Improving Text Classification by Web Corpora*

Rafael Guzmán[1,2], Manuel Montes[3], Paolo Rosso[2], and Luis Villaseñor[3]

[1] FIMEE, Universidad de Guanajuato, Mexico
guzman@salamanca.ugto.mx
[2] DSIC, Universidad Politécnica de Valencia, Spain
prosso@dsci.upv.es
[3] LTL, Instituto Nacional de Astrofísica Óptica y Electrónica, Mexico
{mmontesg,villasen}@inaoep.mx

Summary. A major difficulty of supervised approaches for text classification is that they require a great number of training instances in order to construct an accurate classifier. This paper proposes a semi-supervised method that is specially suited to work with very few training examples. It considers the automatic extraction of unlabeled examples from the Web as well as an iterative integration of unlabeled examples into the training process. Preliminary results indicate that our proposal can significantly improve the classification accuracy in scenarios where there are less than ten training examples available per class.

1 Introduction

Nowadays there is a lot of digital information available from the Web. This situation has produced a growing need for tools that help people to find, filter and analyze all these resources. In particular, text classification [4], the assignment of free text documents to one or more predefined categories based on their content, has emerged as a very important component in many information management tasks.

The state-of-the-art approach for automatic text classification considers the application of a number of statistical and machine learning techniques, including regression models, Bayesian classifiers, support vector machines (SVM), nearest neighbor classifiers (k-NN) and neuronal networks [4]. A major difficulty with this kind of supervised techniques is that they commonly require a great number of labeled examples (training instances) to construct an accurate classifier. Unfortunately, because a human expert must manually label these examples, the training sets are extremely small for many application domains. In order to overcome this problem, recently many researchers

* This work was done under partial support of CONACYT-Mexico (43990), MCyT-Spain (TIN2006-15265-C06-04) and PROMEP (UGTO-121).

have been working on semi-supervised learning algorithms (for an overview see [5]). It has been showed that augmenting the training set with additional information it is possible to improve the classification accuracy using different learning algorithms such as naïve Bayes [3], SVM [1], and k-NN [7].

In this paper we propose a new method for semi-supervised text classification. This method differs from previous approaches in three main concerns. First, it is specially suited to work with very few training examples. Whereas previous methods consider groups of ten and even hundreds of training examples, our method allows working with less than ten labeled examples per class. Second, it does not require a predefined set of unlabeled examples. It considers the automatic extraction of related untagged data from the Web. Finally, given that it deals with very few training examples, it does not aim including a lot of additional information in the training phase; on the contrary, it only incorporates a small group of examples that considerably augment the dissimilarities among classes.

It is important to point out that the Web has been lately used as a corpus in many natural language tasks [2]. In particular, Zelikovitz and Kogan [8] proposed a method for mining the Web to improve text classification by creating a background text set. Our method is similar to this approach in that it also mines the Web for additional information (extra-unlabeled examples). Nevertheless, our method applies finer procedures to construct the set of queries related to each class and to combine the downloaded information.

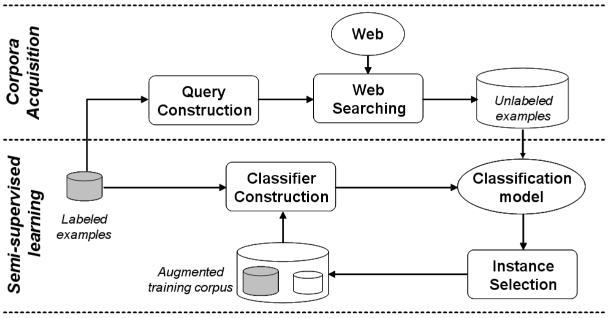

Fig. 1. General overview of the method

2 Proposed Method

Figure 1 shows the general scheme of the proposed method. It consists of two main processes. The first one deals with the corpora acquisition from the Web, while the second one focuses on the semi-supervised learning problem. The following sections describe in detail these two processes.

2.1 Corpora Acquisition

This process considers the automatic extraction of unlabeled examples from the Web. It first constructs a number of queries by combining the most significant words for each class; then, using these queries it looks at the Web for some additional training examples related to the given classes.

