Active Query Driven by Uncertainty and Diversity for Incremental Multi-Label Learning *

Sheng-Jun Huang and Zhi-Hua Zhou
National Key Laboratory for Novel Software Technology
Nanjing University, Nanjing 210023, China
{huangsj, zhouzh}@lamda.nju.edu.cn

Abstract—In multi-label learning, it is rather expensive to label instances since they are simultaneously associated with multiple labels. Therefore, active learning, which reduces the labeling cost by actively querying the labels of the most valuable data, becomes particularly important for multi-label learning. A strong multi-label active learning algorithm usually consists of two crucial elements: a reasonable criterion to evaluate the gain of queried label, and an effective classification model, based on whose prediction the criterion can be accurately computed. In this paper, we first introduce an effective multi-label classification model by combining label ranking with threshold learning, which is incrementally trained to avoid retraining from scratch after every query. Based on this model, we then propose to exploit both uncertainty and diversity in the instance space as well as the label space, and actively query the instance-label pairs which can improve the classification model most. Experimental results demonstrate the superiority of the proposed approach to state-of-the-art methods.

Keywords—active learning; multi-label learning; uncertainty; diversity

I. INTRODUCTION

In many applications, we have plenty of unlabeled data but few labeled data. While labeling is usually expensive since it requires the participation of human experts, training an accurate model with as few labeled data as possible becomes a challenge of great significance. Active learning, which reduces the labeling cost by actively selecting the most valuable data to query their labels, is a leading approach to this goal [23]. The key task in active learning is to design a selection criterion such that queried labels can improve the classification model most. During the past years, many active selection criteria have been proposed. For example, uncertainty measures the confidence of the current model on classifying an instance [26], diversity measures how different an instance is from the labeled data [2], density measures the representativeness of an instance to the whole data set [18], and so on. There are also some other approaches trying to consider different criteria simultaneously [6], [13].

In traditional supervised classification problems, one instance is assumed to be associated with only one label. However, in many real-world applications, an object can have multiple labels simultaneously. Multi-label learning is a framework dealing with such objects [32]. To label the multi-label examples, each of the multiple labels should be decided whether a proper one for an instance. Obviously, the labeling cost is even higher than that of single label learning, and thus active learning under the multi-label setting has attracted more and more attention.

In multi-label active learning, the main efforts focus on exploiting uncertainty, leaving the other active selection criteria rarely considered. Different algorithms are designed to evaluate the uncertainty of unlabeled data. A commonness of them is that they usually evaluate the active selection criterion based on the predicted labels of instances. Thus an effective classifier which can accurately predict the labels for the unlabeled data is crucial for a successful active learning algorithm. Most existing methods decompose the multi-label task into a series of binary classification problems and learn each label independently. However, the prediction values on different labels may not be comparable, and the number of positive labels for each instance is unknown, thus it is still a challenge to decide the threshold for separating positive and negative labels of an instance given the prediction values on each label. To address this challenge, some ad-hoc efforts have been attempted. For example, simply taking the sign of predictions as labels [16], normalizing the predictions on different labels [24], and predicting the number of positive labels via an extra regression model [30].

In this paper, we propose to exploit both uncertainty and diversity in the instance space and label space with an incremental multi-label classification model. First, along with a label ranking algorithm which optimizes the approximated ranking loss to rank positive labels before negative ones, we also employ a dummy label for each instance, and train the model to rank the dummy label between positive and negative labels. Since the dummy label threshold is learned specifically for each instance along with the ranking model, it is expected to provide an accurate separation of positive and negative labels. Based on this model, we then integrate uncertainty with diversity as a new active learning criterion, and select the most valuable instance-label pairs to query. Specifically, in the instance space, we simultaneously evaluate the uncertainty with label cardinality inconsistency and

*This work was partially supported by NSFC (61073097, 61273301) and JiangsuSF (BK2011566)
the diversity with the number of labels not queried; while in
the labels space, the distance from a label to the thresholding
dummy label is employed to evaluate the uncertainty. After
each query, our multi-label model is incrementally updated
based on only the newly added labeled data, avoiding the
retraining from scratch.

