Journal of Software Engineering

ISSN 1819-4311

Academic Journals Inc.

www.academicjournals.com

Journal of Software Engineering 10 (1): 42-53, 2016
ISSN 1819-4311 / DOI: 10.3923/jse.2016.42.53

Text Classification Based on a Novel Cost-Sensitive Ensemble Multi-Label Learning Method

[1]Haifeng Hu, [1]Tao Zhang and [2]Jiansheng Wu
[1]Department of Telecommunication and Information Engineering,
[2]Department of Geographic and Biological Information, Nanjing University of Posts and Telecommunications, Nanjing, 210046, China

Corresponding Author: Haifeng Hu, Department of Telecommunication and Information Engineering, Nanjing University of Posts and Telecommunications, Nanjing, 210046, China Tel: +86 13813814879

ABSTRACT

Text classification is one of the most important tasks in the Natural Language Processing research field. In most cases, text classification is usually a multi-label learning task where, three attributes (i.e., information gain, document frequency and chi-square test values) are widely used to describe documents and the degree of importance of each attribute varies depending on different applications. Hence, it is valuable to improve the prediction performance of text classification by assembling the above attributes. Furthermore, there exists a widespread problem of class imbalance in multi-label learning algorithm. Thus, in this study, a novel cost-sensitive ensemble multi-label learning method CS-EnMLKNN is proposed to assemble the attributes in text classification and deal with the class imbalance problem and a comprehensive framework for solving text classification problems is also proposed accordingly. Finally, experiments on two classic datasets show that our CS-EnMLKNN algorithm outperforms most state-of-the-art multi-label learning algorithms in terms of several learning evaluation criteria.

Key words: Text classification, multi-label learning, cost-sensitive, class imbalance

INTRODUCTION

With the exponential growth of documents available from the Internet, text classification has become a significant tool to organize and manage these texts efficiently (Tan, 2006). Furthermore, text classification is also a hot research field in information retrieval, machine learning and natural language processing (Tan, 2006). Text classification is an application, which aims at assigning one or more predefined labels to free text documents (Manning and Schütze, 1999). Thus, text classification is usually a multi-label learning task. For instance, a news report about 2014, World Cup in Brazil belongs to several predefined labels at the same time, such as “sports”, “World Cup” and “Brazil”. In multi-label learning framework, each instance is described by an attribute vector and each instance in the training set is associated with a set of labels and the task of multi-label learning algorithm is to predict a label set for each unseen instance (Zhang and Zhou, 2005).

During the past decades, multi-label learning is often solved by degrading into Binary Relevance (BR) problems, which are to learn a binary classifier for each label (Boutell *et al*., 2004; Tsoumakas and Katakis, 2006; Yang, 1999). However, such binary relevance methods cannot

consider label correlations and may not able to accurately predict label sets. Recently, several multi-label learning methods have been applied to the task of text classification (Gao *et al*., 2004; McCallum, 1999; Mitra *et al*., 2006; Schapire and Singer, 2000) to improve the performance of classification.

In Year 1999, Macallum had proposed a Bayesian approach for multi-label text classification (McCallum, 1999) based on a mixture probabilistic model and EM (Dempster *et al*., 1977) algorithm is used to learn the weights and the word distributions in mixture component. Schapire and Singer had proposed the BoosTexter method to keep a set of weights on both training instances and their corresponding labels in the training stage where, instances and their labels that are easy to predict will get lower weight. By using independent word-based Bag-of-Words representation, Ueda and Saito (2002) had proposed 2 kinds of probabilistic generative models for multi-label text classification, which are called parametric mixture models PMMs and the basic assumption of PMMs is that a multi-label text has a mixture of characteristic words in a single-label text. In Year 2003, Gao *et al*. (2004) had extended the Maximal Figure-of-Merit (MFoM) (Gao *et al*., 2003) in single-label to multi-label learning area and proposed a method assigning a uniform score function to each label for each given test instance based on classical Bayes decision rules. Especially, in Year 2007, Zhang and Zhou (2007) had proposed the MLKNN method derived from the K-Nearest Neighbor (KNN) method. In the MLKNN model, the neighbors of each new instance are firstly identified and the Maximum A Posteriori (MAP) principle is utilized to determine the label set for the new instance based on the label sets of its neighboring instances. Due to its simplicity and efficiency, MLKNN is a widely used classifier in the multi-label learning tasks, such as in text classification.

