A Generative Probabilistic Model for Multi-Label Classification

Hongning Wang Minlie Huang Xiaoyan Zhu
State Key Laboratory of Intelligent Technology and Systems
Tsinghua National Laboratory for Information Science and Technology
Department of Computer Science and Technology, Tsinghua University, Beijing 100084 China
whn03@mails.tsinghua.edu.cn {aihuang, zxy_dcs}@tsinghua.edu.cn

Abstract

Traditional discriminative classification method makes little attempt to reveal the probabilistic structure and the correlation within both input and output spaces. In the scenario of multi-label classification, most of the classifiers simply assume the predefined classes are independently distributed, which would definitely hinder the classification performance when there are intrinsic correlations between the classes. In this article, we propose a generative probabilistic model, the Correlated Labeling Model (CoL Model), to formulate the correlation between different classes. The CoL model is presented to capture the correlation between classes and the underlying structures via the latent random variables in a supervised manner. We develop a variational procedure to approximate the posterior distribution and employ the EM algorithm for the empirical Bayes parameter estimation. In our evaluations, the proposed model achieved promising results on various data sets.

1. Introduction

1.1 Multi-label classification

In the traditional definition of classification, classes are mutually exclusive:

> *Let* X *denote the domain of possible samples, and* Y *be a finite set of class labels, the goal of the classification is to find an optimal classifier* $H: x \rightarrow y, x \in X, y \in Y$*, which could minimize the misclassification rate.*

However, in most of the real situation, data may associate with multi-classes simultaneously. For example, in the text classification task, a scientific article might be also concerning about the economy and in the scene categorization domain, an image may belong to the semantic concept *beach* and *sunset* together, yielding multiple labels [20]. In that case, a suitable definition for this kind of classification should be the multi-label classification, by modifying *y* in the original definition to be a subset of *Y* rather than a single one, and thus the optimal classifier should be $H: x \rightarrow \mathbf{y}, x \in X, \mathbf{y} \subseteq Y$ to optimize some specific evaluation metric. We should note that, in most cases, there are intrinsic latent correlations between the classes. For example, a document concerning about politics is more likely to be also talking about the economy (positive correlation) but less likely talking about the pop stars (negative correlation). Unfortunately, most of the classification algorithms for the multi-labeling problem simply assume the classes are independently distributed, so that they failed to directly model the correlation between the classes.

A straightforward solution for the multi-label classification is to map the problem to a *one-versus-the rest* manner [15], which constructs a set of binary classifiers obtained by training on each possible class versus all the rest and assigns a real value for each class to indicate the class membership. But the deficiency of this simple mapping is obvious: the rough separation strategy ignores the correlation between the classes; moreover, the traditional discriminative classifiers make little attempt to uncover the probabilistic structure within both input and output spaces.

Researchers have noticed this problem and tried to solve it from different perspectives. Matthew et al. suggested several ways to utilize the multi-label samples for training with binary classifiers and different strategies to predict the class membership [20]. Zhang adapted the traditional *KNN* lazy learning algorithm for multi-label data by utilizing the statistical information gained from the unseen sample's neighborhood [25]. Schapire et al. advanced *BoosTexter* [21], an extended *AdaBoost* algorithm, to address the multi-label text classification problem. In their work, they transformed the multi-labeling issue into a document-class pair ranking problem. *BoosTexter* employed various base classifiers to evaluate every document-class pair and ranked the separate predictions according to the weight settings.

1550-4786/08 $25.00 © 2008 IEEE
DOI 10.1109/ICDM.2008.86

IEEE computer society

However, they noted that it was an open issue to control the model complexity to avoid over-fitting. McCallum proposed a Bayesian mixture model to select the most probable set of classes from the power set of all the classes and used some heuristics to reduce the associated computational complexity [19]. The proposed model tried to capture the relationship between the classes and word occurrences, but it did not consider the correlation within the classes.

1.2 Generative topic model

Nowadays, in the machine learning community, the generative topic model is receiving more and more attentions. Latent Dirichlet Allocation (*LDA*) [11] is one of the most typical models. It reduces the complex process of producing a document into a small number of simple probabilistic steps and thus specifies a probability distribution over all possible documents. Using standard statistical techniques, one can invert the process and infer the set of latent topics responsible for generating a given set of documents [22]. An important contribution of *LDA* is that, it explicitly models the heterogeneity in the grouped data that exhibits multiple latent patterns.

Recent work has employed *LDA* as a building block to address particular modeling problems. Fei-Fei Li advanced a hierarchical generative model to classify natural scene in an unsupervised manner [17]; Blei proposed an image-caption model to capture the correlation between image regions and caption words [9]; Griffiths modeled the documents with both short-range syntactic and long-range semantic dependencies [14].

However, the *LDA* model failed to directly formulate the correlation between topics because of the dependence assumption implicit in the Dirichlet distribution on the topic proportions, which are nearly independent. Several other generative topic models have been recently proposed to capture the correlation between topics, such as Hierarchical Dirichlet Process Model (HDP) [23], Correlated Topic Model (CTM) [10] and Pachinko Allocation Model (PAM) [18].

The advantages of the generative topic model are obvious: 1) it would be easy to postulate complex latent structures responsible for a set of observations; 2) the correlation between the different factors could be easily exploited by introducing the latent variables.

In this paper, to capture the correlations within different classes and words in the multi-label classification, we propose a hierachical generative probabilistic model to formulate the generation of the multi-labeled documents. We model these documents as a finite mixture over the classes and words: different classes exhibit different proportions of latent topics, which are represented by distributions of words over a fixed vocabulary, and the observed words are governed by the latent topic factors accordingly. By this model, we would be able to model the correlations within the classes and words simultaneously.

We should emphasize that the reason we use the language of text classification in the following expatiation is just for intuitive understanding and interpretation about the notions. It is important to note that the proposed model is not narrowly restricted to the text classification field: it could be feasibly applied to any multi-label classification or annotation problem such as scene categorization in image processing and gene function annotation in bioinformatics.

