathbf{x}, y) to the previously available data \mathcal{L} . Since the query

algorithm does not know the true label y in advance, we must instead calculate the length as an expectation over the possible labelings:

$$\mathbf{x}_{EGL}^* = \arg \max_{\mathbf{x} \in \mathcal{U}} \sum_y P(y|\mathbf{x}, \mathbf{w}) \|\nabla f_\theta(\mathcal{L} \cup \{(\mathbf{x}, y)\})\|. \quad (11)$$

This EGL approach is applied in single-instance single-label (SISL) or multi instance learning (MIL). For multi-label learning (MLL), to consider all label sets of one instance, we need to sum of gradient of 2^C possible cases, where C is the number of classes. This leads to issue of computational complexity when C is large. However, since we restrict our attention to selecting the bag-class pair, we manage to avoid the high computation cost associated with the calculation of the EGL criterion for an entire bag. Moreover, with selecting the bag-class pair, we might also reduce the redundant labeled classes due to query the labels of the entire bag.

To that end, we develop a novel extension of the EGL criterion to the MIML-ILL setting to select a bag-class pair for labeling during the query phase. Given the model parameters and the current training data, for each bag-class pair in the unlabeled data, we compute the gradient of the marginal log-likelihood (7) associated with the training data after a new bag-class pair is included. The bag-class pair that has the most impact to the gradient of the marginal log-likelihood on the training data with this pair included is selected for the label querying. This pair is then added to the current training data for model update.

Specifically, given the current label index set $\mathcal{L} = \{(b, c) | Y_{bc} \neq -1\}$ and the current model parameter vector \mathbf{w} , for each pair of bag-class (b, c) in \mathcal{U} , we add this pair to current available index set \mathcal{L} , then compute the gradient of the marginal log-likelihood (ML) on this new available index set. The number of available indices in \mathcal{L} will increase by 1 and the gradient of the marginal log-likelihood for new available index set is:

$$\begin{aligned} G_{\mathcal{L} \cup \{(b, c)\}} &= \frac{1}{|\mathcal{L}| + 1} \sum_{(b', c') \in \mathcal{L} \cup \{(b, c)\}} \nabla \log P(Y_{b'c'} | \mathbf{x}_{b'}, \mathbf{w}) \\ &= \frac{1}{|\mathcal{L}| + 1} \sum_{(b', c') \in \mathcal{L}} \nabla \log P(Y_{b'c'} | \mathbf{x}_{b'}, \mathbf{w}) + \frac{1}{|\mathcal{L}| + 1} \nabla \log P(Y_{bc} | \mathbf{x}_b, \mathbf{w}) \\ &= \frac{|\mathcal{L}|}{|\mathcal{L}| + 1} G_{\mathcal{L}} + \frac{1}{|\mathcal{L}| + 1} \nabla \log P(Y_{bc} | \mathbf{x}_b, \mathbf{w}). \end{aligned} \quad (12)$$

We proceed with the assumption that the minimization of the marginal log-likelihood was successfully accomplished prior to the current query such that $G_{\mathcal{L}}|_{\mathbf{w}} = 0$ for the resulting \mathbf{w} . Consequently, we can simplify the expression for the gradient of marginal log-likelihood after adding the pair (b, c) as:

$$G_{\mathcal{L} \cup \{(b, c)\}} = \frac{1}{|\mathcal{L}| + 1} \|\nabla \log P(Y_{bc} | \mathbf{x}_b, \mathbf{w})\|. \quad (13)$$

Because $Y_{bc} \in \{0, 1\}$, then the EGL of the marginal log-likelihood after adding pair (b, c) is computed as:

$$\begin{aligned} EGL_{bc} &= \frac{1}{|\mathcal{L}| + 1} (P(Y_{bc} = 1 | \mathbf{x}_b, \mathbf{w}) \|\nabla \log P(Y_{bc} = 1 | \mathbf{x}_b, \mathbf{w})\| \\ &\quad + P(Y_{bc} = 0 | \mathbf{x}_b, \mathbf{w}) \|\nabla \log P(Y_{bc} = 0 | \mathbf{x}_b, \mathbf{w})\|), \end{aligned} \quad (14)$$

where

$$\begin{aligned} \log P(Y_{bc}) &= Y | \mathbf{X}_b, \mathbf{w} \\ &= I(Y = 0) \sum_{i=1}^{n_b} \log P(y_{bi} \neq c) + I(Y = 1) \log(1 - e^{\sum_{i=1}^{n_b} \log P(y_{bi} \neq c)}) \end{aligned} \quad (15)$$

as in (7) and

$$\nabla_{\mathbf{w}_b} \log P(Y_{bc}) = Y | \mathbf{X}_b, \mathbf{w} \\ = \sum_{i=1}^{n_b} \left(I(Y = 1) P(Y_{bc} = 0) \frac{P(y_{bi} = c)}{P(Y_{bc} = 1)} - I(Y = 0) P(y_{bi} = c) \right) \left(I(t = c) - I(t \neq c) \frac{P(y_{bi} = t)}{1 - P(y_{bi} = c)} \right) \mathbf{x}_{bi} \quad (16)$$

as in (9).

The beg-class pair that satisfies the following condition is selected for querying:

$$(b^*, c^*) = \arg \max_{b, c} EGL_{bc}. \quad (17)$$

Our selection based on EGL approach requires no use of the labeled data and for each bag-class pair only the features of bag b , \mathbf{X}_b , and the model parameter vector \mathbf{w} are needed. The computational complexity of the selection phase is $O(\sum_{b \in \mathcal{U}} n_b C d)$, where as before n_b is the number of instances in bag b , C is the number of classes, and d is the dimension of the feature vector \mathbf{x}_{bi} .

3.3.2 Uncertainty sampling for MIML

Uncertainty sampling is a common approach to active learning in the standard supervised setting. For probabilistic classifiers, this involves applying the classifier to each unlabeled instance and querying those with most uncertainty about the class label. Specifically, the learner should query instance \mathbf{x} about which it is least certain how to label. Let $P(y|\mathbf{x}, \mathbf{w})$ be the probability that instance \mathbf{x} belongs to class y . A more general uncertainty sampling variant might query the instance whose prediction is the least confident:

$$\mathbf{x}^* = \arg \max_{\mathbf{x} \in \mathcal{U}} (1 - P(\hat{y}|\mathbf{x}, \mathbf{w})), \quad (18)$$

where $\hat{y} = \arg \max_{y \in \{1, \dots, C\}} P(y|\mathbf{x}, \mathbf{w})$. This uncertainty sampling approach is applied in single-instance single-label (SISL) or multi instance learning (MIL). To apply this uncertainty sampling principle directly to multi-label learning, we need to determine $\hat{Y} = \arg \max_{Y \in \{0, 1\}^C} P(Y|\mathbf{x}, \mathbf{w})$. The computational complexity of this step is $O(2^C)$, which leads to the same issue of computational complexity with the aforementioned general EGL when C is large. We restrict our attention to selecting the bag-class pair which (i) avoids the high computation cost associated with the calculation of the uncertainty sampling criterion for an entire bag; (ii) avoids labeling potentially-redundant classes by querying the most informative class only. To do so, we adopt the idea of uncertainty sampling for binary classification. In this setting, uncertainty sampling queries the instance, which is nearest to the boundary. For example, in [26], instance is selected to query its label if this instance satisfies:

$$\mathbf{x}^* = \arg \max_{\mathbf{x} \in \mathcal{U}} 2P(y = 1 | \mathbf{x}, \mathbf{w})(1 - P(y = 1 | \mathbf{x}, \mathbf{w})), \quad (19)$$