Query Construction. In order to form queries for searching the Web, it is necessary to previously determine the set of relevant words for each class in the training corpus. The criterion used for this purpose is based on a combination of the frequency of occurrence and the information gain of words. We consider that a word w_i is relevant for class C if:

1. The frequency of occurrence of w_i in C is greater than the average occurrence of all words (happening more than once) in that class. That is:

$$f_{w_i}^C > \frac{1}{|C'|} \sum_{\forall w \in C'} f_w^C, \ where\ C' = \{w \in C | f_w^C > 1\}$$

2. The information gain of w_i with respect to C is positive ($IG_{w_i}^C > 0$).

Once obtained the set of relevant words per class, it is possible to construct the corresponding set of queries. Founded on the method by Zelikovitz and Kogan [8], we decide to construct queries of three words. This way, we create as many queries per class as all three-word combinations of its relevant words. We measure the significance of a query $q = w_1, w_2, w_3$ to the class C as:

$$\Gamma_C(q) = \sum_{i=1}^{3} f_{w_i}^C \times IG_{w_i}^C$$

Web Searching. The next action is using the defined queries to extract from the Web a set of additional unlabeled text examples. Based on the observation that most significant queries tend to retrieve the most relevant web pages, our method for searching the Web determines the number of downloaded examples per query in a direct proportion to its Γ-value. Therefore, given a set of M queries $q_1, , q_M$ for class C, and considering that we want to download a total of N additional examples per class, the number of examples to be extracted by a query q_i is determined as follows:

$$\Psi_C(q_i) = \frac{N}{\sum_{k=1}^{M} \Gamma_C(q_k)} \times \Gamma_C(q_i)$$

2.2 Semi-supervised learning

As we previously mentioned, the purpose of this process is to increase the classification accuracy by gradually augmenting the originally small training set with the examples downloaded from the Web. Our algorithm for semi-supervised learning is an adaptation of a method proposed elsewhere [6]. It mainly considers the following steps:

1. Build a weak classifier (C_l) using a specified learning method (l) and the training set available (T).
2. Classify the downloaded examples (E) using the constructed classifier (C_l). In order words, estimate the class for all downloaded examples.
3. Select the best m examples ($E_m \subseteq E$) based on the following two conditions:
   a) The estimate class of the example corresponds to the class of the query used to download it. In some way, this filter works as an ensemble of two classifiers: C_l and the Web (expressed by the set of queries).
   b) The example has one of the m-highest confidence predictions.
4. Combine the selected examples with the original training set ($T \leftarrow T \cup E_m$) in order to form a new training set. At the same time, eliminate these examples from the set of downloaded instances ($E \leftarrow E - E_m$).
5. Iterate σ times over steps 1 to 4 or repeat until $E_m = \emptyset$. In this case σ is a user specified threshold.
6. Construct the final classifier using the enriched training set.

3 Experimental Evaluation

3.1 Experimental Setup

Corpus. It is a set of Spanish newspaper articles about natural disasters. It consists of 210 documents grouped in four different categories: forest fires (C1), hurricanes (C2), inundations (C3), and earthquakes (C4). For experimental evaluation we organized the corpus as follows: four different training sets (formed by 1, 2, 5 and 10 examples per class respectively) and a fixed test set of 200 examples (50 per class).

Searching the Web. We used Google as search engine. We downloaded 1,000 additional examples (snippets for these experiments) per class.

Learning methods. We selected two state-of-the-art methods for text classification, namely, support vector machines (SVM) and naïve Bayes (NB) [4].

Evaluation measure. The effectiveness of the method is measured by the classification accuracy, which indicates the percentage of documents that have been correctly classified from the entire document set.

Baseline. Baseline results correspond to the application of the selected classifiers on the test data. Table 2 shows these results for the four different training conditions. They evidence that traditional classification approaches achieve poor performance levels when dealing with *very* few training examples.

3.2 Experimental Results

This section presents some results related to the main processes of the proposed method, namely, the corpora acquisition from the Web and the semi-supervised learning approach.

The central task for corpora acquisition is the automatic construction of a set of queries that express the relevant content of each class. Table 1 shows some numbers on this task. It is noticeable that, because the selection of relevant words relies on a criterion based on their frequency of occurrence and their information gain, there is not the same number of queries per class even thought there were used the same number of training examples. In addition, it is also visible that an increment on the number on examples not necessarily represents a growth on the number of built queries.