The paper is organized as follows: in Section 2, detailed descriptions for the proposed model are presented and we will discuss the inference and parameter estimation procedures for the proposed model in Section 3; in Section 4, extensive experiments are performed in different perspectives to validate the model; we would conclude the work in this paper and demonstrate our contributions in Section 5.

2. Correlated Labeling Model

We present the novel Correlated Labeling Model (*CoL* Model) to address the multi-label classification issue. The graphical representation of the *CoL* model is depicted in Figure 1. Following the standard graphical model formalism [12], nodes represent the random variables, edges indicate the possible dependence and boxes with number N means the unit in this box is repeated N times. Shaded nodes are observed random variables, unshaded nodes are latent random variables. The joint distribution can be obtained from the graph by taking the product of the conditional distribution of nodes given their parents, see Eq(2).

The *CoL* Model can be viewed in the terms of generative process that, to generate a document, we should first select a set of classes (e.g. themes of a document), then select different topics under the classes (e.g. aspects about the themes), and finally employ specific words to build up the contents of the document.

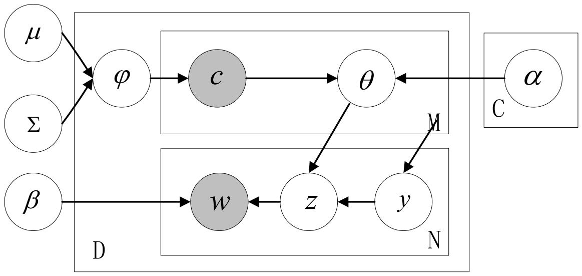

Figure 1. Graphical model representation for the CoL Model.

Formally, we define a corpus consists of *D* documents, *C* classes and *V* words, and a given document consists of *M* classes and *N* words. To simplify the model, we have assumed the topic size *k* is known and fixed on the whole corpus. In the given document *d*, we denote φ as the document-specific distribution of classes; θ as the distribution of topics under each class; $\mathbf{z} = \{z_1, z_2, z_3, \ldots, z_N\}$ as the particular discrete topic assignment for each word and $\mathbf{y} = \{y_1, y_2, y_3, \ldots, y_N\}$ as an indexing variable to indicate which class generates the corresponding topic. These are the latent variables. *c* is a *C*-dimensional vector with $c_i = 1$ to imply document *d* is associating with class *i* and $w = \{w_1, w_2, w_3, \ldots, w_N\}$ are the observed words in *d*. Besides, μ and Σ are the mean and covariance parameters of a multivariate Normal distribution to formulate the class distribution; α are *C* *k*-dimensional Dirichlet parameters to characterize the topic prior distribution under each class; and β are *k* *V*-dimensional Multinomial parameters to represent the word distribution under topics. These are the model parameters.

Conditioned on the model parameters *(*$\mu, \Sigma, \alpha, \beta$*)*, the *CoL* model assumes the following generative process of the classes and words in the document:

1. Sample φ from the Normal distribution: $\varphi \sim N(\mu, \Sigma)$
2. For each class c_m, m ∈ {1,2,3,. . . ,M} :
   a. Sample c_m from the Multinomial distribution : $c_m \sim$ Mul($l(\varphi)$)
   b. Sample θ_m from the Dirichlet distribution conditioned on c_m : $\theta_m \sim$ *Dir(*$\alpha|c_m$*)*
3. For each word w_n, n ∈ {1,2,3,. . . ,N}:
   a. Sample y_n from the Uniform distribution conditioned on M:
   $y_n \sim$ Unif$(1, 2, 3, \ldots, M)$
   b. Sample z_n from the Multinomial distribution conditioned on y_n: $z_n \sim$ Mul$(\theta|y_n)$
   c. Sample w_n from the Multinomial distribution conditioned on z_n : $w_n \sim p(w_n|\beta, z_n)$

where $l(\varphi)$ maps the natural parameter of the class proportions to the mean parameter by logistic Normal [6]:

$$l(\varphi) = \frac{\exp(\varphi)}{1 + \sum_i \exp(\varphi_i)} \tag{1}$$

Note that, the *CoL* model employs a multivariate Normal distribution *N(*μ, Σ*)* to capture the correlation between the classes: for each document, it draws a real valued random vector from *N(*μ, Σ*)* and then maps it to a *C*-1 dimensional simplex to obtain a Multinomial parameter for the document-specific distribution of classes. The mapping is implemented by the logistic Normal $l(\varphi)$, see Eq(1). The covariance matrix Σ induces the dependencies between the components, allowing for a general pattern of variability between its components. Following the general settings of *LDA* model, we assume the topic proportion θ is drawn from the Dirichlet distribution and each topic is represented by a Multinomial distribution of words on a fixed vocabulary. Furthermore, we assume such proportion varies between different classes observed in the documents. Besides, since the relationship between the classes and topics is underlying, we use the indexing variable **y** to indicate the latent structure between them.

The joint probability on the words, classes and the latent variables in one document is thus given by:

$$\begin{aligned} &p(\varphi, \theta, \mathbf{y}, \mathbf{z}, \mathbf{c}, \mathbf{w}|\mu, \Sigma, \alpha, \beta) = \\ &\qquad p(\varphi|\mu, \Sigma) \prod_{m=1}^{M} p(c_m|\varphi) p(\theta_m|\alpha, c_m) \\ &\qquad \prod_{n=1}^{N} p(y_n|M) p(z_n|\theta_{y_n}) p(w_n|\beta, z_n) \end{aligned} \tag{2}$$

From the notion behind the proposed model, we can find obvious distinction between the proposed *CoL* model and the *LDA* model: the *CoL* model is supervised while the *LDA* model is unsupervised. *CoL* model aims to capture the information conveyed in the class membership, to exploit the in-depth relation between the classes and words, and to predict the potential classes in an unseen document. The *LDA* model is not capable to directly formulate such class membership, so that some other regression or classification techniques have to be employed to perform the prediction [11]. Besides, the *LDA* model assumes the proportion of topics is identical in the whole corpus, while in the *CoL* model the mixture is depending on the classes which the document belongs to. In this sense, the *CoL* model can overcome the deficiency in the *LDA* model stems from the strong independence assumptions.