Specifically, instead of considering all classes of a bag, we propose an approach to select a bag-class pair from unlabeled data to query based on uncertainty sampling. Given the model parameters and the current training data, for each

bag-class pair in the unlabeled data, we compute its score based on the probability of the class being present in the bag. The bag-class pair that its class probability is close to the boundary is considered as the most informative pair and will be selected to add to training data for model update. For each $(b, c) \in U$, we compute the score of each unlabeled pair as:

$$S_{bc} = 2P(Y_{bc} = 1|\mathbf{X}_b, \mathbf{w}))(1 - P(Y_{bc} = 1|\mathbf{X}_b, \mathbf{w})), \quad (20)$$

where $P(Y_{bc} = 1|\mathbf{X}_b, \mathbf{w}) = 1 - \prod_{i=1}^{n_b} p(y_{bi} \neq c)$ and $P(Y_{bc} = 0) = 1 - P(Y_{bc} = 1)$. Note that we can also compute the score using the following formula $S_{bc} = |P(Y_{bc} = 1|\mathbf{X}_b, \mathbf{w}) - 0.5|/|P(Y_{bc} = 0|\mathbf{X}_b, \mathbf{w}) - 0.5|$. The pair that satisfies the following condition is selected for querying:

$$(b^*, c^*) = \arg \max_{b,c} S_{bc}. \quad (21)$$

As with EGL, our selection based on uncertainty sampling approach requires only the features of bag b , \mathbf{X}_b , and the model parameter vector \mathbf{w} . The computational complexity of the selection phase is $O(\sum_{b \in \mathcal{U}} n_b C d)$, where n_b is the number of instances in bag b , C is the number of classes, and d is the dimension of the feature vector \mathbf{x}_{bi} .

3.4 Model Update

In this section, we present the model update process after every query. We will begin with the update of training data after one query is performed and proceed with the update of the model parameters.

3.4.1 Training data update

After querying the bag-class pair from the unlabeled pool, we update this pair into the training data. Specifically, let b^{*k}, c^{*k} is the selected pair at the k th query, $t \in \{0, 1\}$ is the value of $Y_{b^{*k}c^{*k}}$, and S_1^k, \dots, S_B^k are the available label set of B bags at the k th query, we have:

$$Y_{bc}^k = \begin{cases} Y_{bc}^{k-1}, & \text{if } (b, c) \neq (b^{*k}, c^{*k}) \\ t, & \text{if } (b, c) = (b^{*k}, c^{*k}). \end{cases}$$

and

$$S_b^k = \begin{cases} S_b^{k-1}, & \text{if } b \neq b^{*k} \\ S_b^{k-1} \cup \{c^{*k}\}, & \text{if } b = b^{*k}; \end{cases}$$

$$\mathcal{U} \leftarrow \mathcal{U} \setminus \{(b^{*k}, c^{*k})\} \quad (22)$$

3.4.2 Model parameter update

We demonstrate how our model is updated after a bag-class pair is queried. Specifically, After each query, our goal is to update the model parameters by minimizing the MML objective that takes into account all the available labels:

$$\bar{F}^k(\mathbf{w}) = \frac{1}{\sum_b^B |S_b^k|} \sum_{i=1}^k f_{b^{*k}c^{*k}}(\mathbf{w}, Y_{b^{*k}c^{*k}}^i), \quad (23)$$

where

$$f_{bc}(\mathbf{w}, l) = -[I(l=0) \sum_{i=1}^{n_b} \log P(y_{bi} \neq c) + I(l=1) \log(1 - e^{\sum_{i=1}^{n_b} \log P(y_{bi} \neq c)})] \quad (24)$$

is the objective associated with the label of bag b and class c . To reduce the computational complexity associated with the minimization of the MML objective that includes the entire available data, we propose a stochastic gradient descent (SGD) approach that involves updating the parameter vector based on the gradient associated with the newly obtained label:

$$\mathbf{w}^k = \mathcal{P}(\mathbf{w}^{k-1} - \eta_k g^k(\mathbf{w}^{k-1})), \quad (25)$$

where $g^k = \nabla f_{b^{*k}c^{*k}}(\mathbf{w}, Y_{b^{*k}c^{*k}}^k)$ is the gradient of $f_{b^{*k}c^{*k}}(\mathbf{w}, Y_{b^{*k}c^{*k}}^k)$, η_k is the step-size, and \mathcal{P} is used to denote a projection onto the feasible solution set. For the gradient, g^k is computed by

$$g^k = - \sum_{i=1}^{n_b} \left(I(Y_{b_k c_k} = 1) P(Y_{b_k c_k} = 0) \frac{P(y_{b_k i} = c_k)}{P(Y_{b_k c_k} = 1)} - I(Y_{b_k c_k} = 0) P(Y_{b_k i} = c_k) \right) \left(I(c = c_k) - I(c \neq c_k) \frac{P(y_{b_k i} = c)}{1 - P(y_{b_k i} = c_k)} \right) \mathbf{x}_{b_k i}. \quad (26)$$

We consider a monotonically decreasing step size that follows this form: $\eta_k = \frac{c'}{\lambda k + c''}$, where c' and c'' are constants, λ is the regularization term. Similar to algorithms such as PEGASOS [40], we can show that the optimal solution is guaranteed to be contained in a sphere of a given radius. For our problem, we can show that the radius is $\tau = \sqrt{\frac{2}{\lambda}} \max(\log(C), \max_b(n_b) \frac{1}{C-1})$ and the solution must reside in the sphere $\mathcal{S}_\tau = \{\mathbf{w} \mid \|\mathbf{w}\| \leq \tau\}$. The projection operator onto the sphere \mathcal{S}_τ is denoted by \mathcal{P} and is give by

$$\mathcal{P}(\mathbf{w}) = \begin{cases} \mathbf{w}, & \|\mathbf{w}\| \leq \tau \\ \tau \frac{\mathbf{w}}{\|\mathbf{w}\|}, & \|\mathbf{w}\| > \tau. \end{cases}$$

The detailed derivation of τ is provided in Appendix A. Consider the MML as the average of bag-class pairs to solve the minimization problem using SGD leads to more randomness and increases the variant of the gradient. Therefore, we introduce another method for update the model parameter after a bag-label pair comes. Instead of update only the queried pair, we update the full bag that contain the bag-label queried pair. Specifically, after each query, we update the model parameters by minimizing the following MML:

$$\bar{F}^k(\mathbf{w}) = \frac{1}{\sum_b^B |S_b^k|} \sum_{b=1}^B \sum_{c \in S_b^k} f_{bc}(\mathbf{w}, Y_{bc}^k), \quad (27)$$

where $f_{bc}(\mathbf{w}, Y_{bc}^k)$ is computed as in (24). The update rule is the same with the bag-class update (25):

$$\mathbf{w}^k = \mathcal{P}(\mathbf{w}^{k-1} - \eta_k g^k(\mathbf{w}^{k-1})), \quad (28)$$

where η_k and $\mathcal{P}(\cdot)$ are described right after (25). The difference here is that g^k is the gradient of the full bag $g^k = \nabla \sum_{c \in S_b^k} f_{bc}(\mathbf{w}, Y_{bc}^k)$ and g^k w.r.t \mathbf{w}_c . This bag contains the queried bag-class pair. The gradient g^k is computed as:

$$g^k = - \sum_{c_k \in S_b^k} \sum_{i=1}^{n_b} \left(I(Y_{b_k c_k} = 1) P(Y_{b_k c_k} = 0) \frac{P(y_{b_k i} = c_k)}{P(Y_{b_k c_k} = 1)} - I(Y_{b_k c_k} = 0) P(Y_{b_k i} = c_k) \right) \left(I(c = c_k) - I(c \neq c_k) \frac{P(y_{b_k i} = c)}{1 - P(y_{b_k i} = c_k)} \right) \mathbf{x}_{b_k i}. \quad (29)$$

From (26) and (29), the only difference between the bag-class pair update and the full bag update is the gradient of the marginal log-likelihood g^k used in updating the model parameters. In the bag-class pair update, g^k is the gradient of the bag-class pair queried. Other available class labels of this bag are not taken into account to update the model. However, in the full bag update, all other available class labels beside the queried bag-class pair are used in updating the model.

