A semi-supervised multi-label classification framework with feature reduction and enrichment*

Thi-Ngan Phama,b, Van-Quang Nguyena, Van-Hien Tranña, Tri-Thanh Nguyenna and Quang-Thuy Haa

aUniversity of Engineering and Technology (VNU-UET), Vietnam National University, Hanoi, Vietnam; bThe Vietnamese People’s Police Academy, Hanoi, Vietnam

ABSTRACT

Multi-label classification (MLC) has drawn much attention, thanks to its usefulness and omnipresence in real-world applications in which objects may be characterized by more than one label as in the traditional approach. Getting multi-label examples is costly and time-consuming; therefore, semi-supervised learning approach should be considered to take advantages of both labelled and unlabelled data. In this work, we propose a semi-supervised MLC algorithm exploiting the specific features of the prominent class label(s) chosen by a greedy approach as an extension of the LIFT algorithm, and unlabelled data consumption mechanism from the TExt classification using Semi-supervised Clustering algorithm. We also make a semi-supervised MLC application framework for Vietnamese texts with several feature enrichment steps including (a) a stage of enriching features by adding hidden topic features and (b) a stage of dimensional reduction for subtracting irrelevant features. Experimental results on a data set of hotel reviews (for tourism) indicate that a reasonable amount of unlabelled data helps to increase the F1 score. Interestingly, with a small amount of labelled data, our algorithm can reach a comparative performance to the case of using a larger amount of labelled data.

ARTICLE HISTORY

Received 1 September 2016
Accepted 10 April 2017

KEYWORDS

Multi-label classification (MLC); semi-supervised multi-label classification; latent Dirichlet allocation (LDA); dimensional reduction; specific features

1. Introduction

Multi-label classification (MLC) is the task of assigning each of the given instances to a set of predefined classes, in a domain where an instance can simultaneously belong to several classes. The MLC has received increasing attention and has presented ubiquitously in many application domains such as web categorization, tag recommendation, gene function prediction, medical diagnosis and video indexing (Elisseeff & Weston, 2001; Rousu, Saunders, Szemédl, & Shawe-Taylor, 2006; Silla & Freitas, 2011; Trohidis, Tsoumakas, Kaliris, & Vlahavas, 2008; Tsoumakas, Katakis, & Vlahavas, 2010; Zhang & Zhou, 2007).

Problem transformation is one of the two common approaches in MLC, which transforms the multi-label problem into one or more single-label problems that are solved with a single-label classification algorithm. It is worth noting that this method assumes that the labels are
independent that leads to the ignorance of correlation among labels; thus, in certain situation, it may down the performance of system. In this approach, all labels are discriminated based on identical feature representation of the instance. This method seems to be suboptimal, as each label is supposed to possess specific characteristics of its own. Therefore, a lot of work was proposed to exploit the specific features by dividing the training data set into two groups based on positive and negative instances to build features in different ways (Huaqiao, Zhang, Liu, & Zhao, 2011; Zhang, Fang, & Li, 2015; Zhang & Wu, 2015). For example, Zhang and Wu (2015) proposed an intuitively effective multi-label learning with Label-specific FeaTures algorithm (named LIFT), which constructs the specific features of each label by conducting clustering analysis on its positive and negative instances, and then carries out training and testing by querying the clustering results. Similarly, Zhang, Tang, and Yoshida (2015) proposed to employ spectral clustering to find the closely located local structures between positive and negative instances, which are assumed to be of more discriminative information, and then transform the original data set by consulting the clustering results in classification. Huaqiao et al. (2011) compute feature density and then select features of high density on the positive and negative instance set for each label, respectively; the label-specific features are constructed by taking the intersection on the corresponding label. Finally, each label is classified based on its specific features.

These approaches used clustering, a fundamentally unsupervised learning technique, to group a set of objects in such a way that objects in the same group are more similar (i.e. regarding to a similarity/distance measure) to each other than those in another groups (Basu, 2005). Clustering can be used to aid the classification in terms of discovering the kinds of structure in training examples. The fact is that manual labelling is time-consuming and expensive; the combination of both labelled and unlabelled data in semi-supervised classification framework provides a more effective and cheaper approach to enhance the performance.

