Expert Systems With Applications 57 (2016) 1–11

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

Ensemble multi-label text categorization based on rotation forest and latent semantic indexing

Haytham Elghazel*, Alex Aussem, Ouadie Gharroudi, Wafa Saadaoui

University of Lyon, Université Lyon 1, LIRIS UMR CNRS 5205, F-69622, France

ARTICLE INFO

Article history:
Received 27 October 2015
Revised 22 March 2016
Accepted 23 March 2016
Available online 24 March 2016

Keywords:
Multi-label classification
Text categorization
Ensemble learning
Rotation forest
Content analysis and indexing

ABSTRACT

Text categorization has gained increasing popularity in the last years due the explosive growth of multimedia documents. As a document can be associated with multiple non-exclusive categories simultaneously (e.g., Virus, Health, Sports, and Olympic Games), text categorization provides many opportunities for developing novel multi-label learning approaches devoted specifically to textual data. In this paper, we propose an *ensemble* multi-label classification method for text categorization based on four key ideas: (1) performing Latent Semantic Indexing based on distinct orthogonal projections on lower-dimensional spaces of concepts; (2) random splitting of the vocabulary; (3) document bootstrapping; and (4) the use of BoosTexter as a powerful multi-label base learner for text categorization to simultaneously encourage diversity and individual accuracy in the committee. Diversity of the ensemble is promoted through random splits of the vocabulary that leads to different orthogonal projections on lower-dimensional latent concept spaces. Accuracy of the committee members is promoted through the underlying latent semantic structure uncovered in the text. The combination of both rotation-based ensemble construction and Latent Semantic Indexing projection is shown to bring about significant improvements in terms of *Average Precision, Coverage, Ranking loss* and *One error* compared to five state-of-the-art approaches across 14 real-word textual data sets covering a wide variety of topics including health, education, business, science and arts.

1. Introduction

Document categorization is the assignment of documents to one or more predefined classes or categories based on their similarity to the conceptual content of the categories. It is a fundamental issue in natural language processing, including information retrieval, information extraction, and text mining. The document categorization problem may be cast as a multi-label classification (MLC) problem.

Formally, MLC amounts to finding a mapping from a space of features (the words) to a space of labels (the categories). Given a multi-label training set $\mathcal{D}$, the goal of multi-label learning is to find a function which is able to map any unseen example to its proper set of labels. Multi-label learning also emerged in other challenging applications such as gene function classification, spam filtering, sentiment analysis and semantic annotation of images (Boutell, Luo, Shen, & Brown, 2004; Clare & King, 2001; Zhang & Zhou, 2006) to cite a few. More recently, the MLC problem, viewed from a statistical perspective, attracted a great deal of interest in the machine learning community (Dembczynski, Waegeman, Cheng, & Hüllermeier, 2012; Gasse, Aussem, & Elghazel, 2015; Madjarov, Kocev, Gjorgjevikj, & Dzeroski, 2012; Zhang & Zhou, 2013) and several approaches have been proposed (see for instance (Madjarov et al., 2012; Zhang & Zhou, 2013) for a comparative review).

Research on multi-label learning was initially motivated by the difficulty of concept ambiguity encountered in text categorization, where each document may belong to one or more topics (labels) simultaneously. However, the existence of an underlying latent structure in textual data capture calls for MLC approaches that are specifically tailored to document categorization. Therefore, developing powerful and scalable MLC approaches devoted to document categorization is still an important issue in the field of machine learning.

In the last decade, there has been a great deal of research focused on ensemble MLC methods in order to improve the robustness and the generalization ability of single MLC learners which suffer from severe limitations in the presence of high-dimensional data, noisy, or imbalanced data. To achieve higher prediction accuracy than individual classifiers, it is crucial that the ensemble consists of highly accurate classifiers which at the same time disagree as much as possible.

* Corresponding author. Tel.: +33426234465.
E-mail addresses: haytham.elghazel@liris.cnrs.fr (H. Elghazel), alexandre.aussem@liris.cnrs.fr (A. Aussem), ouadie.gharroudi@liris.cnrs.fr (O. Gharroudi), wafa.saadaoui@univ-lyon2.fr (W. Saadaoui).

http://dx.doi.org/10.1016/j.eswa.2016.03.041
0957-4174/

With this motivation in mind, we present a novel ensemble multi-label text categorization algorithm, termed Multi Label Rotation Forest (MLRF), based on a combination of Rotation Forest and Latent Semantic Indexing. The combination of both paradigms brings about significant benefits. On the one hand, Rotation Forest (Rodríguez, Kuncheva, & Alonso, 2006) is a one of the most powerful ensemble methods for binary classification problems as shown in a number of recent extensive experimental studies (Bibimoune, Elghazel, & Aussem, 2013; Kuncheva & Rodríguez, 2007) over a wide range of data sets. On the other hand, Latent semantic indexing (LSI) is an efficient indexing and retrieval method that uses a rank-reduced singular value decomposition (SVD) to identify patterns in the relationships between the words (or terms) and the (latent) concepts. The key idea is to apply the LSI on small random subsets of the vocabulary in order to build a collection of training sets with distinct samples and concept representations. The goal is to encourage simultaneously individual accuracy and diversity within the ensemble. Diversity is promoted through the different splits of the set of words that lead to different orthogonal projections on lower-dimensional subspaces, namely the space of concepts. Accuracy is promoted through the underlying latent semantic structure in the text uncovered by LSI. The LSI also reduces noise and other undesirable artifacts of the original space.

The main contribution of this paper is an investigation of the extent to which MRLF is powerful compared to state-of-the-art methods. Extensive experiments are conducted on various benchmark text categorization multi-label data sets. Our results demonstrate that the proposed method enjoys significant advantages compared to other methods.

The rest of the paper is organized as follow: Section 2 reviews recent studies on ensemble learning and multi-label learning methods with special emphasis on the multi-label document categorization methods; Section 3 introduces our multi-label classification method for text categorization; Experiments using relevant benchmark text categorization data sets are presented in Section 4; finally, Section 5 concludes the study and identifies some future research directions.

2. Related work

In this section, we review of the Rotation Forest algorithm and the standard MLC methods with special emphasis on the MLC methods devoted to document categorization.

2.1. Rotation Forest

The idea of exploiting ensemble learning to improve multi-label classification has received an increasing attention in the last few years. Dietterich (2000) states that "A necessary and sufficient condition for an ensemble of classifiers to be more accurate than any of its individual members is if the classifiers are accurate and diverse". Many methods have been proposed to generate accurate, yet diverse, sets of models. Bagging (Breiman, 1996), boosting (Freund & Shapire, 1996), Random Subspaces (Ho, 1998), Random Forest (Breiman, 2001) and Rotation Forest (Rodríguez et al., 2006) are the most popular examples of this methodology. We assume the reader is familiar with these techniques. We point the reader to (Zhou & Zhou, 2012) for a brief review.

Proposed by Rodríguez et al. (2006), Rotation Forest is another successful ensemble classifier generation technique in which the training set for each base classifier is formed by applying PCA (Principal Component Analysis) to rotate the original attribute axes. Specifically, to create the training data for a base classifier, the attribute set of data is randomly split into K subsets and PCA is applied to each subset. All principal components are retained in order to preserve the variability information in the data. Thus, K axis rotations take place to form the new attributes for a base classifier. The main idea of Rotation Forest is to simultaneously encourage diversity and individual accuracy within the ensemble: diversity is promoted through doing feature extraction for each base classifier and accuracy is sought by keeping all principal components and also using the whole data set to train each base classifier. Recent extensive experimental studies Kuncheva and Rodríguez (2007) and Bibimoune et al. (2013) over a wide range of benchmark and real data sets has demonstrated the effectiveness of Rotation Forest in the context of binary classification problems compared to a variety of well-known ensemble methods, including Adaboost, Rotation Forest and Rotboost (Zhang & Zhang, 2008).

2.2. Multi-label learning

As mentioned in the Introduction, the issue of learning from multi-label data has recently attracted significant attention over the last years (Dembczynski et al., 2012; Madjarov et al., 2012; Zhang & Zhou, 2013). While, many efficient approaches have been proposed, their theoretical underpinnings remain weak, at least, as compared to the rather complete theory of binary classification learning. In fact, the benefit of exploiting label dependence is closely dependent on the loss function as discussed in Dembczynski et al. (2012).