4 EXPERIMENTS

In this section, we evaluate our proposed approach for learning an instance/bag-level classifier under the MIML active learning with missing labels by deploying three different experimental settings: (i) model update inference, we compare our online (SGD) approach with offline GD approach in [38] to verify the correctness of our approach; (ii) instance selection criterion and type comparison: we run our model with four different selection criteria; specifically, we compare our proposed bag-class pair selection based on EGL and uncertainty sampling with other two selection criteria: bag selection and bag-then-label selection from [34] and [35], respectively, to verify the effectiveness of our selection criterion and type; (iii) finally, we compare our framework with two other state-of-the-art approaches on the MIML setting with active learning [34], [35]. To perform the comparison, with each compared method, we run our model with the instance selection strategy used by the compared method. In our comparison, we evaluate bag-level metrics as a function of the epochs and the number of bag-class pair queries.

Dataset: We perform a comparison on three benchmark datasets, including: HJA - a bird song audio recordings dataset [37], and two letter datasets [41] (i.e., Letter Carol and Letter Frost). HJA is bird song audio recordings dataset. Each 10-second audio recording is converted into a bag consisting of audio segments obtained via time-frequency domain segmentation and featurized as in [37]. The bags are manually labeled to indicate the presence or absence of bird species. HJA dataset includes 645 bags with a total of 13 bird species. Letter Carol and Letter Frost are also MIML datasets, each is taken from a poem [41]. Each word (bag) contains multiple letters. Each letter is described by a 16-dimensional feature vector and is annotated by one of 26 letter labels¹ from 'a' to 'z'. The labels for each word are the union of its letter labels. On each dataset, we generate 10 cross validation sets and the results are reported based on the average of these 10 sets.

4.1 Model update inference methods - comparison

In this section, we run our online SGD and offline GD approach in [38] on three aforementioned datasets. In our online SGD, we run experiments when model is updated with new coming bag, called bag-SGD and with new coming bag-class pair, called pair-SGD. For each dataset and on each cross validation set, we run all training data to learn the model parameter. We run three algorithms until convergence to show that the performance of our online

SGD approach is comparable to GD approach.

Evaluation metrics: We report the results based on bag accuracy used for MIML learning evaluation in [42] as the function of the number of epochs. Besides, we present the log-likelihood function as the function of number of epochs.

Result analysis: The performance of our two online SGD update method (the bag-class pair update and the full bag update) vs. the full data GD update method in terms of bag accuracy and the log-likelihood is shown in Fig. 3 and Fig. 4. In these two figures, the x-axis is the number of epochs - each epoch is the number of iteration in which all training data is learned to update the model parameters. As can be seen in Fig. 4, the value of the log-likelihood of three methods converges after 1500 epochs. The full data GD update takes more time to converge in comparing to the bag-SGD update and the bag-class pair update, but it converges to a higher value of log-likelihood than our two online update methods, which is expected.

4.2 Instance selection criterion and type

We run our model (MIMLILL-AL) on four selection strategies selected from a combinations of two instance selection criteria: (i) EGL and (ii) uncertainty sampling and three types of query level: (1) bag only, (2) bag-class pair (our approach), and (3) bag-then-label. For (1) bag only criterion, a bag is selected from the unlabeled data based on uncertainty criteria, then all labels of the selected bag are queried to be added into the current training data to update the model. For (2) a bag-class pair criterion (our approach), a bag-class pair is selected simultaneously. For (3) bag-then-label criterion, a bag is selected from the unlabeled data based on the uncertainty sampling and diversity, then a class of the bag(s) is queried based on the distance from the label to the thresholding dummy label. From the six combinations, we ignore two bag only selection strategies based on (i) EGL and (ii) uncertainty sampling to avoid the computational complexity issue mentioned in Section 3.3. To initialize all active learning methods, for each dataset and cross validation, a small number of bags is fully labeled and is used for training the initial model and the remaining bags are unlabeled. Queries are then made to label bags or bag-class pairs from the unlabeled data. Additionally, we consider random instance selection of bag-class pairs as a baseline method.

Evaluation metrics: We report the results based on metrics used for MIML learning evaluation in [42] including: bag accuracy, average precision, Hamming loss, and one-error as the function of the number of bag-class pair queries.

Result analysis: The performance of our framework on different selection criteria is presented in Fig. 5-Fig. 8. In these figures, the x-axis presents the number of bag-class pair label queries, the y-axis shows the bag accuracy, average precision, Hamming loss, and one error of our method on five different selection criteria. From the curves of five different criteria, we can see that the performance of our two proposed bag-class pair selection criteria based on EGL and uncertainty sampling surpasses the state-of-the-art two selection criteria: bag selection [34] and bag-then-label [35] and the baseline random selection. Our two approaches reach to the performance, which is obtained when all

1. Specifically at this point, 24 letters are present in poem

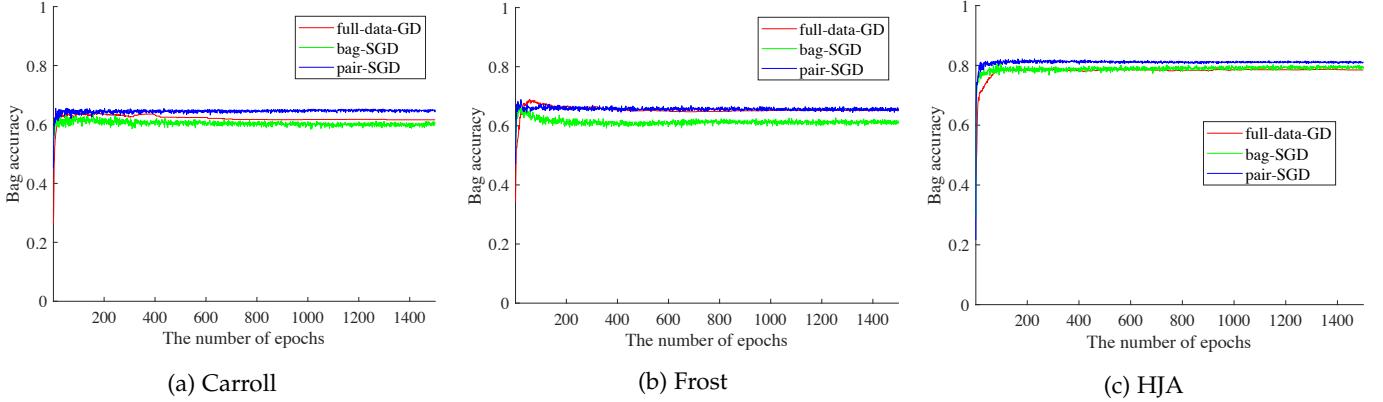


Fig. 3: Bag accuracy as the function of the number of epochs of three model update inference methods: full-data GD, bag-SGD and pair-SGD on three datasets: Carroll, Frost, and HJA.