The rest of this paper is organized as follows. Section II reviews related work. In Section III, the multi-label
classification model as well as the active selection strategy
are presented. Section IV presents the experiments, followed by
the conclusion in Section V.

II. RELATED WORK

The multi-label classification model used in this paper
is extended from an online multi-class ranking model [29].
The method in [29] is designed for single label setting, and
thus can not decide how many labels should be selected as
positive from the ranked label list. In this paper, we employ
a dummy label to automatically learn a threshold for sepa-
rating positive and negative labels, which is previously used
in [12] for multi-instance multi-label learning. Fürnkranz et
al. propose a calibrated label ranking approach for multi-
label classification in [11], where a mechanism similar to
our dummy label is used for separating positive and negative
labels. However, to the best of our knowledge, label ranking
with threshold learning has not been exploited in multi-label
active learning.

Though the main efforts on active learning focused on the
single-label setting, multi-label active learning has attracted
more and more attention due to its importance [14]. Most
existing approaches are based on the binary relevance model,
which decomposes the multi-label task into a series of binary
classification problems. For example, [17] trains a SVM for
each label, and selects the instance which maximizes the
reduction of expected loss to query its labels. [30] also trains
a SVM for each label and aims to maximize the expected
loss reduction, but uses an extra regression model to predict
the number of labels for each instance. Besides, based
on independently trained binary classifiers, the minimal,
average and weighted summarization over the uncertainties
measured on each label are taken as active selection criterion
in [3], [25] and [9], respectively. In [16], label cardinality
inconsistency is combined with the separation margin via a
tradeoff parameter as a new criterion for active selection,
where the labels of instances are directly determined by the
sign of prediction values of individual binary SVMs.

While most multi-label active learning methods try to
query the whole label vector of one instance at a time,
Qi et al. propose a two dimensional approach to iteratively
query whether a label is positive on a specific instance [20].
They derive a multi-labeled Bayesian error bound, and select
the instance-label pairs which maximizes the expected error
reduction to query.

There are also some other multi-label active learning ap-
proaches proposed for different settings, such as the method
for multi-view data [33] and the batch model form algorithm
proposed in [4].

III. THE ALGORITHM

We first introduce an incremental multi-label model by
extending the label ranking method proposed in [29] from
single-label to the multi-label in subsection A, and then
propose a new active learning strategy in subsection B to
iteratively query whether a label is positive on an instance.

A. Multi-Label Ranking with Dummy Label

In [29], an online algorithm was proposed to optimize
approximated ranking loss on single-label data, aiming to
rank the positive label before negative ones for each instance.
In single-label learning, we can easily determine the positive
label for an instance by selecting the one with maximum
prediction value. However, in multi-label learning where one
instance can have more than one label, we do not know how
many labels should be selected as positive from the label list
ranked based on the predictions values. To overcome this
difficulty, inspired by [11], [12], we introduce a dummy label
to each instance, and train the model to rank the dummy
label between positive and negative labels for thresholding.

We denote by \( \{(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\} \) the
training data with \( n \) examples, where each instance \( x_i \) is
a \( d \)-dimensional feature vector. Assuming there are in all
\( K \) possible labels, the label vector of \( x_i \) is denoted by
\( y_i = [y_{i1}, y_{i2}, \ldots, y_{iK}]^\top \), where \( y_{ik} = 1 \) if instance \( x_i \) has
the \( k \)-th label, otherwise \( y_{ik} = -1 \). The classification
model first maps both the feature space and label space into a
shared subspace, and then measures the relevance of a label
to an instance by their similarity in this subspace. Formally,
given an instance \( x \), we define the classification model on
label \( k \) as

\[
f_k(x) = w_k^\top W_0 x,
\]

where \( W_0 \) is a \( m \times d \) matrix which maps the original
feature vectors to the subspace, and \( w_k \) is a \( m \)-dimensional
vector that maps the \( k \)-th label into the subspace. Here \( d \)
and \( m \) are the dimensionalities of the feature space and the
subspace, respectively. Then the ranking error [28], [29] for
the instance \( x \) and one of its positive labels \( y \) is defined as:

\[
\epsilon(x, y) = \sum_{i=1}^{R(x, y)} \frac{1}{i},
\]