Most MLC methods intend to exploit, in one way or the other, dependencies between the class labels. Basically, these methods can be summarized into three categories: (a) algorithm adaptation methods, (b) problem transformation methods, and (c) *ensemble methods* (Madjarov et al., 2012). We discuss each category in what follows.

2.2.1. Algorithm adaptation methods

These methods extend specific learning algorithms (like decision trees, SVM, neural networks and k-nearest neighbors) to handle multi-label data directly. Clare et al. adapted the entropy function in the C4.5 decision tree algorithm to handle the multi-label data (Clare & King, 2001) by summing the label entropies. PCT, in Blockeel, Raedt, and Ramon (1998), is another algorithm adaptation decision tree capable of predicting multiple target attributes at once. The induction process in PCT uses the sum of the Gini indices throughout all labels to identify the best separation at each node. Elisseeff and Weston presented in (Elisseeff, 2005) a ranking approach based SVM to handle multi-label data. They propose to use the average fraction of incorrectly ordered pairs of labels as a cost function. Crammer and Singer propose to use neural network approach called BP-MLL in (Zhang & Zhou, 2006) which is an adaptation of back-propagation in the multi-label setting. The important modification of the algorithm is the use of function error that take considers multiple labels. From the popular k-Nearest Neighbors (k-NN), various multi-label learning have been proposed. Zhang and Zhou proposed in (Zhang, 2007) a lazy learning approach (MLkNN). Their model is similar to the traditional k-NN algorithm. Although, the determination of labels for a new test instance is different. The algorithm use prior and posterior probabilities of each label among the k-NN. More recently, based on the variable precision neighborhood rough sets, two multi-label classification approaches named MLRS and MLRS-LC were proposed in (Yu, Pedrycz, & Miao, 2014). Both approaches consider the aspects of correlation among the labels and uncertainty in the mapping between the feature space and the label space to improve the quality of multi-label classification. MLRS and MLRS-LC respectively provide a global and local view at the label correlation. Although, a series of experiments reported in (Yu et al., 2014) have shown that both approaches perform well in several domains, their performances depend on the nature of the data. As a nearest-neighbors-based method, they have been recognized as having poor perfor-

mance when applied to high-dimensional multi-label data sets (Yu et al., 2014); thus not appropriate for multi-label text categorization.

2.2.2. Problem transformation methods

Problem transformation methods transform the multi-label task, into one or more single-label tasks. Then single-label problems are solved using traditional algorithms. The outputs are then transformed back into the initial representation. Problem Transformation methods can be essentially grouped into two categories: *Binary Relevance methods* (BR) and *Label Power-set methods* (LP). *Binary Relevance methods* switch the multi-label problem into distinct binary classification problems. In BR transformation scheme, for each label λ_j a separate binary classifier h_j is created. For h_j, a data set D_{λ_j} is created, which contains all examples of the original data set labeled as positive if it is labeled with the label λ_j and negative otherwise: $h_j : \mathcal{X} \longrightarrow \{0, 1\}$. Where h_j, is a classification learner. BR approaches are intuitive and easy to implement. Nevertheless, BR methods don't take into consideration dependencies between labels: each label is treated independently. To take into account correlation among label various enhancements of BR methods have been suggested. In the Classifier Chain (CC) (Read, Pfahringer, Holmes, & Frank, 2009) method, each binary classifier is trained to learn its corresponding label given the predicted labels of the previous ones. Recently, a two-steps framework has been proposed in (Wang, Wang, Wang, & Ji, 2014) to capture the labels dependencies. In the first step, a traditional multi-label classifier is used to learn to estimates the class label assignments; and in the second step, a Bayesian network is constructed to model the dependencies among the labels and to refine the multi-label classifier prediction in the prediction phase.

On the other hand, *Label Power-set methods* transform the problem into a standard multi-class classification problem. The class variable encodes all the subsets of labels that are observed in the training data. Therefore, correlation between labels is taken into consideration. However, LP fails when the number of effective label subsets— and thus the number of classes— become too large.

2.2.3. Ensemble methods for multi-label learning

Ensemble methods for multi-label learning are developed on top of the common problem transformation or algorithm adaptation methods. RAndom k-labELsets (RAkEL) (Tsoumakas, Katakis, & Vlahavas, 2011a) splits the labels into small-size random subsets called *k*-labelsets (*k* is the labelset size), and runs LP on each of them. A simple voting scheme determines the final classification set. In this manner, RAkEL takes into account correlation between labels, and at the same time, overcomes the limitations of LP methods, by reducing the number of labels handled by the LP classifiers. Based on the same ensemble construction strategy (Rokach, Schclar, & Itach, 2014) proposed an improved version of RAkEL. The idea is to, (i) aggregate the probabilities provided by the base-classifiers rather than using a crisp label votes as in the original RAkEL, and (ii) use in the labeling step a data-specific threshold calibrated via a cross validation procedure. ECC are other ensemble methods that are based on classifier chains CC. The algorithm use p classifier chains $C_1, C_2, \ldots, C_p$; trained in a random sequence. The decisions of all CC classifiers are combined and a threshold is used to choose the final multi-label set. RAkEL and ECC are problem transformation methods. In contrary, RFPCT (Kocev, Vens, Struyf, & Dzeroski, 2013) is an algorithm adaptation method that uses a standard Random Forest with PCT (Blockeel et al., 1998) as base learner. Each tree performs multi-label predictions, and then predictions are combined using a majority or probability voting scheme. The diversity among the trees is promoted using two strategies; bootstrap sampling of training data and random selection of feature subsets. More recently, a novel multi-label classification framework called Variable Pairwise Constraint projection for Multi-label Ensemble (VPCME) (Li, Li, & Wu, 2013) was proposed to construct a multi-label ensemble for handling multi-label data. This framework involves two inherent components, i.e., the variable pairwise constraint projection and the boosting-like strategy. The variable pairwise constraint projection produces a lower-dimensional data representation which aims at preserving the correlations between samples and labels, while the boosting-like strategy improves the generalization ability of the classifier.

2.3. Multi-label learning for text categorization

The problem of multi-label learning was initially motivated by the difficulty of concept ambiguity encountered in text categorization, where each document may belong to several topics (labels) simultaneously. There are many ways to deal with this problem. BoosTexter is a is powerful approach proposed by Schapire and Singer (2000), which can be regarded as an extension of AdaBoost (Freund & Shapire, 1996). In the training phase, BoosTexter maintains a set of weights over both training examples and their labels, where training examples and their corresponding labels that are hard (easy) to predict correctly get incrementally higher (lower) weights. Other proposals for multi-label text categorization (Fujino, Isozaki, & Suzuki, 2008; Sajnani, Javanmardi, McDonald, & Lopes, 2011; Vilar, Castro, & Sanchis, 2004) are essentially in the BR category. The main caveat is that the statistical relations between the labels are ignored. An online procedure for multi-label text classification was proposed recently in (Nanculef, Flaounas, & Cristianini, 2014). The procedure approximates the geometry of the TF-IDF (Joachims, 2002; Sebastiani, 2002) document representation and then divide the feature space into a set of regions where documents with similar low-dimensional representations collide and the label are assumed to be independent. Finally, based on the label independence assumption a BR naive Bayes is used as a multi-label classifier. The proposed approach aims at improved both time complexity and memory need in textual data analysis and cannot guarantee predictive performance improvements.

3. The MLRF method

We now present the MLRF algorithm for Multi-label text categorization. We review first the LSI method for document indexing and then describe the algorithm in details.

3.1. The Latent Semantic Indexing

Databases have increased many fold in recent years with sometimes hundreds or thousands of explanatory features. In such high-dimensional data sets, traditional information retrieval and pattern recognition approaches suffer from severe limitations in terms of efficiency and computational burden. Therefore, the dimensionality reduction provided by semantic indexing or feature projection is of great importance for document analysis and processing and commonly applied in world problems (He, Cai, Liu, & Ma, 2004; Wiemer-Hastings, 1999; Yu, Yu, & Tresp, 2005). LSI (Deerwester, Dumais, Landauer, Furnas, & Harshman, 1990) is considered effective to overcome the problems of lexical matching by deriving conceptual indices instead of individual terms (or words) for retrieval in a collection of documents. It presumes the existence of an underlying latent structure in the data ant tries to statistically capture this implicit structure in the association between terms across documents.