In Year 2014, we had proposed En-MLKNN (Zhang *et al*., 2014) to solve text classification problem. In this study, we observe that there exists an outstanding class imbalance in the text classification tasks. Class imbalance occurs when the number of instances from one label is much smaller than from another label (Tahir *et al*., 2009), which tends to assign labels with larger number of instances to the test instance and lead to a lower algorithm performance over the minority class (Soda, 2011). It's a natural that some labels (classes) contain many instances and some labels (classes) contain few due to a hot pot problem. Thus, it is worth noting that the research of multi-label algorithms with imbalanced training set is a significant issue (Soda, 2011). Several approaches had been proposed to solve it, such as up-sampling (Chawla *et al*., 2002; Kubat *et al*., 1997; Lee, 2000), down-sampling (Chen *et al*., 2005; Kubat and Matwin, 1997). In particular, cost-sensitive learning has been considered as a good method to the class imbalance problem. It means the misclassifying a minor class test instance costs more than misclassifying a major class (Wu and Zhou, 2013).

In this study, a novel method CS-EnMLKNN has been proposed by integrating the cost-sensitive learning into our previous model En-MLKNN. Experiments on two classic text datasets show that CS-EnMLKNN is superior to the En-MLKNN method and some other state-of-the-art multi-label learning methods.

MATERIALS AND METHODS

Datasets preparation: There are several steps for preparing the datasets for multi-label text classification, mainly including datasets collection, pre-processing, document transformation.

In this study, we have collected two classic datasets, i.e., Reuters-21578 and 20 Newsgroups. Reuters-21578 consists of documents from Reuters column in 1987 and 135 labels, where every document has several labels. There exists an outstanding multi-label class imbalance due to the fact that some labels in the Reuters-21578 dataset only cover very few instances. The 20 News groups dataset contains about 20,000 texts collected from the Usenet News groups and 20 labels where every label covers nearly 1,000 documents.

Documents pre-processing is the first step to transform texts, that is sequence of words, into a representation suitable for the learning algorithm. It includes the following steps: removing tags, removing stop words and word stemming. In the text classification tasks, documents often contain many meaningless tags, which should be removed. Stop words are frequent word and often carry no information, i.e., pronouns, prepositions, conjunctions etc. In the documents, there exist many group words, which have the same concept and we can remove the suffix of words to generate word stems.

Document transformation is an important step to encode the documents for multi-label text classification (Chen *et al*., 2007).

Attributes: Currently, there are three kinds of attributes which is widely used in text classification tasks i.e., Document Frequency (DF), Information Gain (IG) and chi-square test values.

Document frequency for each term is the number of documents in which it occurs (Yang and Pedersen, 1997). In this study, we have selected such terms, whose document frequency is not less than the predetermined threshold. Because the widely accepted hypothesis in information retrieval is that low-document frequency terms are very informative for label prediction. However, document frequency works well for high-document frequency term.

Information gain measures the number of information bits for label prediction by considering a word's presence or absence at document (Yang and Pedersen, 1997). If $\{C_l\}_{l=1}^{Q}$ represents the label in label set y. The Information gain is defined as:

$$G(t) = -\sum_{l=1}^{Q} P_r(C_l)\log P_r(C_l) + P_r(t)\sum_{l=1}^{Q} P_r(C_l|t)\log P_r(C_l|t) + P_r(\bar{t})\sum_{l=1}^{Q} P_r(C_l|\bar{t})\log P_r(C_l|\bar{t}) \quad (1)$$

where the term according to all labels is measured on the average.

Chi-square test values measures the lack of independence of a term t and label and compares them to χ^2 distribution with one degree of freedom to judge extremeness (Yang and Pedersen, 1997). With term t and $\{C_l\}_{l=1}^{Q}$ in y, the value is defined by:

$$\chi^2(t, C_l) = \frac{N(AD-CB)^2}{(A+C)(B+D)(A+B)(C+D)} \quad (2)$$

The three attributes have been applied widely in many applications including text classification. However, the above attributes hold different meanings. Information gain only measures the goodness of a term with respect to all labels as a whole. The CHI is known not to be reliable for low frequency terms and document frequency is typically not suitable for aggressive term removal due to the assumption that low document frequency terms are assumed to be highly informative and should be selected in information retrieval. It is valuable to improve the prediction performance of text classification by assembling the above attributes.

Formulation as a multi-label learning task: Formally, let $\{(X_i, Y_i)\}$ (i = 1,2,…n) be the training dataset of n examples where X_i denotes the i-th text in the training set and Y_i denotes the labels set assigned to X_i. Further, let w_{ij} be the j-th term of the text X_i and $Y = \{C_1, C_2,…C_Q\}$ be the label set of Q labels. A label vector $Y_i = (y_{1i}, y_{2i},…y_{Qi})$ can be defined as:

$$y_{ki} = \begin{cases} 1\ , & X_i \text{ belongs to } C_k \\ 0\ , & X_i \text{ does not belong to } C_k \end{cases} \tag{3}$$