Table 1. Some numbers about query construction

Number of training examples	Relevant words per class				Queries per class			
	C1	C2	C3	C4	C1	C2	C3	C4
1	5	5	7	3	10	10	35	1
2	4	5	6	2	4	10	20	1
5	5	5	6	5	10	10	20	10
10	4	5	5	5	4	10	5	10

Nevertheless, it is important to clarify that using more examples allows to construct more general and consequently more relevant queries. For instance, using only two examples about hurricanes we constructed queries such as <*Baja* + *California* + *hurricane*>, whereas using ten examples we could obtain queries such as <*hurricane* + *kilometers* + *storm*>.

Table 2. Experiment result using $m = 1$ and $m = |T|$

				Our method					
Training	**Baseline**		**m-value**	1^{st} **iteration**		2^{nd} **iteration**		3^{rd} **iteration**	
examples	SVM	NB		SVM	NB	SVM	NB	SVM	NB
1	50.0	51.7	$m = 1$	49.1	**78.3**	51.0	77.3	55.3	76.0
2	58.3	56.7		62.3	70.0	68.1	**86.0**	67.0	**86.1**
5	77.1	80.4		76.4	82.2	80.1	85.1	87.0	**92.1**
10	80.4	77.1		82.1	83.1	85.2	87.2	90.1	**91.3**
1	50.01	51.72	$m = \|T\|$	49.0	**78.2**	51.5	77.5	55.2	76.5
2	58.33	56.71		68.2	86.5	74.0	**87.6**	74.5	86.5
5	77.14	80.41		93.5	**97.0**	92.5	96.5	96.0	95.6
10	80.42	77.14		96.5	97.2	96.1	**97.5**	95.1	96.5

Using these queries we collected from the Web a set of 1,000 snippets per class, obtaining a total of 4,000 additional unlabeled examples. Then, we added some of these examples to the original training set. Mainly, we performed three different experiments by varying the parameter m of the algorithm of Section 2.2.

1. At each iteration we added to the training set one additional example per class (i.e., we set $m = 1$).
2. At each iteration we added to training set as many unlabeled examples as the number of instances in the original set (i.e., we set $m = |T|$).

3. In one single step we added to the training set all unlabeled examples satisfying the condition (a) of the algorithm.

Table 2 shows the results of the first two experiments. They indicate that our method outperformed all base configurations especially when using the naïve Bayes classifier. In particular, setting $m = |T|$ lead to accuracy improvements on the range of 30%. On the other hand, the results of the third experiment do not favor the proposed method. They showed a fall in accuracy around 5 to 25%. In some way these results confirms our intuition that in scenarios having very few training instances it is better to include a small group of unlabeled examples that considerably augments the dissimilarities among classes than including a lot of doubtable-quality information.

4 Conclusions and Future Work

In this paper we proposed a method for semi-supervised text classification that is specially suited to work with very few training examples. This method differs from previous approaches in that: (*i*) it automatically collects from the Web the set of unlabeled examples and, (*ii*) it only incorporates into the training phase a small group of unlabeled examples.

The experimental results on a set of newspaper articles about natural disasters demonstrate the viability of the method. In some way, they confirm our hypothesis that when dealing with very few training instances it is better to add a selected set of unlabeled examples (those that considerably augments the dissimilarities among classes) than incorporate a lot of doubtable-quality information. In particular, our method obtained the best results when we added to the training set as many unlabeled examples as the number of original labeled instances. It was also noticeable that our method achieved the best results only after two or three iterations.

References

1. Joachims T., Transductive inference for text classification using support vector machines, Sixteenth International Conference on Machine Learning, 1999.
2. Kilgarriff A., and Greffenstette G., Introduction to the Special Issue on Web as Corpus, Computational Linguistics, 29(3), 2003.
3. Nigam K., Mccallum A. K., Thrun S., and Mitchell T., Text classification from labeled and unlabeled documents using EM, Machine Learning, 39(2/3), 2000.
4. Sebastiani F., Machine learning in automated text categorization, ACM Computing Surveys, 34(1), 2002.
5. Seeger M., Learning with labeled and unlabeled data. Technical report, Institute for Adaptive and Neural Computation, University of Edinburgh, UK, 2001.
6. Solorio T., Using unlabeled data to improve classifier accuracy, Master Degree Thesis, Computer Science Department, INAOE, Mexico, 2002.
7. Zelikovitz S., and Hirsh H., Integrating background knowledge into nearest-Neighbor text classification, Advances in Case-Based Reasoning (ECCBR), 2002.
8. Zelikovitz S., and Kogan M., Using Web Searches on Important Words to Create Background Sets for LSI Classification, 19th FLAIRS conference, Florida, USA, 2006.