LSI is purely unsupervised and essentially detects the most representative features for document representation rather than the

most discriminative features (He et al., 2004). To apply LSI, documents are represented in a vector space model using a document-by-term matrix $\mathcal{X}_{n,d}$ (n is a number of documents and d is a number of words) and SVD is performed to find a linear mapping from the input space $\mathcal{X}$ to some low-dimensional latent space, while most of the structure in the data can be explained and recovered.

The SVD projection is computed by decomposing the matrix $\mathcal{X}$ into the product of three matrices $\mathcal{X}_{n,d} = U_{n,d} \times S_{d,d} \times V_{d,d}^T$, where U and V are orthogonal matrices that contain the left and right singular vectors of $\mathcal{X}$, respectively, S is the diagonal matrix that contains the singular values of $\mathcal{X}$. The singular values in SVD help you determine what variables (words) are most informative and which ones you can do without. When singular values in S are sorted in descending order, the top k singular vectors corresponding to the k largest singular values ($k \leq min(n, d)$) are used to construct a new lower k-dimensional space. Indeed, the product of reduced matrices $V_{d,k} \times S_{k,k}^{-1}$ is considered as a transformation matrix that can project the higher-dimensional feature matrix $\mathcal{X}_{n,d}$ into a lower-dimensional one $\mathcal{X}'_{n,k} = \mathcal{X}_{n,d} \times V_{d,k} \times S_{k,k}^{-1}$.

Among various methods, LSI turns out to be a successful approach and is widely applied to document analysis and information retrieval (Liu, Chen, Zhang, Ma, & Wu, 2004; Wang, Peng, & Liu, 2015; Zhang, Yoshida, & Tang, 2011) mainly on single classification problem. However, as stated above, LSI is a purely unsupervised transformation method. To remedy this situation, an algorithm called Multi-label informed Latent Semantic Indexing (MLSI), which considers predictions with multivariate labels, was proposed in (Yu et al., 2005) to preserve the information in the data while capturing the correlations between the labels. MLRF will be compared to MLSI in the experimental section.

3.2. The MLRF algorithm

MLRF is a natural extension of the Rotation Forest paradigm (Rodríguez et al., 2006) to multi-labeled data. MLRF aims at building accurate and diverse multi-label classifiers. The main idea consists in applying feature extraction algorithm on different random splits of the feature set to form a new attributes for each base multi-label classifier in the ensemble. We have chosen LSI in this study for reasons explained in the previous section. The pseudo-code of the sketch is shown in Algorithm 1. In the following, we elaborate this framework more clearly.

Let $\mathcal{L} = \{\lambda_1, \lambda_2, \ldots, \lambda_L\}$ be a label space and $\mathcal{X}$ a data set that consists of n documents each taking the form (x_i, y_i) where $x_i = (x_{i1}, \ldots, x_{id})$ is a vector of d descriptive terms (also words or features) and $y_i \in \mathcal{L}$ is the subset of labels associated to x_i (represented by a binary feature vector $(y_{i1}, \ldots, y_{iL}) \in \{0, 1\}^L$). Denote by $\mathcal{H} = \{h_1, h_2, \ldots, h_N\}$ the ensemble committee of multi-label classifiers and by F, the terms set. As with most ensemble methods, we need to pick the size of committee N in advance. To construct a classifier member h_i, we carry out the following steps:

1. As in the classical rotation forest algorithm, we first split F randomly into K subsets (K is a parameter of the algorithm). To enforce diversity, the subsets may be disjoint. For simplicity, suppose that K is a factor of d so that each feature subset contains $m = \frac{d}{K}$ terms.
2. Denote by $F_{i,j}$ the jth subset of features for the training set of multi-label classifier h_i. For every such subset, we draw a bootstrap sample of documents, of size 75% of the number of documents. Run LSI using only the m features in $F_{i,j}$ and the selected documents of $\mathcal{X}$. Store the coefficients of the transformation matrix $T_{i,j} = V_{d,k} \times S_{k,k}^{-1}$ denoted $T_{i,j}^{(1)}, \ldots, T_{i,j}^{(k)}$ each of size $m \times 1$ (k is a parameter of the algorithm and it denotes the size of the low-dimensional latent space chosen for the reduction step in the LSI method). Running LSI on a bootstrap sample of documents instead on the whole document set is a source of diversity in the ensemble construction and is done in a bid to avoid identical latent space if the same terms subset is chosen for different classifiers.
3. Organize the obtained vectors with coefficients in a sparse rectangular orthogonal matrix R_i (i.e., the columns of R_i are orthogonal, R_i is not a rotation matrix anymore as it is rectangular),
$$R_i = \begin{pmatrix} T_{i,1}^{(1)}, \ldots, T_{i,1}^{(k)}, & & \cdots & \\ [0] & T_{i,2}^{(1)}, \ldots, T_{i,2}^{(k)}, & \cdots & [0] \\ \vdots & \vdots & \ddots & \vdots \\ [0] & [0] & \cdots & T_{i,K}^{(1)}, \ldots, T_{i,K}^{(k)} \end{pmatrix}$$
The diagonal matrix will have dimensionality $d \times \Sigma_j k$. To calculate the new data representation with the reduced dimension $\Sigma_i k$ for the multi-label classifier h_i , we first rearrange the columns of R_i (the terms) so that they correspond to the original terms. Denote the rearranged matrix by R_i^a, the new set lower-dimensional feature matrix for h_i is given by $\mathcal{X} \times R_i^a$.
4. A base multi-label classifier h_i is learned in the new data space. In this study, we use the BoosTexter algorithm (Schapire & Singer, 2000).

Algorithm 1 *MLRF*$(\mathcal{X}, \mathcal{Y}, F, d, L, N, K, k, S)$.

Require:
The document in the Training data ($\mathcal{X}$) where $x_i = (x_{i1}, \ldots, x_{id})$, the labels of the training document ($\mathcal{Y}$) where $y_i = (y_{i1}, \ldots, y_{iL})$, the input space ($F = \{f_1, \ldots, f_d\}$), the number of descriptive terms (d), the number of labels (L), the committee size (N), the number of feature subsets (K), the dimensionality reduction parameter for LSI (k) and the number of iterations for the base multi-label learning algorithm BoosTexter (S)

$\mathcal{H} = \emptyset$
for $i = 1 : N$ **do**
  Split the input space F into K subsets $F_{i,j}$ (for $j = 1 \ldots K$)
  for $j = 1 : K$ **do**
    $\mathcal{X}_{i,j}$ = bootstrap sample from $\mathcal{X}$ of size 75% of the number of documents in $\mathcal{X}$ projected onto $F_{i,j}$
    Apply LSI on $\mathcal{X}_{i,j}$ to obtain the coefficients of the k-dimensional latent space transformation in a matrix $T_{i,j}$
  end for
  Arrange the $T_{i,j}$ transformation matrices, for $j = 1 \ldots K$ in a rotation matrix R_i
  Construct R_i^a by rearranging the columns of R_i so as to match the order of features in F
  Learn the individual multi-label classifier $h_i = BoosTexter(\mathcal{X} \times R_i^a, \mathcal{Y}, S)$
  $\mathcal{H} = \mathcal{H} \cup h_i$
end for
return A committee of generated base multi-label classifiers $\mathcal{H}$ jointly producing the label probabilities of the testing documents using the average combination method

For a given document x, let $P_i(x, \lambda_j)$ be the probability assigned by the multi-label classifier member h_i to the hypothesis that label λ_j is relevant for x. The final probability $P(x, \lambda_j)$ returned by the ensemble committee $\mathcal{H}$ for x belonging to a label λ_i is given by the average combination method using the N label probabilities produced by ensemble members $P_i(x, \lambda_j)$ $(i = 1, \ldots, N)$. Indeed, as stated in (Briggs, Fern, & Irvine, 2013; Gharroudi, Elghazel, & Aussem, 2015), it is more reasonable for ensemble performances to aggregate probabilities from each base-classifier rather than 0/1 votes (*i.e.* majority voting).

$$P(x, \lambda_j) = \frac{1}{N} \sum_{i=1}^{N} P_i(x, \lambda_j)$$

3.3. Why should our approach work?

In this Section, we discuss related work and provide several arguments to support the claim that the combination of LSI-based projection, random split of the vocabulary, document bagging and a powerful multi-label base learner (BoosTexter) has a potential for significant benefits for multi-label text categorization in terms of performance.

