Cost-Effective Active Learning for Hierarchical Multi-Label Classification

Yi-Fan Yan and **Sheng-Jun Huang**[*]
College of Computer Science & Technology, Nanjing University of Aeronautics & Astronautics
Collaborative Innovation Center of Novel Software Technology and Industrialization
{yanyifan7,huangsj}@nuaa.edu.cn

Abstract

Active learning reduces the labeling cost by actively querying labels for the most valuable data. It is particularly important for multi-label learning, where the annotation cost is rather high because each instance may have multiple labels simultaneously. In many multi-label tasks, the labels are organized into hierarchies from coarse to fine. The labels at different levels of the hierarchy contribute differently to the model training, and also have diverse annotation costs. In this paper, we propose a multi-label active learning approach to exploit the label hierarchies for cost-effective queries. By incorporating the potential contribution of ancestor and descendant labels, a novel criterion is proposed to estimate the informativeness of each candidate query. Further, a subset selection method is introduced to perform active batch selection by balancing the informativeness and cost of each instance-label pair. Experimental results validate the effectiveness of both the proposed criterion and the selection method.

1 Introduction

Multi-label learning (MLL) is a practical and effective learning framework for objects with complex semantics, where each instance can be simultaneously associated with multiple class labels. While the output space is exponentially larger than that of single-label learning, it usually requires a large labeled dataset to train an effective MLL model. On the other hand, the annotation cost of an multi-label object could be very high when there is a large number of candidate labels. It is thus important to reduce the labeling cost for multi-label learning. Active learning is a primary approach to achieving this goal. It iteratively selects the most important instances to query their labels, and tires to improve the model with less queries.

Active learning for multi-label classification has attracted much research interest in recent years. Most of the previous methods try to extend some selection criteria from single-label to multi-label cases. There are a few other studies try to design novel query types to acquire other supervised information rather than class labels. These methods have achieved decent performances on some tasks, but pay less attention to the label correlations, which is crucial for multi-label learning. In this paper, we study the active learning for multi-label tasks with label correlations, especially in the form of label hierarchical tree.

In many multi-label learning tasks, the labels can be organized into a hierarchical tree structure from coarse to fine. For example, in document categorization, the candidate topics can be organized to a tree structure according to the Open Directory Project (ODP); and the topics of a specific document may cover one or multiple paths of the tree. In medical image analysis, each image is associated with a subset of the labels, which can be organized as part of IRMA hierarchical classification scheme.

The label hierarchies may contain important information, and should be exploited when perform active selection. Firstly, for a specific instance, its relevance to some labels can be directly decided from the others. An instance must have all the ancestor labels of a relevant label, and can never be assigned with any descendant labels of irrelevant labels. Secondly, the labels are typically organized from coarse to fine as the tree goes deeper. So the labels located at deeper layers are expected to contain more detailed information. Lastly, the annotation cost of labels at different levels can differ greatly. For example, it is easy for a labeler to identify whether there is a *dog* in an image, but rather difficult to identify whether the dog is a *poodle*.

In this paper, we propose a novel batch mode active learning approach to exploit the label hierarchies for cost-effective multi-label classification. Specifically, a new criterion is proposed to estimate the informativeness of an instance-label pair, which considers not only the usefulness of the current query, but also the potential contributions of ancestor and descendant labels after the query. Further, labels at different levels of the hierarchical tree are allowed to have different annotation costs. In each iteration, a batch of instance-label pairs is selected by automatically maximizing the informativeness while minimizing the overall annotation cost. The cost-effective selection is achieved with a bi-objective optimization problem. Our empirical study on multiple datasets

[*]This research was supported by JiangsuSF (BK20150754) and NSFC (61503182, 61732006).

and with different cost settings demonstrates the advantage of the proposed approach.

The rest of this paper is organized as follows. In Section 2, we introduce the related work. In Section 3, the propose approach is introduced. Section 4 presents the experiments, followed by the conclusion in Section 5.

2 Related Work

In many multi-label tasks, the labels are organized into a hierarchical tree structure from coarse to fine. The flat classification is the simplest method for hierarchical MLL [Burred and Lerch, 2003]. It ignores the label hierarchy, and makes predictions only for labels at leaf nodes. So traditional multi-label or multi-class approaches can be easily extended to this scenario. There are some other approaches try to exploit the hierarchies locally [Koller and Sahami, 1997] or globally [Blockeel *et al.*, 2002]. Local methods train the model by utilizing the local information, such as the information per node [Eisner *et al.*, 2005], per parent node [Secker *et al.*, 2007] or per tree level [Freitas and de Carvalho, 2007]. Global methods consider the dependencies between different labels and learning a single global model.