Recently, semi-supervised clustering technique has been applied to single classification (Dara, Kermer, & Stacey, 2002; Demirez, Bennett, & Embrechts, 1999; Luo, Liu, Yang, Wang, & Zhou, 2015; Tian, 2014; Zhang, Tang, et al., 2015). Self-organizing mapping technique (Dara et al., 2002) is used to cluster unlabelled instances and to infer possible labellings from the clusters which are presented to a multi-layer perception along with labelled instances. This method significantly improves the classification performance on all the experimental data sets. Demirez et al. (1999) take the approach of incorporating label information into an unsupervised learning approach to group both labelled and unlabelled instances into the clusters where each cluster is as pure as possible in term of class distribution provided by labelled instances. The clusters help characterize segment of the population likely or unlikely to possess the target characteristic represented by the label. This method can be used for both inductive and transductive inference. Zhang, Tang, et al. (2015) introduced a novel semi-supervised learning method, called TExt classification using Semi-supervised Clustering (TESC). In clustering process, TESC uses labelled texts to capture silhouettes of text clusters; next, the unlabelled texts are added to the corresponding clusters to adjust their centroid. These clusters are used for building the model in the classification phase. Given a new unlabelled text, the label associated with it is taken after the label of its nearest text cluster. For MLC, Xiangnan, Ng, and Zhou (2013) proposed a model of transductive multi-label learning by using label set propagation TRAsdu ctive Multi-label classification (TRAM) to assign a set of
multiple labels to each instance. Firstly, TRAM formulates the transductive multi-label learning as an optimization problem of estimating label concept compositions. Then it derives a closed-form solution to this optimization problem and develops an algorithm to assign label sets to unlabelled instances.

In this paper, we propose a framework for MLC with a key contribution of a semi-supervised classification algorithm (called MULTICS) which could exploit specific features of label/label set based on a semi-supervised clustering technique to extract useful information of both labelled and unlabelled instances together. In addition, we expand the framework for MLC with several steps of feature reduction and feature enrichment to evaluate the proposed algorithm and boost up the performance of overall system.

The rest of this paper is organized as follows. In the next section, we will describe the proposed framework of semi-supervised learning approach for MLC focusing on the details of the algorithm MULTICS. An application framework using the proposed algorithm in MLC for Vietnamese texts will be shown in Section 3. We also discuss the experimental results and evaluate the performance of system in this section. The last section presents conclusions and summary.

2. The proposed framework of semi-supervised multi-label classifier

2.1. Problem formulation

2.1.1. Supervised MLC

Let \( D^L \) be the input labelled instance collection with a set \( L = \{l_1, l_2, \ldots, l_q\} \), where each instance in \( D^L \) is assigned a non-empty subset of labels, i.e. \( \text{label}(d_i) \neq \emptyset \), \( \forall d_i \in D^L \). The task of MLC is to construct the classification function \( f: D^L \rightarrow 2^L \), so that, given a new unlabelled instance \( d^u \), the function identifies a set of relevant labels \( f(d^u) \subseteq L \) with an assumption that the labels in \( L \) are sorted in a certain order.

2.1.2. Semi-supervised MLC

Let \( \tilde{D} = D^L \cup D^U \) be an instance collection, where \( D^L \) and \( D^U \) are the collections of labelled and unlabelled instances, respectively. The task of semi-supervised MLC is to construct the classification function \( f: \tilde{D}^L \cup \tilde{D}^U \rightarrow 2^L \). In our proposed semi-supervised MLC algorithm – MULTICS – the goal in building the classifier is to find a partition \( C \) derived from \( \tilde{D} \), such that \( C = \{C_1, \ldots, C_m\} \), where \( C_i = \{d_1^{(i)}, \ldots, d_{|C_i|}^{(i)}\} \ (1 \leq i \leq m) \), \( U_{1 \leq i \leq m} C_i = \tilde{D} \) and \( C_i \cap C_j = \emptyset \ (1 \leq i \neq j \leq m) \). All instances in cluster \( C_i \) are given the same non-empty label set (called cluster label) \( l_{C_i} \).

In the traditional unsupervised clustering method, the number of clusters is often predefined and chosen manually. However, in our algorithm, the number of clusters (\( m \)) is automatically identified based on the label set in combination with the labelled and unlabelled data set.

After we have obtained the partition \( C \), given a new unlabelled instance \( d^u \in D^U \), \( f \) employs the 1-nearest neighbour to get the nearest cluster \( C_j = \arg\min \text{dis}(d^u, c_p) \), where \( c_p \) is the centroid of the cluster \( C_p \) and \( \text{dis}(\cdot) \) is the distance between two instances, then the cluster label of \( C_j \) is assigned to \( d^u \), i.e. \( l(d^u) = l_{C_j} \). Our contribution is to consume both labelled and unlabelled instances to find the partition \( C \) to form the classification model \( f \), which could predict the associated label(s) of unlabelled instances in \( D^U \).
2.2. Brief summary of LIFT and TESC

2.2.1. LIFT algorithm

LIFT was proposed for enhancing the performance of supervised MLC using label-specific features. Concretely, LIFT, at the first step, aims at figuring out features with label-specific characteristics, so as to provide appropriately discriminative information to facilitate its learning as well as classification. For each class label \( l_k \in L \), the set of positive and negative training instances are founded as the set of the training instances with and without label \( l_k \), respectively. After that, clustering analysis is performed on its positive and negative sets to extract the features specific to \( l_k \). In the second step, \( q \) binary classifiers, one for each class label \( l_k \) using \( l_k \)-specific features, are used to check whether a new instance has the label \( l_k \). The approach in LIFT is a supervised method, in which the input is a labelled data set for the training process and the output is a classification model including the family of \( q \) classifiers corresponding to \( q \) labels. Given an unseen instance, its associated label set is predicted by going through \( q \) classifiers to get prediction for each label.