Moreover, X_i can be represented as a vector $(w^p_{i1}, w^p_{i2},...w^p_{iF})$, where F depends on the size of feature space and let $p \in \eta = \{\text{"DF","IG", "CHI"}\}$ be an attributes set. The j-th element of X_i can be obtained by different term weights algorithm. A well known approach to computing term weights is the TF-IDF weighting which assigns the weight to term j in text i in proportion to the number of occurrences of the term j in the text i and in inverse proportion to the size of the text set where the term j occurs at least once. It should be pointed out that TF-IDF does not take into account different length of texts. Thus, TFC-weighting is used instead of TF-IDF to normalize length in term weighting equation (Salton and Buckley, 1988), define:

$$w^p_{ij} = \frac{f_{ij} * \log(\frac{N}{n_i})}{\sqrt{\sum_{k=1}^{F}[f_{kj} * \log(\frac{N}{n_k})]^2}} \tag{4}$$

Moreover, in the training process, the prediction task is to predict a set of proper labels Y_t for a test instance X_t, which can be formalized as a multi-label learning problem (Zhou *et al*., 2012), where multi-label can be obtained by learning a function $f: X \rightarrow 2^Y$ from a training set $\{(X_i, Y_i)\}$. Note that since there is no explicit relationship between an instance X_i and a label $y_{ki} \in Y_i$.

CS-EnMLKNN algorithm: In this study, we propose the CS-EnMLKNN algorithm to assemble three attributes and deal with the class-imbalance problem. As mentioned above, in order to find a solution to class-imbalance problem encountered in En-MLKNN. We employ TFC-weighting to normalize the text length and cost-sensitive learning for improving the performance of classification.

Let $N^p_t = \{X^p_{v_1}, X^p_{v_2},...X^p_{v_K}\}$ be the set of the K-nearest neighbors for the test instance X_t and $n^p_t = \{n^{pt}_1, n^{pt}_2,...n^{pt}_Q\}$ be the label count vector for X_t where:

$$n^{pt}_k = \sum_{r=v_1}^{v_K} y_{kr} (1 \le k \le Q) \tag{5}$$

Moreover, the label-vector Y_t of X_t can be obtained by

$$Y_t = (\sum_p w^p * y^p_{1t}, \sum_p w^p * y^p_{2t},...\sum_p w^p * y^p_{Qt})$$

Subject to

$$0 \le w^p \le 1, \sum_p w^p = 1 \tag{6}$$

where weight can be estimated using cost-sensitive Maximum A Posteriori (MAP) method.

The value of w^p has been specified before the classification and can be changed to other value between test instances. The degree of importance of each attribute varies depending on different applications. Let w^p be different weights assigned to attributes. Various attributes can be assembled to improve the prediction performance of text classification. For example, we can increase the weight of information gain if we use the globally feature space. We decrease the weight of CHI, if low-document frequency terms occur. And we can use document frequency for choosing different feature spaces for each label, when the weight of document frequency is set as 1. Here, we first show how cost-sensitive works in MLKNN as follows:

$$y_{kt}^p = \begin{cases} 1, & \text{if } P(H_{kt} = 1 | E = n_k^{pt}) \ge P(H_{kt} = 0 | E = n_k^{pt}) \\ 0, & \text{if } P(H_{kt} = 0 | E = n_k^{pt}) \ge P(H_{kt} = 1 | E = n_k^{pt}) \end{cases} \tag{7}$$

where H_{kt} is set as 1 when X_t belongs to label C_k and vice-versa and let E be the number of texts in N_t^p belonging to label C_k. According to Bayes' rule, we have

$$P(H_{kt} = b | E = n_k^{pt}) = \frac{P(H_{kt} = b) P(E = n_k^{pt} | H_{kt} = b)}{P(E = n_k^{pt})} \tag{8}$$

where b = 0 or 1. According to MLKNN (Zhang and Zhou, 2007), Eq. 8 can be expressed as

$$y_{kt}^p = \begin{cases} 1, & \text{if } P(H_{kt} = 1) P(E = n_k^{pt} | H_{kt} = 1) > P(H_{kt} = 0) P(E = n_k^{pt} | H_{kt} = 0) \\ 0, & \text{if } P(H_{kt} = 0) P(E = n_k^{pt} | H_{kt} = 0) > P(H_{kt} = 1) P(E = n_k^{pt} | H_{kt} = 1) \end{cases} \tag{9}$$

However, according to MLCKNN (Han and Li, 2014), Eq. 8 can be rewritten as

$$y_{kt}^p = \begin{cases} 1, \text{if } C*P(H_{kt} = 1) P(E = n_k^{pt} | H_{kt} = 1) / (C*P(H_{kt} = 1) P(E = n_k^{pt} | H_{kt} = 1) + P(H_{kt} = 0) P(E = n_k^{pt} | H_{kt} = 0)) \ge 0.5 \\ 0, \text{if } C*P(H_{kt} = 1) P(E = n_k^{pt} | H_{kt} = 1) / (C*P(H_{kt} = 1) P(E = n_k^{pt} | H_{kt} = 1) + P(H_{kt} = 0) P(E = n_k^{pt} | H_{kt} = 0)) < 0.5 \end{cases} \tag{10}$$

where C is the cost assigned to classifier for misclassifying the minor label (class). Therefore, the classifier will be prone to identify minor label and assign the minor label to test instance and the class imbalance problem can be tackled effectively. Notice that $P(H_{tk})$ and $P(E|H_{tk})$ can be obtained with the training data (Zhang and Zhou, 2007). Although our algorithm is based on the text classification, it can be extended to general multi-label learning application.

