MASS: A Semi-supervised Multi-label Classification Algorithm with Specific Features

Thi-Ngan Pham, Van-Quang Nguyen, Duc-Trong Dinh, Tri-Thanh Nguyen and Quang-Thuy Ha

Abstract Multi-label Classification (MLC), which recently has attracted numerous attentions, aims at building classification models for objects assigned with multiple class labels simultaneously. Existing approaches for MLC mainly focus on improving supervised learning which needs a relatively large amount of labeled data for training. In this work, we propose a semi-supervised MLC algorithm to exploit unlabeled data for enhancing the performance. In the training process, our algorithm exploits the specific features per prominent class label chosen by a greedy approach as an extension of LIFT algorithm, and unlabeled data consumption mechanism from TESC. In classification, the 1-Nearest-Neighbor (1NN) is applied to select appropriate class labels for a new data instance. Our experimental results on a data set of hotel (for tourism) reviews indicate that a reasonable amount of unlabeled data helps to increase the F1 score. Interestingly, with a small amount of labeled data, our algorithm can reach comparative performance to a larger amount of labeled data.

Keywords Semi-supervised clustering · Multi-label classification (MLC) · Specific feature · Semi-supervised multi-label classification

T.-N. Pham · V.-Q. Nguyen · D.-T. Dinh · T.-T. Nguyen · Q.-T. Ha (✉)
Vietnam National University (VNU), University of Engineering and Technology (UET), Hanoi, Vietnam
e-mail: thuyhq@vnu.edu.vn

T.-N. Pham
e-mail: nganpt.di12@vnu.edu.vn

V.-Q. Nguyen
e-mail: quangnv_570@vnu.edu.vn

D.-T. Dinh
e-mail: trongdd_58@vnu.edu.vn

T.-T. Nguyen
e-mail: ntthanh@vnu.edu.vn

T.-N. Pham
The Vietnamese People's Police Academy, Hanoi, Vietnam

D. Król et al. (eds.), *Advanced Topics in Intelligent Information and Database Systems*, Studies in Computational Intelligence 710,
DOI 10.1007/978-3-319-56660-3_4

1 Introduction

In the domain where an example can simultaneously belong to multiple classes, MLC aims at identifying a subset of predefined class labels for a given unlabeled instance. The multi-label classification has received increasingly attention and been applied to several domains, including web categorization, tag recommendation, gene function prediction, medical diagnosis and video indexing [1–6].

The most well-known approach to MLC is to use a different classifier for each label. The final labels of each instance are then obtained by using an aggregation scheme where the predictions of the individual classifier are combined. This approach has the advantage of its simplicity and disadvantage of ignoring the correlation among labels, thus, in certain situation, it can show performance degradation. In commonly existing approaches, all the class labels are discriminated based on the same feature representation. In other words, they use the identical feature set for different class label functions in computation. Since each label relies on its own specific characteristics, this approach might be not optimal. In several approaches, a document collection is divided into two groups based on positive and negative instances [7–9], then specific features are built in different ways. For example, Zhang and Lei [7] proposed an intuitively effective *multi-label learning with Label specIfic FeaTures* algorithm (named LIFT), which builds the specific features of each label by applying clustering analysis on its positive and negative instances, and then carry out training and testing by exploiting the clustering results. Similarly, Zhang et al. [8] proposed to employ spectral clustering to figure out the closely located local structures between positive and negative instances, and exploit the clustering results in classification. Huaqiao et al. [9] built the label-specific features by computing and selecting high density features on the positive and negative instance set for each class. Finally, each class label is classified based on its specific features.

Clustering, a basically unsupervised learning technique, groups a data set into clusters such that data in the same clusters are more similar (i.e., regarding to a distance measure) to each other than those in another cluster [10]. Clustering can be used to aid the text classification in term of discovering the kinds of structure in training examples. Due to the fact that obtaining labeled data is costly and time consuming, the combination of both labeled and unlabeled data in semi-supervised classification framework provides a more effective and cheaper approach to increase the performance.