where \( R(x, y) \) counts how many negative labels are ranked
before label \( y \). Obviously, the lower \( y \) is ranked, the larger
\( R(x, y) \) is, and accordingly the ranking error \( \epsilon(x, y) \) be-
comes larger. By minimizing the ranking error on all positive
labels of all training instances, the model is expected to rank
positive labels before negative ones for an unseen instance.
Let $\bar{Y}$ denote the set of all negative labels of $x$, as in [29], the ranking error $\epsilon(x, y)$ can be approximated by a convex surrogate loss

$$
\epsilon(x, y) \approx \sum_{\bar{y} \in \bar{Y}} \epsilon(x, \bar{y}) \frac{1 + f_\bar{y}(x) - f_y(x)}{R(x, y)},
$$

(2)

which is further spread into every negative label $\bar{y}$ in $\bar{Y}$:

$$
\mathcal{L}(x, y, \bar{y}) = \epsilon(x, y) |1 + f_\bar{y}(x) - f_y(x)|_+.
$$

(3)

This surrogate loss is minimized with stochastic gradient descent (SGD) [21]. At each iteration of SGD, we first randomly select an instance $x$ and one of its positive labels $y$, and then iteratively sample negative labels $\bar{y}$ from $\bar{Y}$ until getting a violated label (i.e., $\bar{y}$ is ranked before $y$). Based on the number of samples needed to reach this violated label, $R(x, y)$ can be estimated as $R(x, y) = \sum_{\bar{y} \in \bar{Y}} I[f_\bar{y}(x) > f_y(x) - 1]$ according to [29]. After that, gradient descent is performed to minimize $\mathcal{L}(x, y, \bar{y})$. At last, the updated parameters, i.e., $w_u, w_d$ and each column of $W_0$ are then normalized to have an $\ell^2$-norm smaller than a specific constant $C$.

We then explain how the positive and negative labels are separated with the dummy label. We assume that every instances $x_i$ has a dummy label, denoted by $y_{i0}$. At each iteration of SGD, the sampled label $y$ can be a positive label of $x$ or the dummy label $y_0$. As in [12], we construct the negative label set $\bar{Y}$ depending on the type of $y$. If $y$ is the dummy label, then $\bar{Y}$ consists of all negative labels of instance $x$, otherwise, $\bar{Y}$ contains all negative labels as well as the dummy label. After the training with such a mechanism, the dummy label will be ranked before all negative labels while positive labels will be ranked before both the dummy label and negative labels. So the dummy label provides a nature threshold to separate positive and negative labels. Given an unseen test instance, with the prediction values on each label, we can easily select the labels with larger predictions than that of the dummy label as positive labels.

### B. Active Selection

In this subsection, we present the strategy of active selection based on the previously introduce multi-label classification model. We follow the setting in [20] to iteratively query if a label is positive on an instance. We denote by $\mathcal{D}$ the data set, and divide it into two parts: the labeled data $\mathcal{D}_l$ with $N_l$ instances and unlabeled data $\mathcal{D}_u$ with $N_u$ instances. Since we select instance-label pairs for querying, there will be some instances partially labeled. For convenience, such partially labeled instances are also taken as unlabeled data. In other words, $\mathcal{D}_l$ contains only the fully labeled instances, while $\mathcal{D}_u$ contains both the unlabeled and partially labeled instances. We also introduce $U(x)$ to denote the set of labels that have not been queried for instance $x$. So the task in each iteration is to select an instance $x^*$ from $\mathcal{D}_u$ and then select one label $y^*$ from $U(x^*)$ to query.

Uncertainty is an effective and mostly used criterion for active learning [16], [25], [26]. In this paper, we will try to exploit the uncertainty in both the instance space and label space, and combine it with diversity for active selection. In single-label learning, a classic implementation of uncertainty sampling is to query the instance closest to the decision boundary. This strategy can be easily extend to multi-label learning. For example, the average or minimal margin over all labels can be taken to measure the uncertainty of an instance [25], [9]. Recently, a simple uncertainty criterion, named as label cardinality inconsistency was proposed in [16]. It measures the inconsistency between the number of predicted positive labels of an instance and the average label cardinality on the fully labeled data, and can be formally defined as:

