J Intell Inf Syst (2016) 46:83–97
DOI 10.1007/s10844-014-0352-1

Labelset topic model for multi-label document classification

Ximing Li · Jihong Ouyang · Xiaotang Zhou

Received: 17 April 2014 / Revised: 8 December 2014 / Accepted: 10 December 2014 /
Published online: 20 December 2014

Abstract It has recently been suggested that assuming independence between labels is not suitable for real-world multi-label classification. To account for label dependencies, this paper proposes a supervised topic modeling algorithm, namely labelset topic model (LsTM). Our algorithm uses two labelset layers to capture label dependencies. LsTM offers two major advantages over existing supervised topic modeling algorithms: it is straightforward to interpret and it allows words to be assigned to combinations of labels, rather than a single label. We have performed extensive experiments on several well-known multi-label datasets. Experimental results indicate that the proposed model achieves performance on par with and often exceeding that of state-of-the-art methods both qualitatively and quantitatively.

Keywords Multi-label classification · Topic model · Labelset · Label dependency

1 Introduction

Traditional single-label classification involves instances annotated with a single label. However, many application domains commonly associate multiple labels to each instance. For example, in web page categorization (Ueda and Saito 2002; Kazawa et al. 2004; Ji et al. 2008), a web page can be assigned one or more labels; in image classification (Boutell et al. 2004; Wang et al. 2009), an image can simultaneously be tagged with multiple labels (e.g., "elephant" and "jungle"). Other popular multi-label

X. Li · J. Ouyang (✉) · X. Zhou
College of Computer Science and Technology, Jilin University, Changchun, China
e-mail: ouyj@jlu.edu.cn

X. Li
e-mail: liximing86@gmail.com

X. Li · J. Ouyang · X. Zhou
Key Laboratory of Symbolic Computation and Knowledge Engineering of Ministry of Education, Jilin University, Changchun, China

domains include music classification (Li and Ogihara 2006; Trohidis et al. 2008; Jiang and Ras 2013), video annotation (Qi et al. 2007), and direct marketing (Zhang et al. 2006). Multi-label classification for various domains is a prominent machine learning topic.

To the best of our knowledge, the existing methods for multi-label classification can be grouped into two categories: discriminative approaches (Tsoumakas and Katakis 2007; Yuret et al. 2008) and generative modeling approaches (Rubin et al. 2012). Recently, generative modeling approaches (i.e., supervised topic models) have received increasing attention because of their two advantages (Rubin et al. 2012): (1) predicting labels at the word-level, rather than at the document-level; (2) modeling all of the labels simultaneously, rather than handling each label independently.

A popular supervised topic model for multi-label classification is labeled latent Dirichlet allocation (L-LDA) (Ramage et al. 2009), which establishes a one-to-one correspondence between topics and observed labels. This model can, under certain conditions, achieve performance on par with the state-of-the-art support vector machines (SVMs). However, it ignores dependencies between different labels. In many applications, strong dependencies exist between labels. For example, a "computer" article is likely to cover "software" as well, but unlikely to cover "food" and "sport". Another example involves modeling the topics and authorship of documents: a particular author may focus only on certain topics; therefore it is essential to model them in a coordinated fashion (Ji et al. 2008).

To modify L-LDA to consider label dependencies, we develop a novel supervised topic model for multi-label document classification: labelset topic model (LsTM). In this work, we use the labelset concept described in Boutell et al. (2004) to capture label dependencies. Let W be a multi-label corpus with a set of disjoint labels Y. The labelset is defined as a subset of Y (i.e., any arbitrary combination of labels). In LsTM, documents are represented by distributions over labelsets rather than individual labels. The structure of LsTM is a four-level hierarchy that consists of a document layer, two labelset layers, and a word layer. Adjacent layers are connected. The upper labelsets are obtained by splitting the initial set of labels Y following the intuition that each label is strongly correlated with a few labels in Y. The lower labelsets are subsets of the upper labelsets that exist in the training data and are used to describe the dependencies between labels within each upper labelset. We have conducted extensive experiments on several commonly used multi-label collections. The empirical results demonstrate that the proposed model achieves competitive performance with state-of-the-art approaches both qualitatively and quantitatively.

The remainder of this paper is organized as follows. We discuss related publications in Section 2. Section 3 presents the proposed LsTM model. In Section 4, the experimental results on several well-known multi-label collections are presented. Conclusions and future directions are discussed in Section 5.

2 Related work

First, we formalize our notation with respect to multi-label document classification. Let $W = \{(w_d, y_d) | d = 1, 2, \cdots, D\}$ be a multi-label corpus and $Y = \{1, 2, \cdots, C\}$ be the finite set of labels, where w_d and $y_d \subseteq Y$ are the word vector and the set of labels for document d, respectively.

2.1 Discriminative approach

Discriminative methods for multi-label learning can be further divided into two classes: algorithm adaptation (AA) and problem transformation (PT). Here, we provide an overview of these two types of discriminative approaches.

AA-based methods modify the existing single-label classification methods to extend to multi-label data. In Clare and King (2001), the decision tree (specifically the C4.5 algorithm) is extended to multi-label cases. The simple but powerful k Nearest Neighbors (kNN) algorithm is another popular method for handling multi-label cases (Zhang and Zhou 2007; Brinker and Hullermeier 2007). Multi-label kNN algorithms first calculate the k nearest neighbors, then aggregate the label sets using various schemes. SVMs and neural networks have also been explored for multi-label learning. For example, Elisseeff (2002) proposes an SVM-based method minimizing the ranking loss, Zhang and Zhou (2006) describes a modification of the back-propagation approach (BP-MLL), and Zhang (2009) proposes an extension of radial basis function networks.