Recently, semi-supervised clustering is an approach to semi-supervised classification [11–15]. Self-Organizing mapping (SOM) clustering is used to identify the label of the unlabeled data in non-ambiguous nodes by using the label of their nodes, then data in the clusters are used to train a multi-layer perception classifier [12]. This method significantly improves the classification performance on all the experimental datasets. Demirez et al. [14] presented a semi-supervised clustering algorithm which finds a set of clusters by minimizing a linear combination of cluster dispersion measured by mean square error and cluster impurity measure.

Zhang et al. [15] introduced a novel semi-supervised learning method, called *TExt classification using Semi-supervised Clustering* (TESC). In clustering process, TESC uses labeled texts to capture silhouettes of text clusters, next the unlabeled texts are added the corresponding clusters to adjust the centroid. These clusters are used for classification phase. Given a new unlabeled text, the label of the nearest text cluster is used to assign to the unlabeled text.

Kong et al. [16] proposed a model of transductive multi-label classification by using label set propagation, called TRAM. Firstly, TRAM formulates the transductive multi-label learning as an optimization problem to exploit unlabeled data. Secondly, TRAM develops an efficient algorithm which has a closed-form solution for this optimization problem and assigning label sets to unlabeled instances. The key in this method is to use the test data for optimization. In addition, the test examples are also used as unlabeled examples.

In this paper, we propose a novel *semi-supervised algorithm for Multi-label clASSification* (called MASS), which can exploit both *unlabeled data* and *specific features* to enhance the performance. By determining the prominent label in specific collection, the dataset is then divided into three different subsets; and the semi-supervised clustering is applied in each subset to extract features specific to each label or label set. The method of extracting specific features is an extension of LIFT algorithm proposed by Zhang and Lei [7]. In addition, MASS has some key breakthrough in using semi-supervised clustering to exploit both labeled and unlabeled data together at the same time as mentioned in TESC of Zhang et al. [15].

The rest of this paper is organized as follows. Section 2 introduces our newly proposed algorithm for MLC text classification. This section will give more details about the process of constructing specific features and using semi-supervised clustering in building MLC. Section 3 evaluates the proposed algorithm using experiments. Conclusions are shown in the last section.

2 The Proposed Algorithm

2.1 Problem Formulation

Supervised Multi-label Classification

Let $\overline{D}^L$ be the input labeled document collection with a set L of q labels, i.e., $L = \{l_1, l_2, \ldots, l_q\}$, where each document in $\overline{D}^L$ is assigned a non empty subset of labels $label(d \in \overline{D}^L) \subseteq L$ The task of MLC is to construct the classification function $f: \overline{D}^L \to 2^L$, so that, given a new unlabeled document d^u, the function identifies a set of relevant labels $f(d^u) \subseteq L$.

Semi-supervised Multi-label Classification

Let $\overline{D} = \left\{\overline{D}^L, \overline{D}^U\right\}$ be a document collection, where $\overline{D}$ and $\overline{D}^U$ are the collections of labeled and unlabeled documents, correspondingly. The task of semi-supervised MLC is to construct the classification function $f: \overline{D} \to 2^L$. The goal in the training step is to find a partition C from $\overline{D}$, such that $C = \{C_1, \ldots, C_m\}$, where $C_i = \left\{d_1^{(i)}, \ldots, d_{|C_i|}^{(i)}\right\} (1 \le i \le m)$, $\bigcup_{1 \le i \le m} C_i = \overline{D}$, and $C_i \cap C_j = \emptyset (1 \le i \ne j \le m)$. For all documents in C_i, they are given the same non-empty label set (called cluster-label) l_{C_i}.

In traditional unsupervised clustering method, the number of cluster is often predefined and manually chosen. However, in our model, the number of clusters m is automatically identified based on the label set in combination with the labeled and unlabeled data set.

After we have obtained the partition C, given a new unlabeled document $d^u \in D^U$, f employs the 1-nearest neighbor to get the nearest cluster $C_j = \underset{C_p}{arg\,min}\, dis\left(d^u, c_p\right)$, and c_p is the centroid of the text cluster C_p and $dis(.)$ is the distance between data points, 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 labelled and unlabeled data to find the partition C to form classification model f, which could predict class label set of unlabeled texts D^U.

