Proceedings of the Twenty-Ninth International
Florida Artificial Intelligence Research Society Conference

Semisupervised Text Classification Using Unsupervised Topic Information

Rubén Dorado and **Sylvie Ratté**
Laboratoire d'ingénierie Cognitive et Sémantique
École de Technologie Supérieure, Université du Québec
1100 Rue Notre-Dame Ouest
Montréal, QC H3C 1K3, Canada
ruben.dorado-sanchez.1@ens.etsmtl.ca, Sylvie.Ratte@etsmtl.ca

Abstract

Labeling corpora is a time consuming and recurring problem while developing practical NLP applications. In this paper, we present a semi-supervised method to build a text classifier using unsupervised topic information. The objective is to use the least amount of labeled data to accelerate the creation of corpus for classification in specific domains. We show that it is possible to obtain a performance similar to state-of-the-art methods, despite the limited quantity of data.

Introduction

The problem of automatically classifying text documents is of great practical importance due to the large quantity of digital texts created every day. Commonly, a system is trained on a large quantity of examples, so that it can classify new unseen documents. Methods to develop such systems have been largely researched in recent years; they can be divided roughly into two categories: supervised and unsupervised.

Supervised methods use labeled examples as input to learn; they are used in practice with excellent performance, in general. However, one problem that frequently arises with such systems is the lack of training data for a specific domain or context. This is a major inconvenient since the creation of these tagged data requires the involvement of human annotators to label documents, a time consuming task.

Unsupervised systems try to make use of the dependencies and similarities in the unlabeled training resources to build coherent models. However, they still have several limitations that circumscribe their use in practical systems.

In this paper, we present experimental results of a method capable of dealing with the problem of resource scarcity. Our goal is to increase the speed of resource development by using as less amount of labeled training data as possible. The main objective of our research is to develop a set of similar methods for practical and specific text classification applications. Our contribution in this paper is twofold. First, we present a system that classify documents using a small amount of training examples, and whose performance is as good as the state-of-the-art methods. Second, the experiments show that the new data acquired using a small portion of labeled training examples can be used to test other algorithms.

The rest of this paper is organized as follows. Section 2 describes the background and previous works related to this research. We discuss the proposed models and describe the general framework used in this study, in section 3. The experiments performed to evaluate the system are presented in section 4 along with the discussion of the results obtained. Finally, section 5 points out the conclusions of the study and examines possible ways to extend this work.

Background and related Work

There is a vast amount of work in text categorization (also known as automatic document classification) since it is one of the fields that has gained more popularity in recent years. One of the reasons is probably because of the high quantity of digital text available, and also because it has many practical applications. A huge range of machine learning algorithms and techniques have been used to solve this problem, for example, neural networks (Mello, Senger, and Yang 2005), self-organizing maps (ChandraShekar and Shoba 2009), genetic algorithms (Svigen 1998), Bayesian models (Iwayama and Tokunaga 1995) and support vector machines (Joachims 1998), among many others. Unsupervised methods have also been used to find clusters of related documents, which is a different but related task. There is also a large quantity of work in this field. For a more detailed survey of methods, the reader is referred to (Aggarwal and Zhai 2012).

Within the last decade, the Latent Dirichlet Allocation model (LDA) has received great attention (Blei, Ng, and Jordan 2003). The LDA framework models a set of documents as a generative process, according to which each document is related probabilistically with a set of topics, and words are generated according to them by a mixture of distributions.

The LDA model has been successfully applied to obtain clusters of words extracted from a set of documents that are related to their topics. In this context, each recognized cluster corresponds to a set of keywords that better defines each group of documents. The original LDA model has been extended in many ways. These extensions were used to extract key phrases or sentences from documents (Pasquier 2010), to improve named entity recognition task (Polifroni and Mairesse 2011), or even to chronologically ordered doc-

uments by incorporating in the model temporal information (Bolelli, Ertekin, and Giles 2009).

The LDA model has also been researched extensively in recent years to improve it by including additional information or by combining it with other models. For example, (Gruber, Rosen-Zvi, and Weiss 2007) introduces Markov models to incorporate the assumption that the words depend on the previous topic. (Razavi and Inkpen 2014) proposes the use of LDA to obtain a hierarchical clustering of documents using the results obtained from several LDA classifications varying the number of groups.