PT-based methods transform a multi-label classification task into dozens of single-label tasks. Binary relevance (BR) (Tsoumakas and Katakis 2007) is a popular PT-based method that separately trains C binary classifiers, where each classifier corresponds to a label within Y. Another popular PT-based method is the label powerset (LP) (Boutell et al. 2004), initially used for multi-label image classification. LP defines each subset of Y (labelset) from the new single-class labels in the training data and learns a binary classifier for each labelset. In contrast to BR, LP uses labelsets to capture label dependencies. However, it is time-consuming for collections with many labels and suffers from severely skewed learning problems when given few training instances. To address these problems, a modification of LP, RAndom k labELsets (RAkLE), is proposed in Tsoumakas et al. (2011). The RAkL algorithm performs LP over m iterations on different labelsets of size k. It can harmonize the number of existing labelsets by adjusting parameters k and m. Two other PT-based methods which consider label dependencies include a model based on cyclic-directed graphs (Guo and Gu 2011) and the classifier chains (CC) method (Read et al. 2011).

In addition, hierarchical classification is often linked with label dependency, where lower the nodes in the tree implies stronger dependency level of its labels. The representative works include (Jiang and Ras 2013; Fan et al. 2007).

2.2 Generative modeling approach

Because of the popularity of topic modeling approaches such as probabilistic latent semantic indexing (PLSI) (Hofmann 1999) and latent Dirichlet allocation (LDA) (Blei et al. 2003), a number of supervised topic models have been investigated for multi-label classification in recent years.

L-LDA (Ramage et al. 2009) adapts the unsupervised LDA model to multi-label learning such that topics have a one-to-one correspondence with labels and each document d is restricted to its set of labels y_d. More recently, Kim et al. (2012) proposed a Dirichlet process with mixed random measures (DP-MRM) to further extend L-LDA using nonparametric methods. Although the empirical results indicate that under certain conditions L-LDA and DP-MRM can achieve competitive performance with the state-of-the-art approaches, they use an independent assumption among labels, which is inadequate for many real world examples. Many publications focus on relaxing this assumption and consider label dependencies by: (1) projecting labels onto some latent space, e.g., Dependency-LDA (Rubin et al. 2012), which introduces a set of corpus-wide topic distributions over labels, and

partially LDA (PLDA) (Ramage et al. 2011), where each label is represented by a number of topics; (2) constructing a hierarchy of labels, e.g., TreeLaD (Nguyen et al. 2013), which is based on a tree-structured topic hierarchy.

However, the existing supervised topic models that consider label dependencies have two disadvantages: (1) some methods introduce an extra topic layer in addition to the label layer, but topics can be obscure and difficult to interpret; (2) some assume that each word corresponds to a single label; however, in multi-label settings, words may be associated with one or more labels. In this paper, we develop a supervised multi-label topic model that captures label dependencies. The proposed model, called LsTM, learns the label co-occurrence relationships using the labelset concept defined in LP (Boutell et al. 2004). In contrast to existing methods, LsTM replaces the label layer with the labelset layers. The labelsets are observed and straightforward. Moreover, LsTM allows a word to correspond to a combination of labels since some of the labelsets represent distributions over words. Thus, LsTM overcomes the two aforementioned disadvantages of existing methods. The evaluation results demonstrate the effectiveness of LsTM (see Section 4).

Several unsupervised topic models considering topic dependencies have been investigated during the past decade. The correlated topic model (CTM) (Blei and Lafferty 2007) algorithm represents topic correlation using the logistic normal distribution as prior. Several models, such as the pachinko allocation model (PAM) (Li and McCallum 2006) and hierarchical Dirichlet processes (HDP) (Teh et al. 2006), include an extra layer for topic correlation. Since these models are unsupervised, they are not directly applicable to multi-label classification. Furthermore, in contrast to LsTM, none of them allow a word to correspond to a combination of topics.

3 Proposed model

This section presents the LsTM approach and introduces the model training and testing procedures.

3.1 Labelset topic model

The traditional L-LDA model has no mechanism to capture label dependencies. Intuitively, some word tokens may simultaneously correspond to several dependent labels. For example, the word “code” may correspond to both a “computer” and “software” label. L-LDA assigns each word token to a single label. This constraint can degrade classification performance.

To address this problem, we extend L-LDA to develop LsTM, a novel supervised topic model. Our algorithm uses labelsets to form new class labels denoting combinations of labels. Since the labelsets group dependent labels, they can help capture label interdependencies. We organize LsTM as a four-level hierarchy that consists of a document layer, two labelset layers, and a word layer, where the adjacent layers are connected. We form two labelset layers to both filter and reduce the number of labelsets. We call the labelsets at the upper layer super-labelsets, and the ones at the lower layer sub-labelsets. The super-labelsets are subsets of the finite set of labels Y. They are used to form clusters of K different labels under the assumption that each label is strongly correlated with a few others. The subsets of the super-labelsets that exist in the training data are defined as sub-labelsets (i.e., the co-occurrence combinations of labels). The sub-labelsets layer

connects directly to the words layer and represents the label dependencies within each super-labelset.

