Determining Expert Research Areas with Multi-Instance Learning of Hierarchical Multi-Label Classification Model

Tao Wu, Qifan Wang, Zhiwei Zhang, and **Luo Si**
Computer Science Department, Purdue University
West Lafayette, IN 47907, US
{wu577, wang868, zhan1187, lsi}@purdue.edu

Abstract

Automatically identifying the research areas of academic/industry researchers is an important task for building expertise organizations or search systems. In general, this task can be viewed as text classification that generates a set of research areas given the expertise of a researcher like documents of publications. However, this task is challenging because the evidence of a research area may only exist in a few documents instead of all documents. Moreover, the research areas are often organized in a hierarchy, which limits the effectiveness of existing text categorization methods. This paper proposes a novel approach, Multi-instance Learning of Hierarchical Multi-label Classification Model (MIHML) for the task, which effectively identifies multiple research areas in a hierarchy from individual documents within the profile of a researcher. An Expectation-Maximization (EM) optimization algorithm is designed to learn the model parameters. Extensive experiments have been conducted to demonstrate the superior performance of proposed research with a real world application.

1 Introduction

In many large commercial, academic and government organizations, it is often important to identify experts in specific topics. This leads to extensive research on expert retrieval [Hertzum and Pejtersen, 2000; Balog *et al.*, 2009; 2012; Rybak *et al.*, 2014], where the goal is to uncover associations between experts and topics. Microsoft Academic Search is one example of such an expertise retrieval system. One important task of expert retrieval is expert profiling [Balog and De Rijke, 2007; Berendsen *et al.*, 2013], which answers the question: what topics does a person know about. In academics, expert profiling identifies the research areas of researchers. This is essential for building expertise organizations or search systems, which help build expert profiles or determine knowledgeable people for given topics respectively. In general, based on researchers' expertise like documents of publications and research projects, we can view this task as a multi-label text classification [Elisseeff and Weston, 2001; Zhang and Zhou, 2005] problem that generates a set of research areas. However this task is challenging due to its characteristics. First, the evidence of a research area may only exist in a few instead of all documents of a researcher (multi-instance learning problem). Second, research areas are often organized in a hierarchy, which requires effective methods to deal with the label correlations (hierarchical multi-label problem).

Multi-instance Learning (MIL) [Dietterich *et al.*, 1997] studies the case that each bag is composed of multiple data instances, with the assumption that a bag is labeled positive if at least one of its instances is positive, whereas a negative bag only contains negative instances. In the work of [Andrews *et al.*, 2002], two different large margin methods have been proposed, i.e., mi-SVM for instance level classification and bag level method MI-SVM. In [Chen *et al.*, 2006], bags are embedded into a feature space spanned by all the instances. A more recent work in [Wang *et al.*, 2012] proposes a mixture model approach. Most recently, the work in [Wang *et al.*, 2014] boosts the learning performance by adaptive knowledge transfer.

In Multi-label Classification (MLC), objects are associated with a set of labels. Recent works include RankSVM [Elisseeff and Weston, 2001], ML-kNN [Zhang and Zhou, 2005] and Max-margin multi-label [Hariharan *et al.*, 2010]. The joint multi-instance multi-label problem [Jin *et al.*, 2009; Nguyen *et al.*, 2014] has also attracted a lot of attentions. One specific case of MLC is Hierarchical Multi-Label classification (HML) [Barutcuoglu *et al.*, 2006; Schietgat *et al.*, 2010], in which labels have hierarchical structures. Hierarchical Max-margin Markov (HM^3) Network [Rousu *et al.*, 2006] has been proposed to model the label hierarchy, where it follows the framework Max-Margin Markov network (M^3) [Taskar *et al.*, 2003].

However, none of the existing work addresses the expert profiling task effectively, which requires a combination of MIL and HML. This paper models the expert profiling task as Multi-instance Learning of Hierarchical Multi-label Classification (MIHML) problem. Within this framework, we not only incorporate the correlation between multiple research areas in a hierarchy, but also explore the association between individual documents and research areas. In particular, the document-area pair is modeled using Markov network under a unified probabilistic learning framework. In this model, we need to find the correct research areas for each single document as well as the optimal model parameters. This leads to a mixed integer programming problem that cannot be solved efficiently. We propose a novel optimization method called Expectation and Maximization Hierarchical Maximum-Margin Markov Network (EM-HM^3) algorithm to solve the MIHML problem, which optimizes over the model parameters and instance-label assignments alternatively to obtain an optimal solution.

To evaluate the performance of the proposed EM-HM^3 method on expert profiling, we compare it to multi-label SVM, MIMLSVM [Zhou and Zhang, 2006] and HM^3 with an extensive set of experiments with a real world expertise database system. We demonstrate the advantage of EM-HM^3 with research areas in each level of the research hierarchy, and validate the effectiveness for different sizes of training data. Experimental results show that our approach outperforms several existing MIML and HML methods, in both accuracy and the ability of dealing with small training size.

2 Maximum Margin Hierarchical Multi-label Classification

We first introduce the framework of Max-Margin Markov (M^3) Network [Taskar *et al.*, 2003] and its Hierarchical solution HM^3 [Rousu *et al.*, 2006]. Consider a data domain $\mathcal{X} \times \mathcal{Y}$, where $\mathcal{X}$ is a set of instances, and the label set is $\mathcal{Y} = \mathcal{Y}_1 \times \cdots \times \mathcal{Y}_k$ with $\mathcal{Y}_j = \{+1, -1\}$, $j = 1, \cdots, k$. We call a vector $\mathbf{y} = (y_1, \cdots, y_k) \in \mathcal{Y}$ a multilabel and its component y_j a microlabel. For hierarchical labels, they naturally forms a Markov Network: $\mathcal{G} = (\mathcal{Y}, E)$, where there is an edge $e = (i, j)$ between microlabels y_i, y_j if one microlabel is the parent of the other. Thus the network represents the dependence between microlabels. For each edge $e = (i, j)$, the network potential is $\psi_{ij}(\mathbf{x}, y_i, y_j) = \exp(\mathbf{w}_e^T \boldsymbol{\phi}(\mathbf{x}, \mathbf{y}_e))$, $\mathbf{x} \in \mathcal{X}$, $\mathbf{y}_e = (y_i, y_j)$. The network gives a joint conditional probability distribution:

$$\begin{aligned} P(\mathbf{y}|\mathbf{x}) &= \frac{1}{Z(\mathbf{x},\mathbf{w})}\prod_{e\in E}\exp(\mathbf{w}_e^T\boldsymbol{\phi}_e(\mathbf{x},\mathbf{y}_e)) \\ &= \frac{1}{Z(\mathbf{x},\mathbf{w})}\exp(\mathbf{w}^T\boldsymbol{\phi}(\mathbf{x},\mathbf{y})) \end{aligned} \qquad (1)$$

with normalizing factor $Z(\mathbf{x}, \mathbf{w}) = \sum_{\mathbf{y}} \exp(\mathbf{w}^T \boldsymbol{\phi}(\mathbf{x}, \mathbf{y}))$.

2.1 Max-Margin Learning

Given the training data $\{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^m$, we have the following optimization problem:

$$\begin{aligned} &\operatorname*{argmax}_{\mathbf{w}} \log(\prod_{i=1}^{m} P(\mathbf{y}_i|\mathbf{x}_i;\mathbf{w})) \\ =&\operatorname*{argmax}_{\mathbf{w}} \sum_{i=1}^{m}[\mathbf{w}^T\boldsymbol{\phi}(\mathbf{x}_i,\mathbf{y}_i) - \log Z(\mathbf{x}_i,\mathbf{w})] \end{aligned} \qquad (2)$$

Unfortunately the above problem is difficult to solve directly for a general graph $\mathcal{G}$. An alternative way is to maximize the ratio $P(\mathbf{y}_i|\mathbf{x}_i;\mathbf{w})/P(\mathbf{y}|\mathbf{x}_i;\mathbf{w})$, which is equivalent to maximize the minimum linear margin: $\mathbf{w}^T\boldsymbol{\phi}(\mathbf{x}_i,\mathbf{y}_i) - \mathbf{w}^T\boldsymbol{\phi}(\mathbf{x}_i,\mathbf{y})$, and it leads to the following max-margin optimization problem:

$$\begin{aligned} \min_{\mathbf{w}} \quad & \frac{1}{2}||\mathbf{w}||^2 + C\sum_{i=1}^{m}\xi_i \\ \text{s.t.} \quad & \mathbf{w}^T\Delta\boldsymbol{\phi}(\mathbf{x}_i,\mathbf{y}) \geq \ell(\mathbf{y}_i,\mathbf{y}) - \xi_i, \forall i, \mathbf{y} \end{aligned}$$

where $\Delta\phi(\mathbf{x}_i, \mathbf{y}) = \phi(\mathbf{x}_i, \mathbf{y}_i) - \phi(\mathbf{x}_i, \mathbf{y})$, and $\ell(\mathbf{y}_i, \mathbf{y})$ is the value of loss function for multilabels $\mathbf{y}_i$ and $\mathbf{y}$. The dual form can be written as:

$$\max_{\boldsymbol{\alpha}\geq 0} \boldsymbol{\alpha}^T\ell - \frac{1}{2}\boldsymbol{\alpha}^T\mathbf{K}\boldsymbol{\alpha}, \text{ s.t. } \sum_{\mathbf{y}}\boldsymbol{\alpha}(i,\mathbf{y}) \leq C, \ \forall i$$

where $\mathbf{K} = \Delta\boldsymbol{\Phi}^T\Delta\boldsymbol{\Phi}$ is the joint kernel matrix of the training examples. Marginal dual methods can be applied to reduce the size of dual variables from exponential to polynomial. For an edge $e \in E$, and a restricted labeling $\mathbf{y}_e$, the marginal of $\boldsymbol{\alpha}(i, \mathbf{y})$ is defined as:

$$\boldsymbol{\mu}_e(i,\mathbf{y}_e) = \sum_{\{\mathbf{v}\in\mathcal{Y}\}}[\mathbf{y}_e = \mathbf{v}_e]\boldsymbol{\alpha}(i,\mathbf{v}).$$

We can define a decomposable loss function as $\ell(\mathbf{y}_i, \mathbf{y}) = \sum_{e\in E} \ell_e(i, \mathbf{y}_e)$ and a decomposable feature vector $\boldsymbol{\phi}(\mathbf{x}, \mathbf{y}) = (\boldsymbol{\phi}_e(\mathbf{x}, \mathbf{y}_E))_{e\in E}$, which lead to a decomposable joint kernel $\mathbf{K}(\mathbf{x}, \mathbf{y}; \mathbf{x}', \mathbf{y}') = \sum_{e\in E} \boldsymbol{\phi}_e(\mathbf{x}, \mathbf{y}_e)^T \boldsymbol{\phi}_e(\mathbf{x}', \mathbf{y}'_e) = \sum_{e\in E} \mathbf{K}_e(\mathbf{x}, \mathbf{y}_e; \mathbf{x}', \mathbf{y}'_e)$. Then the original dual problem is equivalent to the following form:

$$\begin{aligned} \max_{\mu\geq 0} \quad & \sum_{e\in E}\boldsymbol{\mu}_e^T\ell_e - \frac{1}{2}\sum_{e\in E}\boldsymbol{\mu}_e^T\mathbf{K}_e\boldsymbol{\mu}_e \\ \text{s.t.} \quad & \sum_{\mathbf{y}_E}\boldsymbol{\mu}_e(i,\mathbf{y}_E) \leq C, \ \forall i, e \in E, \\ & \sum_{y'}\boldsymbol{\mu}_e(i,(y',y)) = \sum_{y'}\boldsymbol{\mu}_{e'}(i,(y,y')), \forall i, y, (e,e') \in E_2 \end{aligned} \qquad (3)$$

where $E_e = \{(e, e') \in E \times E | e = (j', i), e' = (i, j)\}$. The above formula is a polynomial-sized quadratic program, which can be solved efficiently [Rousu *et al.*, 2006].