An intuitive interpretation for the proposed *CoL* model is illustrated in Figure 2. In the traditional approach for the multi-label classification (the left panel in Figure 2), the employed classifiers simply assume the predefined classes are independent between each other. When one class conveys information about another, the traditional classifiers would fail to capture this. Furthermore, those classification algorithms assume all the words are independent when given the observed classes, thus it would ignore to model the latent patterns among the different words under particular classes either.

On the contrary, the *CoL* model (the right panel in Figure 2) formulates the relationship between words and classes within a more throughout consideration: in each document, the classes are drawn from a correlated prior distribution, in our case the multivariate Normal distribution with a non-diagonal covariance, each class exhibits

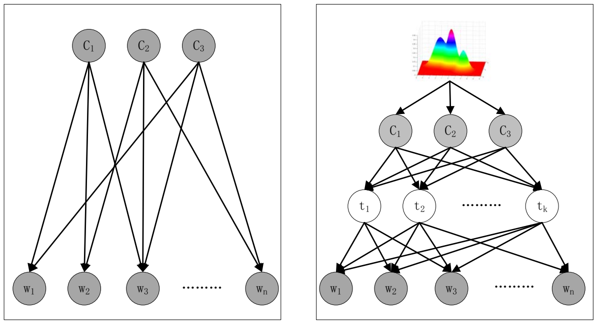

Figure 2. Comparison between the traditional multi-labeling approach and the CoL model. In the above representation, c denotes the class labels associating with the document, w denotes the observed words and t in the right panel denotes the latent topic factors.

different proportion of the topics, and different topics govern dissimilar word occurrences, embedding the correlation among different words.

In the above intuitive representation of the *CoL* model, it is obvious that the correlation between the classes and words is not modeled as an one-to-one mapping, but in a more general manner: via the latent topic factors, words are treated as finite mixtures under a set of classes, so that they are not restricted to any particular classes and multiple words could contribute to the same class. Efficient dimensional decomposition could be explicitly implemented: *V*-dimensional word space is mapped into the *k*-dimensional topic space, in which it will be easier to reveal the latent correlations between the classes and the variant word distributions.

3. Inference and parameter estimation

3.1 Variational inference

In order to utilize the *CoL* model, the key inferential problem is to compute the posterior distribution of the classes in a given document, that is:

$$p(\mathbf{c},\varphi,\theta,\mathbf{y},\mathbf{z}|\mathbf{w},\mu,\Sigma,\alpha,\beta)=\frac{p(\mathbf{c},\mathbf{w},\varphi,\theta,\mathbf{y},\mathbf{z}|\mu,\Sigma,\alpha,\beta)}{p(\mathbf{w}|\mu,\Sigma,\alpha,\beta)}$$

Unfortunately, this posterior distribution is intractable: the couples between φ and α, θ and β induce a combinatorial number of terms and make it impossible to efficiently get the exact inference result. Different from the *LDA* model, where the conjugacy between the Dirichlet and Multinomial distribution provides nice computational convenience; in the *CoL* model, a non-conjugate Normal distribution is employed to capture the correlations between the classes, which does not enjoy the same convenience. Thus we cannot analytically compute the integrals of each term. And the non-conjugacy further precludes most of the *Markov chain Monte Carlo (MCMC)* [7] sampling techniques, especially for the *Gibbs* Sampling, which makes use of the conjugacy to compute the analytical coordinate-wise posteriors. In this case, we develop a variational procedure [8] (in particular, the mean filed approximation) to approximate the desired posterior distribution, which provides nice computational convenience and intuitive interpretation about the middle results.

In particular, we define the following fully factorized distribution on the latent factors:

$$\begin{aligned} &q(\varphi,\theta,\mathbf{y},\mathbf{z}|\lambda,\delta,\gamma,\phi,\sigma)= \\ &\qquad q(\varphi|\lambda,\delta)\prod_{m=1}^{M}q(\theta_m|\gamma_m)\prod_{n=1}^{N}q(y_n|\sigma_n)q(z_n|\phi_n) \end{aligned} \tag{3}$$

In the above variational distribution, the document-specific class distribution φ is governed by a *C* dimensional multivariate Normal distribution $N(\lambda,\delta)$. Since the variational parameters are fit within a single document, there is no advantage to introduce a non-diagonal covariance. The variational topic distribution θ is specified by *M* *k* -dimensional Dirichlet parameters γ, the class-topic indicator $\mathbf{y}$ is conditioned on *N* *M*-dimensional Multinomial parameters σ and the discrete topic assignment $\mathbf{z}$ is controlled by *N* *k*-dimensional Multinomial parameters ϕ.

The meaning of this variational distribution is obvious: we release the dependence among the latent variables by assuming they are independently drawn from the respective distribution. Thus the aim of the variational inference is to find the optimal variational parameters which would maximize the likelihood on the given documents.