The CS-EnMLKNN algorithm is described in Fig. 1.

Framework of text classification tasks with the CS-EnMLKNN model: According to CS-EnMLKNN algorithm, a complete framework is constructed for text classification in Fig. 2.

Y = CS-EnMLKNN (D, K, η, X, C)
Input:
D: The training instances
K: K-nearest neighbors
η: The kind of attributes {"DF", "IG", "CHI"}
X: Test instance
C: Cost parameter
Output:
Y: The predicted label vector of X
Process:
1: Identify N_X^p
2: For p∈η do
3: For k∈y = $\{y_1, y_2,...y_Q\}$ do
4: Compute y_{kX}^p using Eq. 10
5: End for
6: End for
7: For X, its prediction label vector can be obtained by
8: $Y_X = (\sum_p w^p * y_{1X}^p, \sum_p w^p * y_{2X}^p, ... \sum_p w^p * y_{QX}^p)$

Fig. 1: Pseudo code of CS-EnMLKNN

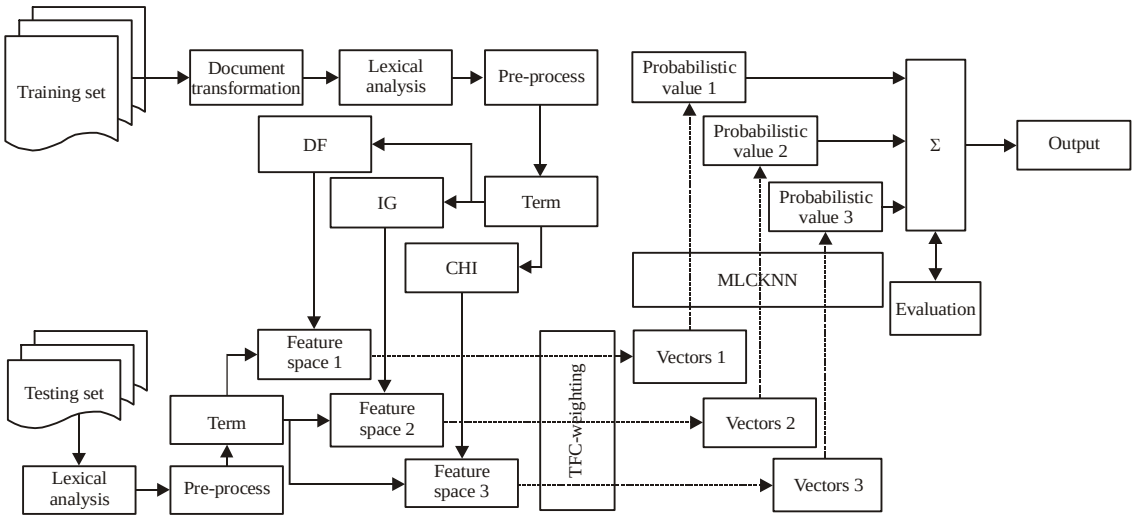

Fig. 2: En-MLKNN text classification framework

There are two stages in the framework: training stage and testing stage. In training stage, the label set of each document in training set is obtained and classify model is learned. In testing stage, test documents and some evaluation criterions are used to test the learned model.

According to the framework, the tasks in training stage include: (1) Document transformation (Chen *et al*., 2007) and pre-process is performed to get training terms, (2) Three feature spaces of information gain, CHI and document frequency are constructed by selecting terms from training terms, respectively, (3) There exist 6 vectors for each text in the above feature spaces, 2 vectors for each feature space and (4) the training set can be obtained based on the training stage of MLKNN. The tasks in testing stage include: (1) Similar to (3) in training stage, there exist 6 vectors for a test document. We assume that the label set is unknown, (2) For test document vectors, the

predicted probabilities under different feature spaces are computed using MLCKNN algorithm, respectively, (3) We assign different weights to different predicted probabilities and compute the weighted sum of all predicted values and (4) The weights can be adjusted for better performance according to evaluation criterion.

Classifier criterion: In single-label learning frame, the evaluation criterions are precision, accuracy, recall and F-measure (Liu *et al*., 2006; Sebastiani, 2002). In multi-label learning frame, the following criterions have come up with in (Schapire and Singer, 2000) for testing collection $S = \{(w_i, y_i) | 1 \le i \le p\}$ and classifier $f(\cdot)$.