Active learning has been successfully applied to many multi-label tasks [Hung and Lin, 2011; Huang *et al.*, 2014a]. Most previous studies focus on designing the selection criterion such that the selected instances can improve the model maximumly. Uncertainty is one of the most effective criteria for active selection [Tong and Koller, 2001]. Some researches exploit the diversity [Brinker, 2003] or density [Nguyen and Smeulders, 2004] of data to estimate the representativeness of instances. Criteria combining different measurements are also studied, and have shown descent performances [Huang *et al.*, 2014b]. Moreover, the query type has been validated to be important for multi-label active learning. Typical query types include querying all the labels of selected instances [Li *et al.*, 2004], querying instance-label pairs [Huang and Zhou, 2013] and querying the relevance ordering of label pairs [Huang *et al.*, 2015].

Recently, a few works applies active learning methods to hierarchical classification. However, these methods focus on the multi-class case, and regard only leaf nodes as labels in the hierarchal tree. Cheng *et al.* [2012] embed the label hierarchy and training data into a latent semantic space, and propose a uncertainty strategy based on the semantic space to query the informative instances. They improve this approach by joining a diversity measure in selection criterion and select a batch size of unlabeled data to query in [Cheng *et al.*, 2014]. Chakraborty *et al.* [2015] propose a batch mode active learning method based on informativeness and diversity, and reduce the batch selection task to a score ranking problem. These methods assume that each instance has only one relevant label, and cannot handle multi-label learning problems. Li *et al.* [2012; 2013] use the uncertainty strategy to query the most informative instance. They pay more attention on minimizing the out-of-domain queries, and propose two strategies to subdivide the unlabeled data based on a top-down active learning framework. This method only explores the information of data while ignoring the different annotation cost of labels.

3 The Proposed Approach

Let $\mathcal{X} = \mathcal{R}^d$ be the input space and $\mathcal{Y} = \{y_1, y_2, \cdots y_L\}$ be the label space with L possible labels. We denote by $\{(\boldsymbol{x}_1, Y_1), (\boldsymbol{x}_2, Y_2), ..., (\boldsymbol{x}_n, Y_n)\}$ the training data that consists of n instances, where each instance $\boldsymbol{x}_i$ is a d-dimensional feature vector, and $Y_i \subseteq \mathcal{Y}$ is the set of labels relevant to $\boldsymbol{x}_i$.

We focus on the batch mode active learning setting, where a small batch of queries are performed at each iteration, and each query is for checking the relevance of an instance-label pair. We propose a new criterion for estimating the informativeness of an instance-label pair with hierarchies in the first subsection, and then present the algorithm for selecting cost-effective queries in the second subsection.

3.1 Informativeness for Hierarchical Labels

It has been well accepted that if the current model is less certain about the prediction on an instance, then the instance is likely to be more informative, and can contribute more for improving the model. Uncertainty is one of the most commonly used criterion to measure the informativeness. Specifically, we estimate the uncertainty of an instance $\boldsymbol{x}_i$ on a label y_j by the reciprocal of the closeness to the decision boundary:

$$U_{ij} = \frac{1}{|f_j(\boldsymbol{x}_i)|}, \tag{1}$$

where f_j is the classification model for label y_j.

Then we extend this definition to incorporate the information embedded in the hierarchical structure. When querying the relevance of an instance-label pair $(\boldsymbol{x}_i, y_j)$, we may get extra supervised information of ancestor or descendant labels. Moreover, the extra information is dependent to the ground-truth relevance of $(\boldsymbol{x}_i, y_j)$. For example, if label y_j is relevant to instance $\boldsymbol{x}_i$, the ancestor nodes of y_j are also relevant to instance $\boldsymbol{x}_i$. On the contrary, if label y_j is irrelevant to instance $\boldsymbol{x}_i$, the descendant nodes of y_j are also irrelevant to instance $\boldsymbol{x}_i$. This characteristic indicates that the informativeness defined in Eq. (1) can not be fully utilized. Thus the information of ancestor and descendant nodes should be considered to contribute the current label nodes, which are denoted by U^{ij}_{anc} and U^{ij}_{des}, respectively. So the enhanced informativeness of each instance-label pair $(\boldsymbol{x}_i, y_j)$ can be discussed for two cases:

$$I_{ij} = \begin{cases} U^{ij}_{anc} + U_{ij}, & \text{if } \boldsymbol{x}_i \text{ is relevant to } y_j; \\ U^{ij}_{des} + U_{ij}, & \text{if } \boldsymbol{x}_i \text{ is irrelevant to } y_j. \end{cases} \tag{2}$$