3 Multi-instance Maximum Margin Hierarchical Multi-label Classification

In this section, we present the novel approach of Multi-instance Learning of Hierarchical Multi-label Classification based on Maximum Margin. For multi-instance learning problems, one can always concatenate the features of multiple instances (individual documents) into a single feature vector (a single vector of an expertise profile), and treat it as a traditional single-instance learning task. However, this procedure fails to utilize the multi-instance structure of this problem. Moreover, in many cases, there exist some labels (e.g., research areas) that are only associated with certain instances (documents) of the bag (the profile). Thus we want to explore the intrinsic relations between multiple instances and hierarchical labels. Given an example bag $\mathbf{x}^{(i)} = (\mathbf{x}_1^{(i)}, \cdots, \mathbf{x}_{R_i}^{(i)})$ (documents of an expert) with labels $\mathbf{y}^i \in \mathcal{Y}$ (labels of research areas), where $\mathbf{x}_1^{(i)}, \cdots, \mathbf{x}_{R_i}^{(i)}$ are the multiple instances in the bag, the ideal case is that we also know the exact instance-label pair $(\mathbf{x}_j^{(i)}, \mathbf{y}_j^{(i)})$ for $j = 1, \cdots, R_i$. In contrast to MIMLBoost [Zhou *et al.*, 2012], which takes $\mathbf{y}_j^{(i)} = \mathbf{y}^{(i)}$, for $j = 1, \cdots, R_i$, we assume that each instance contributes different parts to the bag labels, which is illustrated in Fig 1.

3.1 Problem Formulation

We consider a set of training examples: $\{(\mathbf{x}^{(i)}, \mathbf{y}^{(i)})\}_{i=1}^m$, where each bag $\mathbf{x}^{(i)}$ is a set of R_i instances $\{\mathbf{x}_1^{(i)}, \mathbf{x}_2^{(i)}, \cdots, \mathbf{x}_{R_i}^{(i)}\}$, $\mathbf{x}_j^{(i)} \in \mathcal{X}$, and $\mathbf{y}^{(i)} \in \mathcal{Y}$ is the multilabel of the corresponding training bag. According to formula (1), for each instance $\mathbf{x}_j^{(i)}$ the distribution of its multilabel can be written as:

$$P(\mathbf{y}_j^{(i)}|\mathbf{x}_j^{(i)}) = \frac{1}{Z(\mathbf{x}_j^{(i)}, \mathbf{w})} \exp(\mathbf{w}^T \boldsymbol{\phi}(\mathbf{x}_j^{(i)}, \mathbf{y}_j^{(i)})) \qquad (4)$$

where $Z(\mathbf{x}_j^{(i)}, \mathbf{w}) = \sum_{\mathbf{y}_j^{(i)}} \exp(\mathbf{w}^T \boldsymbol{\phi}(\mathbf{x}_j^{(i)}, \mathbf{y}_j^{(i)}))$ is the normalizing factor. The joint probability of all the training examples is represented as: $\prod_{i=1}^m \prod_{j=1}^{R_i} P(\mathbf{y}_j^{(i)}|\mathbf{x}_j^{(i)}; \mathbf{w})$. In the joint probability, $\mathbf{w}$ is the model parameter which needs to optimize. $\mathbf{y}_j^{(i)}$ $(1 \le i \le m,\ 1 \le j \le R_i)$ are instance labels, which are also unknown. In order to maximize the joint probability, we have to optimize over both $\mathbf{w}$ and $\mathbf{y}_j^{(i)}$:

$$\begin{aligned} &\operatorname*{argmax}_{\mathbf{w}, \mathbf{y}_j^{(i)}} \log(\prod_{i=1}^m \prod_{j=1}^{R_i} P(\mathbf{y}_j^{(i)}|\mathbf{x}_j^{(i)}; \mathbf{w})) \\ =& \operatorname*{argmax}_{\mathbf{w}, \mathbf{y}_j^{(i)}} \sum_{i=1}^m \sum_{j=1}^{R_i} [\mathbf{w}^T \boldsymbol{\phi}(\mathbf{x}_j^{(i)}, \mathbf{y}_j^{(i)}) - \log Z(\mathbf{x}_j^{(i)}, \mathbf{w})] \end{aligned} \qquad (5)$$

Note that the instance labels above are actually constrained as they are highly related to the bag labels. In the task of expert profiling, we assume that a positive research area of a profile implies that at least one of the documents in this profile is positive, while a negative research area of a profile means that all its documents are negative. Based on this assumption, we can have the following constraints on instance labels:

$$(\mathbf{y}_1^{(i)}, \cdots, \mathbf{y}_{R_i}^{(i)}) \in S_i,\ \forall 1 \le i \le m \qquad (6)$$

where $S_i = \{(\mathbf{y}_1, \cdots, \mathbf{y}_{R_i}) | \mathbf{y}^{(i)} = \bigvee_{j=1}^{R_i} \mathbf{y}_j\}$ and the operator $\bigvee$ denotes the boolean union operation.

The optimization problem in Eqns. (5) and (6) leads to a mixed integer programming problem, with a feasible domain size: $\prod_{i=1}^m |S_i|$, which grows exponentially with the training data. On the other hand, we notice that for any fixed instance labels, the optimization problem can be solved efficiently by max-margin method. Therefore, we can solve the optimization problem iteratively.

