Combining labeled and unlabeled data for text classification with a large number of categories

Rayid Ghani
Center for Automated Learning and Discovery
Carnegie Mellon University
Rayid.Ghani@cs.cmu.edu

Accenture Technology Labs
Northbrook, IL 60062
Rayid.Ghani@accenture.com

Abstract

We develop a framework to incorporate unlabeled data in the Error-Correcting Output Coding (ECOC) setup by decomposing multiclass problems into multiple binary problems and then use Co-Training to learn the individual binary classification problems. We show that our method is especially useful for classification tasks involving a large number of categories where Co-training doesn't perform very well by itself and when combined with ECOC, outperforms several other algorithms that combine labeled and unlabeled data for text classification in terms of accuracy, precision-recall tradeoff, and efficiency.

1 Introduction

A major difficulty with supervised learning techniques for text classification is that they often require a large number of labeled examples to learn accurately. One way to reduce the amount of labeled data required is to develop algorithms that can learn from a small number of labeled examples augmented with a large number of unlabeled examples.

There has been recent work in supervised learning algorithms using labeled and unlabeled data in problem domains where the features naturally divide into two disjoint sets, and algorithms that use this division, fall into the co-training setting (Blum & Mitchell, 1998).

Published studies on text classification with Co-training algorithms (Blum & Mitchell, 1998; Nigam & Ghani, 2000) have focused on small, often binary, problems and it is not clear whether their conclusions would generalize to real-world classification tasks with a large number of categories. On the other hand, Error-Correcting Output Codes (ECOC) are well suited for classification tasks with a large number of categories. However, most of the earlier work has focused neither on text classification problems (except our earlier work (Ghani, 2000) and (Berger, 1999)), nor on problems which deal with a large number of categories.

2 Combining ECOC and Co-Training

We propose a new algorithm aimed at combining the advantages that ECOC offers for supervised classification with a large number of categories and that of Co-Training for combining labeled and unlabeled data. Since ECOC decomposes a multiclass problem into multiple binary problems, we incorporate unlabeled data by learning each of these binary problems using Co-training.

The algorithm we propose is as follows:

- Training Phase
  1) Given a problem with m classes, create an m x n binary matrix M. 2) Each class is assigned one row of M. 3) Train n Co-trained classifiers to learn the n binary functions (one for each column since each column divides the dataset into two groups).
- Test Phase
  1) Apply each of the n single-bit Co-trained classifiers to the test example. 2) Combine the predictions to form a binary string of length n. 3) Classify to the class with the nearest codeword

Of course, an m-class problem can be decomposed naively into n binary problems and co-training can then learn each binary problem, but our approach is more efficient since by using ECOC we reduce the number of models that our classifier constructs and our approach scales up sublinearly with the number of classes (More details about using ECOC for efficient text classification using ECOC can be found in (Ghani, 2001). We also believe that our approach will perform better than the naive approach under the conditions that: 1) ECOC can outperform Naive Bayes on a multiclass problem (which actually learns one model for every class), 2)Co-Training can improve a single Naive Bayes classifier on a binary problem by using unlabeled data

The complication that arises in fulfilling condition 2 is that unlike normal binary classification problems where Co-Training has been shown to work well, the use of Co-Training in our case involves binary problems which themselves consist of multiple classes. Since the two classes in each bit are created artificially by ECOC and consist of many "Real" classes, there is no guarantee that Co-Training can learn these arbitrary binary functions.

If Co-training does not contain at least one labeled example from one of the original classes, it is likely that it will never be confident about labeling any unlabeled example from that class. Under the conditions that : 1) the initial

Table 1. Classification accuracies for the two datasets. Naive Bayes and ECOC do not use any unlabeled data whereas all the other algorithms have access to the same amount of labeled and unlabeled data.