- Principal component analysis (PCA) originally used in the rotation forest approach (Rodríguez et al., 2006), is a common feature space transformation method and have a considerable connexion to LSI in the sense that each latent semantic can be viewed as a component to represent the data (Yu et al., 2005). However, both transformation methods are different in the way they consider the data. In the context of text categorization the use of LSI instead of PCA as the inner transformation of the ensemble model is crucial for two reasons: First, PCA turns out to solve the eigenvalue problem on either XX' (correlations among the documents) or $X'X$ (correlations among the terms); thus it analyzes either the documents or the terms independently whereas LSI analyzes both together (Skillicorn, 2007). Second, the computation of the correlation matrices for PCA are expensive and most of the time ill-conditioned especially on textual data.
- As aforementioned in Section 3.1, LSI is a purely unsupervised. To remedy this situation, an algorithm called Multi-label informed Latent Semantic Indexing (MLSI) was proposed in (Yu et al., 2005) to preserve the information in the data while capturing the correlations between the labels. Nonetheless, in a rotation style ensemble, a supervised transformation such as MLSI can hurt the diversity in the ensemble committee. Indeed, the rotated block will be redundant since that all rotations try to map the original features to be as close as possible to target space $\mathcal{Y}$. This is why LSI was chosen as the feature space transformation method in MLRF.
- It is worth noting that many terms have different meanings depending on the context of the considered documents. LSI attempts to map out the relationships between terms in order to help decipher the meaning of the text; thus it can in part solve the synonym problem (one meaning, many terms). Nonetheless, LSI cannot handle the polysemy problem (one word, many meanings) well since each term in the vocabulary is considered as a single point in the input space (Zhou, Zhang, & Hu, 2006). MLRF tackle the polysemy issue by working on distinct "semantic" spaces (of concepts) through different splits of the vocabulary.
- Different splits of the feature set together with the bootstrap steps lead to different rotations and therefore encourage diversity of the base multi-labels classifiers in the ensemble.
- BoosTexter was chosen as our base learner in view of its good predictive performance in text categorization. An accurate baseline learner is usually seen as a necessary for an ensemble approach based on independent components, like bagging or Random Forests, to work well.

4. Performances analysis

In this section, we assess the effectiveness of our MLRF approach for multi-label text categorization, by comparing its performance against conventional multi-label classification methods on several benchmark textual data sets from the *Mulan's repository* (Tsoumakas, Xioufis, Vilcek, & Vlahavas, 2011b). A brief description of the evaluation metrics and the data sets is given. We then report on the experiments performed to evaluate the performance of MLRF.

4.1. Evaluation metrics

In these experiments, we use the following popular multi-label evaluation metrics: *Hamming loss, Ranking loss, Coverage, One-error and Average precision*. Borrowing notation and terminology from (Li et al., 2013; Schapire & Singer, 2000; Zhang, 2007), a test set is denoted by $Te = \{(x_1, y_1), (x_2, y_2), \ldots, (x_p, y_p)\}$, where $y_i \in \mathcal{L}$ is the true label set and $\mathcal{L} = \{\lambda_1, \lambda_2, \ldots, \lambda_L\}$ is the total label set. For a given instance x_i, its predicted label set from the multi-label classifier h is denoted by $h(x_i)$, and the estimated rank of the label λ_j is denoted by $r_i(\lambda_j)$. The most relevant label takes the top rank (1) and the least one only gets the lowest rank (L). The evaluation metrics are discussed and formulated mathematically as follows:

The Hamming loss evaluates the accuracy in a multi-label classification task. It measures the percentage of incorrectly predicted labels to the total number of labels.

$$Hamming\_loss(h) = \frac{1}{p}\sum_{i=1}^{p}\frac{|y_i \Delta h(x_i)|}{L} \tag{1}$$

where the Δ is the symmetric difference between two sets.

While, the Hamming loss is based on the multi-label classifier h, the following metrics are defined based on the estimated rank of r which concern the ranking quality of different labels for each instance (Zhang, 2007):

The Ranking loss evaluates the average fraction of label pairs that are reversely ordered for the instance.

$$Ranking\_loss(h) = \frac{1}{p}\sum_{i=1}^{p}\frac{1}{|y_i||\bar{y}_i|}|\{(\lambda_a, \lambda_b) : r_i(\lambda_a) > r_i(\lambda_b), (\lambda_a, \lambda_b) \in y_i \times \bar{y}_i\}| \tag{2}$$

where the term $\bar{y}_i$ is the complementary set of y_i with respect to $\mathcal{L}$.

The Coverage evaluates how far we need, on the average, to go down the list of labels in order to cover all the proper labels of the instance.

$$Coverage(h) = \frac{1}{p}\sum_{i=1}^{p}\max_{\lambda_a \in y_i} r_i(\lambda_a) - 1 \tag{3}$$

The One-error loss evaluates how many times the top-ranked label is not in the set of proper labels of the instance.

$$One\text{-}error(h) = \frac{1}{p}\sum_{i=1}^{p}\delta(\operatorname{argmin}_{\lambda_a \in \mathcal{L}} r_i(\lambda_a))$$
$$\text{where } \delta(\lambda_a) = \begin{cases} 1, & \text{if } \lambda_a \in y_i \\ 0, & \text{otherwise} \end{cases} \tag{4}$$

The average precision evaluates the average fraction of labels ranked above a particular label $\lambda_a \in y_i$ which actually are in y_i.

$$Average\_precision(h) = \frac{1}{p}\sum_{i=1}^{p}\frac{1}{|y_i|}\sum_{\lambda_a \in y_i}\frac{|\lambda_b \in y_i : r_i(\lambda_b) \le r_i(\lambda_a)|}{r_i(\lambda_a)} \tag{5}$$

For Hamming loss, Ranking loss, Coverage and One-error, the smaller the value, the better the performance. For the Average precision, greater values indicate better performance.

4.2. Data sets

We use 14 different multi-label classification benchmark problems devoted to text categorization. The selected problems were used in various studies and evaluations of methods for multi-label

Table 1
Description of the Benchmark multi-label text categorization data sets used in the experiments.

Data	L	Vocabulary	Training set			Test Set		
		size	#docs	PMC	ANL	#docs	PMC	ANL
Arts	26	462	2000	44.50%	1.627	3000	43.63%	1.642
Business	30	438	2000	42.20%	1.590	3000	41.93%	1.586
Computers	33	681	2000	29.60%	1.487	3000	31.27%	1.522
Education	33	550	2000	33.50%	1.465	3000	33.73%	1.458
Enron	53	1001	1123	87.98%	3.387	579	89.46%	3.363
Entertainment	21	640	2000	29.30%	1.426	3000	28.20%	1.417
Health	32	612	2000	48.05%	1.667	3000	47.20%	1.659
Medical	45	1449	333	23.72%	1.255	645	22.79%	1.240
Recreation	22	606	2000	30.20%	1.414	3000	31.20%	1.429
Reference	33	793	2000	13.75%	1.159	3000	14.60%	1.177
Science	40	743	2000	34.85%	1.489	3000	30.57%	1.425
Slashdot	22	1079	1513	16.19%	1.174	2269	17.36%	1.186
Social	39	1047	2000	20.95%	1.274	3000	22.83%	1.290
Society	27	636	2000	41.90%	1.705	3000	39.97%	1.684

Table 2
Experimental results of each multi-label learning algorithm in terms of *Hamming loss*.

Data	MLkNN	BoosTexter	RAkEL	VPCME	MLSI	MLRF
Arts	0.0612	0.0652	0.0606	0.0608	0.0592	**0.0576**
Business	0.0269	0.0293	0.0282	**0.0267**	0.0272	0.0268
Computers	0.0412	0.0408	0.0384	0.0414	0.0406	**0.0370**
Education	0.0387	0.0457	0.0389	0.0405	0.0401	**0.0386**
Enron	0.0520	0.0510	0.0497	0.0487	0.0539	**0.0470**
Entertainment	0.0603	0.0626	0.0606	0.0610	0.0625	**0.0552**
Health	0.0458	0.0397	0.0362	0.0445	0.0428	**0.0354**
Medical	0.0187	0.0115	**0.0110**	0.0161	0.0181	0.0113
Recreation	0.0618	0.0657	0.0604	0.0610	0.0607	**0.0569**
Reference	0.0314	0.0304	0.0270	0.0312	0.0309	**0.0267**
Science	0.0325	0.0379	**0.0325**	0.0339	0.0325	0.0326
Slashdot	0.0530	0.0434	0.0454	0.0534	0.0502	**0.0431**
Social	0.0218	0.0243	0.0224	0.0256	0.0237	**0.0210**
Society	0.0537	0.0628	0.0535	0.0551	0.0552	**0.0530**
Mean	0.0428	0.0436	0.0403	0.0428	0.0427	**0.0387**

learning. These data sets are already divided into training and testing sets to make comparisons as consistent as possible. Table 1 reports the basic statistics of the data sets. Table 1 shows, for each data set, the number of documents in both the training and test data set (#docs); the number of features (the vocabulary size), the number of labels; the percentage of documents belonging to more than one category (PMC) and the average number of labels for each document (ANL).