Let M be the number of K-sized super-labelsets $\mathrm{A}^{(g)}$, and S be the total number of sub-labelsets $\mathrm{A}^{(l)}$. Each super-labelset $\mathrm{A}_m^{(g)}$ corresponds to an l_m-dimension multinomial distribution $\phi_m^{(l)}$ over its own sub-labelsets, drawn from the Dirichlet prior η. To allow supervised learning with respect to labelsets, each document d is described only by the super-labelsets included in y_d, and these super-labelsets are represented only by their sub-labelsets included in y_d. For simplicity, we use the term "y_d-related" to denote the corresponding super-labelsets and sub-labelsets for document d. For example, suppose that corpus W has five labels (numbered 1 to 5) three super-labelsets, $\mathrm{A}_1^{(g)} = \{1, 2, 3\}$, $\mathrm{A}_2^{(g)} = \{2, 3, 4\}$, and $\mathrm{A}_3^{(g)} = \{3, 4, 5\}$, and eight sub-labelsets, $\mathrm{A}_1^{(l)} = \{1\}$, $\mathrm{A}_2^{(l)} = \{2\}$, $\mathrm{A}_3^{(l)} = \{3\}$, $\mathrm{A}_4^{(l)} = \{4\}$, $\mathrm{A}_5^{(l)} = \{5\}$, $\mathrm{A}_6^{(l)} = \{1, 2\}$, $\mathrm{A}_7^{(l)} = \{2, 4\}$, and $\mathrm{A}_8^{(l)} = \{3, 5\}$. A document d with $y_d = \{1, 2\}$, would be restricted to $\mathrm{A}_1^{(g)}$ and $\mathrm{A}_2^{(g)}$; $\mathrm{A}_1^{(g)}$ would be constrained to $\mathrm{A}_1^{(l)}$, $\mathrm{A}_2^{(l)}$ and $\mathrm{A}_6^{(l)}$; $\mathrm{A}_2^{(g)}$ would be constrained to $\mathrm{A}_2^{(l)}$. Thus, for document d, $\mathrm{A}_1^{(g)}$ and $\mathrm{A}_2^{(g)}$ are the y_d-related super-labelsets (i.e., $y_d^{(g)} = \left\{\mathrm{A}_1^{(g)}, \mathrm{A}_2^{(g)}\right\}$); $\mathrm{A}_1^{(l)}$, $\mathrm{A}_2^{(l)}$ and $\mathrm{A}_6^{(l)}$ are the y_d-related sub-labelsets for $\mathrm{A}_1^{(g)}$ (i.e., $y_{d,1}^{(l)} = \left\{\mathrm{A}_1^{(l)}, \mathrm{A}_2^{(l)}, \mathrm{A}_6^{(l)}\right\}$); $\mathrm{A}_2^{(l)}$ is the y_d-related sub-labelset for $\mathrm{A}_2^{(l)}$ (i.e., $y_{d,2}^{(l)} = \left\{\mathrm{A}_2^{(l)}\right\}$).

LsTM is depicted using graphical model notation in Fig. 1. Step 1 draws the multinomial distribution $\phi_m^{(l)}$ over sub-labelsets for each super-labelset $\mathrm{A}_m^{(g)}$, from Dirichlet prior η. Step 2 draws the multinomial distribution $\phi_s^{(w)}$ over words for each sub-labelset $\mathrm{A}_s^{(l)}$, from Dirichlet prior β. Step 3 is the word generation process as follows: For each document d with y_d, generate the multinomial distribution θ_d over its y_d-related super-labelsets, from Dirichlet prior α, and repeat the following process N_d times for N_d words: first sample a y_d-related super-labelset $z_{d,n}^{(g)}$ from distribution θ_d; then sample a y_d-related sub-labelset $z_{d,n}^{(l)}$ from distribution $\phi_{z_{d,n}^{(g)}}^{(l)}$; finally sample a word $w_{d,n}$ from distribution $\phi_{z_{d,n}^{(l)}}^{(w)}$.

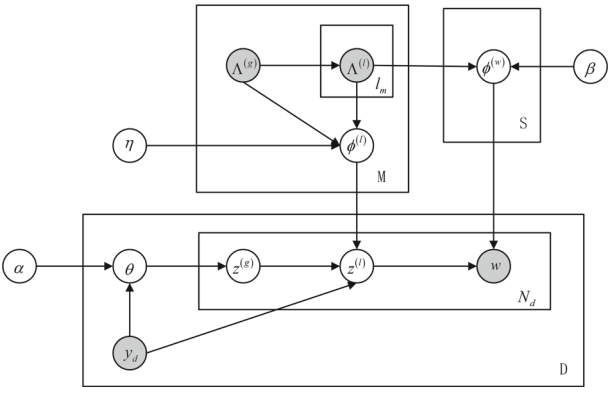

Fig. 1 Graphical models for LsTM

The generative process of LsTM is summarized as:

1. For each super-labelset $A_m^{(g)}$
   a. Choose $\phi_m^{(l)} \sim Dirichlet\,(\eta)$
2. For each sub-labelset $A_s^{(l)}$
   a. Choose $\phi_s^{(w)} \sim Dirichlet\,(\beta)$
3. For each document d with y_d
   a. Choose a distribution over y_d-related super-labelsets $\theta_d \sim Dirichlet\,(\alpha)$
   b. For each of the N_d words $w_{d,n}$
      i. Choose a y_d-related super-labelset $z_{d,n}^{(g)} \sim Multinomial\,(\theta_d)$
      ii. Choose a y_d-related sub-labelset $z_{d,n}^{(l)} \sim Multinomial\left(\phi_{z_{d,n}^{(g)}}^{(l)}\right)$
      iii. Choose a word $w_{d,n} \sim Multinomial\left(\phi_{z_{d,n}^{(l)}}^{(w)}\right)$

Note that this model allows each word token to be assigned either a single label or a combination of labels (e.g., $A_1^{(l)} = \{1\}$ or $A_6^{(l)} = \{1, 2\}$ in the aforementioned example). Hence, LsTM represents the label dependencies at the word-level.