- **One-error:** Calculates the number of times that the top-ranked labels aren't included in the collection of right labeled set for test input. The larger error (f) is, the worse the algorithm performance is:

$$\text{Error}(f) = \frac{1}{p}\sum_{i=1}^{p}\left\|\arg\max_{y\in y} f(w_i, y) \notin y_i\right\| \tag{11}$$

  where, is set to 1 if condition π is met, otherwise should be set as 0
- **Hamming loss:** Calculates the number of times that an input label pair is wrong labeled. The larger ham loss (f), the worse the algorithm performance is

$$\text{Hamloss}(f) = \frac{1}{p}\sum_{i=1}^{p}\frac{1}{Q}\left|h(w_i)\Delta y_i\right| \tag{12}$$

  where, Δ represents the difference between 2 collections
- **Coverage:** Measures how long the list of sets is obtained to overlap entire right labels of the input. The larger cov(f), the worse the algorithm performance is

$$\text{cov}(f) = \frac{1}{p}\sum_{i=1}^{p}\max_{y\in Y_i}\text{rank}_f(w_i, y) - 1 \tag{13}$$

- **Ranking loss:** Calculates the mean fraction of label couples sorted in reverse order for input. The larger rloss(f), the worse the algorithm performance is

$$\text{Rloss}(f) = \frac{1}{p}\sum_{i=1}^{p}\frac{1}{|y_i|\left|\overline{y_i}\right|}\times\left|\left\{\left(y', y''\right) | f(w_i, y') \le f(w_i, y''), \left(y', y''\right) \in y_i \times \overline{y_i}\right\}\right| \tag{14}$$

  where $\overline{y_i}$ denotes the complementary set of y_i
- **Average precision:** Calculates the mean fraction of labels sorted upon a unique label $y \in y$, which are really included in y. The smaller avgprec(f), the worse the algorithm performance is

$$\text{Avgprec}(f) = \frac{1}{p}\sum_{i=1}^{p}\frac{1}{|y|}\times\sum_{y\in Y_i}\frac{\left|\{y | \text{rank}_f(w_i, y) \le \text{rank}_f(w_i, y), y \in y_i\}\right|}{\text{rank}_f(w_i, y)} \tag{15}$$

RESULTS AND DISCUSSION

Performance of the CS-EnMLKNN models: Table 1 and 2 compare the performance of our proposed algorithm with other approaches on Reuters-21578 and 20 Newsgroups. The experiments are implemented on Window 7 platform (x32) with 4×2.6 G CPU processor and 2 G memory. In pre-processing stage, we remove possible punctuations using stop word list in (Lewis, 1992), every letter is transformed into lowercase and numbers are deleted for processing convenience. The number of k-nearest neighbor is set to 10 as recommended in (Zhang and Zhou, 2007). We repeat ten-fold cross validation (Kale *et al*., 2011) for 10 times and all experiment results are averaged.

In our experiments, Hamming Loss (HL), Ranking Loss (RL), One-Error (OE), Coverage and Average Precision (AP) are used as learning evaluation criteria. In the following tables, "↓" means "the larger the worse" and "↑" means "the smaller the worse". The better result of every evaluation criterion is described and represented in bold face. In Table 1 and 2, '1' represents the experiment where, TFC-weighting and MLKNN are used, '2' represents the experiment where TFC-weighting and MLCKNN are used. As shown in the following tables, our proposed algorithm achieves relatively better performance.

Performance comparison with other methods: We compared the CS-EnMLKNN method with state-of-the-art learning methods including BSVM (Boutell *et al*., 2004), RankSVM (Elisseeff and Weston, 2001) and TRAM (Kong *et al*., 2013). The codes of these algorithms are shared by their authors. For BSVM, the SVM (Kecman, 2001; Wang, 2005) is implemented by LIBSVM (Chang and Lin, 2011) package with radial basis function whose parameter "-g" selected from $\{2^{-8}, 2^{-6},\ldots 2^{6}, 2^{8}\}$ and parameters "-c" selected from $\{2^{-4}, 2^{-2},\ldots, 2^{6}, 2^{8}\}$ by tenfold cross validation. The optimal values

Table 1: Comparison result (Mean±STD) of CS-EnMLKNN models based on different FS on reuters-21578