By *Jensen's* inequality, we could estimate the lower bound of the log likelihood as follows (we omit the parameters for simplicity):

$$\begin{aligned} &\log p(\mathbf{c},\mathbf{w}) \\ &=\log\iint\sum_{y_n}\sum_{z_n}\frac{p(\varphi,\theta,y_n,z_n,\mathbf{c},\mathbf{w})}{q(\phi,\theta,y_n,z_n)}q(\phi,\theta,y_n,z_n)\mathrm{d}\varphi\mathrm{d}\theta \\ &\geq\iint\sum_{y_n}\sum_{z_n}q(\phi,\theta,y_n,z_n)\log p(\varphi,\theta,y_n,z_n,\mathbf{c},\mathbf{w})\mathrm{d}\varphi\mathrm{d}\theta \\ &\quad-\iint\sum_{y_n}\sum_{z_n}q(\phi,\theta,y_n,z_n)\log q(\phi,\theta,y_n,z_n)\mathrm{d}\varphi\mathrm{d}\theta \\ &=E_q[\log p(\varphi,\theta,\mathbf{y},\mathbf{z},\mathbf{c},\mathbf{w})]-E_q[\log q(\phi,\theta,\mathbf{y},\mathbf{z})] \end{aligned}$$

It is easy to verify that the difference between two sides of the above inequality is the *Kullback-Leibler* divergence between the variational posterior probability and the true posterior probability. We denote the right side of the above inequality as $L(\lambda, \delta, \gamma, \phi, \sigma; \mu, \Sigma, \alpha, \beta)$ to represent the lower bound of log likelihood. Thus, to maximize $L(\lambda, \delta, \gamma, \phi, \sigma; \mu, \Sigma, \alpha, \beta)$ is equivalent to minimize the *KL* divergence between the variational posterior probability and the true posterior probability.

Following the general recipe for variational approximation, we take derivatives of the expectation likelihood function $L(\lambda, \delta, \gamma, \phi, \sigma; \mu, \Sigma, \alpha, \beta)$ with respect to the variational parameters and obtain the following iterative variational parameter estimation equations:

1. Dirichlet parameter γ:

$$\gamma_{ij} = \alpha_{ij} + \sum_{n=1}^{N} \sum_{m=1}^{M} c_m^i \sigma_{nm} \phi_{nj} \tag{4}$$

   where c_m^i means the *m*th class label in the document belongs to class *i*.

2. Multinomial parameter ϕ:

$$\log \phi_{nj} \propto w_n^s \beta_{js} + \sum_{m=1}^{M} c_m^i \sigma_{nm} \left[\psi(\gamma_{ij}) - \psi(\sum_{s=1}^{k} \gamma_{is})\right] \tag{5}$$

   where w_n^s means the *n*th word in the document is the *s*th one in the vocabulary.

3. Multinomial parameter σ:

$$\log \sigma_{nm} \propto \sum_{j=1}^{k} \phi_{nj} c_m^i \left[\psi(\gamma_{ij}) - \psi(\sum_{s=1}^{k} \gamma_{is})\right] \tag{6}$$

4. Optimize the Normal parameter λ and δ^2 by the Conjugate Gradient algorithm:

$$\frac{\partial L(\lambda)}{\partial \lambda} = -\Sigma^{-1}(\lambda - \mu) + \overline{C} - \frac{M}{\epsilon} \exp\{\lambda + \delta^2/2\} \tag{7}$$

$$\frac{\partial L(\delta^2)}{\partial \delta^2} = -\frac{1}{2} Tr(\Sigma^{-1}) + \frac{1}{2\delta^2} - \frac{M}{2\epsilon} \exp\{\lambda + \delta^2/2\} \tag{8}$$

   where $\epsilon = \sum_{i=1}^{C} exp(\lambda_i + \delta_i^2/2)$, and $\overline{C}$ is the class vector observed in the given document.

These estimations have appealing intuitive interpretations. Because the Multinomial distribution is conjugated with the Dirichlet distribution , estimations (4) – (6) are the posterior updating given the expected observations (sufficient statistics) taken under the variational distribution. But the non-conjugacy between the Multinomial and Normal distribution prevents us to analytically get the update equations, therefore we employ the Conjugate Gradient algorithm to find the optimal parameters in (7) and (8).

The only problem left for the inference procedure is that, when we are in the testing phase, we could not know exactly which classes are assigned to the given document in advance. Without the specific classes, we are not able to tell where the words and topics come from. To solve this problem, we appeal to the maximum a posteriori (*MAP*) criterion to retrieval the most probable classes associating with the given document:

$$\hat{\mathbf{c}} = \max_i p(\mathbf{c_i}, \theta, \varphi, \mathbf{y}, \mathbf{z} | \mathbf{w}, \mu, \Sigma, \alpha, \beta) \tag{9}$$

where $\mathbf{c_i}$ is the subset from the power set of all the possible classes.

Unfortunately, it is unfeasible when the number of classes is large. To simplify the computation complexity, we simply estimate the posterior probability of every single class in the given document and use a pre-estimated threshold to retrieve the most probable ones.

3.2. Parameter estimation

Following the same procedure in the variational inference, in this section, we utilize an empirical Bayesian method to estimate the parameters of the *CoL* model. To maximize the likelihood on the training data, we look for the optimal parameters to tighten the lower bound of $L(\lambda, \delta, \gamma, \phi; \mu, \Sigma, \alpha, \beta)$ estimated by the variational inference. By taking derivatives of $L(\lambda, \delta, \gamma, \phi; \mu, \Sigma, \alpha, \beta)$ with respect to the model parameters $(\mu, \Sigma, \alpha, \beta)$, we obtain the following update equations:

1. Update the mean parameter μ and covariance matrix Σ:

$$\mu = \frac{1}{D} \sum_{d=1}^{D} \lambda_d \tag{10}$$

$$\Sigma = \frac{1}{D} \sum_{d=1}^{D} \left\{ I\delta_d^2 + (\lambda_d - \mu)(\lambda_d - \mu)^T \right\} \tag{11}$$

2. Update the Dirichlet parameter α by the Newton-Raphson algorithm:

$$\frac{\partial L(\alpha)}{\partial \alpha_{ij}} = \sum_{d=1}^{D} \sum_{m=1}^{M_d} c_{dm}^i \Big\{ \big[\sum_{s=1}^{k} \psi(\alpha_{is}) - \psi(\alpha_{ij})\big] + \big[\psi(\gamma_{dij}) - \sum_{s=1}^{k} \psi(\gamma_{dis})\big] \Big\} \tag{12}$$

$$\frac{\partial^2 L(\alpha)}{\partial \alpha_{ij} \partial \alpha_{ik}} = \sum_{d=1}^{D} \sum_{m=1}^{M_d} c_{dm}^i \Big[\psi'(\sum_{s=1}^{k} \alpha_{is}) - \delta(j,k)\psi'(\alpha_{ij})\Big] \tag{13}$$

   where $\delta(j,k) = 1$ when $j = k$, otherwise 0.

