Multi-label text classification algorithm based on semi-supervised learning

Min Tu¹, Shiyang Xu²

¹Jiangxi Police College, Nanchang, Jiangxi, 330100, China
²Patent Office of State Intellectual Property Office, China

Corresponding author and e-mail: Min Tu, 14371789@qq.com

Abstract. The standard Co-Training algorithm can effectively improve the performance of the classifier by using unlabeled data. In fact, however, it is difficult to meet the requirement of sufficient and redundant views. At the same time, the constraints of Co-Training is further relaxed to solve the problem of inaccurate confidence estimation of unlabeled samples. In this paper, immknn and mlkiss-knn are used to train the basic classifiers with great difference. By using unlabeled data in the way of providing Pseudotag sample to each other, a Multi-Label Text Classification Algorithm CT-MLTC based on Co-Training is proposed. Experimental results show that the algorithm has good performance in multi label text classification tasks with more unlabeled data.

1. Introduction

Semi-supervised learning (SSL) is a kind of extended supervised learning which uses unmarked data according to some principles. When the semi-supervised learning method is used in the classification task, it is usually called semi-supervised classification (SSC) [1]. The unlabeled data in SSC can be used as an auxiliary part of the training classifier. SSC can be carried out in two different ways: transductive learning, in which the labeled data is used to construct the classifier, and then the classifier is applied to the unmarked data; In inductive learning, some labeled data are excluded from the training data set as the verification set, and the feedback and induction are carried out by verifying the classification results, then the relevant parameters or rules are updated, and the classifier is optimized iteratively [2].

Cooperative training is a classic semi-supervised learning method. The Standard Co-Training algorithm is a classic semi-supervised learning algorithm based on Flow Pattern Hypothesis [3], which proves that the performance of the classifier can be effectively improved by the co-training algorithm use unlabeled data under the condition that the data of the training classifier are independent of each other. However, in real tasks, the requirement for data set of complete independence often can not met. S. Goldman and Y. Zhou proposed a cooperative training algorithm without sufficient and redundant views. In the algorithm, decision tree is used for cooperative training, and two classifiers are trained on the same attribute set. Each classifier can divide the sample space into several equivalent classes [4]. In the process of Co-Training, statistical techniques are used to estimate the confidence of the classifier to the unlabeled samples, and then the samples with high confidence are selected to mark the unlabeled samples. The samples and the mark results of the samples are put into another classifier.
training set, and the classifier is trained. The process is carried out in two classifiers until a stopping condition is met.

In order to further relax the constraints of collaborative training, Zhihua Zhou et al. proposed the Tri-training [5], which has no restrictions on the sufficient and redundant view of attribute set, and does not require different types of classifiers to be used in collaborative training. The most significant feature of this algorithm is that three classifiers were used for collaborative training. In this way, the confidence estimation problem of unlabeled samples and the prediction problem of unlabeled samples can be simplified, and the idea of integrated learning is introduced, which improves the generalization ability of the classifier in the collaborative training algorithm. However, compared with the explicit marker confidence estimation method in standard cooperative training, the implicit marker confidence estimation method in Tri-Training is often not accurate enough to estimate the confidence, especially when the initialization of the classifier is weak. Unlabeled samples are likely to be mislabeled by two auxiliary classifiers at the same time, which brings noise to the training of the main classifier. However, Subsequent theoretical research have found that data set of this kind of algorithm does not need to satisfy these requirements such as independent, sufficient and redundant view. The generalization performance can be improved by providing pseudo-labeled samples with each other, that is, significant differences between basic classifiers are needed.

From the perspective of Co-Training, imml-knn and mlplsa-knn are used to train the basic classifiers with great difference. By using unlabeled data in the way of providing Pseudotag sample to each other, so CT-MLTC based on Co-Training is proposed.

2. Construct a cooperative training classifier
Cooperative training is essentially a divergence-based method. The basic classifiers that need to be generated not only have significant differences with each other, but also have strong performance. Otherwise, the classifiers with poor performance will continue to misjudge the data labels when using unlabeled data, which will affect each other and eventually reduce the performance of the entire classification model. Therefore, on the one hand, at the beginning of collaborative training, classifiers with large differences are often selected, so that in repeated iterations of collaborative training, the classifier can always maintain large differences. On the other hand, the basic classifier should have a high classification accuracy in multi-label text classification, otherwise it will cause the accumulation of wrong classification.

The effectiveness of the IMML-KNN and MLPLSA-KNN algorithms in multi-label text classification tasks has been verified, the IMML-KNN algorithm is based on the probability of co-occurrence of category labels, and the MLPLSA-KNN algorithm is from a semantic perspective. The two different starting points ensure that the trained classifiers are always different. Therefore, in this paper, IMML-KNN and MLPLSA-KNN are selected as the algorithms for training the basic classifier.