2.2 Brief Summary of LIFT and TESC

LIFT Algorithm

LIFT was proposed for enhancing the performance of supervised multi-label classification using label-specific features. With assumption that label-specific features, i.e. the most specific characteristics, could improve the classification. 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 supervised method in which the input is labeled dataset for training process and the output is a classification model including the family of q classifiers corresponding to q labels. Given an unseen examples, its associated label set is predicted by going through q classifiers to get prediction for each label.

TESC Algorithm

TESC was proposed for single label classification where each instance can be associated with only a single class label. In this work, the task of constructing classification model is based on a semi-supervised clustering. 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 labeled and unlabeled texts. The labeled documents are clustered to find the silhouettes of documents, then, the unlabeled documents are added to adjust the clusters. The label of cluster is assigned to the newly added unlabeled documents.

Let $D = \{D^L, D^U\}$ be a document collection, where D^L and D^U are the collections of (single-label) labeled and unlabeled documents, correspondingly. 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 on D after process of semi-supervised clustering (i.e., the training phase)

$$C \leftarrow TESC(D, L).$$

After this process, the resulted cluster set C is regarded as the model of the classification function. In the classification step, given a new document the label of the nearest cluster is used as the predicted label of the new document, i.e., given an unseen example, the label of its nearest cluster $c_j \in C$ is used to assign to it.

2.3 Proposed Algorithm

In our approach, we construct the specific features for each label and label set based on the idea proposed in LIFT with several improvements. In LIFT, the authors build the features specific to each label in the same manner. In our model, the first step is to find the prominent labels in a cluster following to the greedy approach, i.e., select the best choice at the moment (or local optimization) with an assumption that this would lead to a globally optimal solution. Since, with the labels with few occurrences, it is not good enough to form a cluster, we proposed to select the maximum occurrence label (the prominent label) as clue to build clusters.

Next step in LIFT is to extract features specific to each label by k-means clustering technique on its positive and negative samples. Our model makes some important changes in this stage. We divide a document collection into three different document subsets: (1) documents with expansion of the only prominent label λ, (2) documents with a set of label including λ, and (3) documents without λ. After that, we perform semi-clustering analysis on these three subsets to get a partition on collection of unlabeled and labeled documents. The semi-supervised clustering technique in TESC is applied in our model to consume unlabeled documents, i.e., an unlabeled document is added to its nearest cluster, and its label set is the same as the cluster label. Finally, the partition on the dataset of both labeled and unlabeled documents is used as classification model. No additional classification algorithm is

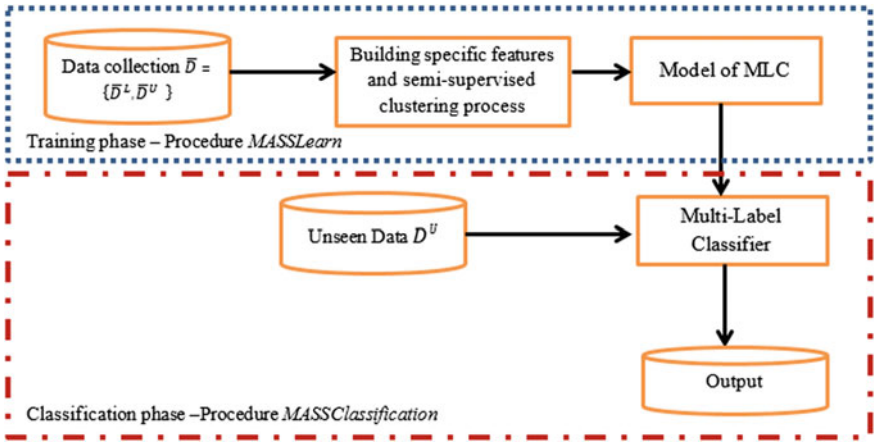

Fig. 1 The proposed semi-supervised MCL model

used in our approach. This is different from the LIFT which uses q (i.e., the cardinality of the label set) binary classifiers with label-specific features in classification phase.