3. Update the Multinomial parameter β:

$$\beta_{js} \propto \sum_{d=1}^{D} \sum_{n=1}^{N_d} w_{dn}^{s} \phi_{dnj} \tag{14}$$

These update equations correspond to find the maximum likelihood estimation with the expected sufficient statistics for each document taken under the variational posterior. We employed an alternating *EM* procedure to find the optimal parameters as follows:

1. *(E-Step)* For each document in the training corpus, optimizing the variational parameters $(\lambda, \delta, \gamma, \phi)$ according to equations (4) – (8);

2. *(M-Step)* Maximizing the resulting lower bound on the variational likelihood with respect to the model parameters $(\mu, \Sigma, \alpha, \beta)$ according to equations (10) – (14).

4. Experiment results

We collect two different types of data with multi-label annotations from: scientific publications and news reports to evaluate the capability of the proposed model in managing various applications. The *macro-precision*, *macro-recall* and *macro-fscore* [13] are employed to evaluate the performance in average.

4.1. Test corpora

Biological literature. In the molecular biology domain, biologists would employ various experiment methods to confirm their findings; and a single document may contain multiple methods simultaneously. It is important to annotate these experiment methods since each method has an implicit degree of reliability. We collect 5319 full text documents from the public biological database *PubMed* [5] with method annotations from another public annotation databases *MINT* [3] and *IntAct* [1]. One thing we should emphasize is that, this collection is heavily unbalanced: the whole corpus consists of 115 unique methods, and each document is labeled with 1.99 different methods in average; unfortunately, there are 5 dominate methods taking up nearly 59.3% occurrences and 86.1% (99 out of 115) of the methods are observed in less than 10% training data.

Reuters-21578. Documents in this collection are collected from the Reuters financial newswire service in 1987 [16]. It is a well-studied benchmark corpus for many text classification algorithms. There are 90 classes and 10,788 documents in the original corpus and the collection is pre-partitioned into a training set of 7769 documents and a testing set of 3019 documents. To get a more balanced data set, we remove the minor classes with less than 50 documents and build up a collection consisting of 36 classes and 10449 documents with 7543 training documents and 2906 testing documents. In this collection, each document is associated with 1.3 classes in average and about 13.9% documents contain multiple labels.

These two data sets are quite different from each other and represent the typical sources in the real text processing task. We perform simple pre-processions on each data set: 1) remove a standard list of 400 stop words, punctuations, and the terms occur less than 50 times; 2) stems the words to original form.

4.2. Effect of topic factors

We first use the perplexity as the criterion to evaluate the effect of the number of topic factors, which is the only arbitrary parameter in the *CoL* model. The perplexity on a set of testing samples is calculated as follows:

$$\textit{perplexity} = \exp\left\{\frac{-\sum_{d=1}^{D}\sum_{m=1}^{M_d}\log p(c_m|\mathbf{w_d})}{\sum_{d=1}^{D} M_d}\right\} \tag{15}$$

Eq(15) is equivalent algebraically to the inverse of the geometric mean per-class likelihood and the better generalization capability is indicated by a lower perplexity over the held-out testing samples. We evaluate the perplexity on both data sets respectively. In the Bio-Literature data set, we held out 20% of collection for the test purpose and used the remaining 80% to train the model, in accordance with 5-fold cross-validation.

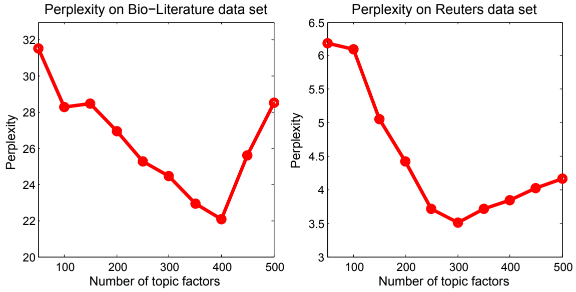

Figure 3. Class perplexity on the number of topics. The left panel illustrates the perplexity on the Bio-Literature data set and the right panel illustrates the perplexity on the Reuters data set.

Figure 3 demonstrates that the generalization power of the *CoL* model gets improved with more topic factors. Since with more topic factors the documents could be partitioned into finer segments, more precise correlations between the classes and words could be captured. But as the number of topics exceeds a limit, the model becomes too specific (higher perplexity). Therefore we could conclude that the topic factors could be treated as the discriminate granularity of the model and it operates as a tradeoff between the generality and specificity. Besides, as the number of topic factors increase, there will be more parameters to be estimated (linearly increase with the number of topics), so that more training data is needed to obtain the reliable parameters. In this sense, when the number of topic factors exceeds a limit, the quality of the estimated parameters decreases and hampers the prediction power.

Besides understanding the impact of the number of topic factors on the generalization capability, we would be more interested in their explicit effect on the classification performance. Here, we evaluate the precision and recall performances under different number of topic factors on the two data sets. We use the same corpus partition as in Figure 3.

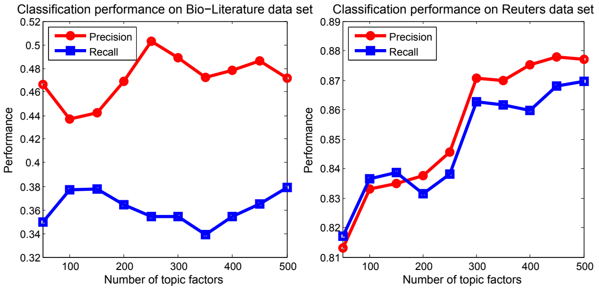

Figure 4. Classification performance on the number of topics.