2.2.2. TESC algorithm

TESC was proposed for single-label classification where each instance can be associated with only a single label. In this work, the task of constructing classification model is based on the semi-supervised clustering technique. The basic assumption is that the data samples come from multiple components. Therefore, in the training step, TESC uses clustering to identify components from both labelled and unlabelled instances. The labelled instances are clustered to find the silhouettes of instances, then, the unlabelled instances are added to adjust the clusters. The label of cluster is assigned to the newly added unlabelled instances.

Let \( D = D^l \cup D^u \) be an instance collection, where \( D^l \) and \( D^u \) are the collections of (single-label) labelled and unlabelled instances, respectively. Let \( L \) be the label set on \( D^l \) including \( q \) labels, i.e. \( L = \{ l_1, l_2, \ldots, l_q \} \), and \( C \) be the partition derived from \( D \) after semi-supervised clustering (i.e. the training phase)

\[
C \leftarrow \text{TESC}(D, L). \tag{1}
\]

After this process, the derived cluster set \( C \) is regarded as the model of the classification function. In the classification step, the predicted label of a new instance is assigned as the label of its nearest cluster, that is, given an unseen example, the label of its nearest cluster \( c_j \in C \) is used to assign to it.

2.3. The proposed framework

Our proposed framework using semi-supervised learning approach for MLC is described in Figure 1. In this framework, we combine several techniques to enhance the performance of the classifier.

The proposed framework for MLC using semi-supervised algorithm consists of several steps of building or/and enriching features, selecting features and applying proposed algorithm MULTICS.

We build the MULTICS algorithm for MLC that constructs the specific features for each label/label set based on the idea proposed in LIFT (Zhang & Wu, 2015) with several improvements. In LIFT, the features specific to each label were built in the same
manner. In our model, the first step is to find the prominent labels in a cluster following the greedy approach, that is, select the best choice at the moment (or local optimization) with the hope that this choice would lead to a globally optimal solution. Since, with the labels with few occurrences, it is not good enough to form a cluster, we propose to select the maximum occurrence label (the prominent label) as clue to build clusters. In case of several labels with maximum occurrences in $\overline{D}_L$, the one with smallest index will be chosen as prominent label.

The next step in LIFT is to extract features specific to each label by the k-means clustering algorithm on its positive and negative samples. Our model makes some important changes in this stage. We divide the instance collection into three different subsets: (1) instances with expansion of the only prominent label $\lambda$, (2) instances with a set of label including $\lambda$, and (3) instances without $\lambda$. After that, we perform semi-clustering analysis on these three subsets to get a partition derived from collection of unlabelled and labelled instances. The semi-supervised clustering technique in TESC (Zhang, Tang, et al., 2015) is applied in our model to consume unlabelled instances, that is, an unlabelled instance is added to its nearest cluster, and its label set is the same as the cluster label. Finally, the partition derived from both labelled and unlabelled instances is used as the classification model. No additional classification algorithm is used in our approach. This is different from LIFT, which uses $q$ (i.e. the cardinality of the label set) binary classifiers with label-specific features in the classification phase.
The proposed algorithm includes two phases: one is the learning phase, which uses clustering to identify the components (i.e. clusters) from both labelled and unlabelled instances based on the prominent label; the other is classification, which identifies the nearest instance cluster to assign label for the unseen instance in \( D_U \).

In the learning phase, we use the semi-supervised clustering method in Tian (2014) to take advantages of the TESC algorithm to get partition derived from the instance collection \( \bar{D} \). The recursive training procedure is called \( \text{MULTICSLearn}(.) \) whose pseudo-code is shown in Figure 2.

In order to find the partition \( C \) (i.e. the model of our classification algorithm), we first initialize \( C = \{\} \), then call \( \text{MULTICSLearn}(\bar{D}, \{\}, L, C) \). The result set of clusters \( C \) is regarded as components and used to predict labels of unlabelled instances in the classification phase as shown in Figure 3.

In the classification process, the input includes the collection \( C \) of labelled data clusters (resulted from the \( \text{MULTICSLearn}(.) \)) and an unlabelled instance \( d_U \) needed to be labelled. The output is a predicted label set \( l_U \) assigned to the \( d_U \). We calculate the distances from unlabelled instance to the centroids of all clusters to find out the nearest centroid. Then the unlabelled instance will be assigned the label set of its nearest cluster.

3. An application frame for semi-supervised MLC

In application, we applied this framework for multi-label text classification with following detail stages: (a) a stage of enriching features by using the hidden topic model (latent Dirichlet allocation, LDA) (Blei, 2012; Blei, Ng, & Jordan, 2003) to exploit information of semantic meaning of text representation; (b) a stage of selecting features with mutual information (MI) (Doquire & Verleysen, 2011) to get the most relevant ones and removing irrelevant ones, and (c) a stage of using proposed semi-supervised MLC algorithm. The application framework for multi-label text classification is described in Figure 4.