3. Multi-label text classification algorithm CT-MLTC based on collaborative training
3.1. Definition
The training sample set $D$ is composed of a labeled sample set $L$ and an unlabeled sample set $U$.

$$D = \{(x_1, y_1), (x_2, y_2) \cdots (x_{|L|}, y_{|L|}), (x_{|L|+1}, y_{|L|+1}) \cdots (x_{|L|+|U|}, y_{|L|+|U|})\}$$ (1)

$$L = \{(x_1, y_1), (x_2, y_2) \cdots (x_{|L|}, y_{|L|})\}$$ (2)

$$U = \{x_{|L|+1}, x_{|L|+2} \cdots x_{|L|+|U|}\}$$ (3)

Among them, $|L|$ is the number of labeled samples, and $|U|$ is the number of unlabeled samples.
3.2. Algorithm input
Labeled sample set L, unlabeled sample set U, the defined number of nearest neighbors K, the number of defined topics NUM_topic, the sample set to be predicted Test, the smoothing parameter s=1.

3.3. Algorithm output
Prediction tag sequence of each text sample in Test

3.4. Algorithmic process
Step 1: The labeled sample set was divided into $L_1$ and $L_2$ by Repeated random sampling of Bootstrap. The initial classifiers $H_1(1)$ and $H_2(1)$ are trained using IMML-KNN algorithm and MLPLSA-KNN algorithm respectively. To ensure the difference of classifiers.

Step 2: For the first time, two initial classifiers $H_1(1), H_2(1)$ are used to traverse the unlabeled sample set $U$, and for the unlabeled sample $x_u(x_u \in U)$, text samples with high self-confidence are selected and given pseudo-labels. The prediction result was used as a pseudo-label for $x_u$ and was saved in the set $Set_1(1), Set_2(1)$. After the first traversal, the classification result set $Set_2(1)$ of $H_2(1)$ is added to the training sample $L_1$ of $H_1$, which is updated to $L_1 + Set_2(1)$. Similarly, the training sample of $H_2$ is also updated to $L_2 + Set_1(1)$.

Step 3: the updated training samples was trained again by the previous algorithm to obtain the classifiers $H_1(2), H_2(2)$ and repeat step 2. However, $Set_1(1)$ and $Set_2(1)$ generated will not be regarded as labeled samples, but will be put into the unlabeled sample set $U$ as unlabeled samples.

Step 4: Repeat steps 2 and 3 until the classification result does not change significantly, and The final classifier $H_1(n), H_2(n)$ is trained by $L_1 + Set_2(n), L_2 + Set_1(n)$.

Step 5: Use the final classifier to classify the test sample set Test.

4. Experiment and analysis
In order to test the classification performance of the algorithm in different proportions of unlabeled data, the public data set Delicious on Mulan, which is a multi label classification research platform was selected in this experiment. By removing the category labels to construct different Delicious data sets with different proportions of unlabeled data. Then, Hamming Loss, Ranking Loss, Coverage, Precision and One Error are used as evaluation indicators.

In order to verify the effectiveness and superiority of the proposed algorithm in dealing with multi-label text classification problems, we compared with other semi-supervised multi-label classification algorithms in experiments. Miha Pavlinek and Vili Podgorelec proposed a self-training-based LDA method STLDA [6], which continuously tried to use unlabeled text data through self-training to further iteratively optimize the performance of the classifier. Tan Qiaoyu et al. proposed a graph-based transductive linear classification algorithm SMILE [7]. The missing category label information can be supplemented by the domain map, which has a good effect when processing incomplete category label data.

Wen Zhang et al. proposed a classification algorithm TESC[8] based on semi-supervised clustering, which calculates the distance between the unlabeled text data and each cluster center to cluster. Thus, the unlabeled text data is converted into a prior knowledge, which helps to improve the classification performance. The experimental results are shown in the following table.

| Evaluation indicators | ST LDA | SMILE | TESC  | CT-MLTC |
|-----------------------|--------|-------|-------|---------|
| Hamming Loss          | 0.049±0.002 | 0.057±0.002 | 0.082±0.005 | 0.036±0.001 |
| Precision             | 0.897±0.003 | 0.864±0.001 | 0.887±0.001 | 0.896±0.002 |
| RankingLoss           | 0.257±0.002 | 0.465±0.002 | 0.393±0.011 | 0.456±0.003 |
| Coverage              | 6.787±0.103 | 6.287±0.173 | 6.436±0.021 | 6.176±0.037 |
| One error             | 0.434±0.001 | 0.514±0.001 | 0.413±0.002 | 0.396±0.003 |
Table 2. Experimental results in Delicious dataset with 50% of unlabeled data.