Figure 4 demonstrates that, both the precision and recall performances get improved as the number of topic factors increase. We could discover that the classification performance peaks close to the place where the perplexity reaches the minimum. And from the results on the Reuters data set (since the Bio-Literature data set is unbalanced), we can see that with a smaller number of topics the model behaves with nice recall performance; while with more topics, the precision performance improves fast. This is consistent with the foregoing perplexity result.

4.3. Comparison with other models

We employ *Naïve Bayes*, *KNN* and *SVM* models as the baseline methods to evaluate the capability of the *CoL* model. We choose *Naïve Bayes* because it is the simplest generative model with complete independence assumptions, and *KNN* model could exploit the correlation between classes among similar documents. These are the two basic notions in the *CoL* model. Besides, *SVM* model is one of the most powerful discriminative model for classification task [15]. All the baseline models are operating on the same feature set as the *CoL* model employs.

In *Naïve Bayes* model, we estimate the posterior probability of the classes in a given document by Eq(16). We use a pre-estimated threshold to retrieval the most probable classes.

$$p(c|\mathbf{w}) \propto \prod_n p(w_n|c)p(c) \tag{16}$$

In *KNN* model, we make the prediction by ranking the candidate classes in the union of the unlabeled sample's *k*-nearest labeled neighbors, and weight the candidate labels by the similarity between the desired unlabeled sample and its neighbors. This strategy is similar with the *ML-kNN* proposed by Zhang in [25].

In *SVM* model, we follow Boutell's strategy [20] to train a set of binary classifiers for each class and predict the unknown classes by the classifiers' output. We use SVM^{light} [24] toolkit to implement a linear kernel *SVM* model with the default parameters.

We first perform the comparison on different proportions of data used for training on the Bio-Literature data set. In this comparison, we set the size of topics in the *CoL* model to be 250 and *k* in *KNN* model to be 37.

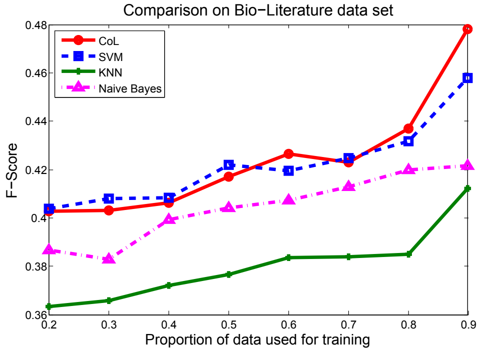

Figure 5. Comparisons with the baseline models on the Bio-Literature data set.

We could discover from the above results that, as the training set increases, the performance of the *CoL* model improves rapidly. The reason for this phenomenon is that in the *CoL* model, there are $C(C+1)+k(C+V)$ parameters to be estimated, when the training set is not large enough, most of the parameters cannot be fully estimated, and it directly hinders the capability of the model.

One thing we should note is that, since the Bio-Literature data set is unbalanced, we should attend the performance on the minor classes as well. In the class-level evaluation, the baseline models only retrieve most of the major classes (e.g. the top 5 methods) but ignoring the other minor ones, while the *CoL* model exhibits superior retrieve power. We demonstrate the coverage performance of each model on the same settings as in Figure 5 to compare their retrieve capability.

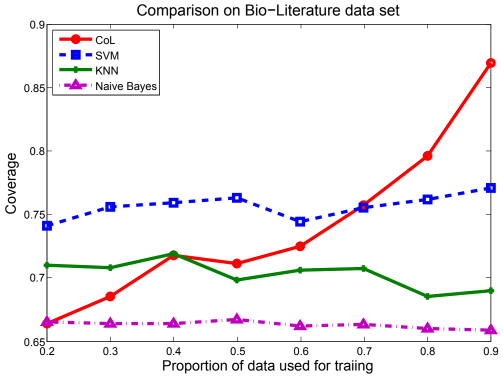

Figure 6. Coverage comparison with baseline models on the Bio-Literature data set.

Figure 6 demonstrates that the *CoL* model possesses better retrieval capability than all the baseline methods when the training set is large enough.

Next, we perform experiments on the well-studied Reuters data set and compare the result with two reported approaches on this data set [21, 19]. This time, we set the size of topics in the *CoL* model to be 300 and *k* in the *KNN* model to be 37.

Table 1. Classification performance on the Reuters data set (36 classes).

	Precision	*Recall*	*F-Score*
KNN	0.795	0.797	0.791
Naïve Bayes	0.751	0.892	0.803
SVM	0.878	0.814	0.848
CoL Model	0.872	0.875	**0.876**

The results on Reuters data set with top 36 classes are demonstrated in Table 1. We can see that the *CoL* model achieved the best *F-Score* performance and both its precision and recall performances are promising.

Table 2. CoL model performance on different class volume.

	Precision	*Recall*	*F-Score*
McCallum's EM [21]	0.839	-	-
Top 10 Classes	**0.901**	0.923	0.898
AdaBoost.MH [19]	-	-	0.851
All Classes	0.867	0.873	**0.866**

McCallum operated his mixture model on the ten largest classes and reported precision performance of 0.839. Schapire classified the original data set with all the classes and reported *F-Score* performance of 0.851. To compare with their achieved performances, we run the *CoL* model on the same training and testing data set as they did respectively. As a result, the *CoL* model achieves competitive performances, illustrated in Table 2.

From the detailed comparisons on these two data sets, we can discover the proposed *CoL* model possesses nice precision and comparative recall performance. We contribute the improvement to the information exploited from the correlation between different classes: the model captures the relationship between the classes from the training set and filters out the false positive combinations in the testing phrase.