3.2 Optimization

Here we apply Expectation-Maximization (EM) algorithm to get an approximate solution of the above optimization problem. For the sake of convenience, we denote the instance labels as $\mathbf{Z} = \mathbf{Z}_1 \times \cdots \times \mathbf{Z}_m$, where $\mathbf{Z}_i = (\mathbf{y}_1^{(i)}, \cdots, \mathbf{y}_{R_i}^{(i)})$, $\forall 1 \le i \le m$, s.t. $\mathbf{Z} \in S$, where $S = S_1 \times \cdots \times S_m$, and we denote the bag labels as $\mathbf{Y} = (\mathbf{y}^{(1)}, \cdots, \mathbf{y}^{(m)})$. Then we can write the joint probability as:

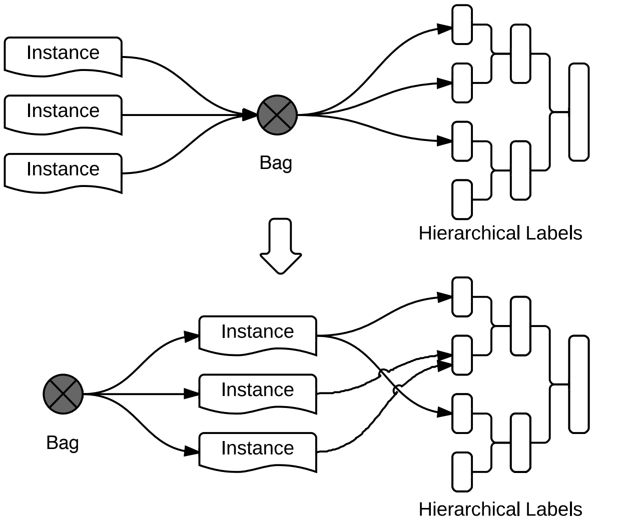

Figure 1: A comparison of HML and the proposed model. The top figure shows the learning model used in HML, where the exact instance-label pairs are unknown. The bottom figure demonstrates the proposed model, which explores the hidden instance-label information based on the bag labels.

$$P(\mathbf{Y}, \mathbf{Z}|\mathbf{w}) = \prod_{i=1}^m \prod_{j=1}^{R_i} \frac{1}{Z(\mathbf{x}_j^{(i)}, \mathbf{w})} \exp(\mathbf{w}^T \boldsymbol{\phi}(\mathbf{x}_j^{(i)}, \mathbf{y}_j^{(i)})).$$

During the M-step in the EM algorithm, we update the $\mathbf{w}^{\text{old}}$ with $\mathbf{w}^{\text{new}}$ from:

$$\mathbf{w}^{\text{new}} = \operatorname*{argmax}_{\mathbf{w}} \sum_{\mathbf{Z} \in S} P(\mathbf{Z}|\mathbf{Y}, \mathbf{w}^{\text{old}}) \log P(\mathbf{Y}, \mathbf{Z}|\mathbf{w}). \qquad (7)$$

The above problem (7) requires us to sum over all feasible $\mathbf{Z}$, which is computational intractable as the size of feasible $\mathbf{Z}$ is exponential with the training size m. It is also difficult to obtain the exact values of $P(\mathbf{Z}|\mathbf{Y}, \mathbf{w})$ and $P(\mathbf{Y}, \mathbf{Z}|\mathbf{w})$ since computing the normalization factors $Z(\mathbf{x}_j^{(i)}, \mathbf{w}) = \sum_{\mathbf{y}_j^{(i)}} \exp(\mathbf{w}^T \boldsymbol{\phi}(\mathbf{x}_j^{(i)}, \mathbf{y}_j^{(i)}))$ is non-trivial.

The general EM method indicates that the parameter $\mathbf{w}^{\text{new}}$ should be calculated based on all feasible values of instance labels $\mathbf{Z}$. However, for less likely instance labels, which means the value of $P(\mathbf{Z}|\mathbf{Y}, \mathbf{w}^{\text{old}})$ is relatively small, it is unlikely to make a big difference for selecting the $\mathbf{w}^{\text{new}}$. Moreover, the model essentially works by focusing on the most relevant instance labels $\mathbf{Z}$, which has the highest posterior probability $P(\mathbf{Z}|\mathbf{Y}, \mathbf{w}^{\text{old}})$. Therefore, a revision of the general EM for our problem is:

1. Choose an initial parameter $\mathbf{w}^{\text{old}}$.
2. **E-Step** Calculate $\mathbf{Z}^* = \operatorname{argmax}_{\mathbf{Z} \in S} P(\mathbf{Z}|\mathbf{Y}, \mathbf{w}^{\text{old}})$.
3. **M-Step** Evaluate $\mathbf{w}^{\text{new}}$ given by $\mathbf{w}^{\text{new}} = \operatorname{argmax}_{\mathbf{w}} \log P(\mathbf{Y}, \mathbf{Z}^*|\mathbf{w})$.

4. Check whether it satisfies the stop criterion. If not, replace the parameter value by $\mathbf{w}^{\text{old}} \leftarrow \mathbf{w}^{\text{new}}$ and return to step 2.

For **M-Step**, max-margin method is used to transform the optimization problem to:

$$
\begin{aligned}
\min_{\mathbf{w}} \quad & \frac{1}{2}||\mathbf{w}||^2 + C\sum_{i=1}^{m}\sum_{j=1}^{R_i}\xi_j^{(i)} \\
\text{s.t.} \quad & \mathbf{w}^T\Delta\boldsymbol{\phi}(\mathbf{x}_j^{(i)},\mathbf{y}) \geq \ell(\mathbf{y}_j^{(i)},\mathbf{y}) - \xi_j^{(i)}, \forall i, j, \mathbf{y}
\end{aligned}
\tag{8}
$$

Then the marginal dual method (3) can be applied to solve the above problem (i.e., by solving the dual variable $\boldsymbol{\mu}$).