Unfortunately, before the querying, it is unknown whether $\boldsymbol{x}_i$ is relevant to y_j or not. Instead, we try to evaluate the relevance by exploiting the information from similar examples. Specifically, majority voting on k nearest neighbors is employed to decide the relevance h_{ij} of $(\boldsymbol{x}_i, y_j)$:

$$h_{ij} = sign\Big(\sum_{\boldsymbol{x} \in nei(\boldsymbol{x}_i, y_j, k)} sign\big(g_j(\boldsymbol{x})\big)\Big), \tag{3}$$

where $nei(\boldsymbol{x}_i, y_j, k)$ consists of k nearest neighbors of $\boldsymbol{x_i}$ with regard to label y_j. If $\boldsymbol{x}$ is an labeled instance, then $g_j(\boldsymbol{x})$

is the ground-truth relevance on y_j, otherwise $g_j(\boldsymbol{x}) = f_j(\boldsymbol{x})$. To avoid negative affection from unreliable instances, we filter out half of the unlabeled instances with low confidence when searching for the k nearest neighbors. Note that the predictions of the same instance on different labels may have different confidences, so $nei(\boldsymbol{x}_i, y_j, k)$ varies with different y_j.

With the estimated relevance h_{ij}, the two cases in Eq. (2) can be integrated into an unified formulation:

$$I_{ij} = \mathbb{I}[h_{ij} == 1] * U_{anc}^{ij} + \mathbb{I}[h_{ij} == -1] * U_{des}^{ij} + U_{ij}, \quad (4)$$

where $\mathbb{I}[\cdot]$ is the indicator function which returns 1 if the argument is true and 0 otherwise. From Eq. (4) we can observe that if $h_{ij} == 1$, labels in deeper level are preferred because it brings more label information of its ancestor; otherwise, labels in upper level are preferred because they have more descendant labels to be additionally identified as irrelevant.

We now discuss the implementation of U_{anc}^{ij} and U_{des}^{ij} for estimating the contribution of ancestor and descendant labels. A straightforward method is to simply summarize the uncertainty of all ancestor or descendant labels, leading to the following definitions:

$$U_{anc}^{ij} = \sum_{k \in anc(j)} U_{ik}, \quad (5)$$

$$U_{des}^{ij} = \sum_{k \in des(j)} U_{ik}, \quad (6)$$

where $anc(j)$ and $des(j)$ denote the set of ancestor and descendant labels of y_j, respectively. Note that if a label y_k has been previously annotated for $\boldsymbol{x}_i$, it will not bring any extra information, and thus U_{ik} will be set to 0. Obviously if the ancestor(or descendant) set is empty, the U_{anc}^{ij}(or U_{des}^{ij}) is 0.

The above definition takes all ancestor or descendant labels into consideration, but neglects the risk of information redundancy. As mentioned before, labels at deeper level usually have more detailed information, which may cover most of those embedded in labels at upper level. Considering this, we redefine U_{anc}^{ij} and U_{des}^{ij} as:

$$U_{anc}^{ij} = U_{ij}, \quad (7)$$

$$U_{des}^{ij} = \sum_{k \in leaf(j)} U_{ik}, \quad (8)$$

where $leaf(j)$ is the set of leaf labels of the subtree rooted at y_j. In other words, the redundant information contained in the internal nodes is filtered out.

3.2 Cost-effective Selection

As discussed before, the annotation cost of labels at different levels can differ greatly. Thus in addition to exploiting the informativeness embedded in label hierarchies, it is also important to consider the annotation cost to achieve cost-effective querying. In this subsection, we introduce a subset selection method, which balances the conflict of informativeness and cost, and selects a batch of instance-label pairs with most information and least cost.

A general subset selection problem aims to select a subset S from a total set V of n elements so that the objective $\mathcal{J}$ can be optimized with the constraint $|S| \le b$, where $|\cdot|$ denotes the size of a set. b is the maximum number of selected elements. This problem can be formalized as

$$\arg\min_{S \subseteq V} \mathcal{J}(S) \qquad s.t. \quad |S| \le b. \quad (9)$$