Parameters	HL↓	RL↓	OE↓	Coverage↓	AP↑
1					
IG-MLKNN	0.1222±0.004	0.0125±0.004	0.0658±0.024	0.2620±0.050	0.9593±0.014
CHI-MLKNN	0.1180±0.004	0.0117±0.003	0.0653±0.019	0.2493±0.042	0.9607±0.011
DF-MLKNN	0.1242±0.005	0.0146±0.004	0.0845±0.019	0.2801±0.047	0.9492±0.011
En-MLKNN	0.1174±0.004	0.0087±0.003	0.0501±0.019	0.2160±0.043	0.9698±0.011
2					
IG-MLCKNN	0.1220±0.006	0.0125±0.004	0.0921±0.118	0.2620±0.050	0.9593±0.014
CHI-MLCKNN	0.1177±0.004	0.0117±0.003	0.0653±0.019	0.2496±0.042	0.9607±0.011
DF-MLCKNN	0.1261±0.008	0.0146±0.004	0.0845±0.019	0.2799±0.047	0.9492±0.011
CS-EnMLKNN	0.1130±0.004	0.0082±0.003	0.0482±0.018	0.2089±0.037	0.9714±0.010

HL: Hamming loss, RL: Ranking loss, OE: One-error and AP: Average precision

Table 2: Comparison result (Mean±STD) of CS-EnMLKNN models based on different FS on news-group

Parameters	HL↓	RL↓	OE↓	Coverage↓	AP↑
1					
IG-MLKNN	0.2960±0.029	0.3710±0.103	0.5913±0.132	5.8554±1.270	0.5237±0.112
CHI-MLKNN	0.3034±0.012	0.3612±0.043	0.5954±0.042	5.7892±0.569	0.5157±0.035
DF-MLKNN	0.2974±0.014	0.3575±0.065	0.5791±0.085	5.7784±0.878	0.5291±0.078
En-MLKNN	0.2907±0.018	0.3221±0.083	0.5436±0.118	5.4262±1.032	0.5557±0.102
2					
IG-MLCKNN	0.4515±0.088	0.3710±0.103	0.5913±0.132	5.8554±1.270	0.5237±0.111
CHI-MLCKNN	0.4439±0.047	0.3612±0.043	0.5954±0.042	5.7892±0.569	0.5157±0.035
DF-MLCKNN	0.3975±0.057	0.3575±0.065	0.5790±0.085	5.7785±0.878	0.5291±0.078
CS-EnMLKNN	0.4234±0.068	0.3243±0.085	0.5441±0.110	5.4575±1.074	0.5514±0.101

HL: Hamming loss, RL: Ranking loss, OE: One-error and AP: Average precision

Table 3: Comparison result (Mean±STD) with three state-of-the art methods on reuters-21578

Parameters	HL↓	RL↓	OE↓	Coverage↓	AP↑
IG					
BSVM	0.072±0.022	0.094±0.038	0.001±.001	2096±710	0.622±0.140
RankSVM	0.082±0.004	0.102±0.011	0.302±.030	1.119±.101	0.795±0.021
TRAM	0.078±0.003	0.643±0.011	0.683±0.055	0.662±0.025	0.561±0.024
CHI					
BSVM	0.079±0.027	0.103±0.034	0.067±0.058	2308±295	0.632±0.130
RankSVM	0.128±0.032	0.216±0.081	0.568±0.181	2.108±0.681	0.601±0.133
TRAM	0.081±0.002	0.639±0.023	0.638±0.058	0.637±0.018	0.595±0.042
DF					
BSVM	0.064±0.027	0.087±0.030	0.001±0.001	1759±513	0.706±0.127
RankSVM	0.126±0.036	0.215±0.080	0.567±0.187	2.090±0.703	0.602±0.137
TRAM	0.082±0.003	0.637±0.014	0.618±0.027	0.627±0.020	0.598±0.010
CS-EnMLKNN	0.113±0.004	0.008±0.003	0.048±0.018	0.209±0.037	0.971±0.010

HL: Hamming loss, RL: Ranking loss, OE: One-error and AP: Average precision, IG: Information gain, CHI: Chi-square test values, DF: Document frequency

Table 4: Comparison result (Mean±STD) with three state-of-the art methods on 20 news group

	HL↓	RL↓	OE↓	Coverage↓	AP↑
IG					
BSVM	0.236±0.002	0.455±0.004	0.591±0.450	1621±1.414	0.295±0.015
RankSVM	0.277±0.012	0.388±0.010	0.623±0.042	5.299±0.121	0.538±0.006
TRAM	0.315±0.010	0.327±0.024	0.317±0.044	0.322±0.034	0.357±0.009
CHI					
BSVM	0.232±0.002	0.383±0.001	0.333±0.278	1615±5.160	0.344±0.009
RankSVM	0.271±0.011	0.440±0.062	0.612±0.002	6.444±1.277	0.485±0.053
TRAM	0.308±0.007	0.355±0.009	0.371±0.010	0.363±0.009	0.364±0.005
DF					
BSVM	0.238±0.002	0.484±0.005	0.303±0.105	1620±2.572	0.282±0.006
RankSVM	0.276±0.004	0.473±0.018	0.609±0.010	7.058±.334	0.454±0.003
TRAM	0.318±0.011	0.338±0.017	0.361±0.020	0.349±0.018	0.287±0.023
CS-EnMLKNN	0.423±0.068	0.324±0.085	0.544±0.110	5.457±1.074	0.551±0.101