There are also several important works in the area of resource scarcity. Most of those propose a model that can learn with a small amount of data by making several assumptions similar to supervised and semi-supervised models. Naive Bayes model has been trained with EM algorithm (Nigam, McCallum, and Mitchell 2006), and active learning has been used to train models faster guided by human assistance (Tsuruoka, Tsujii, and Ananiadou 2008). The difference with this work is the use of an LDA model together with a semi-supervised approach to augment automatically the quantity of labeled data.

This work presents a semi-supervised classification system that uses a small number of labeled data to increase its learning examples from unlabeled data. The objective is to verify if the topic information can be used to obtain the categories and if so, obtain preliminary information about the initial amount of training data necessary to categorize a large quantity of documents with acceptable performance.

Models and framework description

The framework presented in this paper is based on two hypotheses. First, it is possible to categorize text documents using the information acquired from an unsupervised model, in this case, LDA. Second, it is possible to obtain a system whose performance compares to state-of-the-art methods, while using the smallest number of examples as possible. The methodology goes as follows. First, we acquire a set of keywords using LDA and train a Bayesian classifier using these keywords as features to augment the data. Second, we study the size of the labeled dataset that is needed to train different classifiers using either the labeled data or the labeled data along with the augmented data.

Latent Dirichlet Allocation

Latent Dirichlet Allocation (LDA) is a generative probabilistic model composed of a set of probabilistic mixtures that represent distributions over words. The appearance of a given word is conditioned to a set $\mathcal{K}$ of k topics, where each of these topics is defined by a set of keywords. The LDA model assumes that a probabilistic process generates a collection of M documents by sampling words from a dictionary of size V. This process goes as follows: for each of the M documents, the process starts by selecting a number of words $N \sim Poisson(\xi)$, and a k-dimensional multinomial variable $\theta \sim Dir(\alpha)$, which represents the influence of each topic in the document. Then, each word is randomly chosen by first selecting a topic $z_n \sim Multinomial(\theta)$, and posteriorly, each word w_n is sampled from $p(w_n|z_n, \beta)$, where β is a $k \times V$ multinomial distribution of words conditioned on the topics.

All parameters can be obtained from a given corpus using different inference methods, such as variational Bayes approximation of the posterior distribution or sampling methods. Once all the parameters have been learnt, numerous information can be obtained. In the present work, we are interested in using the words forming each topic to increase the number of features acquired with discriminative or supervised models.

Naive Bayes model for data augmentation

The application of LDA results in a set of keywords according to the topic distribution on the full data. After the first step, we use a small quantity of labeled examples to increase the number of training data. This data expansion is made by a Naive Bayes classifier, using the set of keywords as features. The main reason for choosing naive Bayes is the probabilistic nature of the model.

Specifically, given a set of training examples and a set of topics $\mathcal{K}$, a naive Bayes model is trained on the available training data, and used to augment this set with an unlabelel set of documents by classifying them. In the model, a document D is represented by a vector of word counts $\mathbf{w} = (w_1, w_2, .., w_n)$, where each of the counts w_i represents a keyword in the vocabulary of $\mathcal{K}$. Having this representation, the probability of a category given a document is:

$$p(c|D) = \frac{p(D|c)p(c)}{p(D)} = \frac{p(\mathbf{w}|c)p(c)}{p(\mathbf{w})},$$

where, D is a given arbitrary document, and c is a category in the set of possible categories.

Under the conditional independence assumption and naive Bayes, the category of a given document is calculated by:

$$p(c|D) = \frac{p(c)}{p(\mathbf{w})} \prod_{i=1}^{n} p(w_i|c)$$

The category of a document, chosen by selecting the highest probability is:

$$\hat{c} = \underset{c \in C}{\operatorname{argmax}}\, p(c) \prod_{w \in V} p(w|c).$$