For convenience of presentation, a binary vector $s \in \{0,1\}^n$ is introduced to indicate the subsets membership, where $s_i = 1$ if the i-th element in V is selected, and $s_i = 0$ otherwise. Follow the method in [Qian *et al.*, 2015], the subset selection problem in Eq. (9) can be reformulated as a bi-objective minimization problem:

$$\arg\min_{s \in \{0,1\}^n} (\mathcal{J}_1(s), \mathcal{J}_2(s)), \quad (10)$$

$$\mathcal{J}_1(s) = \begin{cases} +\infty, & s = \{0\}^n \text{ or } |s| \ge 2b \\ \mathcal{J}(s), & \text{otherwise} \end{cases},$$

$$\mathcal{J}_2(s) = |s|.$$

The original target of the method in [Qian *et al.*, 2015] is to achieve sparse selection while minimizing the objective function J. Here, we extend it to perform active selection, which maximizes the informativeness of selected instance-label pairs, and at the same time minimizes the annotation cost. Then the two objectives can be defined as the negative informativeness and annotation cost of the selected instance-label pairs, respectively. We denote by C_{ij} the cost of annotating the relevance of y_j on instance $\boldsymbol{x}_i$, and have the following bi-objective optimization problem.

$$\arg\min_{s \in \{0,1\}^n} (\mathcal{J}_1(s), \mathcal{J}_2(s)), \quad (11)$$

$$\mathcal{J}_1(s) = \begin{cases} +\infty, & s = \{0\}^n \text{ or } \mathcal{J}_2(s) \ge 2b \\ -\sum_{ij} s(i,j) \cdot I_{ij}, & \text{otherwise} \end{cases},$$

$$\mathcal{J}_2(s) = \sum_{ij} s(i,j) \cdot C_{ij}.$$

Here s is a n-dimensional binary vector, n is the number of unlabeled instance-label pairs, and $s(i,j)$ returns the element of s corresponding to the instance-label pair $(\boldsymbol{x}_i, y_j)$. b is the cost budget of the batch.

We then employ the bi-objective optimization method POSS in [Qian *et al.*, 2015] to solve Eq. (11). The POSS method maintains a set of candidate solutions for s. In each iteration, it generates a new solution s' from the candidate solution set by random flipping, then compares it with all the other candidate solutions. s' is then added into the candidate solution set if it is both better in two objectives. This process repeats until the maximum number of iterations is reached. Then the optimal solution in the remaining solution set is selected. By solving Eq. (11), a cost-effective batch of instance-label pairs are expected to be selected.

Finally, we summarize pseudo code of the proposed algorithm HALC (Hierarchical Active Learning with Cost) in Algorithm 1. In each iteration, we first compute the informativeness for each instance-label pair, the POSS method then balances the computed informativeness and pre-defined cost. After that, a batch of instance-label pairs is selected to query

Algorithm 1 The HALC Algorithm

```
**Input:** Labeled Data $D_l$, Unlabeled Data $D_u$
**Initialize:** train one model $f_j$ for each label $y_j$ on $D_l$
**repeat**
    compute the uncertainty for $(\boldsymbol{x}_i, y_j)$ in $D_u$ as Eq. 1
    estimate the relevance for $(\boldsymbol{x}_i, y_j)$ in $D_u$ as Eq. 3
    compute the informativeness of ancestor and descendant labels for $(\boldsymbol{x}_i, y_j)$ as Eqs. 7 and 8
    compute the informativeness of $(\boldsymbol{x}_i, y_j)$ as Eq. 4
    select a batch of instance-label pairs $Q$ by solving Eq. 11
    query the labels of $Q$
    annotate the labels for the ancestor or descendant labels $Q'$
    $D_l = D_l \cup (Q \cup Q'), D_u = D_u \setminus (Q \cup Q')$
    update the models with new labeled data $D_l$
**until** the maximum budget reached
```

the oracle, and we can further obtain the annotation of the ancestor or descendant simultaneously without any cost. The complexity of active selection is $O(n)$, where n is the size of unlabeled set.

4 Experiments

4.1 Settings

To examine the effectiveness of the proposed approach, we perform the experiments on four datasets. The statistical information of these datasets are summarized in Table 1, including number of instances, number of labels, feature space dimensionality and depth of the label hierarchy. It is worth notice that we convert the DAG structure of Yeast-go[Barutcuoglu *et al.*, 2006] and Scop-go[Clare, 2003] to hierarchical tree structure by removing subtrees with multiple parents. We also remove labels associated with too few instances. After removing the labels, the size of label set is large enough for experiments. Meanwhile, the class-imbalance problem still exists and imbalance rate is larger than 100:1.

On each data set, we randomly divide it into two parts with 70% as training set and 30% as test set. In the training set, we randomly sample 5% as initial labeled data and the rest as unlabeled pool for active learning. We randomly partition the data for 10 times and report the average results over the 10 repetitions. At each iteration of active learning, a small batch of instance-label pairs is selected by the active learning approaches from unlabeled pool, and then added into the labeled data.

We use one-vs-all linear SVM as baseline classification model for each label. LibSVM is used to implement the classifier in our experiments [Chang and Lin, 2011]. We evaluate the performances of the compared approaches with Hamming Loss and Micro-F1, among which F1 measure is commonly used in hierarchical classification.