HL: Hamming loss, RL: Ranking loss, OE: One-error and AP: Average precision, IG: Information gain, CHI: Chi-square test values, DF: Document frequency

for parameters "-g" and "-c" are 2^{-4} and 2^{4}, respectively. The RankSVM algorithm is assigned the best setup-parameter's described in study (Elisseeff and Weston, 2001). As for RankSVM, cost parameter C is set to 1 and 8 degree is selected for polynomial kernels (Elisseeff and Weston, 2001). TRAM is also assigned with the recommended value described in study (Kong *et al*., 2013). For the number of nearest neighbors k is set as 10 and the number of dimension is determined by setting the threshold parameter of MDDM as preserving 99.99%. Table 3 describes the performance of every compared algorithm on two datasets. For each criterion, ↑ (↓) means that the smaller (larger) of the number, the worse the algorithm performance. In addition, the better result of every evaluation criterion is described and represented in bold face.

In Table 3 and 4, IG, CHI and DF represents the information gain, chi-square test values and document frequency used independently. Like symbol in Table 1 and 2, ↑ (↓) means that the smaller (larger) of the number, the worse the algorithm performance. In addition, the better result of every evaluation criterion is described and represented in bold face. As discussed in the above tables, our algorithm achieves the better performance than the state-of-the-art multi-label learning methods for two of five criteria. However, all the other algorithms in three feature spaces can achieve better performance only once at most. In this sense, our proposed algorithm achieve the best on the whole compared with the other algorithms.

CONCLUSION

In the study, text classification task can be considered as a multi-label learning problem. Based on En-MLKNN and cost-sensitive learning, we proposed CS-EnMLKNN to perform text classification task by assembling the advantages of different feature selection algorithms and assigning different weights to cost-sensitive predicted probabilities. Experiments on two data collections, Reuters-21578 and 20 Newsgroups indicate that the algorithm CS-EnMLKNN demonstrates better performance than most existing multi-label learning algorithms.

ACKNOWLEDGMENT

This study was supported by the National Science Foundation of China (61203289), China Postdoctoral Science Foundation (2013T60523) and Huawei Innovation Research Program (YB2014010003).

REFERENCES

Boutell, M.R., J. Luo, X. Shen and C.M. Brown, 2004. Learning multi-label scene classification. Pattern Recognit., 37: 1757-1771.

Chang, C.C. and C.J. Lin, 2011. LIBSVM: A library for support vector machines. ACM Trans. Intell. Syst. Technol., Vol. 2. 10.1145/1961189.1961199

Chawla, N.V., K.W. Bowyer, L.O. Hall and W.P. Kegelmeyer, 2002. SMOTE: Synthetic minority Over-sampling technique. J. Artificial Intell. Res., 16: 321-357.

Chen, X.W., B. Gerlach and D. Casasent, 2005. Pruning support vectors for imbalanced data classification. Proceedings of the IEEE International Joint Conference on Neural Networks, Volume 3, July 31-August 4, 2005, Lawrence, KS., USA., pp: 1883-1888.

Chen, W., J. Yan, B. Zhang, Z. Chen and Q. Yang, 2007. Document transformation for multi-label feature selection in text categorization. Proceedings of the 7th IEEE International Conference on Data Mining, October 28-31, 2007, Omaha, NE., pp: 451-456.

Dempster, A.P., N.M. Laird and D.B. Rubin, 1977. Maximum likelihood from incomplete data via the EM algorithm. J. R. Stat. Soc. Ser. B (Methodol.), 39: 1-38.

Elisseeff, A. and J. Weston, 2001. A Kernel Method for Multi-Labelled Classification. In: Advances in Neural Information Processing Systems, Kearns, M.S., S.A. Solla and D.A. Cohn (Eds.). Bradford Bks, USA., pp: 681-687.

Gao, S., W. Wu, C.H. Lee and T.S. Chua, 2003. A maximal figure-of-merit learning approach to text categorization. Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval, July 28-August 1, 2003, Canada, pp: 174-181.

Gao, S., W. Wu, C.H. Lee and T.S. Chua, 2004. A MFoM learning approach to robust multiclass multi-label text categorization. Proceedings of the 21st International Conference on Machine Learning, July 2004, Alberta, Canada.

Han, L. and M. Li, 2014. Open source software classification using cost-sensitive multi-label learning. J. Software, 25: 1982-1991.

Kale, S., R. Kumar and S. Vassilvitskii, 2011. Cross-validation and mean-square stability. Proceedings of the 2nd Symposium on Innovations in Computer Science, January 7-9, 2011, Beijing, China, pp: 487-495.