The proposed algorithm comprises of two phases as described in Fig. 1: one is the training phase, which uses clustering to identify the components (i.e., clusters) from both labeled and unlabeled texts based on the prominent label; The other phase is classification, which identifies the nearest text cluster to label the unlabeled text D^U.

In training phase, we use the semi-supervise clustering method in [15] to take advantages of TESC algorithm to get partition on the text collection $\overline{D}$. We name training procedure *MASSLearn*(.), of which the pseudo-code is shown in Fig. 2.

In order to find the partition C (i.e., the model of our classification algorithm), we first initialize $C = \{\}$; then call *MASSLearn*$(\overline{D}, \{\}, L, C)$. The resulted set of text clusters C is regarded as components and used to predict labels of unlabeled texts in classification phase as shown in Fig. 3.

In classification process, the input includes unlabeled texts need labeling. The output is the collection of labels corresponding to each text in unlabeled texts. We calculate the distances from unlabeled text to the centroids of all clusters to find out the nearest centroid. Then the label set of the nearest cluster will assign to the unlabeled text.

Procedure $MASSLearn\ (\overline{D}, L_1, L_2, C)$
Inputs:
$\overline{D}$: collection of labeled and unlabeled texts, $\overline{D} = \{\overline{D}^L, \overline{D}^U\}$, where $\overline{D}^L$ and $\overline{D}^U$ are the set of labeled and unlabeled documents, correspondingly
L_1: set of available default labels for all the instances in $\overline{D}^L$
L_2: set of possible labels which could be labeled for some instances in $\overline{D}^L$
Output:
C: a collection of labeled text clusters

1. Let λ be the label in L_2 with the maximum occurrences in $\overline{D}^L$
2. Create a new label set L^* including three assistant labels: $\lambda_1 \leftarrow L_1 \cup \{\lambda\}$, $\lambda_2 \leftarrow L_1 \cup \{\lambda\} \cup \{other\ labels\}$, and $\lambda_3 \leftarrow \{label\ set\ without\ \lambda\}$
   /* Annotate the data set $\overline{D}^L$ as single label data according to three macro assistant labels λ_1, λ_2, and λ_3 to use TESC algorithm*/
   $\overline{D}' \leftarrow Annotate(\overline{D}, \{\lambda_1, \lambda_2, \lambda_3\});\ C^* \leftarrow TESC(\overline{D}', L^*)$
3. Divide $\overline{D}$ into three subsets of labeled and unlabeled texts regarding to the partition C^*:
   a. $\overline{D}_1$: subset of labeled and unlabeled texts in which labeled texts are assigned the label λ_1
   b. $\overline{D}_2$: subset of labeled and unlabeled texts in which labeled texts are assigned the label λ_2
   c. $\overline{D}_3$: subset of labeled and unlabeled texts in which labeled texts are assigned the label λ_3
4. On $\overline{D}_1$:
   $C \leftarrow C \cup TESC(\overline{D}_1, L_1 \cup \{\lambda\})$
5. On $\overline{D}_2$:
   a. If all instances in $\overline{D}_2$ have the same label set L3
      $C \leftarrow C \cup TESC(\overline{D}_2, L_1 \cup \{\lambda\} \cup \text{L3})$
   b. Else
      // Instances in $\overline{D}_2$ are not identical in set of labels.
      // Move λ from L2 to L1
      $MASSLearn(\overline{D}_2, L_1 \cup \{\lambda\}, \text{L2} \backslash \{\lambda\}, \text{C})$
6. On $\overline{D}_3$:
   a. If all instances in $\overline{D}_3$ are assigned by the set of labels L4
      $C \leftarrow C \cup TESC(\overline{D}_3, \text{L4})$
   b. Else
      // Instances in $\overline{D}_3$ are not identical in set of labels.
      // Delete λ from L2
      $MASSLearn(\overline{D}_3, \text{L1}, \text{L2} \backslash \{\lambda\}, C)$