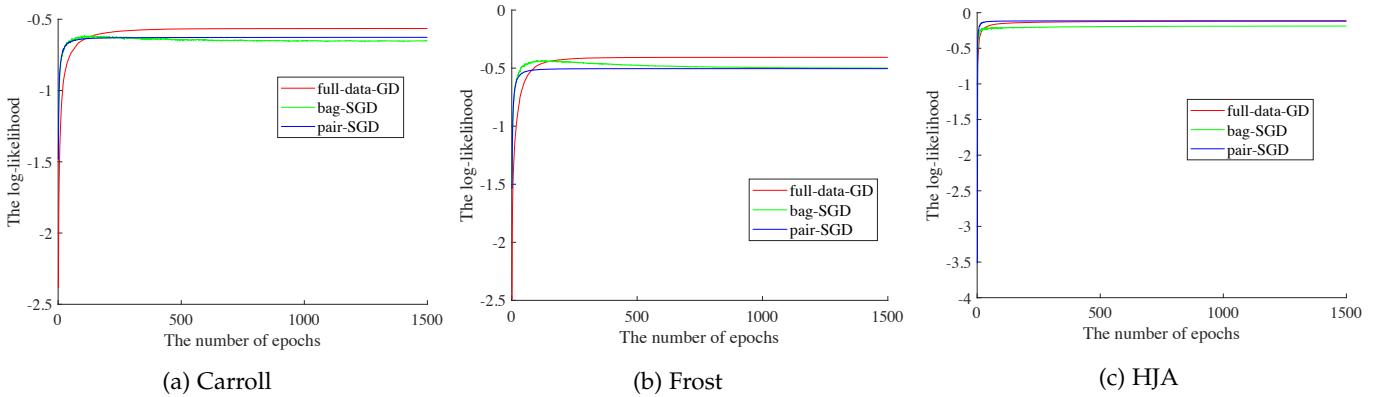


Fig. 4: Log-likelihood as the function of the number of epochs of three model update inference methods: full-data GD, bag-SGD and pair-SGD on three datasets: Carroll, Frost, and HJA.

training data is available, very earlier than the rest methods. Note, in the entire bag selection, after each query, all classes of the selected bag are labeled. To reflect the increase in labeling cost that is proportional to the number of target classes, we plot the performance of each method against the number of bag-class queries rather than queries. When an entire bag is queried, the number of associated individual bag-class queries is C/k , where k is the number of classes in a bag that takes the equivalent labeling cost with a single class in the bag. In our illustration, $k = 1$, this means that the labeling cost for a single class in a bag equals to the labeling cost for a bag-class pair. We can select $k \geq 1$ when the number of classes is small. Hence, it should be noted that the performance curves in Figs. 5-8 for bag query methods appear to be sampled less densely than the methods that rely on bag-class queries.

4.3 Comparison with alternative methods

We compare our model (MIMLL-AL) with the state-of-the-art algorithms for MIML active learning: MIML-AL [33] and MIMLSVM-AL [34]. On each dataset and on each cross validation, a small part of training data is available, the rest is considered as unlabeled data for active learning. Each query request data to label from the unlabeled data.

Algorithms: The method in [34] is specifically designed

based on MIMLSVM. It firstly degenerates bags to single-instance representation and then directly employ traditional active learning method for label querying (select a bag to label all classes every query), which does not truly exploit the characteristics of MIML tasks. The authors in [35] propose an approach for active learning in MIML setting based on the work in [33]. For each query, the bag is selected first based on the uncertainty (the gap between the predicted number of positive labels of the bag and the average number of positive labels of the training data) and diversity of the bag (how many labels of the bag was queried before). Then a class is pointed out to be labeled for the selected bag based on the distance from the label to the thresholding dummy label. In their methods, not only one selected label is queried, but also the key instance which is most relevant to queried label is asked.

Evaluation metrics: We report the results based on metrics used for MIML learning evaluation in [42] including: bag accuracy and Hamming loss as the function of the number of queries.

Result analysis: The performance of our approach and the MIML-AL [35] method that is based on bag-then-label selection criterion is shown in Fig. 9 and Fig. 10. The x-axis is the number of bag-class pair label queries and the y-axis is bag accuracy or Hamming loss. From these figures, our framework appears superior to MIML-AL in

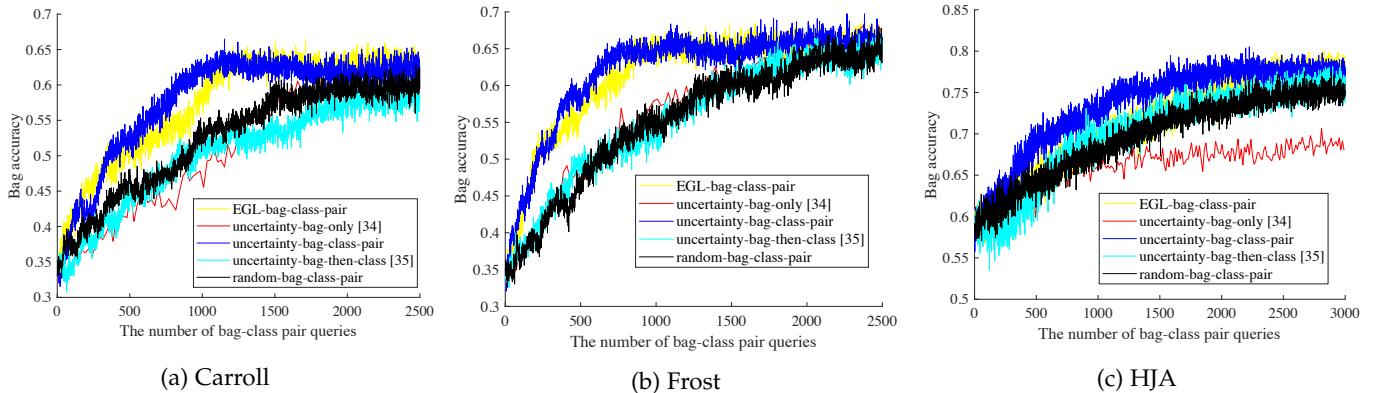


Fig. 5: Bag accuracy as the function of the number of bag-class pair queries of five selection criteria on three datasets: Carroll, Frost, and HJA.

