Leveraging Sub-class Partition Information in Binary Classification and Its Applications[1]

Baoli Li[2] **and Carl Vogel**[3]

Abstract Sub-class partition information within positive and negative classes is often ignored by a binary classifier, even when these detailed background information is available at hand. It is expected that this kind of additional information can help to improve the differentiating capacity of a binary classifier. In this paper, a binary classification strategy via multi-class categorization is proposed to leverage sub-class partition information when they are available. Empirical studies on the 20 newsgroups dataset demonstrate the benefits of this strategy. Furthermore, a preliminary application of this binary classification strategy for multi-label classification problem is given with promising results.

1. Introduction

Binary classification, which aims at classifying an item positively or negatively with respect to a class, is important. On one hand, it can be directly applied to solve some practical problems, e.g. spam filtering. On the other hand, binary classifiers can be assembled to solve multi-class classification problems [1] and multi-label classification problems [4].

In multi-class classification, an item belongs to only one of a set of predefined classes. Error Correcting Output Codes (ECOC) provides a general framework to transform a multi-class problem into a set of binary classification problems [2]. In this framework, each class is assigned a unique codeword, which is a binary string of length *N*. With each bit *i* of these codewords, the original multi-class dataset is split into two mixed classes: one contains all samples of the classes that have value 1 at bit *i* of their codewords, and the other has all the remaining samples. The *N* binary classifiers corresponding to each bit are therefore learned for

[1] This research is supported by the Science Foundation Ireland (Grant 07/CE/I1142) as part of the Centre for Next Generation Localisation (www.cngl.ie) at Trinity College Dublin.

[2] Trinity College Dublin, Dublin 2, Ireland. Email: baoli.li@cs.tcd.ie

[3] Trinity College Dublin, Dublin 2, Ireland. Email: vogel@cs.tcd.ie

classifying a new sample and producing a codeword for it. The predicted class is the one whose codeword is closest to the codeword produced by the *N* binary classifiers.

In multi-label classification, an item may be assigned to more than one class. A commonly used approach to address multi-label problem is the so-called one-vs-rest (a.k.a. one-vs-all) strategy, in which each potential class is examined by a binary classifier. To train a binary classifier for a class, the samples of this class in the training set are used as positive samples, where the rest samples from all other classes form the negative class.

As shown in the above situations, when we build a binary classifier for solving a complex problem, positive and/or negative classes may be derived by artificially or randomly combining several sub-classes, and the information of sub-class partition may be at hand. Unfortunately, the current binary classification methods never take into account this kind of information. Intuitively, this kind of additional information could help to improve the performance of a binary classifier.

In this research, we propose a simple strategy to improve binary classification via multi-class categorization for applications where sub-class partition information within positive and/or negative classes is available. Based on sub-class partition information, a multi-class classifier is built and a new item is labeled according to its prediction. As multi-class categorization may implicitly capture the interactions between sub-classes, we expect that the detailed sub-classes will help differentiating the positive and negative classes with high accuracy.

In the following sections, we explain our proposed strategy and empirically investigate whether this binary classification strategy with sub-class information is better than the traditional binary classification strategy. Experiments on the 20 newsgroups dataset demonstrate that this intuitive strategy can lead to better performance on average, especially the macro-averaging scores. Then, in section 4, we further explore to apply this strategy in solving a multi-label classification problem. Section 5 concludes the paper with planned future work.

2. Binary Classification via Multi-class Categorization

Our proposed strategy targets at solving a special kind of binary classification, where positive and/or negative classes artificially consist of several sub-classes. Suppose that the positive and negative classes in a binary classification problem contain $|P|$ and $|N|$ sub-classes, respectively, where $P=\{p_1, p_2, \ldots, p_{|P|}\}$ and $N=\{n_1, n_2, \ldots, n_{|N|}\}$. Our strategy then works as follows:

a). Build a multi-class classifier C_m, which considers $|P|+|N|$ sub-classes.

b). Classify a new item α with the above learned classifier C_m and suppose its prediction is c.

c). If $c\in P$, α is labled as positive;otherwise, α belongs to negative class.

If the multi-class classifier C_m supports probability output, the probability sums of sub-classes within P and N will be used for final decision. This binary classification strategy is expected to work with any multi-class categorization algorithm.