Finally, the parameters of the model are calculated through the *MAP* framework using a set of training documents composed by pairs (D_i, c_i), where D_i is a document and c_i is its associated category. First, the probability of a given document $p(c)$ is calculated as:

$$\hat{p}(c) = \frac{count(c)}{M}.$$

Second, the probabilities of a word given a document $p(w|c)$ are calculated as

$$\hat{p}(w|c) = \frac{count(w, c)}{\sum_{w \in V} count(w, c)}.$$

Figure 1 displays the pipeline of the data augmentation process. The training data consists of two parts: a small portion (A) forms the labeled training data, and the rest consists of unlabeled examples (B). In step one (1), LDA is applied to obtain a set of keywords using the whole training data (labeled and unlabeled). In step two (2), the set of keywords obtained previously becomes the set of features for the NB classifier. In this step, we train the classifier using the labeled part of the training data (A). In the third and final step, we use the classifier to predict the categories of the unlabeled part of the training data (C). It can be seen in the figure that the augmented data is composed of the labeled examples (A) and the predicted results obtained by the system (C).

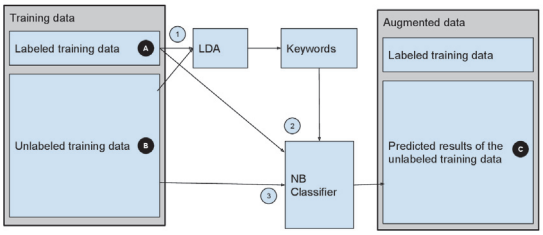

Figure 1: Pipeline of the performed experimentation

Experiments and results

We performed various experiments on the *20Newsgroups* corpus. This dataset consists of a collection of documents taken from newsgroups about 20 different topics. We used the *scikit-learn* library, which contains implementations of Naive Bayes and SVM classifiers. For the LDA topic acquisition, we used the Blei's implementation, which is a software publicly available on the author's webpage. Finally, we implemented the data augmentation model proposed in section 3. The performance is measured using macro-precision, macro-recall, macro-F1 measure and accuracy. We use 10-fold cross validation in all experimentations. We preprocessed the text removing stop words and converted them to lower case. We did not performed stemming or other linguistic methods such as NER or POS.

First, we tested the system with small quantities of training data and compared these results with the ones obtained using a state-of-the-art SVM classifier. For the keywords acquisition with LDA, we set the same number of topics as the number of output classes. We obtained the sets of keywords related by topic and then we used the 20 most important by topic to train the classifier. We compared the results with state-of-the-art methods, SVM and naive Bayes, with data augmentation (A+C) and without it (A). Figure 2 shows the performance of the system and the percentage of original labeled data that is used (portion A). We started with 0.5% (3 examples per category), and gradually increased it to 5% (30 examples per category).

As expected, increasing the training data results in better performance measures. Most interestingly, the number

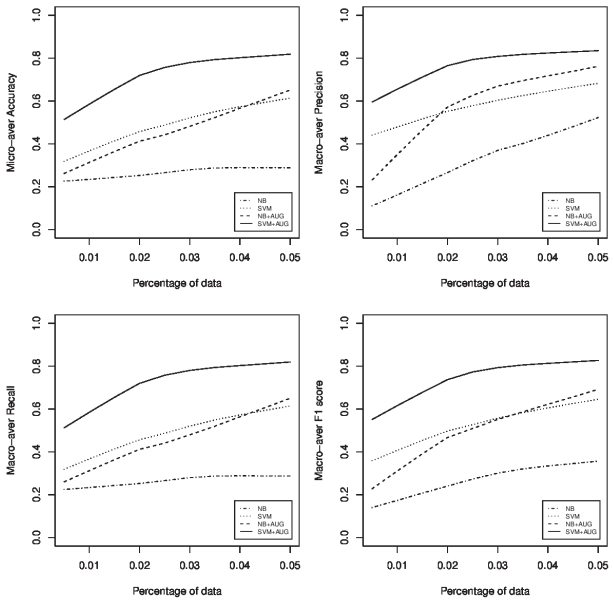

Figure 2: Results by varying the number of training examples and keywords used. NB and SVM curves correspond to the results from training the models with a small portion of training data (see Figure 1, (A)). NB+AUG and SVM+AUG curves correspond to results from training the models with augmented data (see Figure 1 (A + C)).

of training examples needed to achieve an acceptable performance is about 3% (20 examples per category).