$$
LCI(x_i) = \left(\sum_{k=1}^{K} I[\hat{y}_{ik} > 0] - \frac{1}{N_l} \sum_{j=1}^{N_l} \sum_{k=1}^{K} I[y_{jk} > 0]\right)^2,
$$

where $\hat{y}_{ik} (k = 1 \ldots K)$ is the predicted labels for instance $x_i$, and $I[\cdot]$ is the indicator function. In [16], LCI was combined with margin with a tradeoff parameter to measure the uncertainty, and has been shown to be effective. However, with the increase of labeled data, the difference of LCI over different instances may get smaller and smaller, and thus it becomes more difficult to identify the most uncertain instance based on the LCI criterion. In this paper, we extend LCI to incorporate with diversity, and define a new criterion:

$$
C_1(x_i) = \left[\frac{\sum_{k=1}^{K} I[\hat{y}_{ik} > 0] - \frac{1}{N_l} \sum_{j=1}^{N_l} \sum_{k=1}^{K} I[y_{jk} > 0]}{\text{max}\{\xi, K - \text{card}(U(x_i))\}}\right],
$$

where $\text{card}(\cdot)$ counts the number of elements in a set, and thus $K - \text{card}(U(x_i))$ is the number of queried labels of $x_i$. $\xi \in (0, 1)$ is a constant to avoid the zero divisor. The motivation here is that instances with less queried labels may contain more unknown information and should be preferentially queried. As we want to query the instance with maximum uncertainty and diversity, the instance $x_*$ achieves maximum $C_1$ value is selected. Note in our experiments, we set $\xi = 0.5$ and randomly select one if multiple instances achieved the maximal $C_1$ value.

After selecting the instance $x^*$, we need to decide which label to query. Since the dummy label stands for the separating threshold of the positive and negative labels, the uncertainty of a label $y$ can be naturally measured by the distance from it to the dummy label, which is formally defined as:

$$
C_2(x^*, y) = |f_y(x^*) - f_{y_0}(x^*)|
$$

(5)

The pseudo code of the proposed algorithm, termed AUDI (Active learning based on Uncertainty and Diversity for
Algorithm 1: The AUDI algorithm

1: Input:
2: data set \(D\)
3: Initialize:
4: Divide \(D\) to \(D_1\) and \(D_u\)
5: train a model \(f\) on \(D_1\)
6: Repeat:
7: get predictions and labels for instances in \(D_u\) with \(f\)
8: compute \(C_1(x)\) for all \(x \in D_u\) as Eq. 4
9: select the instance \(x^*\) with maximum \(C_1\) value
10: compute \(C_2(x^*, y)\) for all \(y \in U(x^*)\) as Eq. 5
11: select the label \(y^*\) with minimal \(C_2\) value
12: query if label \(y^*\) is a positive one for instance \(x^*\)
13: remove \(y^*\) from \(U(x^*)\)
14: move \(x^*\) from \(D_u\) to \(D_1\) if \(|U(x^*)| = 0\)
15: update the model \(f\) with \(x^*\) and its labels
16: until the number of queries reached

Incremental multi-label learning, is presented in Algorithm 1. First, a subset of the data set \(D\) is randomly sampled to initialize the labeled data \(D_1\). Note in the experiments, to be fair, for all algorithms, \(D_1\) is initialized with the same set of fully labeled instances rather than instance-label pairs. After the initialization, we apply the algorithm introduced in the previous subsection on \(D_1\) to train a multi-label classification model. Then at each iteration of active learning, we select an instance-label pair \((x^*, y^*)\) according to:

\[
x^* = \text{argmax}_{x \in D_u} C_1(x)\]

\[
y^* = \text{argmin}_{y \in U(x^*)} C_2(x^*, y)
\]

and query if \(y^*\) is a positive label of \(x^*\). After that, \(y^*\) is removed from \(U(x^*)\) and \(x^*\) is moved from \(D_u\) to \(D_1\) if it is fully labeled. Since the multi-label classification model can be trained incrementally, we do not need to retrain the model from scratch, but only update \(f\) based on the newly labeled data. Note that adding a label to an instance may affect the rank of all labels on it, so \(f\) is updated on \(x^*\) and all of its labels, instead of only on the pair \((x^*, y^*)\). This active querying and model updating process is repeated until enough data labeled.