For **E-Step**, we search among the domain S and find the optimal instance labels $\mathbf{Z}^*$ with highest posterior probability $P(\mathbf{Z}|\mathbf{Y},\mathbf{w}^{\text{old}})$. This is also transformed into the max-margin framework, where we set the instance labels to minimize $C\sum_{i=1}^{m}\sum_{j=1}^{R_i}\xi_j^{(i)}$. This is done by labeling the instances based on the parameter $\mathbf{w}^{\text{old}}$. Thus the objective of (8) decreases for the fixed $\mathbf{w}^{\text{old}}$ we obtained from **M-Step**. Since the objective (8) decreases after each iteration, and it has a lower bound 0, the algorithm will converge to an local minimal. In our implementation, we terminate the algorithm at the iteration where the decrease of the objective is small enough. The detailed stop criterion is discussed at the end of this section.

E-Step is an inference problem (i.e., predicting multilabels) which is equivalent [Rousu *et al.*, 2006] to solve the following problem for each instance i:

$$
\operatorname*{argmax}_{\mathbf{y}\in\mathcal{Y}} \mathbf{g}_i^T\boldsymbol{\mu}(\mathbf{y}) \tag{9}
$$

where $\mathbf{g}_i = \ell_i - \mathbf{K}_i\boldsymbol{\mu}$ is the gradient of the objective function in (3). Here we use both the notations $\mathbf{w}$ and $\boldsymbol{\mu}$, but they are equivalent since $\boldsymbol{\mu}$ is the dual variable of $\mathbf{w}$. We denote by $\mathcal{G}_j = (V_j, E_j)$ the subtree of $\mathcal{G}$ rooted at the microlabel y_j. Inspired by a botton-up procedure, we define the following two functions:

- $T_{y_j}(i,j)$ the best objective (9) value obtained for instance i in the subtree rooted at the node j when the microlabel y_j has been fixed.
- $G_{y_j}(i,e)$ the best objective (9) value obtained for instance i in the subtree rooted by the edge $e = (j,j')$ when the microlabel y_j has been fixed.

These two functions are computed in the following manner:

$$
T_{y_j}(i,j) = \begin{cases} \sum_{e=(j,j')\in E_j} G_{y_j}(i,e), & \text{if } E_j \neq \emptyset \\ 0, & \text{otherwise} \end{cases}
$$

$$
G_{y_j}(i,e) = \max_{y_{j'}} g_e(i,y_j,y_{j'})\mu_e(i,y_j,y_{j'}) + T_{y_{j'}}(i,j')
$$

where $g_e(i,y_j,y_{j'})$ is the element value of $\mathbf{g}_i$ on the edge $(y_j,y_{j'})$.

At the root, $\max_y T_y(i,root)$ gives the optimum, and we can track back the labels based on this optimum condition. Since we have the domain constraints, we will assign $T_{y_j}(i,j)$ to $-\infty$ for all i, y_j if $y_j = +1$ while the corresponding microlabel y_j for the bag is -1. This way each instance i will not be labeled positive if the corresponding bag is negative.

Then for each positive microlabel y_j of each bag, if all its instances are labeled negative at y_j, we will select the instance that has the minimum decrease of the objective value (9) when changing its mircolabel y_j to positive, and make y_j positive for this instance. After these procedures, the instance labels satisfy the domain constraints.

Algorithm 1 EM-HM^3 for Multi-instance Hierarchical Multi-label Classification

```
Require:
    Training Data: $\mathbf{X} = \{\mathbf{x}_j^{(i)}\}_{i=1,j=1}^{m,R_i}$, $\mathbf{Y} = \{\mathbf{y}^{(i)}\}_{i=1}^{m}$
    Hierarchical Max-margin Multi-label classification function: $\mathbf{w} = \text{HM}^3(\mathbf{X},\mathbf{Y})$;
Ensure:
    Optimal parameter $\mathbf{w}^*$.
Initialize instance labels $\tilde{\mathbf{Z}}$: $\mathbf{y}_j^{(i)} = \mathbf{y}^{(i)}, \forall i,\ j.$
REPEAT:
    $\mathbf{Z} = \tilde{\mathbf{Z}}$, Compute parameter: $\mathbf{w} = HM^3(\mathbf{X},\mathbf{Z})$
    Make $T_{y_j}(i,j) = -\infty$ if the corresponding bag label is negative
    Update labeling $\tilde{\mathbf{Z}}$: $\mathbf{y}_j^{(i)} = \text{argmax}_{\mathbf{y}\in\mathcal{Y}}\, \mathbf{g}_i^T\mu(\mathbf{y})$
    FOR every $i$ in $\{1,2,\cdots,m\}$:
        IF $\tilde{\mathbf{Z}}_i \notin S_i$ as for some microlabel $y_k$:
            $\Delta t_{j,k}^{(i)} = \mathbf{g}_i^T\mu(\mathbf{y}_j^{(i)}) - \mathbf{g}_i^T\mu(\mathbf{y}_j'^{(i)})$,
              where $\mathbf{y}_j'^{(i)}$ only differs from $\mathbf{y}_j^{(i)}$ in $y_k$
            Select $j^* = \text{argmin}_j\, \Delta t_{j,k}^{(i)}$
            Adjust $y_k$ of $\mathbf{y}_{j^*}^{(i)}$ to positive
WHILE ($\tilde{\mathbf{Z}} \neq \mathbf{Z}$)
return w;
```

The EM-HM^3 algorithm is presented in **Algorithm** 1. Once we obtain the solution of the parameter $\mathbf{w}$, for the expert profile $\mathbf{x}^{(i)}$ with unknown research areas, we can acquire the research areas $\mathbf{y}_j^{(i)}$ for each document in the profile, and then predict research areas of the profile as $\mathbf{y}^{(i)} = \bigvee_{j=1}^{R_i}\mathbf{y}_j^{(i)}$.