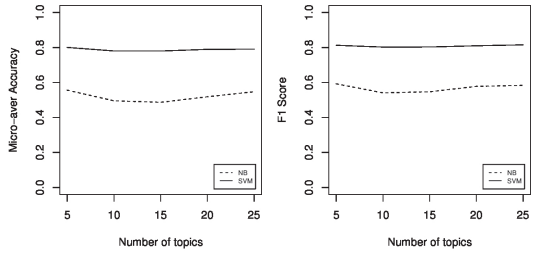

Figure 3: Results varying the number of topics used

Since LDA is an unsupervised system, we varied the number of topics acquired using 3% of the data (20 examples per category), starting with 5 topics and increased them to 25. We selected the first 50 keywords as features for the classification. Figure 3 shows the results from varying the number of topics. It can be seen that it does not have a significant effect, especially in the case of SVM.

Discussion and further work

In this paper, we have presented preliminary results of a general method that build a text classifier with small amounts of labeled training data. We have shown that it is possible to obtain an acceptable performance with a semi-supervised

approach using a relatively low quantity of training data. Specifically, we show that an accuracy of 80% can be achieved with 3% (20 examples per category) of a 600 examples dataset.

This method is the initial step for developing a general iterative process to quickly construct efficient corpora for text categorization. In our future work, we will explore the usage of online algorithms to obtain the augmented data. The idea is to transform the LDA-Naive Bayes approach into an active learning process, when the data elicited from the expert becomes available.

References

Aggarwal, C. C., and Zhai, C. 2012. *Mining Text Data*. chapter A Survey of Text Clustering Algorithms, 77–128.

Blei, D. M.; Ng, A. Y.; and Jordan, M. I. 2003. Latent dirichlet allocation. *Journal of Machine Learning Research* 3:993–1022.

Bolelli, L.; Ertekin, S.; and Giles, C. L. 2009. Topic and trend detection in text collections using latent dirichlet allocation. In *ECIR '09 Proceedings of the 31th European Conference on IR Research on Advances in Information Retrieval*, 776 – 780.

ChandraShekar, B., and Shoba, G. 2009. Classification of documents using kohonen's self-organizing map. *International Journal of Computer Theory and Engineering* 1(5):610–613.

Gruber, A.; Rosen-Zvi, M.; and Weiss, Y. 2007. Hidden topic markov models. In *Artificial Intelligence and Statistics (AISTATS)*.

Iwayama, M., and Tokunaga, T. 1995. Hierarchical bayesian clustering for automatic text classification. In *IJCAI'95 Proceedings of the 14th international joint conference on Artificial intelligence*, volume 2, 1322–1327.

Joachims, T. 1998. Text categorization with support vector machines: Learning with many relevant feature. In *Machine Learning: ECML-98. 10th European Conference on Machine Learning*, 137–142.

Mello, R.; Senger, L.; and Yang, L. 2005. Automatic text classification using an artificial neural network. *High Performance Computational Science and Engineering* 172:215–238.

Nigam, K.; McCallum, A.; and Mitchell, T. 2006. *Semi-Supervised Learning*. MIT Press: Boston. chapter Semi-supervised Text Classification Using EM, 31–51.

Pasquier, C. 2010. Task 5: Single document keyphrase extraction using sentence clustering and latent dirichlet allocation. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, 154–157.

Polifroni, J., and Mairesse, F. 2011. Using latent topic features for named entity extraction in search queries. In *INTERSPEECH 2011, 12th Annual Conference of the International Speech Communication Association*, 27–31.

Razavi, A. H., and Inkpen, D. 2014. Text representation using multi-level latent dirichlet allocation. In *27th Canadian Conference on Artificial Intelligence (Canadian AI 2014)*, 215–226. Montral, Canada: Springer.

Svigen, B. 1998. Using genetic programming for document classification. In *FLAIRS-98. Proceedings of the Eleventh International Florida Artificial Intelligence Research*, 63–67.

Tsuruoka, Y.; Tsujii, J.; and Ananiadou, S. 2008. Accelerating the annotation of sparse named entities by dynamic sentence selection. In *Proceedings of the Workshop on Current Trends in Biomedical Natural Language Processing*, 30–37. Association for Computational Linguistics.