3.2 Model training

During training time, we randomly initialize M different K-sized super-labelsets. Because the labels in the training data are observed, the sub-labelsets are also known. Thereby training LsTM only requires estimating: (1) the M l_m-dimensional multinomial distributions $\phi^{(l)}$ of the super-labelsets over sub-labelsets; (2) the S V-dimensional multinomial distributions $\phi^{(w)}$ of the sub-labelsets over words.

Because the posterior distribution of hidden variables is computationally intractable, we perform approximate estimation using Gibbs sampling (Griffiths and Steyvers 2004), a Markov chain Monte Carlo (MCMC) algorithm. To use this sampling in LsTM, we sequentially update the latent assignments $z_{d,n}^{(g)}$ and $z_{d,n}^{(l)}$ for all words in the training data. The update rule of Gibbs sampling is given as follows:

$$P\left(z_{d,n}^{(g)} = m, z_{d,n}^{(l)} = s | W, z_{d,-n}^{(g)}, z_{d,-n}^{(l)}, \alpha, \eta, \beta\right) \propto \frac{N_{-n}^{m/d} + \alpha}{N_{-n}^{d} + \left|y_d^{(g)}\right| \alpha} \times \frac{N_{-n}^{s/m} + \eta}{N_{-n}^{m} + l_m \eta} \times \frac{N_{-n}^{w/s} + \beta}{N_{-n}^{s} + V\beta} \tag{1}$$

where $N^{m/d}$ and N^d are the number of times that super-labelset m has occurred in document d and the total number of words in document d, respectively; $N^{s/m}$ and N^m are the number of times that sub-labelset s has assigned to super-labelset m and the total number of sub-labelsets tagged with super-labelset m, respectively; $N^{w/s}$ and N^s are the number of times that the position n's corresponding word w has been assigned to sub-labelset s and the total number of words under sub-labelset s, respectively. The subscript "-n" denotes a quantity except for the token in position n.

With the above samples, the model parameters $\phi^{(l)}$ and $\phi^{(w)}$ can be computed as follows:

$$\phi^{(l)}_{m,s} = \frac{N^{s/m} + \eta}{N^m + l_m \eta} \tag{2}$$

$$\phi^{(w)}_{s,w} = \frac{N^{w/s} + \beta}{N^s + V\beta} \tag{3}$$

3.3 Inference for test documents

During testing (when the labels of documents are unobserved), each document may sample from any of the M super-labelsets, and each super-labelset $\mathrm{A}^{(g)}_m$ can sample from any of its l_m sub-labelsets.

We estimate the test documents one by one. Given the optimal distributions $\phi^{(l)}$ and $\phi^{(w)}$ obtained in the training procedure, the Gibbs sampling equation for a test document d' is:

$$P\left(z^{(g)}_{d',n} = m, z^{(l)}_{d',n} = s | d', z^{(g)}_{-n}, z^{(l)}_{-n}, \alpha, \phi^{(l)}, \phi^{(w)}\right) \propto \frac{N^{m/d'}_{-n} + \alpha}{N^{d'}_{-n} + M\alpha} \times \phi^{(l)}_{m,s} \times \phi^{(w)}_{s,w} \tag{4}$$

where $z^{(g)}_{d',n}$ and $z^{(l)}_{d',n}$ are super-labelset and sub-labelset assignments for the nth word in document d', respectively; $N^{m/d'}$ and $N^{d'}$ are the number of times that super-labelset m has occurred and the number of words in document d', respectively.

With the sample results, the posterior distribution $\theta_{d'}$ over super-labelsets for document d' can be estimated as:

$$\theta_{d',m} = \frac{N^{m/d'} + \alpha}{N^{d'} + M\alpha} \tag{5}$$

For prediction, the final C-dimensional labels proportion $\varphi_{d'}$ of the test document d' is calculated as follows:

$$\varphi_{d',c} = \sum_{c \in \mathrm{A}^{(g)}_m}^{m=1,\cdots M} \sum_{c \in \mathrm{A}^{(l)}_{m,s}}^{s=1,\cdots l_m} \frac{1}{\left|\mathrm{A}^{(l)}_{m,s}\right|} \theta_{d',m} \phi^{(l)}_{m,id(m,s)} \tag{6}$$

where $id(m, s)$ is the identifier of the s-th sub-labelset in super-labelset $\mathrm{A}^{(g)}_m$.

3.4 Comparison with related algorithms

Here, we compare LsTM with several related algorithms.

3.4.1 Related discriminative approaches

The LP (Boutell et al. 2004) and RAkLE (Tsoumakas et al. 2011) approaches also use labelsets to capture label dependencies. In particular, RAkLE performs the same process as LsTM to generate the two levels of labelsets. However these two discriminative approaches consider the labelsets at the document-level. When training the binary classifier for each labelset, all the words of the positive samples are assigned to this labelset. In contrast, LsTM applies the labelsets at the word-level. This provides more flexibility because labelsets can be assigned to each word instead of each document.

3.4.2 Related generative modeling approaches

To predict multi-label corpora, supervised topic models such as L-LDA (Ramage et al. 2009) and Dependency-LDA (Rubin et al. 2012) connects observed labels and words using the label-word distributions. The state-of-the-art Dependency-LDA introduces an unobserved topic layer beyond the label layer to represent label dependencies. In our work, LsTM incorporates the label supervision capturing label dependencies simultaneously when forming the labelset-word distributions. Furthermore, it introduces a new kind of labelset (i.e., super-labelset) to cluster several strongly correlated labels together. In contrast to existing algorithms, the two labelset levels are observed and easily interpreted and the words can be assigned to a combination of labels.