IV. EXPERIMENTS

A. Settings

In the experiments, the following six multi-label active learning approaches are compared:

- Random: the baseline which randomly selects instance-label pairs.
- 2DAL: the two dimensional active learning method proposed in [20], which selects instance-label pairs with the expected classification error reduction criterion.
- MML: the mean max loss strategy proposed in [17].
- MMC: the method proposed in [30], which uses the maximum loss reduction with maximal confidence as selection criterion.
- Adaptive: the adaptive method proposed in [16], which combines the max-margin prediction uncertainty and the label cardinality inconsistency as the criterion for active selection.
- AUDI: the method proposed in this paper.

Experiments are performed on 9 data sets, most of which can be download at the web page of MULAN project\(^1\). Detailed characteristics of these data sets are summarized in Table I, including number of instances, number of labels, feature space dimensionality and label cardinality (LC), where LC counts the average number of labels per instance.

<table>
<thead>
<tr>
<th>Data</th>
<th># instance</th>
<th># label</th>
<th># feature</th>
<th>cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Corel5K [7]</td>
<td>5000</td>
<td>374</td>
<td>499</td>
<td>3.52</td>
</tr>
<tr>
<td>Emotions [27]</td>
<td>593</td>
<td>6</td>
<td>72</td>
<td>1.87</td>
</tr>
<tr>
<td>Genebase [5]</td>
<td>662</td>
<td>27</td>
<td>1185</td>
<td>1.25</td>
</tr>
<tr>
<td>Image [31]</td>
<td>2000</td>
<td>5</td>
<td>294</td>
<td>1.24</td>
</tr>
<tr>
<td>Medical [19]</td>
<td>978</td>
<td>45</td>
<td>1449</td>
<td>1.25</td>
</tr>
<tr>
<td>Reuters [22]</td>
<td>2000</td>
<td>7</td>
<td>243</td>
<td>1.15</td>
</tr>
<tr>
<td>Scene [1]</td>
<td>2407</td>
<td>6</td>
<td>294</td>
<td>1.07</td>
</tr>
<tr>
<td>Yeast [8]</td>
<td>2417</td>
<td>14</td>
<td>103</td>
<td>4.24</td>
</tr>
</tbody>
</table>

\(^1\)http://mulan.sourceforge.net/datasets.html
regression model is also implemented with LIBLINEAR. For AUDI, we use constant step size for SGD and set $m = 100$ as default. The other parameters are selected via 5-folds cross validation on the initial labeled data. For the other approaches, parameters are determined in the same way if no values suggested in their literatures.

### B. Comparison Results

We plot the curves of micro-F1 with the number of queried instance-label pairs increasing in Fig. 1. Note that three approaches: Random, 2DAL and AUDI, which query instance-label pairs, are plotted in solid line, while the other three methods which query the whole label vector of instances are plotted in dashed line.

Generally speaking, methods querying instance-label pairs are more effective than those query instances, which is consistent with the results in [20]. This phenomenon is probably because of that multiple labels may be correlated, and thus redundancy of information may exist among the multiple labels of the same instance. This also explains why random selection can be better than some active approaches on some data. Among the methods query instances only, Adaptive and MMC tend to be more effective than MML.

When comparing the proposed AUDI with other methods, no matter querying instance-label pairs or instances only, our method achieves the best performance in most cases. Especially on data sets with more labels, such as Corel5K and Enron data sets, the superiority of AUDI gets more significant. The only special case is on yeast data set, where AUDI achieves comparable performance with the best baseline.

### V. Conclusion

An effective classification model along with a good selection criterion are the key factors of a successful active learning approach. In this paper, we propose a new multi-label active learning approach to iteratively query whether a label is positive on an instance. First, we present an incremental model for effective multi-label classification by incorporating label ranking with a threshold mechanism. Based on the model, we then propose to combine uncertainty and diversity as the criterion to select the most valuable instance-label pairs to query. The experimental results demonstrate the superiority of our algorithm. In the future, we will try to study other active query strategies based on the label ranking model.
REFERENCES