First of all, we make use of the hidden topic model of LDA to build the features of hidden topic probabilities for each document. This kind of features provides a much richer semantic meaning of text representation.

Next, a feature selection method based on MI is applied to improve the features for the classifier.

In the last step, a multi-label classifier is built based on the MULTICS algorithm. This classifier will be used to classify new documents.

3.1. Building features by applying the LDA model

Hidden topic probability models (Blei, 2012; Blei et al., 2003) are useful to improve the semantic meaning of the text representations, such as (Blei, 2012; Pham, Phan, Nguyen, & Ha, 2013; Ramage, Hall, Nallapati, & Manning, 2009; Rubin, Chambers, Smyth, & Steyvers, 2012; Trejo, Sidorov, Miranda-Jiménez, Ibarra, & Martínez, 2015). Ramage et al. (2009) proposed the labelled LDA (L-LDA) model specifically for multi-label settings. In L-LDA, the training of the LDA model is adapted to account for multi-label corpora by putting ‘topics’ in 1-1 correspondence with labels and then restricting the sampling of topics for each document to the set of labels that were assigned to the document. Rubin et al. (2012) proposed some more flexible LDA-based models including the prior-LDA model.
Procedure `MULTICSLearn (\(\overline{D}, L_1, L_2, C\))`

Inputs:
\(\overline{D}\): a collection of labeled and unlabeled instances, \(\overline{D} = \overline{D}^l \cup \overline{D}^u\), where \(\overline{D}^l\) and \(\overline{D}^u\) are the sets of labeled and unlabeled instances, respectively. Initially, \(\overline{D}\) is the whole training dataset.

\(L_1\): the set of predefined labels for all the instances in \(\overline{D}^l\). Initially, \(L_1\) is set as empty.

\(L_2\): the set of possible labels in \(L\) excluding \(L_1\) which could be assigned for some instances in \(\overline{D}^l\), in other words \(L_2 = L \setminus L_1\). Initially, \(L_2\) is the set of all labels.

Output:
\(C\): a collection of labeled instance clusters

1. Let \(\lambda\) be the smallest index label in \(L_2\) with the maximum occurrences in \(\overline{D}^l\)
2. Create a simulative label set \(L^*\) including three macro labels: \(\lambda_1 \leftarrow L_1 \cup \{\lambda\}\), \(\lambda_2 \leftarrow L_1 \cup \{\lambda\} \cup \{\text{any label } \in L_2\}\), and \(\lambda_3 \leftarrow L_2 \setminus \{\lambda\}\) //label set without \(\lambda\)
   \(\overline{D}' \leftarrow \text{Annotate}(\overline{D}, \{\lambda_1, \lambda_2, \lambda_3\})\)

   /* Annotate(\(\overline{D}, \{\lambda_1, \lambda_2, \lambda_3\}\): Annotate the dataset \(\overline{D}^l\) as single label data, i.e. three macro labels \(\lambda_1, \lambda_2,\) and \(\lambda_3\) for adapting the TESC algorithm*/

   \(C^* \leftarrow \text{TESC}(\overline{D}', L^*)\)

3. Divide \(\overline{D}\) into three subsets of labeled and unlabeled instances regarding to the partition \(C^*\):
   a. \(\overline{D}_1\): the subset of labeled and unlabeled instances in which labeled instances are assigned the label \(\lambda_1\)
   b. \(\overline{D}_2\): the subset of labeled and unlabeled instances in which labeled instances are assigned the label \(\lambda_2\)
   c. \(\overline{D}_3\): the subset of labeled and unlabeled instances in which labeled instances are assigned the label \(\lambda_3\)

4. On \(\overline{D}_1\):
   \(\overline{C} \leftarrow C \cup \text{TESC}(\overline{D}_1, L_1 \cup \{\lambda\})\)

5. On \(\overline{D}_2\):
   a. If all instances in \(\overline{D}_2\) have the same label set \(L_3\)
      \(\overline{C} \leftarrow \overline{C} \cup \text{TESC}(\overline{D}_2, L_3)\)
   b. Else
      // Instances in \(\overline{D}_2\) do not have an identical set of labels.
      // Move \(\lambda\) from \(L_2\) to \(L_1\)
      \(\text{MULTICSLearn}(\overline{D}_2, L_1 \cup \{\lambda\}, L_2 \setminus \{\lambda\}, C)\)

6. On \(\overline{D}_3\):
   a. If all instances in \(\overline{D}_3\) have the same label set \(L_4\)
      \(\overline{C} \leftarrow \overline{C} \cup \text{TESC}(\overline{D}_3, L_4)\)
   b. Else
      // Instances in \(\overline{D}_3\) do not have an identical set of labels.
      // Delete \(\lambda\) from \(L_2\)
      \(\text{MULTICSLearn}(\overline{D}_3, L_1 \setminus \{\lambda\}, C)\)