Fig. 2 Pseudo-code of clustering process

Procedure *MASSClassification*
Input:
C: collection of labeled text clusters $C = \{C_1, \ldots, C_m\}$
D^U: collection of unlabeled texts
Output:
L^U: a collection of labels corresponding to each text in D^U

1. For each $d^u \in D^U$
2. $C_{temp} \leftarrow C_0$ // C_0 is the first cluster in C
3. $l_{d^u} \leftarrow l_{C_{temp}}$ //$l_{C_{temp}}$ is the label set of cluster C_{temp}
4. For each $C_j \in C$
5. $Dis(d^u, C_j) \leftarrow \|d^u - C_j\|$ //we use Euclidean distance here
6. If $Dis(d^u, C_{temp}) > Dis(d^u, C_j)$
7. $C_{temp} \leftarrow C_j$
8. End if
9. End for
10. $l_{d^u} \leftarrow l_{C_{temp}}$
11. Add $l_{C_{temp}}$ to L^U
12. End for

Fig. 3 Pseudo code of classification procedure

3 Experiments and Results

3.1 The Datasets

We built three datasets of labeled, unlabeled, and testing data from thousands of reviews retrieved from several famous Vietnamese websites on tourism and hotels. After some preprocessing steps on the datasets, i.e., main text content extraction, word segmentation, and stop word removal, we got about 1800 reviews. 1500 reviews were manually tagged to create the labelled set of 1250 reviews, and the testing set of 250 reviews. The rest of 300 reviews were left intact to create unlabeled set. We considered reviews on five aspects: (a) location and price, (b) service, (c) facilities, (d) Room Standard, and (e) Food.

3.2 Experimental Results

We took several experiments with different configurations to evaluate the effect of the proposed algorithm. In order to analyze the contribution of the labeled data, we also generated some subsets of size of 500, 750, 1000, 1250 reviews. The contribution of unlabeled data is also evaluated in each category with different size of 0, 50, 100, 200, 300 reviews.

We also built one baseline using supervised algorithm of SVM for MLC, i.e., five binary SVM classifiers, one for each class label. The baseline worked best on the training set of 750 reviews, hence, we used this result for later comparison.

In our model, we used the label-based measures for evaluation [4]. 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., $B \in \{P, R, F1\}$, where $P = TP_j/(TP_j + FP_j)$, $R = TP_j/(TP_j + FN_j)$, and $\mathrm{F1} = \frac{P*R}{2(P+R)}$. The micro-averaging measures are calculated as follows:

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

where q is the total number of labels. For these metrics, the bigger value, the better classification performance.

The results of the experiments are reported in the Table 1. We observed that the proposed solution's results are very promising in all experiments in comparison

Table 1 The results of experiments

Training dataset size	Unlabeled dataset size	$\mathrm{Precision}_{\mathrm{micro}}$ (%)	$\mathrm{Recall}_{\mathrm{micro}}$ (%)	$F1_{\mathrm{micro}}$ (%)
Baseline		68.50	60.00	63.90
500	0	77.40	81.10	79.20
	50	81.40	77.70	79.50
	100	80.60	78.70	79.70
	200	83.00	82.50	**82.70**
	300	79.60	80.40	80.00
750	0	77.70	81.50	79.60
	50	82.40	81.30	81.80
	100	82.10	82.30	**82.20**
	200	80.70	82.50	81.60
	300	79.00	82.30	80.60
1000	0	80.10	79.60	79.80
	50	80.70	81.00	80.90
	100	81.30	83.30	82.30
	200	81.00	84.40	82.60
	300	82.40	83.90	**83.20**
1250	0	79.40	82.70	81.00
	50	80.70	80.70	80.70
	100	80.90	79.90	80.40
	200	81.60	83.30	**82.40**
	300	78.50	82.70	80.50

with the baseline. It indicates that the proposed algorithm may take reasonable contribution to the multi-label classification approaches. We found that in each category of training dataset, the system outperforms in experiments of using different unlabeled sets than the case of using no unlabeled texts. This is the reason that various selections of unlabeled texts involved in MASS can improve the performance of text classification. These experiments also show the role of labeled data in proposed model characterizing the silhouettes of the text clusters. Although the increase in size of labeled dataset also makes some contribution to the performance of system in general, the best result in each category seems to be stable with different number of unlabeled texts. By dividing the dataset into three different sub-datasets, MASS also overcomes the limitation in computational complexity.