4 Experiments

In this section, we evaluate LsTM qualitatively and quantitatively.

4.1 Experimental setting

4.1.1 Collections

The experiments were performed on six commonly used multi-label collections,[1] including two Yahoo! subdirectory datasets (i.e., Arts and Health), enron, rcv1subset1, bibtex and a random subset of bookmarks. These collections contain dozens or hundreds of labels, and they are all skewed in various degrees. Table 1 summarizes some basic statistics, such as the number of documents, number of unique words and labels, *cardinality* (i.e., the average number of labels per document) and *MaxL/MinL* (i.e., the maximum/minimum number of documents for each label).

4.1.2 Metric

In the experiments, we used two popular performance metrics: Micro-F1 and Macro-F1. For both of them, the larger value implies better performance.

The F1 metric is the harmonic mean of *Precision* and *Recall*. Given the number of true positives (TP), false positives (FP) and false negatives (FN), the F1 metric is given:

$$F1 = \frac{2 \times TP}{2 \times TP + FP + FN}$$

According to descriptions in Tsoumakas et al. (2011), Micro-F1 and Macro-F1 are the micro-averaged and macro-averaged versions with respect to the F1 metric, respectively. They are used to measure the binary prediction performance across labels. Let TP_c, FP_c and FN_c be the number of true positives, false positives and false negatives after binary evaluations for a label c. We can compute Micro-F1 and Macro-F1 metrics as follows:

$$Micro{-}F1 = \frac{2 \times \sum_{c=1}^{C} TP_c}{2 \times \sum_{c=1}^{C} TP_c + \sum_{c=1}^{C} FP_c + \sum_{c=1}^{C} FN_c}$$

[1] http://mlkd.csd.auth.gr/multilabel.html

Table 1 Statistics of the selected collections, where "Card" is the shortening of *cardinality*

Dataset	*D*	V	*C*	Card	MaxL	MinL
Arts	7484	23416	26	1.6	1838	1
Health	9205	30605	32	1.6	4703	1
enron	1694	1001	53	3.4	913	1
rcv1subset1	6000	47236	101	2.9	882	1
bibtex	7395	1836	159	2.4	1024	51
bookmarks	10000	2150	208	2.0	565	19

$$Macro{-}F1 = \frac{1}{C}\sum\nolimits_{c=1}^{C} \frac{2 \times TP_c}{2 \times TP_c + FP_c + FN_c}$$

4.2 Text modeling

We qualitatively evaluated LsTM with respect to its text modeling performance. Table 2 shows some examples of the top 10 frequent words for single labels and label combinations across the bookmarks dataset. Intuitively, words listed for each label combination are highly associated with each individual label in the set.

4.3 Classification performance

This section exhibits the quantitative results for multi-label classification. For each dataset, we randomly selected 33 % of the documents as the training data and used the remaining documents as the test data. We repeat this process 10 times to obtain 10 training/test splits.

4.3.1 Evaluation of LsTM

We conduct a number of experiments to investigate the parameters of LsTM, including the Dirichlet priors α, η, and β, the size of the super-labelsets K, the number of super-labelsets M, and the settings of the Gibbs sampler. The evaluation results indicate that LsTM is relatively sensitive to parameters M and K over different datasets. Thus, we report the results with respect to these two significant parameters. In the experiments presented in this paper the remaining parameters are set as follows: $\alpha = 50/C$, $\eta = 1$, and $\beta = 0.01$. To train the model, we ran 10 independent MCMC chains, and used the end samples of the MCMC chains after a burn-in of 500 iterations. We averaged the 10 selected samples to estimate the

Table 2 The top ten frequent words over single labels and their combinations across bookmarks dataset

Label	The most frequent words
ajax	ajax; javascript; browser; web; css; internet; document; html; open; project
software	software; download; support; home; version; fax; free; source; site; open
ajax&software	software; source; ajax; version; offer; project; document; open; money; web
publications	vote; understand; offer; format; improve; zur; die; mix; language; side
research	information; research; home; contact; search; based; page; web; related; data
publications&research	research; information; journal; data; language; improve; viewer; truth; page; web

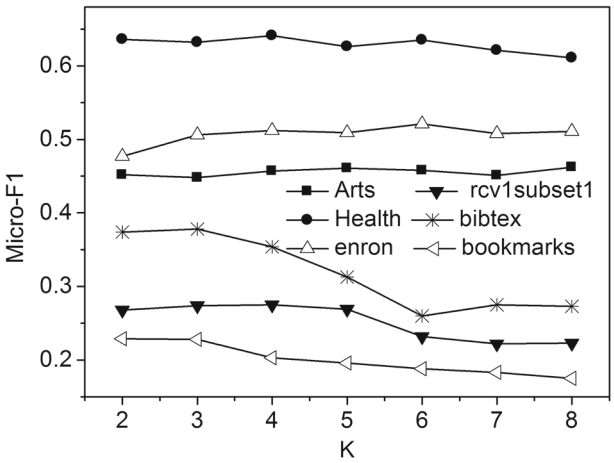

Fig. 2 Micro-F1 scores of LsTM with respect to parameter K

distributions $\phi^{(l)}$ and $\phi^{(w)}$ according to Eqs. (2) and (3). During the testing phase, we ran 5 independent MCMC chains with 200 iterations for each unseen document. We allowed a burn-in of 150 iterations and a lag of 10 iterations between samples for each MCMC chain. We averaged the resulting 25 samples to predict the test documents according to Eq. (6).