---

Figure 2. Pseudo-code of clustering process.
Procedure \textsc{MULTICSClassifier} 

Input:
\begin{itemize}
  \item $C$: a collection of labeled data clusters $C = \{C_1, \ldots, C_m\}$
  \item $d^U$: an unlabeled instance
\end{itemize}

Output:
\begin{itemize}
  \item $l^U$: a collection of possible labels to assign to $d^U$
\end{itemize}

\begin{itemize}
  \item 1. $C_{\text{temp}} \leftarrow C_0$ // $C_0$ is the first cluster in $C$
  \item 2. $l_{d^U} \leftarrow l_{C_{\text{temp}}}$ // $l_{C_{\text{temp}}}$ is the label set of cluster $C_{\text{temp}}$
  \item 3. For each $C_j \in C$
    \begin{itemize}
      \item 4. $\text{Dis}(d^U, C_j) \leftarrow \|d^U - C_j\|$
      \item 5. If $\text{Dis}(d^U, C_{\text{temp}}) > \text{Dis}(d^U, C_j)$
        \begin{itemize}
          \item 6. $C_{\text{temp}} \leftarrow C_j$
        \end{itemize}
    \end{itemize}
  \item 7. End if
  \item 8. End for
  \item 9. $l_{d^U} \leftarrow l_{C_{\text{temp}}}$
\end{itemize}

\textbf{Figure 3.} Pseudo-code of the classification procedure.

\textbf{Figure 4.} A semi-supervised MLC framework with feature reduction and enrichment.

that takes into account prior label frequencies, and dependency-LDA that can additionally account for label dependencies. These LDA-based models were applied to multi-label documents that associate individual word tokens with different labels. In this work, we
take after the idea of Pham et al. (2013) and Trejo et al. (2015) to enrich features for Vietnamese short documents by combining various features like traditional Term Frequency Inverse Document Frequency (TFIDF), bigrams, unigrams and LDA features.

The hidden topic model of LDA has been determined by applying the GibbsLDA++ tool (Trejo et al., 2015) on a universal data set with different numbers of topics. Due to the number of classes, we select the hidden topic numbers of 10, 15, 25, 50 and 100 to evaluate. These models will be applied on the training/testing data to generate probabilities of assigning hidden topics for each document.

Denote \( p(d, j) \) as the probability that a review \( d \) belongs to the hidden topic \( j \) (i.e. \( j = 1, \ldots, k \), where \( k \) is the number of hidden topics). The vector \( (p(d, 1), p(d, 2), \ldots, p(d, k)) \) is called the hidden topic feature vector of the review. These features will be combined with other features of documents to build the feature set for the classifier.

### 3.2. Selecting features based on the MI

Feature selection is one of the fundamental steps in the fields of machine learning and data mining for reducing dimensionality, choosing a small subset of the relevant features from the original ones, which usually leads to better learning performance, lower computational cost and better model interpretability.

In this work, we apply the method of MI (Doquire & Verleysen, 2011) to perform feature selection in MLC problems. MI measures the amount of information contained in variable \( X \) in order to predict variable \( Y \) in any relation, not only linear ones. In addition, the MI concept is directly applicable to groups of variables. The MI is given in Gómez-Verdejo, Verleysen, and Fleury (2007) as follows:

\[
I(X, Y) = \int \int p_{x, y}(x, y) \log \frac{p_{x, y}(x, y)}{p_x(x)p_y(y)},
\]

where \( p_x(x) \) and \( p_y(y) \) are the marginal probability density function of \( X \) and \( Y \), respectively, and \( p_{x, y}(x, y) \) is the joint probability density function of \( X \) and \( Y \).

The method of achieving feature selection based on MI (Doquire & Verleysen, 2011) includes the following steps. First, the multi-label problem is transformed using the pruned problem transformation method (Read, 2008). Then, for each single-label classification task, a forward/backward selection algorithm based on MI is employed to choose the 'optimal' feature subset. The search strategy used in this model is considered as backward elimination, which starts with the set of all features and recursively removes them one at a time. The procedure ends when the predefined number of features has been reached. Another search strategy called a greedy forward feature selection algorithm that begins with an empty set of features and first selects the feature whose MI to the class vector is the highest. Then, the algorithm sequentially selects the feature not yet selected whose addition to the current subset of selected features leading to the set having the highest MI to the output.

### 3.3. The data sets

We applied the proposed framework on the set of customer’s reviews about Vietnamese hotels retrieved from several famous Vietnamese websites on tourism and hotels. We build
labelled, unlabelled and testing data sets with different numbers of documents to evaluate the effectiveness of labelled and unlabelled data on the model. After some pre-processing steps on these data sets, that is, main text context extraction, word segmentation, and stop word removal, we got about 1800 reviews. Thousand five hundred reviews were manually tagged to create the labelled data set of 1250 reviews, and the testing set of 250 reviews. The remaining 300 reviews were left intact to create the unlabelled data set. We considered reviews on five aspects: (a) location and price, (b) service, (c) facilities, (d) room standard, and (e) food.