To examine the effectiveness of selection method, we compare the proposed approach with three methods. To the best of our knowledge, there is no existing study can be directly applied to our setting. The following methods are compared in our experiments:

Data	Instance	Label	Feature	Depth
ImageCLEF07D	11006	46	80	3
ImageCLEF07A	11006	51	80	3
Yeast-go	1607	55	5930	4
Scop-go	8282	79	2003	4

Table 1: Statistics on datasets used in the experiments.

- **HALC**: The cost-effective approach proposed in this paper.
- **Top-down**: The method proposed in [Li *et al.*, 2013], which selects instance-label pairs from top to down level of hierarchy based on uncertainty sampling.
- **Uncertainty**: Selects the most uncertain instance-label pairs, we use *Unc* for short.
- **Random**: Randomly selects instance-label pairs, we use *Ran* for short.

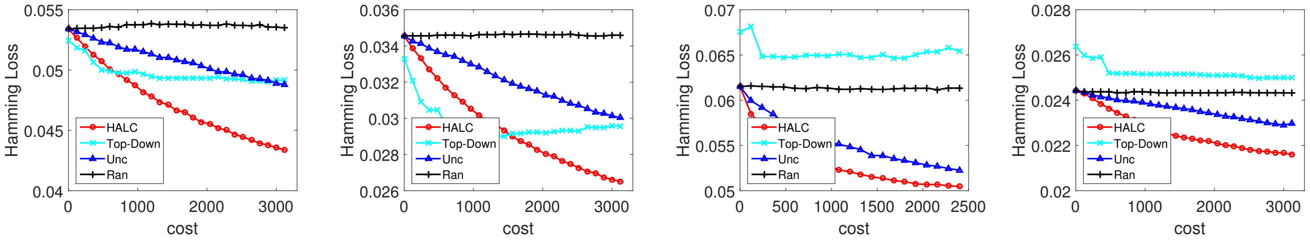
Note that in the top-down method, all the groundtruth of descendant nodes are received simultaneously when querying the selected label. This obviously leads to extra cost. To be fair, the oracle returns only one label of each query in our experiments.

4.2 Performance Comparison

As mentioned before, labels at deeper level cost higher, we manually set the cost of labels at 1:5:10:15 for 4-level hierarchy, and 1:5:10 for 3-level hierarchy. Labels on the same level have the same cost. The performance curves with the cost increasing are plotted in Figure 1 and Figure 2, respectively. Note that the star point of Top-down method is quite different from other methods because of the varies baseline classifiers. The Top-down adopted the hierarchical SVMs as the base classifiers due to the special top-down framework. We can observe that our method achieves the best performance in most cases on all datasets with regard to both measures. The Uncertainty approach can outperform the Random approach, but is worse than our proposed method. Random approach has little improvement on most datasets. The performance of Top-down method is improved quickly at the beginning, but loses its edge as the querying goes on. This is probably because that it simply querys from top to deep labels along the tree path, ignoring the trade-off between informativeness and cost. Further, The improvement of models in deeper level would cost much higher than that in upper nodes, leading to the slowly growth. These results valid that the proposed approach is cost-effective for active selection in hierarchical multi-label learning.

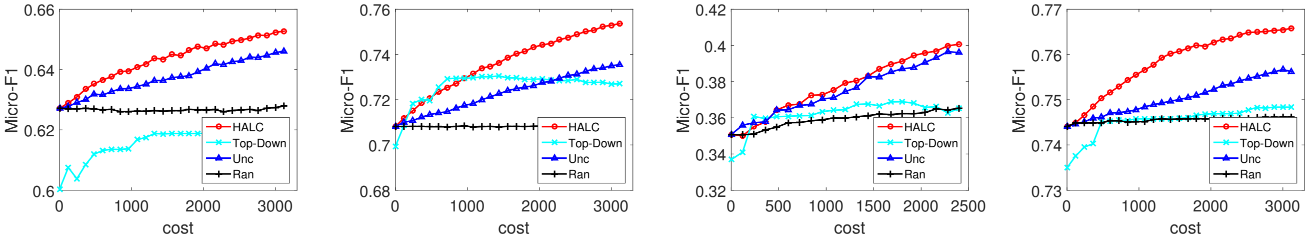
4.3 Study on Relevance Rate

In this subsection, we further compare the relevance rate on different levels. The degenerated version of our algorithm without considering cost , denoted by HALC-i, is an additional comparison method. HALC-i focuses only on hierarchical informativeness. The comparison results are presented in Figure 3. We can observe that our approach, especially