We fixed $M = 5C/3$ and used the Micro-F1 metric to evaluate the parameter K with different values (from value 2 to 8). As shown in Fig. 2, we observe that the effect on performance varies between the different datasets. The three datasets that contain a smaller number of labels C (i.e., Arts, Health, and enron) achieve very steady Micro-F1 scores. However, the performance for the other datasets with a relatively larger number of labels C decreases as K increases. In practice, a larger K leads to more sub-labelsets, yet many of these are very rare, especially for datasets with larger C. Moreover, we observe that higher scores are achieved when K is similar to the *cardinality* of each dataset. This setting can keep the number of rare sub-labelsets under control. In practice, this is an effective guide to set parameter K.

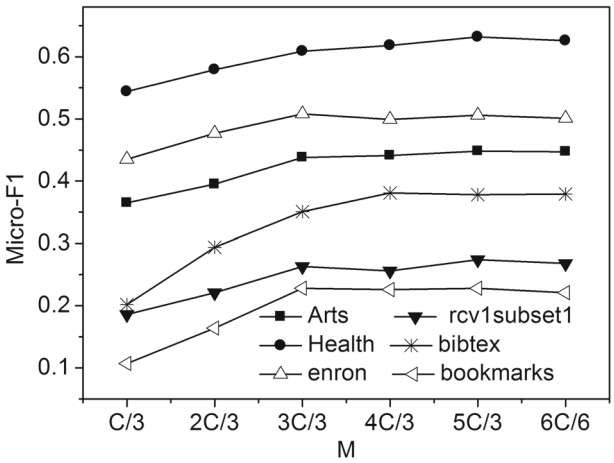

Fig. 3 Micro-F1 scores of LsTM with respect to parameter M

Table 3 Fix $M = 5C/3$, so the value of S with respect to different K across Arts and bookmarks

Dataset	3	4	5	6	7	8
Arts	101	197	238	349	463	504
bookmarks	404	581	781	1028	1270	1622

We fixed $K = 3$ and again used the Micro-F1 metric to study the influence of parameter M. The experimental results are shown in Fig. 3. We observe that the Micro-F1 scores for all of the datasets increased as the number of super-labelset M increased. We contend that this is because a low M value can generate few sub-labelsets, losing many frequent co-occurrence labelsets in the training data. We also observe that in most cases the performance stabilized when $M \in [C, 2C]$.

We acknowledge that we need to balance between conflicting recommendations when combining these two parameters. On one hand, we suggest a smaller K to obtain a smaller S; on the other hand we suggest a larger M to obtain a larger S. In fact, we hope to obtain more frequently occurring sub-labelsets while reducing the number of rare sub-labelsets. With the same random initial seed, some examples of the total number of sub-labelsets S with respect to different K and M values are shown in Tables 3 and 4. As K increases, the value of S becomes unmanageably large since plenty of rare sub-labelsets exist. With small M values, the value of S approaches the number of labels C, leading to a loss of some important sub-labelsets.

In practice, we recommend setting parameters K and M as follows: (1) K equal to the ceiling of the cardinality; (2) M between C and $2C$.

4.3.2 Comparison with other methods

In this section we study the performance of LsTM for multi-label classification. We set K to the ceiling of the *cardinality* of each dataset (e.g., value 2 for Arts and value 4 for enron), and set M to $5C/3$ as discussed above. The other parameters are the same as in the parameter experiments in Section 4.3.1.

We selected several existing approaches as the performance baselines: an AA-based discriminative approach, i.e., ML*k*NN (Zhang and Zhou 2007); two PT-based discriminative algorithms without and with label dependency considerations, i.e., SVMs (Lewis et al. 2004) and RA*k*LE (Tsoumakas et al. 2011); and two supervised topic models without and with label dependency considerations, i.e., L-LDA (Ramage et al. 2009) and Dependency-LDA (Rubin et al. 2012). For the three discriminative approaches, the documents were encoded using the normalized TF-IDF representation. The settings of all of these baseline methods were as follows:

- ML*k*NN is a multi-label version of *k*NN. As suggested in Zhang and Zhou (2007), the number of neighbors was set to 10 and the smoothing factor was set to 1.
- For SVMs, we used the well-known LibSVM tool,[2] and tuned the parameters using a linear search over the set $\{10^i | i = -5, -4, \cdots, 4, 5\}$.

[2] http://www.csie.ntu.edu.tw/~cjlin/libsvm/

Table 4 Fix K =3, so the value of S with respect to different M acorss Arts and bookmarks

Dataset	$C/3$	$2C/3$	C	$4C/3$	$5C/3$	$2C$
Arts	47	55	83	94	101	113
bookmarks	283	305	358	373	404	452

– For RAkLE, we used publicly available code in the Mulan tool (Tsoumakas et al. 2011), a library designed for multi-label learning. As recommended in Tsoumakas et al. (2011), the size of the labelsets was set to 3 and the number of labelsets was set to $2C$, and the C4.5 decision tree was used as the base level algorithm.
– We implemented an in-house code for L-LDA. Following reported experience, the document-label Dirichlet prior was set to $50/C$ and the label-word Dirichlet prior was set to 0.01.
– We implemented an in-house implementation (using the fast inference scheme) for Dependency-LDA. All of the parameters were set and tuned according to the suggestions in Rubin et al. (2012).