In order to train the LDA model for generating the hidden topic models, the universal data set of 24,000 articles, introductions, comments about hotels in Vietnam (from the above sources) were also crawled. The pre-processing step is applied to all data sets for LDA construction and classification.

3.4. Experimental results

We took several experiments with different configurations to evaluate the effect of the proposed algorithm. In order to analyse the contribution of the labelled data sets, we also generated some subsets with the size of 500, 750, 1000 reviews. The contribution of unlabelled data sets is also evaluated in each category with different size of 0, 50, 100, 200, 300 reviews, where the unlabelled data set size of 0 is used as the baseline.

We performed four groups of experiments with different settings to evaluate the effectiveness of the proposed framework as follows:

- Experiment 1 (denoted by the BN): run MULTICS on the set of binary features only.
- Experiment 2 (denoted by the BN + MI): run MULTICS on the set of binary features with the MI feature selection technique.
- Experiment 3 (denoted by the TFIDF + LDA): run MULTICS on the set of TFIDF and hidden topic probability features.
- Experiment 4 (denoted by the TFIDF + LDA + MI): run MULTICS on the set of TFIDF, hidden topic features, and with the MI feature selection technique.

For the hidden topic probability features, we build the hidden topic model of LDA (Phan & Nguyen, 2007) with different topic numbers, that is, 10, 15, 25, 50 and 100. In order to calculate distance between instances, we can apply different kinds of distance measure, but in our experiments below, we chose the Euclidean distance - a popular measure. We should try another measure in the future.

We used the label-based measures (Tsoumakas et al., 2010) for evaluating the performance of the proposed model. For each class label $y_j$, $TP_j$, $FP_j$, $TN_j$, and $FN_j$ which are the number of true-positive, false-positive, true-negative and false-negative test samples were recorded. Let $B(TP_j, FP_j, TN_j, FN_j)$ be some specific binary classification measures (e.g. $B(\cdot) = \{P, R, F1\}$, where $P = TP_j/(TP_j + FP_j)$, $R = TP_j/(TP_j + FN_j)$, and $F1 = P*R/(P + R)$). The micro-averaging measures are calculated as follows:

$$B_{micro} = B\left(\sum_{j=1}^{q}TP_j, \sum_{j=1}^{q}FP_j, \sum_{j=1}^{q}TN_j, \sum_{j=1}^{q}FN_j\right),$$

(3)
where $q$ is the total number of labels. For these metrics, the bigger the value, the better the classification performance. In Tables 1 and 2, the worst cases are formatted in italic, and the best cases are formatted in bold.

The results of Experiments 1 and 2 are described in Table 1 in which we compare the performance of using binary features (BN), and binary features with MI feature selection. The results show the contribution of unlabelled data sets, that is, the performance of all the cases using unlabelled data sets is better than that of the baseline method. Also with the contribution of labelled data sets, when the amount of labelled data increases, the model gets better results. In terms of features, the performances of binary features without feature selection are all better than that of the case using both binary features and MI feature selection. In our opinion, the way of building specific features for prominent label/label set is already a kind of feature selection. Thus, the Experiment 2 (BN + MI) has two feature selection steps which do not help to increase the performance.

In the experiments with hidden topic probabilities, we realize that the combination of binary features with hidden topic features does not increase the performance of the classifier. The reason may be the different kinds of discrete features (of binary representation) and real-valued features (of hidden topic probabilities). Therefore, in the Experiments 3 and 4, we combine the features of TFIDF with the features of the hidden topic probabilities to build continuous features. Because the combination of the different unlabelled data set sizes (5), with different numbers of hidden topics (5), and different labelled data set sizes (3) is big, we only selected the best result of the labelled data set for comparison.

Table 2 shows the best results of Experiments 3 and 4 in which the labelled data set size of both the Experiments 3 and 4 is 1000 documents. When comparing the results in Tables 1 and 2, we see that the performances of enriched features with hidden topic probabilities in Experiments 3 and 4 are better than those of Experiments 1 and 2. The best $F_1$ in Experiments 3 and 4 is 85.3% and 84%, respectively, while the best $F_1$ in Experiment 1 is 83.2%.