Kecman, V., 2001. Learning and Soft Computing: Support Vector Machines, Neural Networks and Fuzzy Logic Models. 1st Edn., The MIT Press, Cambridge, MA, USA., ISBN: 0-262-11255-8.

Kong, X., M.K. Ng and Z.H. Zhou, 2013. Transductive multilabel learning via label set propagation. IEEE Trans. Knowledge Data Eng., 25: 704-719.

Kubat, M. and S. Matwin, 1997. Addressing the curse of imbalanced training sets: One-sided selection. Proceedings of the 14th International Conference on Machine Learning, July 1997, Nashville, USA., pp: 179-186.

Kubat, M., R. Holte and S. Matwin, 1997. Learning when negative examples abound. Proceedings of the 9th European Conference on Machine Learning Prague, April 23-25, 1997, Czech Republic, pp: 146-153.

Lee, S.S., 2000. Noisy replication in skewed binary classification. Comput. Stat. Data Anal., 34: 165-191.

Lewis, D.D., 1992. Representation and learning in information retrieval. Ph.D. Thesis, University of Massachusetts Amherst, MA, USA.

Liu, B., C. Wan and L. Wang, 2006. An efficient semi-unsupervised gene selection method via spectral biclustering. IEEE Trans. NanoBiosci., 5: 110-114.

Manning, C. and H. Schutze, 1999. Foundations of Statistical Natural Language Processing. MIT Press, Cambridge.

McCallum, A., 1999. Multi-label text classification with a mixture model trained by EM. Proceedings of the AAAI Workshop on Text Learning, July 18-22, 1999, Pittsburgh, PA., pp: 1-7.

Mitra, V., C.J. Wang and S. Banerjee, 2006. Lidar detection of underwater objects using a neuro-SVM-based architecture. IEEE Trans. Neural Networks, 17: 717-731.

Salton, G. and C. Buckley, 1988. Term-weighting approaches in automatic text retrieval. Inform. Process. Manage., 24: 513-523.

Schapire, R. and Y. Singer, 2000. BoosTexter: A boosting-based system for text categorization. Mach. Learn., 39: 135-168.

Sebastiani, F., 2002. Machine learning in automated text categorization. ACM Comput. Surveys, 34: 1-47.

Soda, P., 2011. A multi-objective optimisation approach for class imbalance learning. Pattern Recognit., 44: 1801-1810.

Tahir, M.A., J. Kittler, K. Mikolajczyk and F. Yan, 2009. A Multiple Expert Approach to the Class Imbalance Problem using Inverse Random under Sampling. In: Multiple Classifier Systems, Benediktsson, J.A., J. Kittler and F. Roli (Eds.). Springer, New York, pp: 82-91.

Tan, S., 2006. An effective refinement strategy for KNN Text classifier. Expert Syst. Applic., 30: 290-298.

Tsoumakas, G. and I. Katakis, 2006. Multi-label classification: An overview. Department of Informatics, Aristotle University of Thessaloniki, Greece.

Ueda, N. and K. Saito, 2002. Parametric Mixture Models for Multi-Labeled Text. In: Advances in Neural Information Processing Systems, Dietterich, T.G., S. Becker and Z. Ghahramani (Eds.). MIT Press, USA., ISBN: 9780262042062, pp: 721-728.

Wang, L., 2005. Support Vector Machines: Theory and Applications. Springer Science and Business Media, New York, ISBN: 9783540243885, Pages: 431.

Wu, J.S. and Z.H. Zhou, 2013. Sequence-based prediction of microRNA-binding residues in proteins using cost-sensitive laplacian support vector machines. IEEE/ACM Trans. Comput. Biol. Bioinformatics, 10: 752-759.

Yang, Y., 1999. An evaluation of statistical approaches to text categorization. Inform. Retrieval, 1: 69-90.

Yang, Y.M. and J. Pedersen, 1997. A comparative study on feature selection in text categorization. Proceedings of the 14th International Conference on Machine Learning, July 8-12, 1997, Nashville, TN., USA., pp: 412-420.

Zhang, M.L. and Z.H. Zhou, 2005. A k-nearest neighbor based algorithm for multi-label classification. Proceedings of the IEEE International Conference on Granular Computing, Volume 2, July 25-27, 2005, China, pp: 718-721.

Zhang, M.L. and Z.H. Zhou, 2007. ML-KNN: A lazy learning approach to multi-label learning. Pattern Recognit., 40: 2038-2048.

Zhang, T., J. Wu and H. Hu, 2014. Text classification based on a novel ensemble multi-label learning method. Proceedings of the 2nd International Conference on Systems and Informatics, November 15-17, 2014, Shanghai, pp: 964-968.

Zhou, Z.H., M.L. Zhang, S.J. Huang and Y.F. Li, 2012. Multi-instance multi-label learning. Artificial Intell., 176: 2291-2320.