4.3. Experimental protocol

We compared MLRF to several state-of-the-art algorithms, including MLkNN (Zhang, 2007), the Multi-label informed Latent Semantic Indexing (MLSI) (Yu et al., 2005) and three other ensemble multi-label classification approaches: BoosTexter (Schapire & Singer, 2000), RAkEL (Tsoumakas et al., 2011a), and VPCME (Li et al., 2013) (see Section 2.2 for more details about these approaches). To make fair comparisons, the same experimental settings in (Yu et al., 2005) was adopted here for MLSI, (i.e., the dual form of this approach is used for which the parameter β is set to 0.5 and the γ value is simply fixed as 0, where they was found to yield the most satisfactory performances. As in (Li et al., 2013), the instance-based learning method MLkNN (Zhang, 2007) was used as the base classifier for VPCME due to its excellent predictive performance, and the number of nearest neighbor k was set to 10 (Zhang, 2007). For VPCME, the variable pairwise constraint threshold was empirically set to 0.6 as in (Li et al., 2013). Following the experimental settings in (Zhang & Zhang, 2008), the ensemble size for VPCME and BoosTexter was tuned to 100. With respect to MLRF, the number of iterations performed in our ensemble strategy N and BoosTexter Baselearn algorithm S were both taken to be 10 so that the ensemble consist of 100 classifiers. For MLRF, we did not fix K but we fixed the number of features in each subset to 100. The dimensionality reduction parameter k for LSI was set to 10.

4.4. Results

In this section, we report the results in terms of the metrics discussed previously and draw some conclusions from these observations. Tables 2–6 report the results on the benchmark data sets.

The experimental results on each evaluation criterion are reported in Tables 2–6, where the best result on each data set is shown in bold face. Bottom row of the table present the mean of each multi-label evaluation metric used in the experiments.

The methodology proposed by (Demsar, 2006) was adopted to assess the results obtained for each algorithm on each metric. It offers a principled approach to compare several algorithms over multiple data sets. A non-parametric Friedman test is used first to evaluate the rejection of the hypothesis that all the classifiers perform equally well at a certain risk level. The algorithm are ranked separately for each data set, the best performing algorithm being ranked first, the second best ranked second, etc. In case of ties, the average rank is assigned. Then, the Friedman test amounts to compare the average ranks of the algorithms. If a statistically significant difference in the performance is detected, the *post hoc* Nemenyi test is used to compare all the classifiers to each other. A significantly different performance is observed if the average ranks differ more than some critical distance (CD) (Bibimoune et al., 2013). The latter depends on the number of algorithms, the num-

Table 3
Experimental results of each multi-label learning algorithm in terms of *Ranking loss*.

Data	MLkNN	BoosTexter	RAkEL	VPCME	MLSI	MLRF
Arts	0.1520	0.1458	0.1942	0.6988	0.1302	**0.1261**
Business	**0.0373**	0.0416	0.0894	0.1836	0.0389	0.0412
Computers	0.0923	0.0950	0.1624	0.4721	0.0917	**0.0818**
Education	**0.0800**	0.0938	0.1435	0.5321	0.0879	0.0821
Enron	0.0938	0.0912	0.1655	0.3248	0.0984	**0.0784**
Entertainment	0.1150	0.1132	0.4027	0.5045	0.1128	**0.0983**
Health	0.0605	0.0521	0.2708	0.2624	0.0618	**0.0452**
Medical	0.0585	0.0680	0.1504	0.1526	0.0569	**0.0265**
Recreation	0.1913	0.1599	0.2290	0.7134	0.1893	**0.1453**
Reference	0.0919	0.0811	0.1488	0.4260	0.0847	**0.0696**
Science	0.1166	0.1312	0.1737	0.5290	0.1262	**0.1097**
Slashdot	0.1872	0.1101	0.2551	0.5157	0.1736	**0.1017**
Social	**0.0561**	0.0684	0.1213	0.2299	0.0627	0.0566
Society	0.1339	0.1483	0.1940	0.4436	0.1392	**0.1329**
Mean	0.1048	0.1000	0.1929	0.4278	0.1039	**0.0854**

Table 4
Experimental results of each multi-label learning algorithm in terms of *Coverage*.

Data	MLkNN	BoosTexter	RAkEL	VPCME	MLSI	MLRF
Arts	5.4453	5.2973	5.8453	11.4233	5.4290	**4.8443**
Business	**2.1837**	2.4123	3.6410	7.0120	2.2433	2.3270
Computers	4.4183	4.4887	6.0337	13.4793	4.4330	**4.0693**
Education	**3.4947**	4.0673	5.2160	14.3070	3.7607	3.6287
Enron	13.2055	13.3921	16.5112	23.8566	13.4611	**11.8135**
Entertainment	3.1467	3.0883	7.2710	8.9780	3.3233	**2.8170**
Health	3.3047	3.0780	9.0510	8.9307	3.1550	**2.7667**
Medical	3.5039	2.4295	3.9070	4.0915	3.4116	**1.6186**
Recreation	5.1027	4.4737	5.2153	10.8447	5.0387	**4.1460**
Reference	3.5420	3.2100	4.9443	10.9233	3.2983	**2.7820**
Science	6.0423	6.6907	7.6203	14.6677	6.3940	**5.7903**
Slashdot	4.4244	2.7589	3.7435	5.7409	4.0755	**2.5637**
Social	**3.0333**	3.6870	4.9433	7.7227	3.3583	3.0800
Society	**5.3650**	5.8463	6.5487	9.9153	5.4927	5.4173
Mean	4.7295	4.5864	6.4637	10.8495	4.7768	**4.1189**

Table 5
Experimental results of each multi-label learning algorithm in terms of *One-error*.

Data	MLkNN	BoosTexter	RAkEL	VPCME	MLSI	MLRF
Arts	0.6323	0.5550	0.5233	**0.3640**	0.5530	0.5220
Business	0.1213	0.1307	0.1283	**0.1207**	0.1220	0.1250
Computers	0.4367	0.4287	0.4200	0.4390	0.4303	**0.4003**
Education	0.5210	0.5587	0.5173	**0.4520**	0.5580	0.5243
Enron	0.3040	0.2988	0.2297	0.2625	0.3437	**0.2159**
Entertainment	0.5297	0.4750	0.4497	0.4737	0.5957	**0.4407**
Health	0.4207	0.3210	0.3017	0.3490	0.4020	**0.2923**
Medical	0.3504	0.1628	**0.1519**	0.2217	0.3436	0.1566
Recreation	0.7057	0.5557	0.5097	**0.3587**	0.6790	0.5130
Reference	0.4730	0.4427	0.3993	0.4473	0.4540	**0.3960**
Science	0.5803	0.6100	**0.5410**	0.5273	0.6220	0.5450
Slashdot	0.6743	0.4636	0.4852	0.5610	0.6428	**0.4275**
Social	0.3270	0.3437	0.3267	0.3140	0.3617	**0.3013**
Society	0.4370	0.4877	0.4257	**0.4223**	0.4490	0.4240
Mean	0.4652	0.4167	0.3864	0.3795	0.4683	**0.3774**

ber of data sets and the significance level (see (Demsar, 2006) for further details).

Interestingly, the Friedman test reveals statistically significant differences ($p < 0.1$) for all evaluation metrics. In addition, we report the Nemenyi posthoc test results in the form of average rank diagrams as suggested by Demsar (2006). These are given on Fig. 1. The ranks are depicted on the axis, the best ranking algorithms being at the rightmost side of the diagram. The algorithms that do not differ significantly (at $p = 0.1$) are connected with a horizontal bar. Here, the critical difference CD is equal to 1.8307 and is reported above the diagram.