Additionally, Table 2 shows that the Experiment 3 outperforms the Experiment 4. In other words, the Experiment 4 using the feature selection technique on the feature set of TFIDF and hidden topic probabilities is not as good as the Experiment 3 using the

| Training dataset size | Unlabelled dataset size | BN | BN + MI |
|-----------------------|-------------------------|----|---------|
|                       |                         | Precision$_{\text{micro}}$ (%) | Recall$_{\text{micro}}$ (%) | $F_1_{\text{micro}}$ (%) | Precision$_{\text{micro}}$ (%) | Recall$_{\text{micro}}$ (%) | $F_1_{\text{micro}}$ (%) |
| 500                   | 0                       | 77.4 | 81.1 | 79.2 | 74.3 | 75.8 | 75 |
|                       | 50                      | 81.4 | 77.7 | 79.5 | 76.3 | 78.7 | 77.5 |
|                       | 100                     | 80.6 | 78.7 | 79.7 | 73.1 | 78.5 | 75.7 |
|                       | 200                     | 83   | 82.5 | **82.7** | 74.8 | 76.4 | 75.6 |
|                       | 300                     | 79.6 | 80.4 | 80   | 81   | 76.1 | **78.5** |
| 750                   | 0                       | 77.7 | 81.5 | 79.6 | 75.7 | 76.8 | 76.2 |
|                       | 50                      | 82.4 | 81.3 | 81.8 | 77   | 78.2 | 77.6 |
|                       | 100                     | 82.1 | 82.3 | **82.2** | 74.8 | 78.2 | 76.4 |
|                       | 200                     | 80.7 | 82.5 | 81.6 | 74.1 | 81.2 | 77.5 |
|                       | 300                     | 79   | 82.3 | 80.6 | 79.9 | 80.4 | **80.2** |
| 1000                  | 0                       | 80.1 | 79.6 | 79.8 | 80.3 | 78.1 | 79.1 |
|                       | 50                      | 80.7 | 81   | 80.9 | 78.4 | 81.2 | 79.8 |
|                       | 100                     | 81.3 | 83.3 | 82.3 | 83.1 | 82.7 | **82.8** |
|                       | 200                     | 81   | 84.4 | 82.6 | 75.9 | 84.6 | 80 |
|                       | 300                     | 82.4 | 83.9 | **83.2** | 78.2 | 80.9 | 79.5 |

Note: worst cases formatted in italic; best cases formatted in bold.
feature set of TFIDF and hidden topic probability. These results confirm that MULTICS builds the features good enough for the classifier. We may apply MULTICS directly on the original feature set without feature selection step.

4. Conclusions and future work

In this paper, we build a framework for MLC including the process of enriching features, the process of feature selection, and the process of classification of MULTICS – an approach for semi-supervised MLC to exploit label-specific features. Using two basic assumptions including the effect of label-specific features in the learning phase and the multiple components in each label which can be identified by clustering, our proposed model brings major contribution in building label-specific features for multi-label learning with an approach of semi-supervised clustering technique. The experiments show that MULTICS gives promising results for MLC, and the combination with the hidden topic probability features also contributes better performance.

For future direction, we are making some more improvements, for example, method to effectively select unlabelled instances, or post-processing to prune the result clusters to remove outliers, to evaluate the proposed approach.

Note

1. A novel semi-supervised MULTI-label ClasSification which can exploit both unlabelled data and specific features to enhance the performance.

Disclosure statement

No potential conflict of interest was reported by the authors.

Funding

This work was supported in part by Vietnam National University Grant QG-15-22.

Notes on contributors

Thi-Ngan Pham received her bachelor’s degree in Faculty of Mathematics and Information, the Vietnam People’s Security Academy in 2006, and a master’s degree in Faculty of Information Technology (FIT), Vietnam National University (VNU), Hanoi, in 2011. She is now a PhD student in FIT, VNU,
Hanoi. She has been working as a lecturer in Faculty of Mathematics and Information, the Vietnamese People’s Police Academy since 2006.

**Van-Quang Nguyen** is a fourth-year student in an honors program in FIT, VNU, Hanoi.

**Van-Hien Tran** received his bachelor’s degree in FIT, VNU, Hanoi, in 2014. He is also a master’s student in FIT. He is now an assistant lecturer in FIT, VNU, Hanoi.

**Tri-Thanh Nguyen** is a lecturer in FIT, University of Engineering and Technology (UET) in VNU, Hanoi.

He received his master’s degree from the Asian Institute of Technology (AIT), Thailand, in 2002, and a PhD at the Japan Advanced Institute of Science and Technology (JAIST) in 2008, and the position of associate professor in 2014.

**Quang-Thuy Ha (Ha Quang Thuy)** is an associate professor in Information System Department and Data Science and Knowledge Technology Laboratory (DSKTLab), FIT, UET, VNU, Hanoi. He received his PhD at the Hanoi University of Science (HUS), VNU, in 1997. He was a Vice-Dean of the FIT and a Vice-Rector of VNU-UET.

**References**

Basu, S. (2005). *Semi-supervised clustering: Probabilistic models, algorithms and experiments (PhD thesis).* University of Texas at Austin.

Blei, D. M. (2012). Probabilistic topic models. *Communications of the ACM, 55*(4), 77–84.

Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent Dirichlet allocation. *Journal of Machine Learning Research, 3*, 993–1022.

Dara, R., Kermer, S., & Stacey, D. (2002). *Clustering unlabeled data with SOMs improves classification of labeled real-world data.* Proceedings of the 2002 international joint conference on neural networks (pp. 2237–2242). Honolulu, Hawaii.