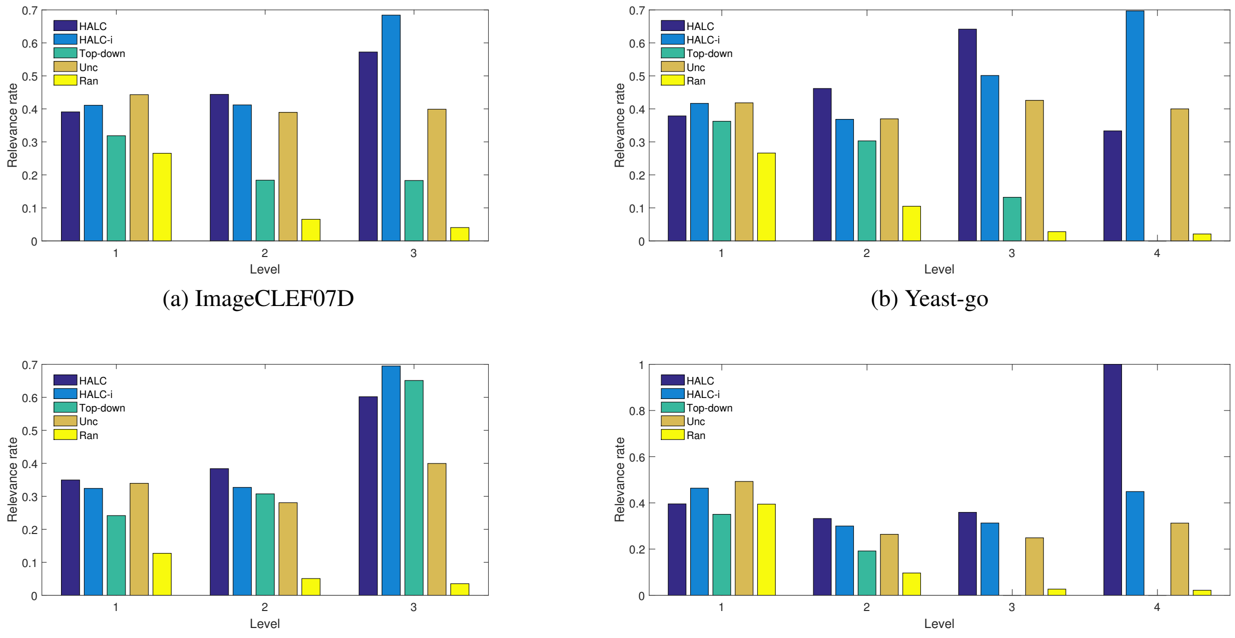
(a) ImageCLEF07D (b) ImageCLEF07A (c) Yeast-go (d) Scop-go

Figure 1: Comparison results on Hamming loss.



(a) ImageCLEF07D (b) ImageCLEF07A (c) Yeast-go (d) Scop-go

Figure 2: Comparison results on Micro-F1.



(a) ImageCLEF07D

(b) Yeast-go

(c) ImageCLEF07A

(d) Scop-go

Figure 3: Comparison results of relevance rate.

HALC-i, queries more relevance instances in deeper level, which illustrates that data with more supervised information has been queried with the same annotation cost, including not only the information itself but the ancestors or descendants. The top-down method queries rarely relevance instances in deep levels in some cases, this may be because it subdivides the unlabeled pool from top to down, and it may mistakenly remove the relevance instances. The relevance rate of Uncer-

cost	Algorithms	ImageCLEF07D	ImageCLEF07A	Yeast-go	Scop-go
1 : 5 : 10 : 15	HALC	**4.72 ± 0.031**	**2.94 ± 0.030**	**5.30 ± 0.047**	**2.26 ± 0.011**
	Top-down	4.96 ± 0.016	3.12 ± 0.020	6.46 ± 0.038	2.41 ± 0.014
	Uncertainty	5.08 ± 0.028	3.20 ± 0.020	5.54 ± 0.040	2.36 ± 0.011
	Random	5.37 ± 0.009	3.46 ± 0.007	6.13 ± 0.014	2.43 ± 0.005
1 : 3 : 6 : 9	HALC	**4.72 ± 0.028**	**3.07 ± 0.024**	**5.41 ± 0.066**	**2.30 ± 0.007**
	Top-down	4.97 ± 0.019	3.13 ± 0.026	6.47 ± 0.046	2.41 ± 0.015
	Uncertainty	5.04 ± 0.021	3.19 ± 0.022	5.55 ± 0.042	2.35 ± 0.011
	Random	5.36 ± 0.009	3.46 ± 0.008	6.13 ± 0.015	2.43 ± 0.005
1 : 2 : 3 : 4	HALC	**4.92 ± 0.026**	**3.08 ± 0.032**	**5.37 ± 0.046**	**2.32 ± 0.009**
	Top-down	4.98 ± 0.021	3.13 ± 0.025	6.47 ± 0.050	2.42 ± 0.019
	Uncertainty	5.02 ± 0.021	3.10 ± 0.019	5.45 ± 0.051	2.34 ± 0.009
	Random	5.36 ± 0.011	3.46 ± 0.010	6.13 ± 0.022	2.43 ± 0.006