Tables 5 and 6 exhibit the mean and standard deviation of the Micro-F1 and Macro-F1 metrics for all of the algorithms. For Micro-F1, different datasets show significantly different results. We observe that the datasets with fewer labels always show better performance than the datasets with more labels (e.g., Arts, Health and enron are about 40 %∼60 %, but rcv1subset1, bibtex and bookmarks are only about 20 %∼30 %). Among the three datasets with few labels, Health achieves the highest Micro-F1 level of 60 % since it contains many categories that are both dominant and characteristic .The enron dataset has mostly higher scores than the Arts dataset and we believe this is because enron has a cleaner vocabulary. Among the three datasets with more labels, the balanced dataset bibtex (the gap between its *MaxL* and *MinL* is relatively small) performs better than the other two. Macro-F1 focuses on the performance of each label, so its variation among different datasets is much smaller than Micro-F1. The datasets with both fewer labels and lower *cardinality* values, i.e., Arts and Health, achieve 20 %∼30 %. The balanced dataset bibtex also achieves a relatively high level of 22 %∼25 %.

Compared with the other algorithms, we observe that LsTM almost always achieves the best performance across these distinct datasets. These results indicate the robustness of LsTM. We provide more detailed comparisons below.

We begin by comparing LsTM with the three discriminative approaches (i.e., MLkNN, SVMs, and RAkLE). LsTM is unquestionably better than MLkNN across all six datasets for

Table 5 Experiment results in terms of Micro-F1 measurement

Dataset	LsTM	MLkNN	SVMs	RAkLE	L-LDA	Dep-LDA
Arts	**.448±.007**	.046±.009	.395±.008	.376±.012	.378±.014	.401±.009
Health	**.632±.009**	.361±.016	.621±.011	.617±.017	.586±.026	.614±.008
enron	**.506±.013**	.263±.028	.433±.015	.467±.011	.403±.022	.442±.025
rcv1subset1	**.274±.003**	.174±.009	.245±.002	.237±.005	.212±.011	.246±.004
bibtex	.378±.008	.188±.013	.371±.009	**.379±.008**	.358±.011	.369±.009
bookmarks	**.228±.007**	.159±.027	.202±.006	.199±.013	.189±.014	.192±.009

The bold values mean the best performance

Table 6 Experiment results in terms of Macro-F1 measurement

Dataset	LsTM	ML*k*NN	SVMs	RA*k*LE	L-LDA	Dep-LDA
Arts	**.302±.008**	.017±.003	.211±.008	.214±.011	.254±.009	.258±.007
Health	**.285±.006**	.065±.005	.263±.007	.253±.015	.247±.019	.269±.006
enron	**.124±.005**	.041±.003	.116±.007	**.124±.008**	.092±.013	.113±.009
rcv1subset1	**.141±.013**	.084±.007	.138±.009	.139±.006	.127±.015	.129±.012
bibtex	**.265±.006**	.047±.004	.241±.005	.241±.012	.222±.006	.243±.005
bookmarks	**.136±.005**	.024±.002	.072±.003	.061±.003	.092±.009	.109±.007

The bold values mean the best performance

both Micro-F1 and Macro-F1. This is because the simpler ML*k*NN algorithm is unsuited for these high-dimensional text document collections. Compared with the two state-of-the-art SVMs and RA*k*LE, LsTM achieves better performance for almost everything except the Micro-F1 score for the bibtex dataset. With respect to Macro-F1, LsTM clearly outperforms the others for the Arts, Health, and bookmarks datasets. We believe that this is due to the relatively low *cardinality* values of these three datasets. In this case, it is easy to assign preferable labelsets to word tokens during model training. It is interesting to observe that SVMs slightly outperform RA*k*LE despite ignoring label dependencies. This is mainly because the SVMs perform the additional parameter searching processes.

We now compare LsTM with the two supervised topic models (i.e., L-LDA and Dependency-LDA). Overall, LsTM outperforms the two models across all six datasets for both Micro-F1 and Macro-F1 metrics. Compared with L-LDA, LsTM achieves a 2 %∼10 % improvement in terms of Micro-F1, and a 2 %∼5 % improvement in terms of Macro-F1. The main difference between L-LDA and LsTM lies in allowing word tokens to be assigned to labelsets. Our empirical results show that LsTM successfully incorporates label dependency knowledge into L-LDA using labelsets. Compared with Dependency-LDA, LsTM improves the performance by 1 %∼5 % on both F1 measures. For Micro-F1, LsTM also performs particularly well with datasets with lower *cardinality* values (i.e., Arts, Health and bookmarks). This further underscores that assigning preferable labelsets to word tokens significantly boosts the final classification performance. Overall, based on our evaluation results, we conclude that our method to capture the label dependencies is more effective than using Dependency-LDA.

5 Conclusion

In this paper, we relax the label independence assumption for multi-label document classification. To achieve this, we propose an extension of L-LDA, namely LsTM, which uses the concept of labelsets described in Boutell et al. (2004) to capture label dependencies. LsTM uses two observed labelset layers: the super-labelset and the sub-labelset. The super-labelsets group several related labels and the sub-labelsets assign combinations of these labels to each word. Compared with existing supervised topic modeling algorithms, LsTM is more straightforward and effective. We perform empirical evaluations on six well-known multi-label collections. Experimental results indicate that LsTM achieves competitive performance with both the state-of-the-art discriminative approaches and the supervised topic models.

In the future, we will focus on adaptively determining the two significant parameters (i.e., K and M). Another potential research direction is to apply LsTM to large-scale multi-label data.

Acknowledgments This work was supported by National Nature Science Foundation of China (NSFC) under the Grant No. 61170092, 61133011, and 61103091.

References

Blei, D.M., Ng, A.Y., Jordan, M.I. (2003). Latent Dirichlet allocation. *Journal of Machine Learning Research*, 993–1022.

Blei, D.M., & Lafferty, J.D. (2007). A correlated topic model fo science. *The Annals of Applied Statistics*, 17–35.