Demirez, A., Bennett, K., & Embrechts, M. (1999). *Semi-supervised clustering using genetic algorithms.* Proceedings of artificial neural networks in engineering (ANNIE-99), (pp. 809–814). Missouri, USA.

Doquire, G., & Verleysen, M. (2011). *Feature selection for multi-label classification problems.* International work-conference on artificial neural networks (pp. 9–16). Berlin: Springer.

Elisseeff, A., & Weston, J. (2001). *A kernel method for multi-labelled classification.* NIPS (pp. 681–687). British Columbia, Canada.

Gómez-Verdejo, V., Verleysen, M., & Fleury, J. (2007). *Information-theoretic feature selection for the classification of hysteresis curves.* International work-conference on artificial neural networks (pp. 522–529). Berlin: Springer.

Huqiao, Q., Zhang, S., Liu, H., & Zhao, J. (2011). A multi-label classification algorithm based on label-specific features. *Wuhan University Journal of Natural Sciences, 16*(6), 520–524.

Luo, X., Liu, F., Yang, S., Wang, X., & Zhou, Z. (2015). Joint sparse regularization based sparse semi-supervised extreme learning machine (S3ELM) for classification. *Knowledge-Based Systems, 73*, 149–160.

Pham, T.-N., Nguyen, V.-Q., Dinh, D.-T., Nguyen, T.-T., & Ha, Q.-T. (2017). MASS: A semi-supervised multi-label classification algorithm with specific features. In D. Król, N.T. Nguyen, & K. Shirai (Eds.), *Advanced topics in intelligent information and database systems* (Vol. 710, pp. 37–47). AICIIDS 2017. Studies in computational intelligence. Kanazawa: Springer International Publishing.

Pham, T.-N., Phan, T.-T., Nguyen, P.-T., & Ha, Q.-T. (2013). Hidden topic models for multi-label review classification: An experimental study. In C. Badica, N.T. Nguyen, & M. Brezovan (Eds.), *Computational collective intelligence. Technologies and applications* (Vol. 8083, pp. 603–611). Lecture notes in computer science. Berlin, Heidelberg: Springer.

Phan, X.-H., & Nguyen, C.-T. (2007). *GibbsLDA++.* Retrieved from http://gibbslda.sourceforge.net/

Ramage, D., Hall, D., Nallapati, R., & Manning, C. D. (2009). *Labeled LDA: A supervised topic model for credit attribution in multi-labeled corpora.* EMNLP (pp. 248–256). Singapore.

Read, J. (2008). *A pruned problem transformation method for multi-label classification.* New Zealand computer science research student conference (NZCSR 2008) (pp. 143–150). Christchurch, New Zealand.
Rousu, J., Saunders, C., Szedmák, S., & Shawe-Taylor, J. (2006). Kernel-based learning of hierarchical multilabel classification models. *Journal of Machine Learning Research, 7*, 1601–1626.

Rubin, T. N., Chambers, A., Smyth, P., & Steyvers, M. (2012). Statistical topic models for multi-label document classification. *Machine Learning, 88*(1–2), 157–208.

Silla, C. N. Jr., & Freitas, A. A. (2011). A survey of hierarchical classification across different application domains. *Data Mining and Knowledge Discovery, 22*(1–2), 31–72.

Tian, D. (2014). Semi-supervised learning for refining image annotation based on random walk model. *Knowledge-Based Systems, 72*, 72–80.

Trejo, J. V. C., Sidorov, G., Miranda-Jiménez, S., Ibarra, M. M., & Martínez, R. C. (2015). Latent Dirichlet allocation complement in the vector space model for multi-label text classification. *International Journal of Combinatorial Optimization Problems and Informatics, 6*(1), 7–19.

Trohidis, K., Tsoumakas, G., Kalliris, G., & Vlahavas, I. P. (2008). *Multi-label classification of music into emotions*. ISMIR (pp. 325–330). Philadelphia, USA.

Tsoumakas, G., Katakis, I., & Vlahavas, I. P. (2010). Mining multi-label data. In O. Maimon, L. Rokach (Eds.), *Data mining and knowledge discovery handbook* (2nd ed., pp. 667–685). Springer.

Xiangnan, K., Ng, M. K., & Zhou, Z.-H. (2013). Transductive multilabel learning via label set propagation. *IEEE Transactions on Knowledge and Data Engineering, 25*(3), 704–719.

Zhang, J.-J., Fang, M., & Li, X. (2015). Multi-label learning with discriminative features for each label. *Neurocomputing, 154*, 305–316.

Zhang, M.-L., & Wu, L. (2015). LIFT: Multi-label learning with label-specific features. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 37*(1), 107–120.

Zhang, M.-L., & Zhou, Z.-H. (2007). ML-KNN: A lazy learning approach to multi-label learning. *Pattern Recognition, 40*(7), 2038–2048.

Zhang, W., Tang, X., & Yoshida, T. (2015). TESC: An approach to text classification using semi-supervised clustering. *Knowledge-Based Systems, 75*, 152–160.