The results in Tables 2–6 show that MLRF exhibit higher predictive performances considering all five comparative metrics together. MLRF obtains the best mean values among the compared algorithms and it ranks first most of the time. The key observations we may draw from these results are five-fold:

- Regarding the Hamming loss, the performances of MLRF are not statistically distinguishable from the performances of RAkEL. The statistical tests we use are rather conservative. To further support these rank comparisons, we also compared these algorithms with the Wilcoxon signed-rank test. BMLRF is significantly better than RAkEL at $p = 0.05$ based on this test,

Table 6
Experimental results of each multi-label learning algorithm in terms of *Average precision*.

Data	MLkNN	BoosTexter	RAkEL	VPCME	MLSI	MLRF
Arts	0,5093	0,5448	0,5719	0,3980	0,5421	**0,5845**
Business	**0,8798**	0,8697	0,8601	0,8156	0,8775	0,8789
Computers	0,6333	0,6449	0,6439	0,5254	0,6390	**0,6701**
Education	0,5993	0,5654	0,5883	0,4651	0,5686	**0,6013**
Enron	0,6232	0,6639	0,6535	0,6171	0,6187	**0,7001**
Entertainment	0,6016	0,6368	0,5968	0,4829	0,5603	**0,6645**
Health	0,6812	0,7408	0,7127	0,6732	0,7032	**0,7705**
Medical	0,7256	0,8677	0,8366	0,8008	0,7478	**0,8864**
Recreation	0,4548	0,5572	0,5816	0,3889	0,4686	**0,5934**
Reference	0,6194	0,6578	0,6256	0,5674	0,6368	**0,6942**
Science	0,5328	0,5006	0,5106	0,4410	0,5015	**0,5592**
Slashdot	0,4743	0,6459	0,6192	0,5092	0,5098	**0,6745**
Social	0,7482	0,7262	0,7282	0,7134	0,7234	**0,7646**
Society	0,6125	0,5717	0,6078	0,5661	0,6002	**0,6168**
Mean	0,6211	0,6567	0,6526	0,5689	0,6212	**0,6899**

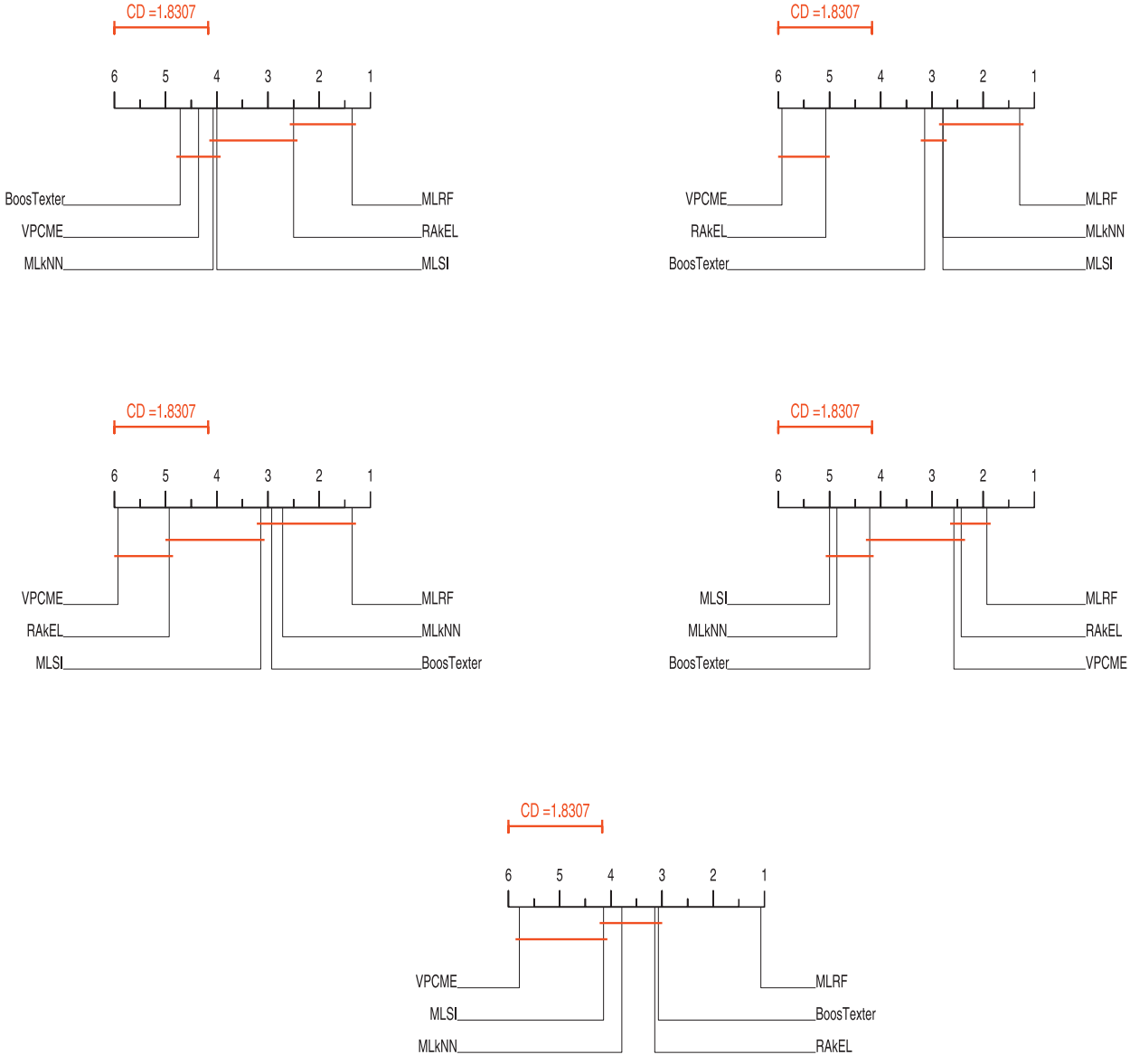

Fig. 1. Average ranks diagrams of the compared multi-label algorithms in terms of Hamming loss, Ranking loss, Coverage, One error and Average precision (from left to right).

- Considering the Ranking loss measure, MLRF achieves a performance superior to all other approaches but comparable to MLkNN. Here again, the Wilcoxon signed-rank test reveals that MLRF significantly outperforms MLkNN at $p = 0.05$.
- As far as the Coverage is concerned, MLRF ranks first. However, its performance is not statistically distinguishable from the performance of MLkNN, BoosTexter and MLSI. According to the Wilcoxon signed-rank test, MLRF significantly outperforms these three approaches at $p = 0.05$.
- A closer inspection of the results in terms of One error criterion reveals that ensemble methods, *i.e.* MLRF, RAkEL, VPCME and BoosTexter, achieve better performances than those of single methods including MLkNN and MLSI. This confirms the effectiveness of these approaches to rank properly the first relevant label. The Wilcoxon signed-rank test does not reveal significant differences at $p = 0.05$ within the first group of algorithms (*i.e.* MLRF, RAkEL and VPCME). These approaches have seemingly similar performances.
- Regarding the Average precision results reported in Table 6 and Fig. 1, MLRF significantly outperforms the other approaches by a noticeable margin.

Overall, the conclusion that can be drawn upon inspection of the previous results is the ability of our approach to construct a multi-label ensemble classifier beneficial for text categorization. MLRF ranks the labels more satisfactorily thanks to more accurate label probability estimates. Compared to BoosTexter and VPCME ensemble approaches for which base-classifiers are combined using majority 0/1 voting scheme, the probability voting strategy in MLRF shows promise for obtaining a ML ensemble classification framework that enjoys significant improvements especially in terms of probability-based ranking metrics (*I.e.* Ranking loss, Coverage, One error, Average precision).