Table 2: Average of Hamming loss with different cost settings(mean ± std e-02).

cost	Algorithms	ImageCLEF07D	ImageCLEF07A	Yeast-go	Scop-go
1 : 5 : 10 : 15	HALC	**0.643 ± 0.002**	**0.737 ± 0.002**	**0.378 ± 0.005**	**0.759 ± 0.001**
	Top-down	0.613 ± 0.001	0.713 ± 0.002	0.370 ± 0.003	0.748 ± 0.001
	Uncertainty	0.638 ± 0.001	0.723 ± 0.001	0.375 ± 0.005	0.751 ± 0.001
	Random	0.627 ± 0.001	0.708 ± 0.000	0.361 ± 0.003	0.746 ± 0.001
1 : 3 : 6 : 9	HALC	**0.647 ± 0.001**	**0.731 ± 0.002**	**0.379 ± 0.006**	**0.756 ± 0.001**
	Top-down	0.612 ± 0.001	0.711 ± 0.002	0.370 ± 0.004	0.749 ± 0.001
	Uncertainty	0.642 ± 0.001	0.724 ± 0.001	0.374 ± 0.006	0.752 ± 0.001
	Random	0.627 ± 0.001	0.708 ± 0.001	0.360 ± 0.003	0.745 ± 0.001
1 : 2 : 3 : 4	HALC	**0.643 ± 0.002**	**0.726 ± 0.002**	**0.385 ± 0.003**	**0.754 ± 0.001**
	Top-down	0.612 ± 0.001	0.710 ± 0.002	0.371 ± 0.003	0.748 ± 0.001
	Uncertainty	0.640 ± 0.001	0.725 ± 0.001	0.381 ± 0.006	0.753 ± 0.001
	Random	0.627 ± 0.001	0.709 ± 0.001	0.365 ± 0.004	0.746 ± 0.001

Table 3: Average of Micro-F1 with different cost settings(mean ± std).

tainty is similar in different levels.

4.4 Study with Different Cost Settings

To examine the robustness of the proposed method to different cost ratios, we further perform experiments in 3 different cost settings, i.e., 1:5:10:15, 1:3:6:9 and 1:2:3:4. Due to space limitation, we cannot plot all the performance curves for all datasets, instead, we only report the average value of the performance curve in Table 2 and Table 3, respectively. The best result is highlighted in boldface in both tables. It can be observed from the tables that the proposed approach consistently achieves the best performance with various cost settings. The Top-down approach and uncertainty method are comparable in most cases. These results validate that the cost-effective approach can effectively save the annotation cost, and is robust to different cost settings.

5 Conclusion

In this paper, we propose a multi-label active learning approach to exploit the label hierarchies for cost-effective queries. To incorporate the potential contribution of ancestor and descendant labels in the label hierarchy, a novel criterion is proposed to estimate the informativeness of each instance-label pair. To balance the conflict of informativeness and annotation cost of the query, a bi-objective optimization method is used for subset selection. Experiments on multiple datasets and different performance measures validate the effectiveness of both the informativeness criterion and active selection method. Moreover, the proposed approach maintains the best performance in various annotation cost settings. In the future, we plan to design other criteria to exploit the label hierarchy. Also, other label correlations rather than tree structure will be considered.

References

[Barutcuoglu *et al.*, 2006] Zafer Barutcuoglu, Robert E Schapire, and Olga G Troyanskaya. Hierarchical multi-label prediction of gene function. *Bioinformatics*, 22(7):830–836, 2006.

[Blockeel *et al.*, 2002] Hendrik Blockeel, Maurice Bruynooghe, Sašo Džeroski, Jan Ramon, and Jan Struyf. Hierarchical multi-classification. In *Workshop Notes of the KDD'02 Workshop on Multi-Relational Data Mining*, pages 21–35, 2002.

[Brinker, 2003] Klaus Brinker. Incorporating diversity in active learning with support vector machines. In *Proceedings of the 20th International Conference on Machine Learning*, pages 59–66, 2003.

[Burred and Lerch, 2003] Juan José Burred and Alexander Lerch. A hierarchical approach to automatic musical genre classification. In *Proceedings of the 6th International Conference on Digital Audio Effects*, pages 8–11, 2003.