4 Conclusions

In this paper, we proposed MASS—an approach for semi-supervised MLC to exploit label-specific features. Using two basic assumptions including the effect of label-specific features in learning process 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 experimental results show the promising trends in MASS for the MLC. Our work is currently seen as the initial step, more improvements, e.g. the method to effectively select unlabeled instances, or post-processing to prune the resulted clusters to remove outliers, should be done to evaluate the proposed approach.

Acknowledgements This work was supported in part by VNU Grant QG-15-22.

References

1. Elisseeff, A., Weston, J.: A kernel method for multi-labelled classification. NIPS **2001**, 681–687 (2001)
2. Rousu, J., Saunders, C., Szedmák, S., Shawe-Taylor, J.: Kernel-based learning of hierarchical multilabel classification models. J. Mach. Learn. Res. **7**, 1601–1626 (2006)
3. Silla Jr., C.N., Freitas, A.A.: A survey of hierarchical classification across different application domains. Data Min. Knowl. Discov. (DATAMINE) **22**(1–2), 31–72 (2011)
4. Tsoumakas, G., Katakis, I., Vlahavas, I.P.: Mining Multi-label Data. Data Min. Knowl. Discov. Handb. **2010**, 667–685 (2010)
5. Trohidis, K., Tsoumakas, G., Kalliris, G., Vlahavas, I.P.: Multi-label classification of music into emotions. ISMIR **2008**, 325–330 (2008)
6. Zhang, M.-L., Zhou, Z.-H.: ML-KNN: a lazy learning approach to multi-label learning. Pattern Recognit. (PR) **40**(7), 2038–2048 (2007)
7. Zhang, M.-L., Lei, W.: LIFT: multi-label learning with label-specific features. IEEE Trans. Pattern Anal. Mach. Intell. **37**(1), 107–120 (2015)

8. Zhang, J.-J., Fang, M., Li, X.: Multi-label learning with discriminative features for each label. Neurocomputing **154**, 305–316 (2015)
9. Huaqiao, Q., Zhang, S., Liu, H., Zhao, J.: A multi-label classification algorithm based on label-specific features. Wuhan Univ. J. Natl. Sci. **16**(6), 520–524 (2011)
10. Basu, S.: Semi-supervised clustering: probabilistic models, algorithms and experiments. University of Texas at Austin (2005)
11. Tian, D.: Semi-supervised learning for refining image annotation based on random walk model. Knowl. Based Syst. 72–80 (2014)
12. Dara, R., Kermer, S., Stacey, D.: Clustering unlabeled data with SOMs improves classification of labeled real-world data. In: Proceedings of the 2002 International Joint Conference on Neural Networks, pp. 2237–2242 (2002)
13. Luo, X., Liu, F., Yang, S., Wang, X., Zhou, Z.: Joint sparse regularization based sparse semi-supervised extreme learning machine (S3ELM) for classification. Knowl. Based Syst. **73**, 149–160 (2015)
14. Demirez, A., Bennett, K., Embrechts, M.: Semi-supervised clustering using genetic algorithms. In: Proceedings of Artificial Neural Networks in Engineering (ANNIE-99), pp. 809–814 (1999)
15. Zhang, W., Tang, X., Yoshida, T.: TESC: An approach to text classification using semi-supervised clustering. Knowl. Based Syst. **75**, 152–160 (2015)
16. Kong, X., Ng, M.K., Zhou, Z.-H.: Transductive multilabel learning via label set propagation. IEEE Trans. Knowl. Data Eng. **25**(3), 704–719 (2013)