The diversity-accuracy analysis performed in the next section will shed broader light on this comparison and attempt to explain the reasons why MLRF works apparently better than the two other multi-label ensembles (BoosTexter and VPCME),

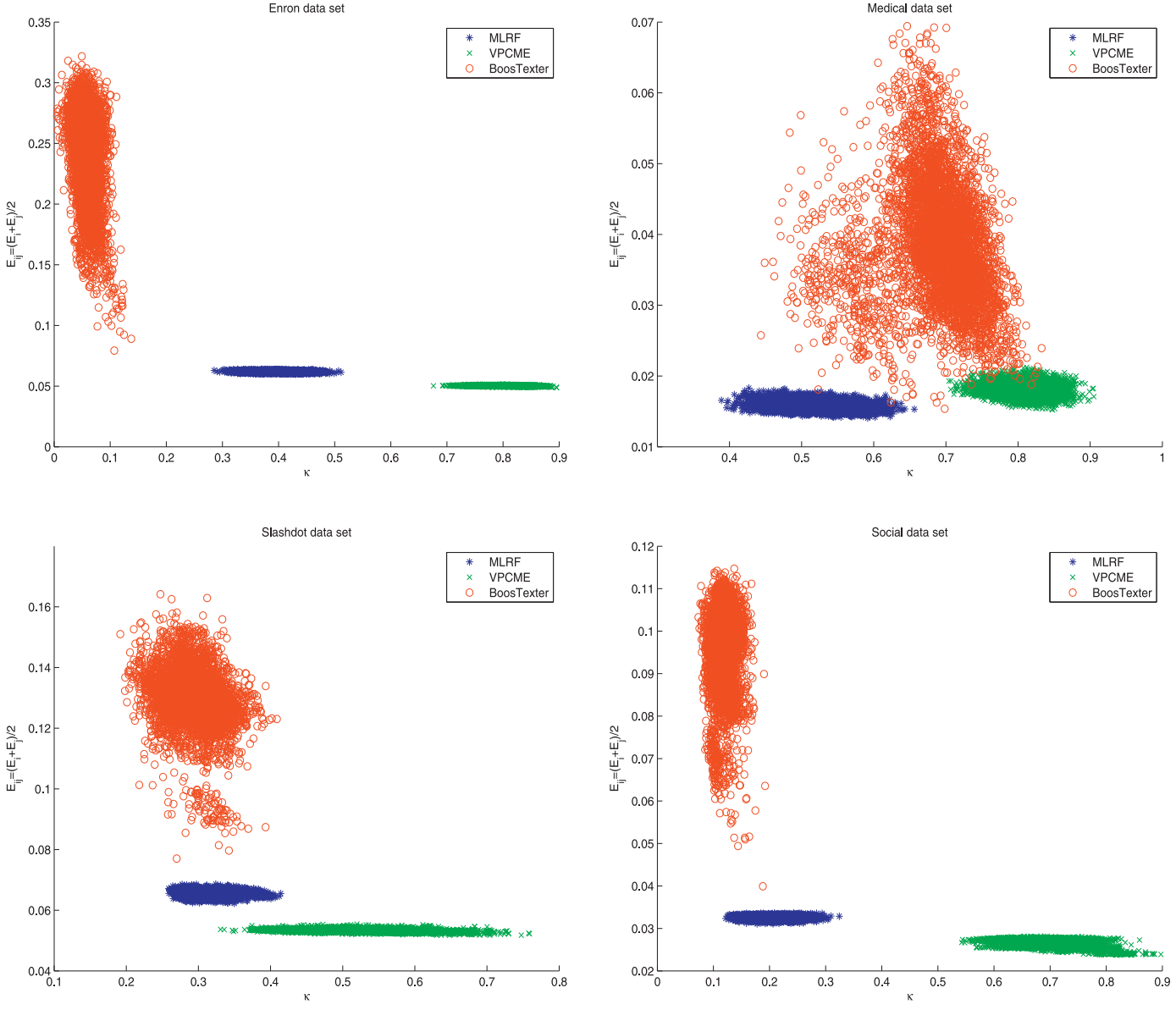

Fig. 2. Kappa-error diagrams of multi-label ensemble approaches on four representative data sets.

4.5. Diversity-error diagrams analysis

To achieve better predictions than individual classifiers, it is widely agreed that the ensemble should consist of base learners that are as accurate and diverse as possible. We use the kappa-error diagrams proposed in (Margineantu & Dietterich, 1997) to illustrate the diversity-accuracy dilemma of the ensemble. Kappa-error diagrams are scatterplots with $N \times (N-1)/2$ points, where N stands for the ensemble size. Each pair of classifiers is depicted as a point in the diagram. On the x-axis is a measure of pairwise diversity, κ. The y-axis denotes the averaged individual error of the two classifiers in the pair, $E_{i,j} = (E_i + E_j)/2$ (Kuncheva & Rodríguez, 2007). In this paper, we propose an extended version of the kappa-error diagram for multi-label learning. To measure the diversity between a multi-label pair, we consider the mean of κ over all labels. The individual errors E_i and E_j are given by the Hamming-loss metric. As small values of κ indicate higher diversity and small values of $E_{i,j}$ indicate higher performance; the dots of a good ensemble should lie in the bottom left corner (Rodríguez et al., 2006).

Fig. 2 shows the kappa-error diagrams of the three multi-label ensemble-based approaches (BoosTexter, MLRF and VPCME) for four representative data sets. They consist of Social, a data set among the 11 Yahoo ones as well as the three remaining Mulan (Tsoumakas et al., 2011b) data sets: Enron, Medical and Slashdot. These data sets were chosen since they have more than a thousand of features and for which the number of labels varies from 20 to over 50 with an increment of 10. For each data set, the ensembles were built on the training sets and the kappa-error plots were performed on the testing sets. As in the previous section, the ensemble size for all ensemble-based approaches was tuned to 100, thus there are 4950 points in each kappa-error diagram.

A closer inspection of plots in Fig. 2 reveals the following: (1) not surprisingly, the more accurate the individual classifiers are, the less the diversity, and vice versa. (2) MLRF builds individual multi-label classifiers that are slightly less diverse, on average, than those in BoosTexter but rather more accurate. On the other hand, they are slightly less accurate than those in VPCME but rather more diverse. The plots indicate that MLRF has the potential to improve on individual accuracy significantly without compromising the diversity. This validates the motivation behind MLRF that enhancing both criteria (diversity and accuracy) pays off as the differences in multi-label ensemble performances for text categorization (*c.f* Section 4.4) are found to be statistically significant in favor of our approach.

5. Conclusion

Nowadays, with the explosive growth of numeric documents, the text categorization faces great challenge, especially when documents are assigned to multiple categories. In this work, we presented a multi-label text categorization approach called MLRF, combining ideas from Rotation forest and LSI. The main novelty is in the approach taken to construct the committee. It is based on four key ideas: (1) performing LSI based on distinct orthogonal projections on lower-dimensional spaces of concepts, (2) random splitting of the vocabulary, (3) document bootstrapping, and (4) the use of a powerful multi-label base learner (BoosTexter) for text categorization. The combination of these components allows to simultaneously encourage diversity and individual accuracy in the committee; thus leading to improve the discrimination power of our approach with better understanding of the semantic meaning of textual data. Diversity of the ensemble is promoted through random splits of the vocabulary that leads to different orthogonal projections on lower-dimensional latent concept spaces. Accuracy of the committee members is promoted through the underlying latent semantic structure uncovered in the text. Both the LSI-based projection on random splits of the vocabulary and the training of the base learner classifiers can be conducted independently. The method was shown to bring about significant improvements in terms of *Average Precision, Coverage, Ranking loss* and *One error* compared to five state-of-the-art approaches across 14 real-word textual data sets covering a wide variety of topics including health, education, business, science and arts.

The downside is that the committee size, the number of iterations performed by the base multi-label learners and the dimensionality reduction parameter for the LSI are hyperparameters that need to be fixed or estimated prior to learning. It is unknown to what degree these parameters impact the final performances. This requires further investigations. On the other hand, MLRF requires a sufficient amount of labeled data for training a high quality ensemble model. Nevertheless, labeling text data with more than one label is extremely expensive and time-consuming in many real-world applications. Future work will also aim to extend MLRF to the semi-supervised text categorization context (Wei, Yang, Junping, & Wang, 2009; Yang, Sun, Wang, & Chen, 2009). Finally, as MLRF is amenable to parallel execution, we also plan to implement MLRF on the Apache Opensource Hadoop Map/Reduce framework for handling large textual data sets.

References

Bibimoune, M., Elghazel, H., & Aussem, A. (2013). An empirical comparison of supervised ensemble learning approaches. In *International workshop on complex machine learning problems with ensemble methods copem@ecml/pkdd'13* (pp. 123–138).