[Chakraborty *et al.*, 2015] Shayok Chakraborty, Vineeth Balasubramanian, Adepu Ravi Sankar, Sethuraman Panchanathan, and Jieping Ye. Batchrank: A novel batch mode active learning framework for hierarchical classification. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 99–108, 2015.

[Chang and Lin, 2011] Chih-Chung Chang and Chih-Jen Lin. Libsvm: a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2(3):27, 2011.

[Cheng *et al.*, 2012] Yu Cheng, Kunpeng Zhang, Yusheng Xie, Ankit Agrawal, and Alok Choudhary. On active learning in hierarchical classification. In *Proceedings of the 21st ACM International Conference on Information and knowledge Management*, pages 2467–2470, 2012.

[Cheng *et al.*, 2014] Yu Cheng, Zhengzhang Chen, Hongliang Fei, Fei Wang, and Alok Choudhary. Batch mode active learning with hierarchical-structured embedded variance. In *Proceedings of the SIAM International Conference on Data Mining*, pages 10–18, 2014.

[Clare, 2003] Amanda Clare. Machine learning and data mining for yeast functional genomics. *Aberystwyth: The University of Wales.(Doctor of Philosophy)*, 2003.

[Eisner *et al.*, 2005] Roman Eisner, Brett Poulin, Duane Szafron, Paul Lu, and Russell Greiner. Improving protein function prediction using the hierarchical structure of the gene ontology. In *Proceedings of the IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology*, pages 1–10, 2005.

[Freitas and de Carvalho, 2007] Alex A Freitas and Andre CPFL de Carvalho. A tutorial on hierarchical classification with applications in bioinformatics. In *D. Taniar (Ed.) Research and Trends in Data Mining Technologies and Applications, Idea Group*, 2007.

[Huang and Zhou, 2013] Sheng-Jun Huang and Zhi-Hua Zhou. Active query driven by uncertainty and diversity for incremental multi-label learning. In *Proceedings of the 13th IEEE International Conference on Data Mining*, pages 1079–1084, 2013.

[Huang *et al.*, 2014a] Sheng-Jun Huang, Wei Gao, and Zhi-Hua Zhou. Fast multi-instance multi-label learning. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence*, pages 1868–1874, 2014.

[Huang *et al.*, 2014b] Sheng-Jun Huang, Rong Jin, and Zhi-Hua Zhou. Active learning by querying informative and representative examples. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(10):1936–1949, 2014.

[Huang *et al.*, 2015] Sheng-Jun Huang, Songcan Chen, and Zhi-Hua Zhou. Multi-label active learning: Query type matters. In *Proceedings of the 24th International Joint Conference on Artificial Intelligence*, pages 946–952, 2015.

[Hung and Lin, 2011] Chen Wei Hung and Hsuan Tien Lin. Multi-label active learning with auxiliary learner. In *Asian Conference on Machine Learning*, pages 315–332, 2011.

[Koller and Sahami, 1997] Daphne Koller and Mehran Sahami. Hierarchically classifying documents using very few words. In *Proceedings of the 14th International Conference on Machine Learning*, pages 170–178, 1997.

[Li *et al.*, 2004] Xuchun Li, Lei Wang, and Eric Sung. Multilabel svm active learning for image classification. In *International Conference on Image Processing*, pages 2207–2210, 2004.

[Li *et al.*, 2012] Xiao Li, Da Kuang, and Charles X Ling. Active learning for hierarchical text classification. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 14–25, 2012.

[Li *et al.*, 2013] Xiao Li, Charles X Ling, and Huaimin Wang. Effective top-down active learning for hierarchical text classification. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 233–244, 2013.

[Nguyen and Smeulders, 2004] Hieu T Nguyen and Arnold Smeulders. Active learning using pre-clustering. In *Proceedings of the 21st International Conference on Machine Learning*, page 79, 2004.

[Qian *et al.*, 2015] Chao Qian, Yang Yu, and Zhi-Hua Zhou. Subset selection by pareto optimization. In *Advances in Neural Information Processing Systems*, pages 1774–1782, 2015.

[Secker *et al.*, 2007] Andrew D Secker, Matthew N Davies, Alex A Freitas, Jon Timmis, Miguel Mendao, and Darren R Flower. An experimental comparison of classification algorithms for hierarchical prediction of protein function. *Prediction of Protein Function Expert Update*, 9(3):17–22, 2007.

[Tong and Koller, 2001] Simon Tong and Daphne Koller. Support vector machine active learning with applications to text classification. *Journal of Machine Learning Research*, 2(Nov):45–66, 2001.