Dataset	Naive Bayes		ECOC		EM	Co-Training	ECOC + Co-Training
	10% Labeled	100% Labeled	10% Labeled	100% Labeled	10% Labeled	10% Labeled	10% Labeled
Jobs-65	50.1	68.2	59.3	71.2	58.2	54.1	64.5
Hoovers-255	15.2	32.0	24.8	36.5	9.1	10.2	27.6

labeled examples cover every "original" class, 2) the target function for the binary partition is learnable by the underlying classifier, 3) the feature split is redundant and independent so that the co-training algorithm can utilize unlabeled data, theoretically, our combination of ECOC and Co-Training should result in improved performance by using unlabeled data.

3 Datasets

Hoovers dataset (used previously in (Ghani et al., 2001)) consists of web pages from 4285 corporate websites organized into 255 industry sectors (classes). Since there is no natural feature split, we randomly divide the vocabulary in two equal parts and apply Co-Training to the two feature sets. We have previously (Nigam & Ghani, 2000) shown that this random partitioning works reasonably well in the absence of a natural feature split.

We also use a dataset obtained from WhizBang! Labs consisting of Job titles and Descriptions organized in a two level hierarchy with 15 first level categories and 65 leaf categories. In all, there are 132000 examples and each example consists of a Job Title and a corresponding Job Description which we consider as two independent and redundant feature sets for Co-Training.

4 Experimental Results

All the codes used are BCH codes (31-bit codes for the Jobs dataset and 63-bit codes for the Hoovers Dataset) and are similar to those used in (Ghani, 2000).

Table 1 shows the results of the experiments comparing our proposed algorithm with EM and Co-Training. The baseline results with Naive Bayes and ECOC using no unlabeled data are also given, as well as those when all the labels are known which serve as an upper bound for the performance of our algorithm.

From results reported in recent papers (Blum & Mitchell, 1998; Nigam & Ghani, 2000), it is not clear whether co-training will perform well by itself and give us any leverage out of unlabeled data on a dataset consisting of a large number of classes. We can see that both Co-Training and EM did not improve the classification accuracy by using unlabeled data on the Hoovers-255 dataset; rather they had a negative effect and resulted in decreased accuracy. The accuracy reported for EM and Co-Training was decreasing at every iteration and since the experiments were stopped at different times, they are not comparable to each other. On the other hand, our proposed combination of ECOC and Co-Training does indeed take advantage of the unlabeled data much better than EM and Co-Training and outperforms both of those algorithms on both datasets. It is also worth noting that ECOC outperforms Naive Bayes for both datasets and this is more pronounced when the number of labeled examples is small.

We also evaluate our results in terms of precision and recall but do not show the graphs because of space limitations. The figures can be found in (Ghani, 2001). We find that our method performs extremely well and can provide very high precision results unlike Naive Bayes and EM. This is a property of ECOC and is discussed further in (Ghani, 2001).

5 Conclusions

The results described in this paper lead us to believe that the combination of ECOC and Co-Training algorithms is indeed useful for learning with labeled and unlabeled data. We have shown that our approach outperforms both Co-Training and EM algorithms, which have previously been shown to work well on several text classification tasks. Our approach not only performs well in terms of accuracy but also provides a smooth precision-recall tradeoff which is useful in applications requiring high-precision results. Furthermore, we have shown that the framework presented in this paper results in text classification systems that are both computationally efficient (through the use of short error-correcting codes) and need very few labeled examples to learn accurately.

References

Berger, A. (1999). Error-correcting output coding for text classification. *IJCAI-99: Workshop on machine learning for information filtering*.

Blum, A., & Mitchell, T. (1998). Combining labeled and unlabeled data with co-training. *COLT 1998.*

Ghani, R. (2000). Using error-correcting codes for text classification. *ICML-00.*

Ghani, R. (2001). *Using error-correcting codes for efficient text classification with a large number of categories. masters thesis* (Technical Report). Center for Automated Learning and Discovery, Carnegie Mellon University.

Ghani, R., Slattery, S., & Yang, Y. (2001). Hypertext categorization using hyperlink patterns and meta data. *ICML-01.*

Nigam, K., & Ghani, R. (2000). Analyzing the applicability and effectiveness of co-training. *CIKM-00.*