Blockeel, H., Raedt, L. D., & Ramon, J. (1998). Top-down induction of clustering trees. In *Icml* (pp. 55–63).

Boutell, M. R., Luo, J., Shen, X., & Brown, C. M. (2004). Learning multi-label scene classification. *Pattern Recognition, 37*(9), 1757–1771.

Breiman, L. (1996). Bagging predictors. *Machine Learning, 26*(2), 123–140.

Breiman, L. (2001). Random forests. *Machine Learning, 45*(1), 5–32.

Briggs, F., Fern, X. Z., & Irvine, J. (2013). Multi-label classifier chains for bird sound. In *Workshop on Machine Learning for Bioacoustics ICML 2013*.

Clare, A., & King, R. D. (2001). Knowledge discovery in multi-label phenotype data. In *Pkdd* (pp. 42–53).

Deerwester, S. C., Dumais, S. T., Landauer, T. K., Furnas, G. W., & Harshman, R. A. (1990). Indexing by latent semantic analysis. *Journal of the American Society for Information Science, 41*(6), 391–407.

Dembczynski, K., Waegeman, W., Cheng, W., & Hüllermeier, E. (2012). On label dependence and loss minimization in multi-label classification. *Machine Learning, 88*(1-2), 5–45.

Demsar, J. (2006). Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research, 7*, 1–30.

Dietterich, T. (2000). Ensemble methods in machine learning. In *First international workshop on multiple classier systems* (pp. 1–15).

Elisseeff, A., & Weston, J. (2005). A kernel method for multi-labelled classification. In *Proceedings of the annual ACM conference on research and development in information retrieval* (pp. 274–281).

Freund, Y., & Shapire, R. (1996). Experiments with a new boosting algorithm. In *13th international conference on machine learning* (pp. 276–280).

Fujino, A., Isozaki, H., & Suzuki, J. (2008). Multi-label text categorization with model combination based on f1-score maximization. In *Ijcnlp* (pp. 823–828).

Gasse, M., Aussem, A., & Elghazel, H. (2015). On the optimality of multi-label classification under subset zero-one loss for distributions satisfying the composition property. In *32nd international conference on machine learning, ICML* (pp. 2531–2539).

Gharroudi, O., Elghazel, H., & Aussem, A. (2015). Ensemble multi-label classification: a comparative study on threshold selection and voting methods. In *27th IEEE international conference on tools with artificial intelligence (ictai)* (pp. 377–384).

He, X., Cai, D., Liu, H., & Ma, W.-Y. (2004). Locality preserving indexing for document representation. In *ACM SIGIR* (pp. 96–103).

Ho, T. K. (1998). The random subspace method for constructing decision forests. *IEEE Transactions Pattern Analysis and Machine Intelligence, 20*(8), 832–844.

Joachims, T. (2002). *Learning to classify text using support vector machines: Methods, theory and algorithms*. Kluwer Academic Publishers.

Kocev, D., Vens, C., Struyf, J., & Dzeroski, S. (2013). Tree ensembles for predicting structured outputs. *Pattern Recognition, 46*(3), 817–833.

Kuncheva, L. I., & Rodríguez, J. J. (2007). An experimental study on rotation forest ensembles. In *7th international workshop of multiple classifier systems (MCS)* (pp. 459–468).

Li, P., Li, H., & Wu, M. (2013). Multi-label ensemble based on variable pairwise constraint projection. *Information Science, 222*, 269–281.

Liu, T., Chen, Z., Zhang, B., Ma, W., & Wu, G. (2004). Improving text classification using local latent semantic indexing. In *ICDM* (pp. 162–169).

Madjarov, G., Kocev, D., Gjorgjevikj, D., & Dzeroski, S. (2012). An extensive experimental comparison of methods for multi-label learning. *Pattern Recognition, 45*(9), 3084–3104.

Margineantu, D. D., & Dietterich, T. G. (1997). Pruning adaptive boosting. In *ICML* (pp. 211–218).

Nanculef, R., Flaounas, I., & Cristianini, N. (2014). Efficient classification of multi-labeled text streams by clashing. *Expert Systems with Applications, 41*(11), 5431–5450.

Read, J., Pfahringer, B., Holmes, G., & Frank, E. (2009). Classifier chains for multi-label classification. In *Proceedings of the 20th European conference on machine learning* (pp. 254–269).

Rodríguez, J. J., Kuncheva, L. I., & Alonso, C. J. (2006). Rotation forest: A new classifier ensemble method. *IEEE Trans. Pattern Anal. Mach. Intell., 28*(10), 1619–1630.

Rokach, L., Schclar, A., & Itach, E. (2014). Ensemble methods for multi-label classification. *Expert Systems with Applications*.

Sajnani, H., Javanmardi, S., McDonald, D. W., & Lopes, C. V. (2011). Multi-label classification of short text: A study on wikipedia barnstars. In *Analyzing microtext (aaai workshops)*.

Schapire, R. E., & Singer, Y. (2000). Boostexter: A boosting-based system for text categorization. *Machine Learning, 39*(2/3), 135–168.

Sebastiani, F. (2002). Machine learning in automated text categorization. *ACM computing surveys (CSUR), 34*(1), 1–47.

Skillicorn, D. (2007). *Understanding complex datasets: data mining with matrix decompositions*. CRC press.

Tsoumakas, G., Katakis, I., & Vlahavas, I. P. (2011a). Random k-labelsets for multilabel classification. *IEEE Transactions Knowledge and Data Engineering, 23*(7), 1079–1089.

Tsoumakas, G., Xioufis, E.S., Vilcek, J., & Vlahavas, I. (2011b). MULAN multi-label dataset repository.

Vilar, D., Castro, M. J., & Sanchis, E. (2004). Multi-label text classification using multinomial models. In *EsTAL* (pp. 220–230).

Wang, J., Peng, J., & Liu, O. (2015). A classification approach for less popular webpages based on latent semantic analysis and rough set model. *Expert Systems with Applications, 42*(1), 642–648.

Wang, S., Wang, J., Wang, Z., & Ji, Q. (2014). Enhancing multi-label classification by modeling dependencies among labels. *Pattern Recognition, 47*(10), 3405–3413.

Wei, Q., Yang, Z., Junping, Z., & Wang, Y. (2009). Semi-supervised multi-label learning algorithm using dependency among labels. In *International conference on machine learning and computing mining (ICDM 2010)* (pp. 112–116).

Wiemer-Hastings, P. (1999). How latent is latent semantic analysis? In *IJCAI'* (pp. 932–937).

Yang, B., Sun, J., Wang, T., & Chen, Z. (2009). Effective multi-label active learning for text classification. In *ACM SIGKDD* (pp. 917–926).

Yu, K., Yu, S., & Tresp, V. (2005). Multi-label informed latent semantic indexing. In *SIGIR* (pp. 258–265).

Yu, Y., Pedrycz, W., & Miao, D. (2014). Multi-label classification by exploiting label correlations. *Expert Systems with Applications, 41*(6), 2989–3004.

Zhang, C.-X., & Zhang, J.-S. (2008). Rotboost: a technique for combining rotation forest and adaboost. *Pattern Recognition Letters, 29*(10), 1524–1536.

Zhang, M.-L., & Zhou, Z.-H. (2006). Multilabel neural networks with applications to functional genomics and text categorization. *IEEE Transactions on Knowledge and Data Engineering, 18*(10), 1338–1351.

Zhang, M.-L., & Zhou, Z.-H. (2013). A review on multi-label learning algorithms. *IEEE Transactions on Knowledge and Data Engineering, 99*, 1.

Zhang, W., Yoshida, T., & Tang, X. (2011). A comparative study of tf*idf, LSI and multi-words for text classification. *Expert Systems with Applications, 38*(3), 2758–2765.

Zhang, M. L., & Zhou, Z.-H. (2007). Ml-knn: a lazy learning approach to multi-label learning. In *Pattern recognition: 40* (pp. 2038–2048).

Zhou, X., Zhang, X., & Hu, X. (2006). Using concept-based indexing to improve language modeling approach to genomic IR. In *ECIR* (pp. 444–455).

Zhou, Z., & Zhou, Z. (2012). Ensemble methods: foundations and algorithms. *Chapman & Hall/CRC Data Mining and Knowledge Discovery Serie*. Taylor & Francis.