In practice, we use a different stop criterion. We notice that the number of different microlabels between $\tilde{\mathbf{Z}}$ and $\mathbf{Z}$ has a decreasing trend, and after first several iterations it will fluctuate around some small number. So we terminate the algorithm when $|\tilde{\mathbf{Z}} \neq \mathbf{Z}| \leq \alpha|\mathbf{Z}|$. Also the optimization problem (8) is solved by the marginal dual method which gives an approximate solution, so we terminate the algorithm when the objective of the dual problem does not have enough increase: $o\tilde{b}j - obj \leq \beta * obj$.

3.3 Discussions

In the determining expert research areas task, there exist correlations among multiple research areas, and between documents and research areas. HM^3 only models the hierarchical relations between multiple research areas, which fails to handle the MIHML problem. Our EM-HM^3 also explores the correlation between documents and research areas via opti-

Table 1: F1 value for each level of the hierarchy for different algorithms.

	The value of F1 (%) with the training size 400				
Methods	Level 1	Level 2	Level 3	Level 4	Level 5
MLSVM	100	$\mathbf{85.3} \pm 0.7$	56.1 ± 0.8	21.5 ± 1.4	14.8 ± 1.9
MIMLSVM	100	84.7 ± 0.8	55.2 ± 1.4	19.9 ± 1.3	15.6 ± 2.2
HM^3	100	84.5 ± 0.8	56.7 ± 1.7	23.8 ± 1.1	19.4 ± 2.4
EM-HM^3	100	84.5 ± 0.7	$\mathbf{58.0} \pm 1.5$	$\mathbf{27.1} \pm 1.2$	$\mathbf{23.1} \pm 2.0$

mizing document based research areas under the constraint that these research areas are consistent with the profile. This procedure enables us to uncover the evidence inside the documents for research areas. We notice that research areas in lower levels (levels far from the root) of the hierarchy tend to have weaker evidence, which may only exist in very few documents, so it is more challenging to predict them correctly. In this case, EM-HM^3 achieves a boost of the accuracy in lower levels compared with other methods.

Moreover, if training examples are insufficient, the classification models learned from the sparse information by previous MIL and HML methods will be less effective. On the other hand, our model fully explores the problem structure using multi-instance and hierarchical multi-label learning. Therefore it achieves much better results under the situation that the number of training examples is small.

4 Experimental Evaluation

We test our algorithm with a public expertise database INDURE[1]. In this dataset, an expertise profile describes professional information of a researcher, including his/her background, publication list, funded projects, and theses as individual documents. The research areas are organized by the hierarchy of National Research Council[2]. For instance, biochemistry, public science and neuroscience are branches of life science, while biochemistry also has branches like molecular biology and structural biology. Funding section provides the information of funded projects; thesis section is displayed with the abstracts of thesis documents. We extract features from these documents by TF-IDF [Salton, 1970] weighing method with word stemming and stop-word removal. Each document feature is a 2,000-dimension vector.

There are 1,132 research areas and 1,930 different expert profiles (bags), associated with 22,049 documents (instances) in total. Labels are consistent in this dataset, that is if a microlabel is positive, then its parent is positive. We notice that most of the research areas are only present in very few profiles [3]. This is a common problem of tail labels in MLC. Since this work focuses on the Multi-instance and Multi-label aspects rather than the zero-short learning problem [Palatucci *et al.*, 2009], we remove the sparse labels which are labeled fewer than 30 times in expert profiles. The label set consists of 87 microlabels that form a hierarchy of depth 5: one root microlabel with 3, 8, 58, 17 microlabels in level 2 to level 5 respectively. Among the 1,930 profiles, 33.4% of them belong to more than one leaf node (i.e., the bottom level) of the hierarchy. The minimum number of microlabels associated with a profile is 2, while the maximum is 16, and the average number is around 6.

We want to answer the following questions in the experiments: 1). Whether the hierarchical multi-label classification scheme helps to improve the performance compared to traditional multi-label methods? 2). Whether the multi-instance scheme outperforms single-instance learning? 3). How effective is the proposed approach compared to existing methods under different training sizes?

4.1 General Result

Since both HM^3 and EM-HM^3 are max-margin based algorithms, we select multi-label SVM (MLSVM) and MIMLSVM [Zhou and Zhang, 2006] for comparisons, both of which have achieved excellent performances in the multi-label classification and the multi-instance multi-label setting respectively [Zhou *et al.*, 2012]. MLSVM learns each research area separately with the constraint that child node can be positive only when its parent node is positive. It is implemented by LIBSVM [Chang and Lin, 2011]. MIMLSVM is an algorithm that first maps the original multi-instance to a new feature space, $f : \mathbf{x}^{(i)} \to \mathbf{z}_i$, where the dimension of $\mathbf{z}_i$ is a ratio γ of the training size, then applies MLSVM to learn the classification problem $\{\mathbf{z}_i, \mathbf{y}^{(i)}\}$.

We use the standard information retrieval statistic F1 to evaluate the performance, where $\mathrm{F1} = 2\mathrm{PR}/(\mathrm{P} + \mathrm{R})$, and P denotes precision, R denotes recall. In order to highlight the differences between research areas in different levels of the hierarchy, the results are presented in level-wise.

Linear kernel is used in all four methods to conduct fair comparisons. Ten-fold cross validations are performed, where the regularization parameter is tuned by maximizing the sum of F1 values of all levels. For MIMLSVM, the ratio γ is set to be 20%[4]. For EM-HM^3 and HM^3, we use the following loss function: $\ell_{(}\mathbf{y}, \mathbf{v}) = \sum_j c_j[y_j \neq v_j]$ where $c_{\mathrm{root}} = 1$, $c_j = c_{\mathrm{pa(j)}}/|\mathrm{sibl(j)}|$, in which pa(j) and sibl(j) denote the parent and the set of siblings of research area y_j respectively. Since MLSVM and HM^3 are not designed for multi-instance learning, we create a feature vector for each profile by treating the affiliated documents as a single one and implement MLSVM and HM^3 by using profile features.