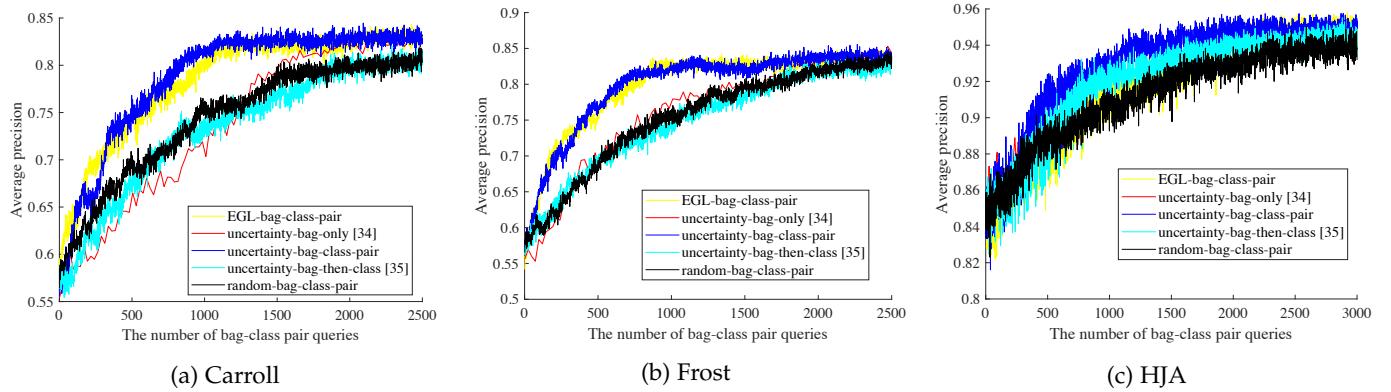


Fig. 6: Average precision as the function of the number of bag-class pair queries of five selection criteria on three datasets: Carroll, Frost, and HJA.

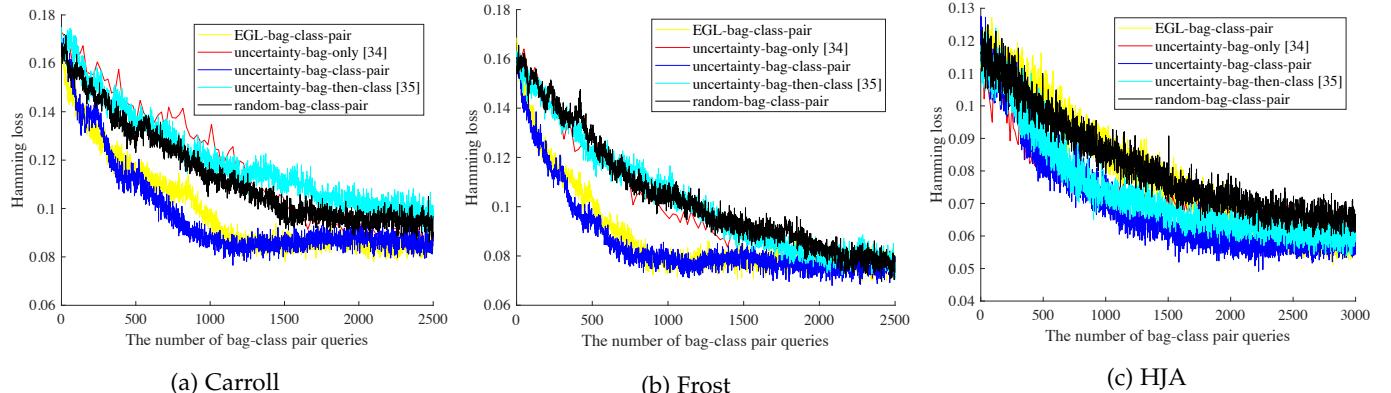


Fig. 7: Hamming loss as the function of the number of bag-class pair queries of five selection criteria on three datasets: Carroll, Frost, and HJA.

both initialized performance and during querying process. This may be due to the difference of the base classifiers used in our framework and MIML-AL. In MIML-AL, the base classifier is based on fastMIML [43], which constructs a low-dimensional subspace shared by all labels, and then trains label specific linear models to optimize approximated ranking loss via stochastic gradient descent. Note that the performance of MIML-AL in terms of Hamming loss shown in Fig. 10 is comparable to the one reported in MIML-AL paper [35] in Carroll and Frost datasets. The performance

of our approach and the MIMLSVM-AL [34] method that is based on bag only selection criterion is shown in Fig. 11 and Fig. 12. The x-axis is the number of bag queries and the y-axis is bag accuracy or Hamming loss. From these figures, the performance of our framework surpasses the performance of MIMLSVM-AL in Carroll and Frost datasets in terms of bag accuracy and Hamming loss. In case of HJA dataset, the performance of MIMLSVM-AL in terms of Hamming loss and bag accuracy is greater than ours in earlier queries. When the number of queries increases,

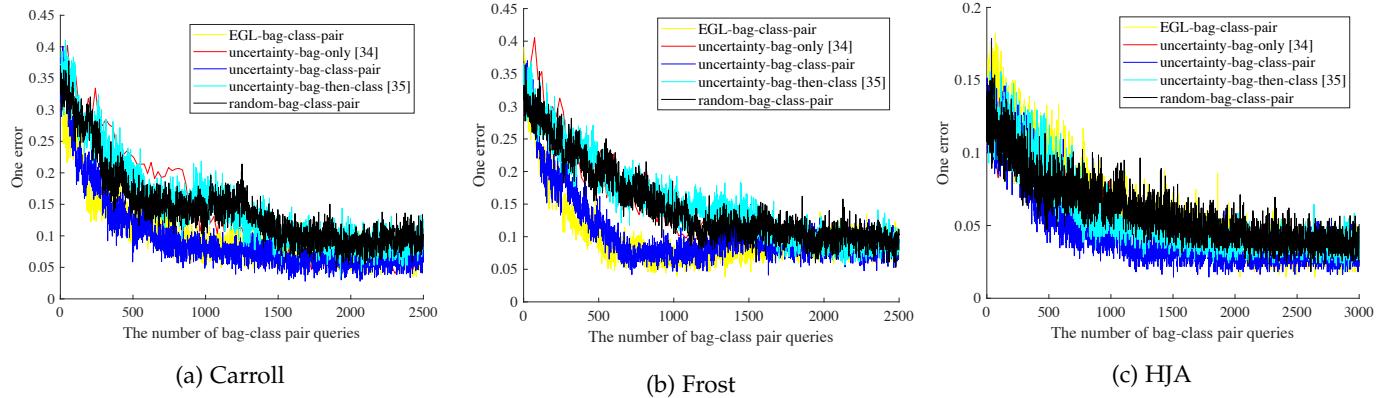


Fig. 8: One error as the function of the number of bag-class pair queries of five selection criteria on three datasets: Carroll, Frost, and HJA.

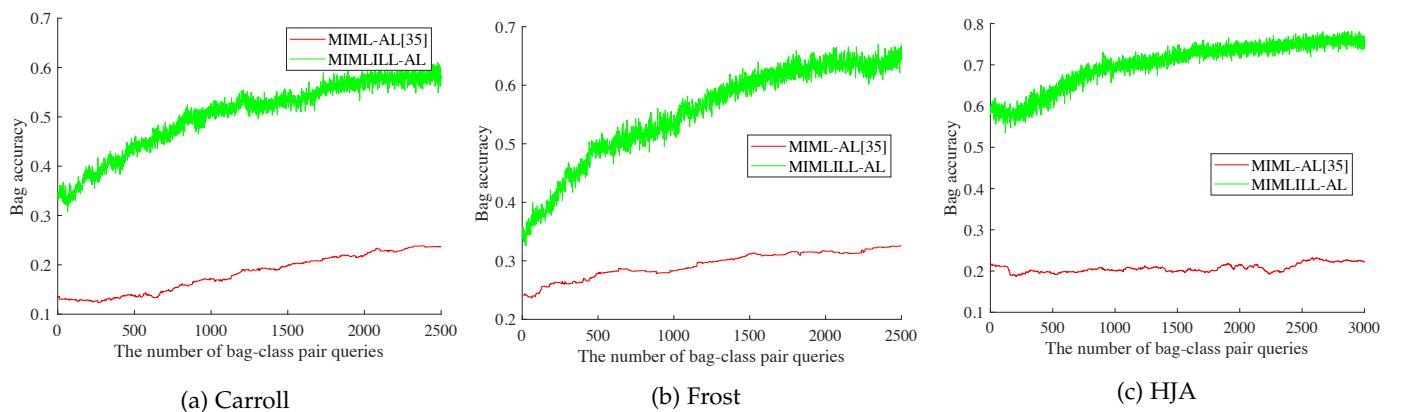


Fig. 9: Bag accuracy as the function of the number of bag-class pair queries of our method and MIML-AL [35] on three datasets: Carroll, Frost, and HJA.