Boutell, M.R., Luo, J., Shen, X., Browna, C.M. (2004). Learning multi-label scene classification. *Pattern Recognition*, 1757–1771.

Brinker, K., & Hullermeier, E. (2007). Case-based multilabel ranking. In *International joint conference on artificial intelligence* (pp. 702–707).

Clare, A., & King, R.D. (2001). Knowledge discovery in multi-label phenotype data. *Principles of Data Mining and Dnowledge Discovery*, 42–53.

Elisseeff, A. (2002). JasonWeston: a kernel method for multi-labelled classification. In *Neural information processing systems*.

Fan, J., Gao, Y., Luo, H. (2007). Hierarchical classifcation for automatic image annotation. In *International ACM SIGIR conference on research and development in information retrieval* (pp. 111–118).

Griffiths, T.L., & Steyvers, M. (2004). Finding scientific topics. In *National academy of sciences of the United States of America* (Vol. 101–101, pp. 5228–5235).

Guo, Y., & Gu, S. (2011). Multi-label classification using conditional dependency networks. In *International joint conference on artificial intelligence* (pp. 1300–1305).

Hofmann, T. (1999). Probabilistic latent semantic indexing. In *ACM SIGIR international conference on research and development in information retrieval* (pp. 50–57).

Ji, S., Tang, L., Yu, S., Ye, J. (2008). Extracting shared subspace for multi-label classification. In *ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 381–389).

Jiang, W., & Ras, Z.W. (2013). Multi-label automatic indexing of music by cascade classifiers. *Web Intelligence and Agent Systems International Journal*, 149–170.

Kazawa, H., Izumitani, T., Taira, H., Maeda, E. (2004). Maximal margin labeling for multi-topic text categorization. In *Neural information processing systems* (pp. 649–656).

Kim, D., Kim, S., Oh, A. (2012). Dirichlet process with mixed random measures: a nonparametric topic model for labeled data. In *International conference on machine learning* (pp. 727–734).

Lewis, D.D., Yang, Y., Rose, T.G., Li, F. (2004). Rcv1: a new benchmark collection for text categorization research. *Journal of Machine Learning Research*, 361–397.

Li, T., & Ogihara, M. (2006). Towards intelligent music information retrieval. *IEEE Transactions on Multimedia*, 564–574.

Li, W., & McCallum, A. (2006). Pachinko allocation: dag-structured mixture models of topic correlations. In *International conference on machine learning* (pp. 577–584).

Nguyen, V.A., Boyd-Graber, J., Chang, J., Resnik, P. (2013). Tree-based label dependency topic models. In *Neural information processins systems workshop on topic models*.

Qi, G.J., Hua, X.S., Rui, Y., Tang, J., Mei, T., Zhang, H.J. (2007). Correlative multi-label video annotation. In *International conference on music information retrieval* (pp. 17–26).

Ramage, D., Hall, D., Nallapati, R., Manning, C.D. (2009). Labeled LDA: a supervised topic model for credit attribution in multi-labeled corpora. In *Conference on empirical methods in natural language processing* (pp. 248–256).

Ramage, D., Manning, C.D., Dumais, S. (2011). Partially labeled topic models for interpretable text mining. In *ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 457–465).

Read, J., Pfahringer, B., Holmes, G., Frank, E. (2011). Classifier chains for multi-label classification. *Machine Learning*, 1–27.

Rubin, T.N., Chambers, A., Smyth, P., Steyvers, M. (2012). Statistical topic models for multi-label document classification. *Machine learning*, 157–208.

Teh, Y.W., Jordan, M.I., Beal, M.J., Blei, D.M. (2006). Hierarchical Dirichlet processes. *Journal of the American Statistical Association*, 1566–1581.

Trohidis, K., Tsoumakas, G., Kalliris, G., Vlahavas, I. (2008). Multilabel classification of music into emotions. In *International conference on music information retrieval*.

Tsoumakas, G., & Katakis, I. (2007). Multi label classification: an overview. *International Journal of Data Warehousing and Mining*, 1–13.

Tsoumakas, G., Katakis, I., Vlahavas, I. (2011). Random k-labelsets for multi-label classification. *IEEE Transactions on Knowledge and Data Engineering*, 1079–1089.

Tsoumakas, G., Spyromitros-Xioufis, E., Vilcek, J., Vlahavas, I. (2011). Mulan: a java library for multi-label learning. *Journal of Machine Learning Research*, 2411–2414.

Ueda, N., & Saito, K. (2002). Single-shot detection of multiple categories of text using parametric mixture models. In *ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 626–631).

Wang, C., Yan, S., Zhang, L., Zhang, H.J. (2009). Multi-label sparse coding for automatic image annotation. In *IEEE conference on computer vision and pattern recognition* (pp. 1643–1650).

Yuret, D., Yatbaz, M.A., Ural, A.E. (2008). Discriminative vs. generative approaches in semantic role labeling. In *Conference on computational natural language learning* (pp. 223–227).

Zhang, M.L., & Zhou, Z.H. (2006). Multi-label neural networks with applications to functional genomics and text categorization. *IEEE Transactions on Knowledge and Data Engineering*, 1338–1351.

Zhang, Y., Burer, S., Street, W.N. (2006). Ensemble pruning via semi-definite programming. *Journal of Machine Learning Research*, 1315–1338.

Zhang, M.L., & Zhou, Z.H. (2007). Ml-knn: a lazy learning approach to multi-label learning. *Pattern Recognition*, 2038–2048.

Zhang, M.L. (2009). Ml-rbf: Rbf neural networks for multi-label learning. *Neural Processing Letters*, 61–74.