| Evaluation indicators | ST LDA       | SMILE       | TESC        | CT-MLTC      |
|-----------------------|--------------|-------------|-------------|--------------|
| Hamming Loss          | 0.069±0.001  | 0.076±0.001 | 0.103±0.007 | 0.097±0.007  |
| Precision             | 0.734±0.001  | 0.754±0.003 | 0.799±0.005 | 0.867±0.001  |
| Ranking Loss          | 0.358±0.007  | 0.537±0.005 | 0.499±0.014 | 0.517±0.003  |
| Coverage              | 8.103±0.035  | 8.284±0.053 | 8.537±0.011 | 8.276±0.012  |
| One error             | 0.517±0.001  | 0.543±0.003 | 0.524±0.001 | 0.496±0.001  |

Table 3. Experimental results in Delicious dataset with 80% of unlabeled data.

| Evaluation indicators | ST LDA       | SMILE       | TESC        | CT-MLTC      |
|-----------------------|--------------|-------------|-------------|--------------|
| Hamming Loss          | 0.274±0.034  | 0.256±0.023 | 0.186±0.033 | 0.147±0.023  |
| Precision             | 0.517±0.001  | 0.664±0.003 | 0.787±0.007 | 0.821±0.005  |
| Ranking Loss          | 0.486±0.001  | 0.554±0.001 | 0.513±0.007 | 0.537±0.001  |
| Coverage              | 12.787±1.103 | 18.287±1.927| 9.176±0.118 | 9.072±0.102  |
| One error             | 0.651±0.003  | 0.614±0.004 | 0.671±0.003 | 0.598±0.003  |

The evaluation index values of four different algorithms on the Delicious dataset with different proportions of unlabeled data can be found in Figure 1. Coverage is too large compared to other values. In order to visually see the differences between algorithms, the coverage value is processed, so that the five evaluation indicators can be reflected in the following figure.

![Figure 1. Experimental results in Delicious dataset with 20% of unlabeled data.](image-url)
Figure 2. Experimental results in Delicious dataset with 50% of unlabeled data.

Figure 3. Experimental results in Delicious dataset with 80% of unlabeled data.

When the unmarked data accounts for 20%, the Hamming Loss Rate, Coverage Rate, Accuracy and Error Rate are all optimal, only the Sorting Loss rate is higher, which is 0.199 more than STLDA algorithm.

When the ratio of unlabeled data is 50%, the Accuracy, Coverage and Error Rate of the algorithms are all the best, Hamming Loss Rate is 0.028 more than STLDA algorithm, and Sorting Loss Rate is 0.199 more than stlda algorithm.

When the unmarked data accounts for 80%, Hamming Loss Rate, Coverage Rate, Sorting Loss Rate, Accuracy and Error Rate are all optimal.

In summary, the experiment proves the effectiveness and superiority of the CT-MLTC algorithm proposed in this paper when processing multi-label text classification tasks with unlabeled data. And compared with other algorithms based on semi-supervised learning, the CT-MLTC algorithm not only performs well in terms of classification performance, but also has stronger performance than some algorithms in using unlabeled data.
5. Conclusions
In this paper, an initial classifier for the collaborative training is constructed by the IMML-KNN and MLPLSA-KNN algorithms. Therefore, the divergence of the classifier can be guaranteed and the unlabeled data can be used effectively. From that a multi-label text classification algorithm CT-MLTC based on collaborative training is proposed. Through the experiments in different density datasets of unlabeled data, the performance in the task of multi label text classification with more unlabeled data has been proved. There are still many directions worth exploring. For example, in different task scenarios, the reliability and robustness of the algorithm proposed need to be verified in more data sets.

Acknowledgement
This research is supported by the 2019 Science and Technology Project of the Ministry of Education "Public Security Big Data Analysis for early warning and pre-control" (GJJ191026), thanks again.

References
[1] Olivier C, Bernhard S, Alexander Z. Semi-supervised learning[M]. Springer US, 2009.
[2] Chen K, Wang S. Semi-Supervised Learning via Regularized Boosting Working on Multiple Semi-Supervised Assumptions[J]. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2011, 33(1):129.
[3] Blum A, Mitchell T M. Combining Labeled and Unlabeled Sata with Co-Training[C]// Eleventh Conference on Computational Learning Theory, COLT 1998, Madison, Wisconsin, Usa, July. DBLP, 1998:92-100.
[4] Goldman S A, Zhou Y. Enhancing Supervised Learning with Unlabeled Data[J]. Proc. 17th Int'l Conf. on Machine Learning, 2000, 2000:327-334.
[5] Zhou Z H, Li M. Tri-Training: Exploiting Unlabeled Data Using Three Classifiers[J]. IEEE Transactions on Knowledge & Data Engineering, 2005, 17(11):1529-1541.
[6] Pavlinek M, Podgorelec V. Text classification method based on self-training and LDA topic models[M]. Pergamon Press, Inc. 2017.
[7] Tan Q, Yu Y, Yu G, et al. Semi-Supervised Multi-Label Classification using Incomplete Label Information[J]. Neurocomputing, 2017.
[8] Zhang W, Tang X, Yoshida T. TESC: An approach to TExt classification using Semi-supervised Clustering[J]. Knowledge-Based Systems, 2015, 75:152-160.