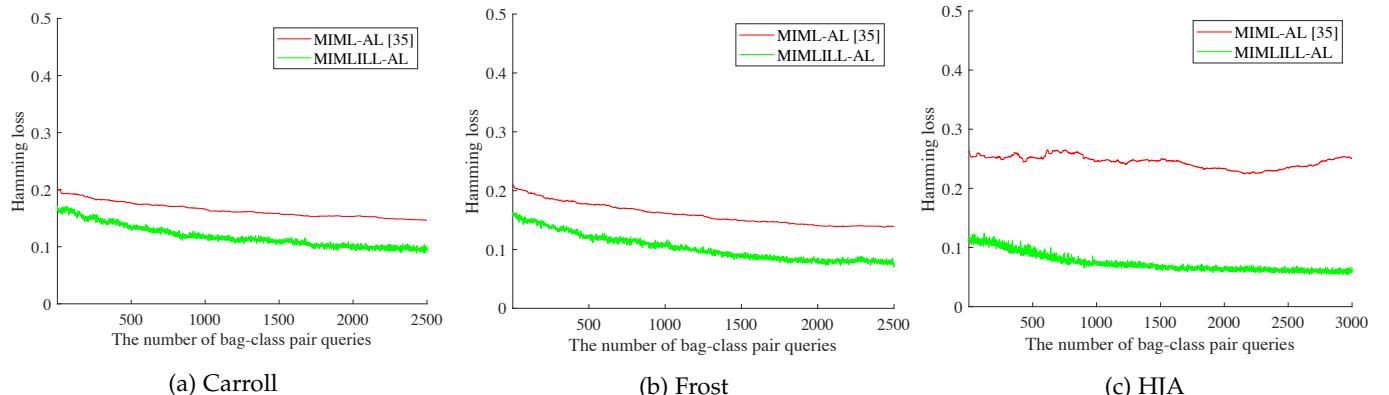


Fig. 10: Hamming loss as the function of the number of bag-class pair queries of our method and MIML-AL [35] on three datasets: Carroll, Frost, and HJA.

the performance of our approach reaches and surpasses the performance of MIMLSVM-AL. We suspect that the size of HJA dataset is big enough and the number of classes of HJA dataset is small enough in comparison to Carroll and Frost datasets such that SVM can obtain good classification performance during the initialization step. After that, the performance of MIMLSVM-AL increases gradually when the number of queries increases. The rate of performance increase for MIMLSVM-AL is slower than the rate for our

approach. Therefore, our approach achieves better performance in terms of Hamming loss and bag accuracy than MIMLSVM-AL after some queries.

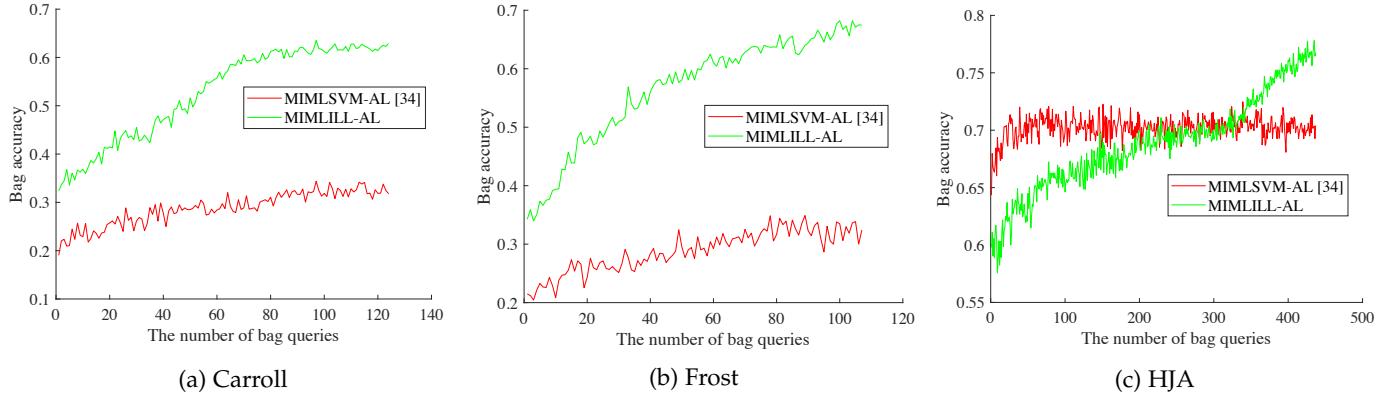


Fig. 11: Bag accuracy as the function of the number of bag queries of our method and MIMLSVM-AL [34] on three datasets: Carroll, Frost, and HJA.

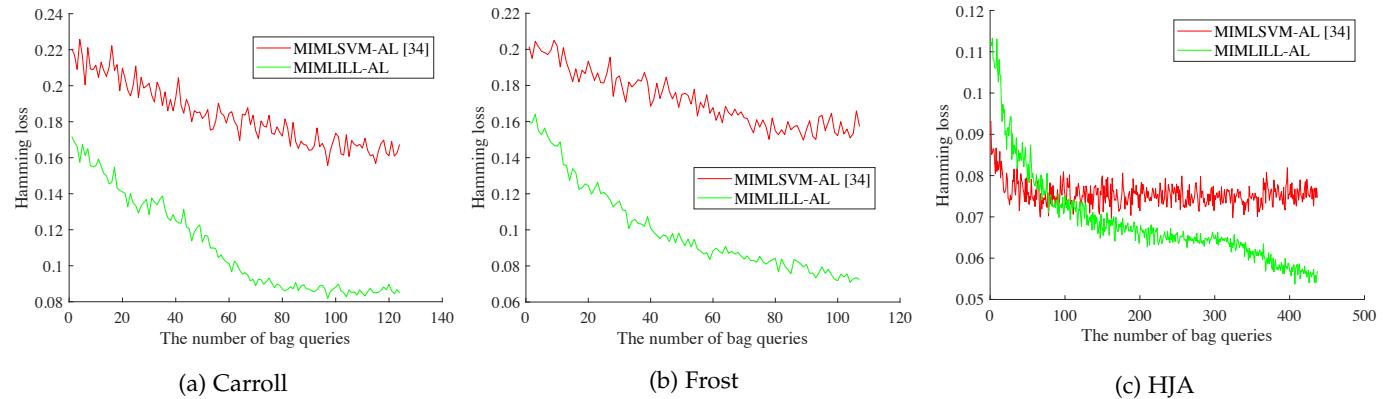


Fig. 12: Hamming loss as the function of the number of bag queries of our method and MIMLSVM-AL [34] on three datasets: Carroll, Frost, and HJA.