[1] www.indure.org

[2] http://www.rackham.umich.edu/nrc/taxonomy/

[3] Most of the research areas are labeled fewer than 5 times.

[4] We tried several γ from 20% to 50%, and it turned out that the higher value of γ does not have a big impact on the result.

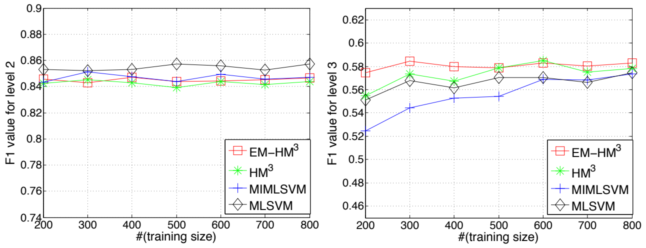

Figure 2: The F1 value of labels in level 2 (left) and level 3 (right) varies with different training sizes: from 200 to 800.

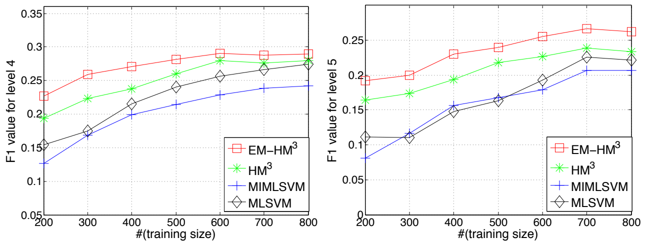

Figure 3: The F1 value of labels in level 4 (left) and level 5 (right) varies with different training sizes: from 200 to 800.

Table 1 depicts the level-wise result of different algorithms for a fixed training size 400, together with the standard deviations. It can be seen from this table that MLSVM gives the best result for level 2, but the differences among the four methods are very small. For the later 3 levels, EM-HM^3 has the highest F1 values, with an increasing advantage from level 3 to level 5. The reason is that for higher levels (i.e., levels close to the root of the tree), there are fewer research areas, which make the classification task easier than that in lower levels (i.e., levels far from the root of the tree), so even less sophisticated methods can achieve comparable results.

We also conduct paired t-test with significance level 0.05 on Table 1 to compare the two hierarchical multi-label methods HM^3 and EM-HM^3 with the other two traditional multi-label methods MLSVM and MIMLSVM. The results show that the former two methods significantly outperform the latter ones especially in lower levels. In hierarchical multi-label learning, the correct prediction of parent nodes will help the prediction on children nodes, which is consistent with our expectation of using hierarchical multi-label learning. We can also observe from this table that the proposed EM-HM^3 approach obtains much better results compared with HM^3, which is already very competitive in levels 4 and 5. This demonstrates the advantage of combining multiple instance with hierarchical multiple labels against the other methods, which only model one of these two aspects.

4.2 Result for Different Training Sizes

In this set of experiments, we further evaluate the performance of all compared methods on different training sizes. Usually, the classification results drop with the decreasing of the training examples. There is also a trend that when providing massive training data, the performance gaps of various algorithms diminish. However, a robust modeling strategy should work with a limited amount of training data. Therefore, we compare the proposed EM-HM^3 approach to the other three methods in this aspect by modifying the number of training examples. Specifically, we evaluate these four methods on different training sizes from 200 to 800 examples. The comparison results for all levels are shown in Figure 2 and Figure 3. It can be seen from Figure 2 that, for level 2, lines are close to each other and fluctuate between 0.840 and 0.857. There is no evidence that the F1 values get higher with the increase of the training size. Our hypothesis is that there are only three nodes in level 2, so it is easy to classify the profile into one of the three nodes even with small numbers of training data. In the figure for level 3, EM-HM^3 achieves the best performance, especially with training size 200. The F1 value is 0.574, which is significantly better than 0.525 of MIMLSVM and 0.551 of MLSVM.

It is clear that in Figure 3, lines are more distinct. From the shapes of the graphs, we can generally have the conclusion: for level 4 and level 5, EM-HM^3 $>$ HM^3 $>$ $\{$MLSVM, MIMLSVM$\}$. Unlike the results for level 3 that the advantage of EM-HM^3 narrows to zero when providing sufficient training data, EM-HM^3 obtains strictly higher F1 value than all other methods for labels in levels 4 and 5. And this gap becomes larger with the decrease of the training size. To achieve the same performance of EM-HM^3 for level 4 or 5 with training size 200, HM^3 needs nearly 400 training examples, while MLSVM and MIMLSVM require almost 600 training examples. These results indicate that EM-HM^3 is much more robust with different training sizes compared to the other methods, which is consistent with our expectation. Another interesting observation is that the other multi-instance algorithm MIMLSVM does not perform well in most cases. Our explanation is that for MIMLSVM there is a trade-off between utilizing multi-instance information and preserving its feature dimension. Since it maps multiple instances into a single bag with the feature dimension to be a ratio (20%-30%) of the training size, this process causes MIMLSVM a loss of information because the feature vector is in low dimension space, especially if multiple instances do not have enough discriminations.

5 Conclusion

This paper proposes the Multi-instance Learning of Hierarchical Multi-label Classification (MIHML) approach to handle the expert profiling task, which aims to determine expert research areas. In this task, we have modeled two types of correlations. One is the correlation between multiple research areas, which have a hierarchical structure. The other one is the association between documents and research areas, where the evidence of a research area may only exist in a few instead of all documents of the profile. we propose an optimization method based on Expectation-Maximization (EM) algorithm to learn the model parameters. Experiments on the expertise profiles from a real world dataset show that EM-HM^3 outperforms traditional MIML and HML algorithms, in both accuracy and the ability of dealing with a small training size.