4.4. Classes correlation analysis

With the *CoL* model, we formulate the correlation between different classes via the latent topic factors, which enable us to analyze the relationship between the classes in the latent space. Meanwhile, in the biological domain, there is a well-defined language describing the relationship among the biological concepts, named ontology and organized in a directed acyclic graph *(DAG)*. The Molecular Interaction (*MI*) ontology [4] is such a concept hierarchy in the molecular interaction domain, which includes the terms describing the molecular interaction types and the experiment detection methods.

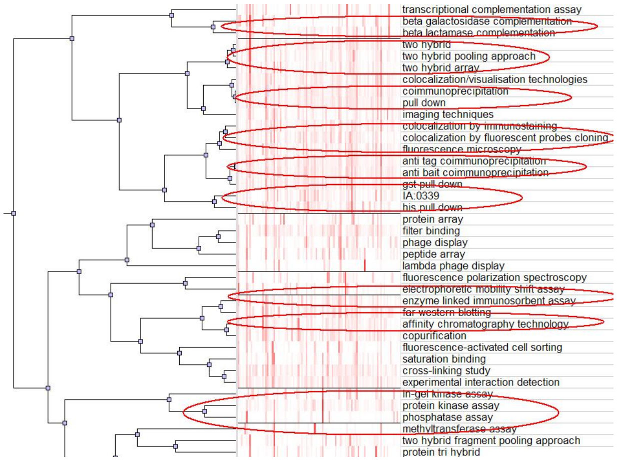
transcriptional complementation assay
beta galactosidase complementation
beta lactamase complementation
two hybrid
two hybrid pooling approach
two hybrid array
colocalization/visualisation technologies
coimmunoprecipitation
pull down
imaging techniques
colocalization by immunostaining
colocalization by fluorescent probes cloning
fluorescence microscopy
anti tag coimmunoprecipitation
anti bait coimmunoprecipitation
gst pull down
IA:0339
his pull down
protein array
filter binding
phage display
peptide array
lambda phage display
fluorescence polarization spectroscopy
electrophoretic mobility shift assay
enzyme linked immunosorbent assay
far western blotting
affinity chromatography technology
copurification
fluorescence-activated cell sorting
saturation binding
cross-linking study
experimental interaction detection
in-gel kinase assay
protein kinase assay
phosphatase assay
methyltransferase assay
two hybrid fragment pooling approach
protein tri hybrid

Figure 7. Detection methods clustering tree.

To represent the given detection methods in the latent topic space, we average the variational posterior Dirichlet parameters over all documents associating with method *i*:

$$r(c_i) = \frac{\sum_{d \in D} \gamma_{di}/Z}{\sum_{d \in D} 1} \tag{17}$$

where Z is a normalization factor to normalize the variational parameter γ in each document, D is the document

Table 3. Relevant terms for methods in the Bio-Literature data set

Method	*Terms*
x-ray	crystal, trypsin, residue, structure, interface, surface, enzyme, substrate, structural, helix, helical, conformation, strand, segment, protease
two hybrid	yeast, pp, record, two-hybrid, site, assay, fusion, acid, amino, contact, saccharomyces, screen, plasmid, pcr, mole
pull down	pp, pull-down, yeast, interact, fusion, wash, sequence, buffer, expression, resin, gst, window, transfect, luciferase, antibody
anti tag coip	pp, record, anti-flag, window, panel, strain, expression, sequence, yeast, stain, extract, sds, lysate, tumor, domain
coip	antibody, pp, record, extract, yeast, sequence, expression, cdna, clone, luciferase, growth, sirna, domain, mmedta, link

Table 4. Relevant terms for classes in the Reuters data set

Class	*Terms*
gas	gasoline, oil, crude, supply, contract, barrel, stock, price, sell, gas, rise, company, industry, energy, import
interest	pct, rate, bank, monetary, economist, market, money, prime, reserve, feed, inflation, deposit, bill, repurchase, federal
trade	trade, u.s, dlrs, deficit, japan, surplus, mln, export, japanese, currency, import, february, bill, gatt, market
ship	u.s, gulf, ship, iran, strike, attack, oil, union, port, vessel, seaman, tanker, tonne, missile, shipment
bop	mln, dlrs, trade, surplus, deficit, currency, export, pct, account, import, current, economic, quarter, growth, u.s

set associating with category c_i. Recall that, the variational parameter γ_i is approximate to the posterior topic distribution under category c_i in the given document. By averaging it over all the relevant documents, we can approximate the posterior distribution of classes over the latent topic factors.

Based on this approximate representation, we employ the accumulative clustering algorithm to perform hierarchical clustering and utilize a visualization tool *gCluto*[2] to demonstrate the captured "pedigree" tree. (We only illustrate part of the clustering result because of the page limit.)

From the clustering result in Figure 7, we can discover that most of the sibling nodes defined in the MI ontology are successfully clustered with the correct hierarchy and high confidence (red circles mean the correct clusters). The promising result confirms that the correlation between different classes exploited by the *CoL* model is reasonable and the model does capture the in-depth semantic relations.

4.5. Correlation between classes and words

It would be interesting to investigate the words posterior distribution under the given classes. Especially in the biological domain, particular terms and phrases convey crucial domain dependent information. To mine relevant words within a given class from the corpus, we utilize a class-specific distribution over words by the conditional distribution $p(w|c)$ to retrieval the most relevant terms under each desired class by the following evaluation function:

$$s(w|c) = \frac{\sum_{d \in D} \log p(w_d|c_d)}{\sum_{d \in D} 1} \tag{18}$$

where *D* is the document set associating with the desired class *c*.