5 CONCLUSION

In this paper, we developed a comprehensive framework for active learning under the MIML-ILL setting. We considered the MIML-ILL model for the classifier use in this paper. To alleviate the computational complexity associated with model update after each query, we developed an online version for maximizing the marginal log-likelihood of the MIML-ILL model. For the query stage, we proposed a novel approach for selecting a bag-class pair by extending EGL and uncertainty sampling to the MIML setting. The experimental evaluation demonstrated the effectiveness and efficiency of the proposed approach.

REFERENCES

- [1] D. Angluin, "Queries and concept learning," *Machine learning*, vol. 2, no. 4, pp. 319–342, 1988.
- [2] D. A. Cohn, Z. Ghahramani, and M. I. Jordan, "Active learning with statistical models," *Journal of artificial intelligence research*, vol. 4, pp. 129–145, 1996.
- [3] R. D. King, K. E. Whelan, F. M. Jones, P. G. Reiser, C. H. Bryant, S. H. Muggleton, D. B. Kell, and S. G. Oliver, "Functional genomic hypothesis generation and experimentation by a robot scientist," *Nature*, vol. 427, no. 6971, pp. 247–252, 2004.
- [4] R. D. King, J. Rowland, S. G. Oliver, M. Young, W. Aubrey, E. Byrne, M. Liakata, M. Markham, P. Pir, L. N. Soldatova *et al.*, "The automation of science," *Science*, vol. 324, no. 5923, pp. 85–89, 2009.
- [5] D. Cohn, L. Atlas, and R. Ladner, "Improving generalization with active learning," *Machine learning*, vol. 15, no. 2, pp. 201–221, 1994.
- [6] I. Dagan and S. P. Engelson, "Committee-based sampling for training probabilistic classifiers," in *Machine Learning Proceedings 1995*. Elsevier, 1995, pp. 150–157.
- [7] T. M. Mitchell, "Generalization as search," *Artificial intelligence*, vol. 18, no. 2, pp. 203–226, 1982.
- [8] H. S. Seung, M. Opper, and H. Sompolinsky, "Query by committee," in *Proceedings of the fifth annual workshop on Computational learning theory*, 1992, pp. 287–294.
- [9] S. Dasgupta, D. J. Hsu, and C. Monteleoni, "A general agnostic active learning algorithm," in *Advances in neural information processing systems*, 2008, pp. 353–360.
- [10] V. Krishnamurthy, "Algorithms for optimal scheduling and management of hidden markov model sensors," *IEEE Transactions on Signal Processing*, vol. 50, no. 6, pp. 1382–1397, 2002.
- [11] H. Yu, "Svm selective sampling for ranking with application to data retrieval," in *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*, 2005, pp. 354–363.
- [12] A. Fujii, T. Tokunaga, K. Inui, and H. Tanaka, "Selective sampling for example-based word sense disambiguation," *Computational Linguistics*, vol. 24, no. 4, pp. 573–597, 1998.
- [13] C. A. Thompson, M. E. Calif, and R. J. Mooney, "Active learning for natural language parsing and information extraction," in *ICML*. Citeseer, 1999, pp. 406–414.
- [14] R. Moskovich, N. Nissim, D. Stopek, C. Feher, R. Englert, and Y. Eliovici, "Improving the detection of unknown computer worms activity using active learning," in *Annual Conference on Artificial Intelligence*. Springer, 2007, pp. 489–493.
- [15] D. D. Lewis and W. A. Gale, "A sequential algorithm for training text classifiers," in *SIGIR'94*. Springer, 1994, pp. 3–12.
- [16] A. K. McCallum and K. Nigam, "Employing em and pool-based active learning for text classification," in *Proc. International*

- Conference on Machine Learning (ICML).* Citeseer, 1998, pp. 359–367.
- [17] S. C. Hoi, R. Jin, and M. R. Lyu, "Large-scale text categorization by batch mode active learning," in *Proceedings of the 15th international conference on World Wide Web*, 2006, pp. 633–642.
- [18] S. Tong and E. Chang, "Support vector machine active learning for image retrieval," in *Proceedings of the ninth ACM international conference on Multimedia*, 2001, pp. 107–118.
- [19] C. Zhang and T. Chen, "An active learning framework for content-based information retrieval," *IEEE transactions on multimedia*, vol. 4, no. 2, pp. 260–268, 2002.
- [20] J. Yang *et al.*, "Automatically labeling video data using multi-class active learning," in *Proceedings Ninth IEEE international conference on computer vision*. IEEE, 2003, pp. 516–523.
- [21] A. G. Hauptmann, W.-H. Lin, R. Yan, J. Yang, and M.-Y. Chen, "Extreme video retrieval: joint maximization of human and computer performance," in *Proceedings of the 14th ACM international conference on Multimedia*, 2006, pp. 385–394.
- [22] G. Tur, D. Hakkani-Tür, and R. E. Schapire, "Combining active and semi-supervised learning for spoken language understanding," *Speech Communication*, vol. 45, no. 2, pp. 171–186, 2005.
- [23] Y. Liu, "Active learning with support vector machine applied to gene expression data for cancer classification," *Journal of chemical information and computer sciences*, vol. 44, no. 6, pp. 1936–1941, 2004.
- [24] K. Salmani and M. Sridharan, "Multi-instance active learning with online labeling for object recognition," in *The Twenty-Seventh International Flairs Conference*, 2014.
- [25] D. Zhang, F. Wang, Z. Shi, and C. Zhang, "Interactive localized content based image retrieval with multiple-instance active learning," *Pattern Recognition*, vol. 43, no. 2, pp. 478–484, 2010.
- [26] B. Settles, M. Craven, and S. Ray, "Multiple-instance active learning," in *Advances in neural information processing systems*, 2008, pp. 1289–1296.
- [27] X. Li and Y. Guo, "Active learning with multi-label svm classification," in *Twenty-Third International Joint Conference on Artificial Intelligence*, 2013.
- [28] J. Tang, Z.-J. Zha, D. Tao, and T.-S. Chua, "Semantic-gap-oriented active learning for multilabel image annotation," *IEEE Transactions on Image Processing*, vol. 21, no. 4, pp. 2354–2360, 2011.
- [29] X. Li, L. Wang, and E. Sung, "Multilabel svm active learning for image classification," in *2004 International Conference on Image Processing, 2004. ICIP'04.*, vol. 4. IEEE, 2004, pp. 2207–2210.
- [30] C.-W. Hung and H.-T. Lin, "Multi-label active learning with auxiliary learner," in *Asian conference on machine learning*, 2011, pp. 315–332.
- [31] J. Wu, V. S. Sheng, J. Zhang, P. Zhao, and Z. Cui, "Multi-label active learning for image classification," in *2014 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2014, pp. 5227–5231.
- [32] G.-J. Qi, X.-S. Hua, Y. Rui, J. Tang, and H.-J. Zhang, "Two-dimensional active learning for image classification," in *2008 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2008, pp. 1–8.
- [33] S.-J. Huang and Z.-H. Zhou, "Active query driven by uncertainty and diversity for incremental multi-label learning," in *2013 IEEE 13th International Conference on Data Mining*. IEEE, 2013, pp. 1079–1084.
- [34] R. Retz and F. Schwenker, "Active multi-instance multi-label learning," in *Analysis of Large and Complex Data*. Springer, 2016, pp. 91–101.
- [35] S.-J. Huang, N. Gao, and S. Chen, "Multi-instance multi-label active learning," in *IJCAI*, 2017, pp. 1886–1892.
- [36] J. Wu, W. Zhu, Y. Jiang, G. Sun, and Y. Gao, "Predicting protein functions of bacteria genomes via multi-instance multi-label active learning," in *2018 IEEE 3rd International Conference on Integrated Circuits and Microsystems (ICICM)*. IEEE, 2018, pp. 302–307.
- [37] F. Briggs, B. Lakshminarayanan, L. Neal, X. Z. Fern, R. Raich, S. J. Hadley, A. S. Hadley, and M. G. Betts, "Acoustic classification of multiple simultaneous bird species: A multi-instance multi-label approach," *The Journal of the Acoustical Society of America*, pp. 4640–4650, 2012.
- [38] T. Nguyen and R. Raich, "Incomplete label multiple instance multiple label learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.
- [39] B. Settles, "Active learning literature survey," University of Wisconsin-Madison Department of Computer Sciences, Tech. Rep., 2009.
- [40] S. Shalev-Shwartz, Y. Singer, N. Srebro, and A. Cotter, "Pegasos: Primal estimated sub-gradient solver for svm," *Mathematical programming*, vol. 127, no. 1, pp. 3–30, 2011.
- [41] F. Briggs, X. Z. Fern, and R. Raich, "Rank-loss support instance machines for mimi instance annotation," in *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2012, pp. 534–542.
- [42] G. Tsoumakas, I. Katakis, and I. Vlahavas, "Mining multi-label data," in *Data Mining and Knowledge Discovery Handbook*, 2009, pp. 667–685.
- [43] S.-J. Huang, W. Gao, and Z.-H. Zhou, "Fast multi-instance multi-label learning," *IEEE transactions on pattern analysis and machine intelligence*, vol. 41, no. 11, pp. 2614–2627, 2018.