3. Experiments and Discussions

3.1 Dataset and Experimental Design

To show the effectiveness of the proposed strategy, we experiment with the 20 Newsgroups dataset. The dataset is nearly evenly partitioned across 20 different newsgroups, each corresponding to a different topic. Among the different versions of this dataset, we use the so called bydate version[4] which is sorted by date and divided into a training set (60%) and a test set (40%), without cross-posts (duplicates or multi-labeled documents) and newsgroup-identifying headers. The total number of newsgroup documents in the "bydate" version is 18,846, with 11,314 for training and 7,532 for testing. We choose this dataset for experiments because it is almost balanced and without multi-label cases. We hope to remove the effects caused by these two factors.

To get binary datasets, we randomly choose one or more original classes and combine them together as a positive class and take the rest to form its complementary negative class.

With the 20 newsgroups dataset, we have totally 616,665 (=$C_{20}^{1}+C_{20}^{2}+...+C_{20}^{10}$) possible separations, which can be classified into ten types: 1vs19 (1 class as positive and the rest 19 classes as negative), 2vs18, 3vs17, …, and 10vs10. As the number of possible separations is so huge, to make our experiments tractable, we decide to randomly choose at most 100 separations from different types. Obviously, we only have 20 different separations of type 1vs19, and so we totally experiment with 920 different separations[5]. Separations from the same type roughly have the same class distribution. For example, in the separations of type 1vs19, the ratio of positive documents to negative documents is around 1 to 19.

We compare our proposed strategy with the traditional binary classification strategy that doesn't take into account sub-class information even though it is available. We label the traditional strategy as BIN, and our proposed one considering sub-class information as 2vM.

[4] http://www.ai.mit.edu/~jrennie/20Newsgroups/.

[5] The 920 binary separations of the 20 newsgroup dataset are available at http://www.cs.tcd.ie/Carl.Vogel/BTCvM/.

We experiment with a widely used text categorization algorithm: Naïve Bayes with multinomial model [3]. At the preprocessing stage, we remove stop words, but without stemming. Words appearing only in one document are ignored. After these processing, we get totally 49,790 words as features. The two common metrics, Micro-averaging and macro-averaging F-1 measures, are used to evaluate performance.

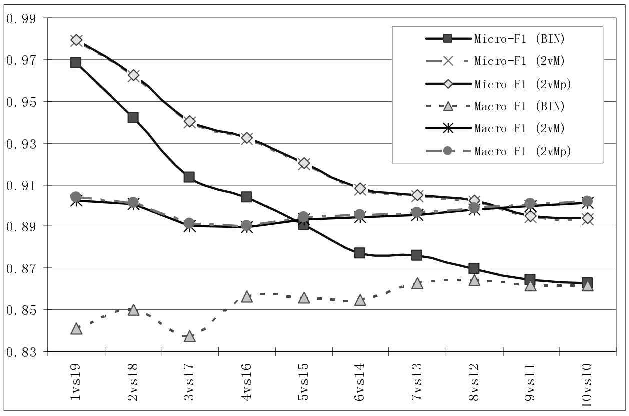

Figure 1. F-1 measures of the two strategies on the 20 newsgroups dataset.

3.2 Results and Discussions

Figures 1 shows the performance of the two binary text classification strategies (BIN and 2vM). The values are the averages of all separations of the same type. Figure 1 also shows the performance of a variant of 2vM (labeled as 2vMp), which uses the probability output for decision. 2vMp and 2vM have very close performance, although the former is statistically significantly better than the latter.

The figure does demonstrate the effectiveness of the simple strategy, i.e. binary classification with sub-class information. As the dataset becomes more balanced, the difference of Mic-F1 between BIN and 2vM grows larger, where that of Mac-F1 gets smaller. With the extremely imbalanced separation (1vs19), the Mic-F1 of the 2vM strategy is just a little higher than that of the traditional BIN strategy (0.9789 vs 0.9683). Comparatively speaking, the performance of 2vM is more stable than that of BIN. In the 920 runs, 2vM beats BIN 901 times on Mic-F1 with average gain 0.02863 and 805 times on Mac-F1 with average gain 0.04857. The average gains achieved by BIN over 2vM are 0.00376 on Mic-F1 and 0.01729 on Mac-F1, respectively.

As the dataset changes from imbalanced to balanced, Mic-F1 is getting worse, while Mac-F1 is steadily becoming better. For example, in figure 1, the Mic-F1 of BIN drops 10.87% from 0.9683 to 0.8630, while that of 2vM goes down 8.75%

from 0.9789 to 0.8933. At the same time, the Mac-F1 of BIN increases 2.42% from 0.8413 to 0.8617, and that of 2vM drops a bit from 0.9025 to 0.9013.