6 Acknowledgments

This work is partially supported by NSF research grants IIS-0746830, DRL-0822296, CNS-1012208, IIS-1017837, CNS-1314688 and a research grant from Office of Naval Research (ONR-11627465). This work is also partially supported by the Center for Science of Information (CSoI), an NSF Science and Technology Center, under grant agreement CCF-0939370.

References

[Andrews *et al.*, 2002] Stuart Andrews, Ioannis Tsochantaridis, and Thomas Hofmann. Support vector machines for multiple-instance learning. In *NIPS*, pages 561–568, 2002.

[Balog and De Rijke, 2007] Krisztian Balog and Maarten De Rijke. Determining expert profiles (with an application to expert finding). In *IJCAI*, volume 7, pages 2657–2662, 2007.

[Balog *et al.*, 2009] Krisztian Balog, Leif Azzopardi, and Maarten de Rijke. A language modeling framework for expert finding. *Information Processing & Management*, 45(1):1–19, 2009.

[Balog *et al.*, 2012] Krisztian Balog, Yi Fang, Maarten de Rijke, Pavel Serdyukov, and Luo Si. Expertise retrieval. *Foundations and Trends in Information Retrieval*, 6(2-3):127–256, 2012.

[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.

[Berendsen *et al.*, 2013] R. Berendsen, M. de Rijke, K. Balog, T. Bogers, and van den Bosch. On the assessment of expertise profiles. *J. Am. Soc. Inf. Sci.*, 64:2024–2044, 2013.

[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*, 2:27:1–27:27, 2011.

[Chen *et al.*, 2006] Yixin Chen, Jinbo Bi, and James Ze Wang. Miles: Multiple-instance learning via embedded instance selection. *TPAMI*, 28(12):1931–1947, 2006.

[Dietterich *et al.*, 1997] Thomas G. Dietterich, Richard H. Lathrop, and Tomás Lozano-Pérez. Solving the multiple instance problem with axis-parallel rectangles. *Artif. Intell.*, 89(1-2):31–71, 1997.

[Elisseeff and Weston, 2001] André Elisseeff and Jason Weston. A kernel method for multi-labelled classification. In *Advances in neural information processing systems*, pages 681–687, 2001.

[Hariharan *et al.*, 2010] Bharath Hariharan, Lihi Zelnik-Manor, Manik Varma, and Svn Viswanathan. Large scale max-margin multi-label classification with priors. In *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, pages 423–430, 2010.

[Hertzum and Pejtersen, 2000] Morten Hertzum and Annelise Mark Pejtersen. The information-seeking practices of engineers: searching for documents as well as for people. *Information Processing & Management*, 36(5):761–778, 2000.

[Jin *et al.*, 2009] Rong Jin, Shijun Wang, and Zhi-Hua Zhou. Learning a distance metric from multi-instance multi-label data. In *CVPR*, pages 896–902, 2009.

[Nguyen *et al.*, 2014] C Nguyen, Xiaoliang Wang, Jing Liu, and Zhi-Hua Zhou. Labeling complicated objects: Multi-view multi-instance multi-label learning. In *Twenty-Eighth AAAI Conference on Artificial Intelligence*, 2014.

[Palatucci *et al.*, 2009] Mark Palatucci, Dean Pomerleau, Geoffrey E Hinton, and Tom M Mitchell. Zero-shot learning with semantic output codes. In *Advances in neural information processing systems*, pages 1410–1418, 2009.

[Rousu *et al.*, 2006] Juho Rousu, Craig Saunders, Sandor Szedmak, and John Shawe-Taylor. Kernel-based learning of hierarchical multilabel classification models. *The Journal of Machine Learning Research*, 7:1601–1626, 2006.

[Rybak *et al.*, 2014] Jan Rybak, Krisztian Balog, and Kjetil Nørvåg. Temporal expertise profiling. In *Advances in Information Retrieval*, pages 540–546. Springer, 2014.

[Salton, 1970] Gerald Salton. Automatic text processing. *Science*, 168(3929):335–343, 1970.

[Schietgat *et al.*, 2010] Leander Schietgat, Celine Vens, Jan Struyf, Hendrik Blockeel, Dragi Kocev, and Sašo Džeroski. Predicting gene function using hierarchical multi-label decision tree ensembles. *BMC bioinformatics*, 11(1):2, 2010.

[Taskar *et al.*, 2003] Benjamin Taskar, Carlos Guestrin, and Daphne Koller. Max-margin markov networks. In Sebastian Thrun, Lawrence K. Saul, and Bernhard Schölkopf, editors, *NIPS*. MIT Press, 2003.

[Wang *et al.*, 2012] Qifan Wang, Luo Si, and Dan Zhang. A discriminative data-dependent mixture-model approach for multiple instance learning in image classification. In *ECCV (4)*, pages 660–673, 2012.

[Wang *et al.*, 2014] Qifan Wang, Lingyun Ruan, and Luo Si. Adaptive knowledge transfer for multiple instance learning in image classification. In *Twenty-Eighth AAAI Conference on Artificial Intelligence*, 2014.

[Zhang and Zhou, 2005] Min-Ling Zhang and Zhi-Hua Zhou. A k-nearest neighbor based algorithm for multi-label classification. In *Granular Computing, 2005 IEEE International Conference on*, volume 2, pages 718–721. IEEE, 2005.

[Zhou and Zhang, 2006] Zhi-Hua Zhou and Min-Ling Zhang. Multi-instance multi-label learning with application to scene classification. In *Advances in Neural Information Processing Systems*, pages 1609–1616, 2006.

[Zhou *et al.*, 2012] Zhi-Hua Zhou, Min-Ling Zhang, Sheng-Jun Huang, and Yu-Feng Li. Multi-instance multi-label learning. *Artificial Intelligence*, 176(1):2291–2320, 2012.