Supplemental Material- "Active Learning in Incomplete Label Multiple Instance Multiple Label Learning", Tam Nguyen and Raviv Raich.

A. BOUNDING THE OPTIMAL PARAMETER VECTOR

In the following, we derive a bound on the l_2 -norm of the parameter vector \mathbf{w} . We begin by expressing the regularized negative marginal log-likelihood objective as follows:

$$f(\mathbf{w}) = f_0(\mathbf{w}) + \frac{\lambda}{2} \|\mathbf{w}\|^2 \quad (30)$$

where $f_0(\mathbf{w})$ is the negative marginal log-likelihood objective function, λ is the quadratic regularization parameter, and $\|\cdot\|$ denotes the l_2 -norm. Specifically the negative marginal log-likelihood is given by

$$f_0(\mathbf{w}) = \frac{1}{\sum_b |S_b|} \sum_{b=1}^B \sum_{c \in S_b} f_{bc}^o(\mathbf{w}, Y_{bc}) \quad (31)$$

where

$$f_{bc}^o(\mathbf{w}, l) = -I(l=0) \sum_{i=1}^{n_b} \log P(y_{bi} \neq c) - I(l=1) \log(1 - e^{\sum_{i=1}^{n_b} \log P(y_{bi} \neq c)}). \quad (32)$$

Let \mathbf{w}^* the model parameter vector, which minimizes $f(\mathbf{w})$. Consequently, we have

$$f(\mathbf{w}^*) \leq f(\mathbf{w}) \quad \forall \mathbf{w}. \quad (33)$$

Since (33) holds for any \mathbf{w} , replacing $\mathbf{w} = 0$ into (33) and replacing $f(\mathbf{w}^*)$ with the RHS of (30) with \mathbf{w}^* in place of \mathbf{w} yields

$$f_0(\mathbf{w}^*) + \lambda \frac{\|\mathbf{w}^*\|^2}{2} \leq f_0(\mathbf{0}) = f_0(\mathbf{0}). \quad (34)$$

Reorganizing (34), we obtain

$$\|\mathbf{w}^*\| \leq \sqrt{\frac{2}{\lambda} (f_0(\mathbf{0}) - f_0(\mathbf{w}^*)}). \quad (35)$$

Since $f_0(\mathbf{w}^*) \geq 0$, we can upper bound the RHS of (35) by $\sqrt{\frac{2}{\lambda} f_0(\mathbf{0})}$ and obtain the following bound on $\|\mathbf{w}^*\|$:

$$\|\mathbf{w}^*\| \leq \sqrt{\frac{2}{\lambda} f_0(\mathbf{0})}. \quad (36)$$

Next, we proceed by bounding $f_0(\mathbf{0})$ to further simplify the RHS of (36). Substituting

$$P(y_{bi} \neq c | \mathbf{w})|_{\mathbf{w}=0} = \frac{C-1}{C}. \quad (37)$$

into (32), we obtain

$$f_{bc}^o(\mathbf{0}, l) = (1 - I(l=1))K_b + I(l=1)(-\log(1 - e^{-K_b})), \quad (38)$$

where $K_b = n_b \log \frac{C}{C-1}$. We can simplify the bound by replacing the indicators in (38) with the max function as follows:

$$f_{bc}^o(\mathbf{0}, l) \leq \max(K_b, -\log(1 - e^{-K_b})). \quad (39)$$

To bound the first term in the max function of (39), we bound K_b as follows:

$$\begin{aligned} K_b &= n_b \log \frac{C}{C-1} \\ &\leq \max_b n_b \log \frac{C}{C-1} \\ &\leq \frac{\max_b n_b}{C-1} \end{aligned} \tag{40}$$

where the last inequality uses $\log(1+x) \leq x$ with $x = \frac{1}{C-1}$. To bound the second term in the max function of (39), we start by lower bounding K_b by $K_b \geq \log \frac{C}{C-1}$ and then bound $-\log(1 - e^{-K_b})$, which is monotonically decreasing in K_b as follows:

$$\begin{aligned} -\log(1 - e^{-K_b}) &\leq -\log(1 - e^{-\log \frac{C}{C-1}}) \\ &= -\log(1 - e^{\log \frac{C-1}{C}}) \\ &= -\log(1 - \frac{C-1}{C}) \\ &= -\log(\frac{C-(C-1)}{C}) \\ &= -\log(\frac{1}{C}) \\ &= \log(C). \end{aligned} \tag{41}$$

Substituting the bounds on the first and second term within the max of (39), respectively, (40) and (41), back into (39), we obtain:

$$f_{bc}^o(\mathbf{0}, l) \leq \max(\log(C), \max_b(n_b) \frac{1}{C-1}). \tag{42}$$

Substituting the bound on $f_{bc}^o(\mathbf{0}, l)$ in (42) into (31), we obtain

$$f_0(\mathbf{0}) \leq \max(\log(C), \max_b(n_b) \frac{1}{C-1}). \tag{43}$$

Finally, by substituting the bound on $f_0(\mathbf{0})$ in (43) in (36), we obtain the following bound on the l_2 -norm of the optimal parameter vector:

$$\|\mathbf{w}^*\| \leq \sqrt{\frac{2}{\lambda} \max(\log(C), \max_b(n_b) \frac{1}{C-1})}. \tag{44}$$

Let $\tau = \sqrt{\frac{2}{\lambda} \max(\log(C), \max_b(n_b) \frac{1}{C-1})}$, we have $\|\mathbf{w}^*\| \leq \tau$ and τ is the bound on the l_2 -norm of the parameter vector \mathbf{w} . Note that this bound holds regardless of the value of the data.