The trend of Mic-F1 conforms to our intuition, as we can easily get higher accuracy with an extremely imbalanced dataset by simply outputting the major one of the two classes. If the two classes within a dataset have more equal size, the problem will become harder because the uncertainty of such a dataset becomes higher. As Mac-F1 score is more influenced by the performance on rare classes, the overall average scores are poor on imbalanced datasets because classifiers often perform poorly on rare classes.

The imbalance of datasets also contributes to the following phenomenon: the deviation between Mic-F1 and Mac-F1 is larger for imbalanced separations than for balanced separations. When dataset is getting balanced, Mic-F1 and Mac-F1 will become very close.

We need to point out that the above analyses are based on the overall tendency, as the values are averages of many runs. The performance difference between 2vM and BIN is statistically significant (P values of t-test for Mic-F1 and Mac-F1 are 4.3E-269 and 8.2E-116, respectively), but it doesn't mean that 2vM can beat BIN on every possible separation. With 2vM, we are more likely to get better results. It is therefore sensible to choose 2vM strategy if we are not allowed to compare these two strategies offline.

Actually, we also tested the proposed strategy with another dataset (Reuters-21578-SL-8Class, also available at http://www.cs.tcd.ie/Carl.Vogel/BTCvM/) and other two categorization algorithms: SVM and kNN. We obtained similar conclusions. Due to space limit, we have to omit all these results in this paper.

4. Application: Multi-label Classification

Empirical study shows that 2vM is a good substitute for BIN when sub-class partition information is available. We then explore to apply this 2vM strategy to solve some practical problems. In a preliminary application, we considered a multi-label text classification problem.

As we mentioned earlier, one-vs-rest is a widely used strategy for multi-label classification [4]. It trains a set of binary classifiers, each of which corresponds to a class. When classifying a new document, these binary classifiers are applied in turn and decide whether the new document could be classified into each class. The most possible class will be given to those documents that are not assigned any class label by those binary classifiers.

We experiment with the two binary classification strategy (BIN and 2vM) and Naïve Bayes algorithm on the Reuters-21578 dataset (Apte' Split[6]) with 91 classes (90 classes with an unknown class). The widely used metrics for multi-label

[6] http://kdd.ics.uci.edu/databases/reuters21578/reuters21578.html.

classification, Hamming Loss [4] and Micro-Averaging F1, are used for measuring performance. Hamming Loss captures the differences between a system's output and the golden answer. The less the hamming loss is, the better the system's performance is. To build the multi-class classifier required by the 2vM strategy, we take each existing label combination as a new class and thus transform a multi-lable problem into a multi-class single-label problem.

Table 1 gives the experimental results. With the 2vM strategy, we get better results. The Hamming Loss value drops 35.87% from 0.009323 to 0.005979, where Mic-F1 increases 8.48% from 0.703087 to 0.762723. The probablity variant 2vMp can boost the results a little bit.

Table 1. Results of BIN and 2vM for multi-label classification.

Strategy	BIN	2vM	2vMp
Hamming Loss	0.009323	0.005979	0.005959
Micro-Avg. F-1	0.703087	0.762723	0.763454

5. Conclusion and Future Work

In this paper, we empirically demonstrate the effectiveness of a simple strategy for improving binary classification via multi-class categorization when sub-class information is available. On average, the proposed 2vM strategy brings better results over the traditional binary classification strategy (BIN), especially for macro-averaging scores and on imbalanced datasets. In a preliminary application, we employed the proposed strategy to solve a multi-label classification problem and got promising results. In the future, we are planning to experiment with more datasets. We assume the proposed strategy could be applicable to not only text data but also other classification data. We also expect to use this strategy to enhance those multi-class categorization algorithms based on binary classification, e.g. error correcting output codes.

References

1. Erin L. Allwein et. al. 2000. Reducing Multiclass to Binary: A Unifying Approach for Margin Classifiers. Journal of Machine Learning Research, 1: 113-141.
2. Rayid Ghani. 2000. Using Error-Correcting Codes for Text Classification. In Proceedings of ICML-2000.
3. Andrew Mccallum and Kamal Nigam. 1998. A Comparison of Event Models for Naive Bayes Text Classification. In Proceedings of AAAI/ICML-98 Workshop on Learning for Text Categorization.
4. Grigorios Tsoumakas and Ioannis Katakis. 2007. Multi-Label Classification: An Overview. Journal of Data Warehousing and Mining, 3(3):1-13.