We collect top 15 terms for 5 different methods from the Bio-Literature data set in Table 3 and top 15 terms for 5 different classes from the Reuters data set in Table 4. We can see from Table 3, most of the terms are appropriately gathered with the given classes. For example *"crystal"*, *"helix"*, *"structure"* are gathered to *x-ray*, and *"two-hybrid"*, *"yeast"*, *"site"* are gathered to *two hybrid*, which are the informative phrases in the *MI* ontology definition of these methods. And in Table 4, terms are also properly clustered to the desired classes. For example, *"gasoline"*, *"oil"*, *"energy"* are gathered to *gas*, and *"surplus"*, *"deficit"*, *"currency"* are gathered to *bop*(balance of payments). The reasonable word distribution under classes confirms that the *CoL* model captures the proper correlation not only between the different classes but also between classes and words.

5. Conclusions

In this paper, we propose a generative probabilistic model, the *CoL* model, to formulate the correlation between the different classes, and exploit the in-depth semantic relationship within the classes and word occurrences. By applying the model on various data sets, we achieved encouraging results comparing to the traditional classification algorithms. The experiment results confirm that it is necessary to model the correlation among different classes in the multi-label classification issue, and the proposed model properly modeled the latent correlations which benefit the classification performance.

One obvious distinction between the *CoL* model and the *LDA* model is that, the *CoL* model performs supervised learning while the *LDA* model is unsupervised. In this sense, the *CoL* model is capable to capture the information conveyed by the class labels while the *LDA* model fails to do so. Besides, because the *CoL* model assumes the topic proportion is governed by the classes the document belongs to, it can overcome the deficiency in the *LDA* model stemming from the strong independence assumption introduced by the Dirichlet distribution.

It is important to note that the *CoL* model is not narrowly restricted to the text classification task; instead, it could be feasibly applied to various application areas such as scene categorization, opinion mining and gene function annotation.

Our contributions in this paper lie in: 1) properly modeling the correlation among classes for the multi-label classification problem, which is ignored by most of previous approaches; 2) proposing a generative probabilistic model with proper underlying probabilistic semantics for the multi-labeling issue, which can be feasibly adopted to various applications.

6. Acknowledgements

This work was supported by the Chinese Natural Science Foundation under grant No. 60572084 and 60621062, as well as Tsinghua Basic Research Foundation under grant No. 052220205 and No. 053220002.

References

[1] Intact home: [http://www.ebi.ac.uk/intact/site/index.jsf].

[2] Matt rasmussen, gcluto home: [http://www-users.cs.umn.edu/ mrasmus/gcluto/index.shtml].

[3] Mint home: [http://mint.bio.uniroma2.it/mint/welcome.do].

[4] Molecular interaction ontology lookup service: [http://www.ebi.ac.uk/ontology-lookup/browse.do?ontname=mi].

[5] Pubmed home: [http://www.ncbi.nlm.nih.gov/pubmed/].

[6] J. Aitchision and S. M. SHEN. Logistic normal distributions: Some properties and uses. *Biometrika*, 67(2):261 – 272, March 1980.

[7] C. Andrieu, N. de Freitas, A. Doucet, and M. I. Jordan. An introduction to mcmc for machine learning. *Machine Learning*, 50(1-2):5 – 43, January 2003.

[8] H. Attias. A variational bayesian framework for graphical models. *Advances in Neural Information Processing Systems*, 12(1-2):209 – 215, December 2000.

[9] D. M. Blei and M. I. Jordan. Modeling annotated data. *SIGIR '03*, 70(2-3):127 – 134, March 2003.

[10] D. M. Blei and J. D. Laferty. A correlated topic model of science. *The Annals of Applied Statistics*, 1(1):17 – 35, April 2007.

[11] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *The Journal of Machine Learning Research*, 3(2-3):993 – 1022, March 2003.

[12] W. L. Buntine. Operations for learning with graphical models. *Journal of Artificial Intelligence Research*, 2:159 – 225, December 1994.

[13] K. M. A. Chai, H. L. Chieu, and H. T. Ng. Bayesian online classifiers for text classification and filtering. *SIGIR '02*, pages 97 – 104, 2002.

[14] T. Griffiths, M. Steyvers, D. Blei, and J. Tenenbaum. Integrating topics and syntax. *Advances in Neural Information Processing Systems*, 70(2-3):537 – 544, March 2005.

[15] T. Joachims. Text categorization with support vector machines: learning with many relevant features. *European Conference on Machine Learning (ECML)*, 1398(7):137–142, October 1998.

[16] D. D. Lewis. Reuters-21578 text categorization test collection distribution 1.0. 2004.

[17] F.-F. Li and P. Perona. A bayesian hierarchical model for learning natural scene categories. *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 524 – 531, 2005.

[18] W. Li and A. McCallum. Pachinko allocation: Dag-structured mixture models of topic correlations. *Proceedings of the 23 rd International Conference on Machine Learning*, pages 577 – 584, 2006.

[19] A. McCallum. Multi-label text classification with a mixture model trained by em. *AAAI99 Workshop on Text Learning*, 39(2-3):135–168, November 2004.

[20] M. R.Boutell, X. S. Jiebo Luo, and C. M.Brown. Learning multi-label scene classification. *Pattern Recognition*, 37(9):1757–1771, September 2004.

[21] R. Schapire and Y. Singer. Boostexter: a boosting-based system for text categorization. *Machine Learning*, 39(2-3):135–168, November 2004.

[22] M. Steyvers and T. Griffiths. Probabilistic topic models. *Latent Semantic Analysis: A Road to Meaning*, 2005.

[23] Y. W. Teh, M. I. Jordan, M. J. Beal, and D. M. Blei. Hierarchical dirichlet processes. *Jorunal of the American Statistical Association*, 476(101):1566 – 1581, 2006.

[24] J. Thorsten. *Learning to Classify Text Using Support Vector Machines*. Springer, HeidelbergGermany, 2002.

[25] M. Zhang and Z. Zhou. Ml-knn: A lazy learning approach to multi-label learning. *Pattern Recognition*, 40(7):2038 – 2048, July 2007.