Text Classification from Positive and Unlabeled Examples

François Denis
Équipe BDA, LIF
Marseille, FRANCE
fdenis@cmi.univ-mrs.fr

Rémi Gilleron
Équipe Grappa
Lille, FRANCE
gilleron@univ-lille3.fr

Marc Tommasi
Équipe Grappa
Lille, FRANCE
tommasi@univ-lille3.fr

Abstract

This paper shows that binary text classification is feasible with positive examples and unlabeled examples. This is important because in many text classification problems hand-labeling examples is expensive while examples of one class and unlabeled examples are readily available. We introduce a naive Bayes algorithm for learning from positive and unlabeled documents. Experimental results show that performance of our algorithm is comparable with naive Bayes algorithm for learning from labeled data.

Keywords: text mining, text classification, semi-supervised learning, positive data.

1 Introduction

Recently there has been significant interest in text learning algorithms that combine information from labeled and unlabeled data. For the labeled data, supervised learning algorithms apply, but their performance can be poor for a small labeled data set and they cannot take advantage of the unlabeled data. For the unlabeled data, unsupervised learning algorithms apply, but they do not use the labels. Thus, learning with labeled and unlabeled data – sometimes named as semi-supervised learning – falls between supervised and unsupervised learning. This research area is motivated by the fact that it is often tedious and expensive to hand-label large amount of training data, specially for text learning tasks, while unlabeled data are freely available.

Several learning algorithms have been defined for text learning tasks in the semi-supervised setting. We only consider supervised learning algorithms with the help of unlabeled data. Such approaches include using Expectation Maximization to estimate maximum a posteriori parameters [11], using transductive inference for support vector machines [5], using the unlabeled data to define a metric or a kernel function [4], using a partition of the set of features into two disjoint sets [1, 10].

We address the issue of learning from positive and unlabeled data where positive data are examples of one fixed target class. We have given in previous papers theoretical and experimental results [2, 7]: we have proven that every class learnable in the Statistical Query model [6] is learnable from positive statistical queries (estimates of probabilities over positive instances) and instance statistical queries (estimates of probabilities over the instance space) when a lower bound on the positive class probability is given; we have also designed a decision tree induction algorithm from positive and unlabeled examples.

In the present paper, we design text learning algorithms from positive and unlabeled documents. Let us consider two examples of applications. A first example is learning to classify web pages as "interesting" for a specific user. His bookmarks define a set of positive examples because they correspond to interesting web pages for this user. Unlabeled examples are easily available on the World Wide Web. A second example is mail filtering. For a given mailing list and a specific user, positive examples are mails from the mailing list which have been saved by the user in his mailboxes. Again, unlabeled examples can easily be obtained by storing all mails from the mailing list, say during one week. It is interesting to note that no hand-labeling is needed in our framework.

In Section 2, we design a naive Bayes algorithm from positive and unlabeled examples. The key step is in estimating word probabilities for the negative class because negative examples are not available. This is possible according to the following assumption: an estimate of the positive class probability (the ratio of positive documents in the set of all documents) is given as input to the learner. In practical situations, the positive class probability can be empirically estimated or provided by domain knowledge.

In Section 3, we give experimental results on the WebKB Course data set [1]. The results show that error rates of naive Bayes classifiers obtained from p positive examples completed with enough unlabeled examples are lower than error rates of naive Bayes classifiers obtained from p labeled documents. The experiments suggest that positive examples may have a high value in context of semi-supervised learning.

2 Naive Bayes from positive and unlabeled examples

2.1 Naive Bayes

Naive Bayes classifiers are commonly-used in text classification [8]. The basic idea is to use the joint probabilities of words and classes to estimate the probabilities of classes given a document. The naive part is the assumption that the presence of each word in a document is conditionnally independent of all other words in the document given its class. This conditional independence assumption is clearly violated in real-world problems. Nevertheless, Naive Bayes classifiers are among the most effective text classification systems [3, 9].

We only consider binary classification problems with a set of classes $\{0, 1\}$ where 1 corresponds to the *positive class*. We consider *bag-of-words* representations for documents. Naive Bayes is given in Table 1. It assumes an underlying generative model. In this model, first a class is selected according to class prior probabilities. A document length is chosen independently of the class. Then, the generator creates each word in a document by drawing from a multinomial distribution over words specific to the class.

Given a vocabulary V and a set D of labeled documents, let us denote by PD (respectively ND) the set of positive documents (respectively negative documents) in the set D. The class probabilities $P(c)$ are estimated by:

$$\hat{P}(0) = \frac{Card(ND)}{Card(D)};\ \hat{P}(1) = \frac{Card(PD)}{Card(D)} \quad (1)$$

where $Card(X)$ is the cardinality of set X.

A key step in implementing naive Bayes is estimating the word probabilities $Pr(w_i|c)$. The word probabilities $Pr(w_i|c)$ are estimated by counting the frequency that word w_i occurs in all word occurrences for documents in class c:

$$\hat{Pr}(w_i|0) = \frac{N(w_i, ND)}{N(ND)}$$
$$\hat{Pr}(w_i|1) = \frac{N(w_i, PD)}{N(PD)}$$

where $N(w_i, X)$ is the total number of times word w_i occurs in the documents in the set X and $N(X)$ the total number of word occurrences in set X. A document cannot be classified as a member of class c as soon as it contains a word w which does not occur in any labeled document of class c. To make the probability estimates more robust with respect to infrequently encountered words, smoothing methods are used or equivalently a prior distribution over multinomials is assumed. We consider the classical *Laplace smoothing*, and the class probability estimates are:

$$\hat{Pr}(w_i|0) = \frac{1 + N(w_i, ND)}{Card(V) + N(ND)} \quad (2)$$

$$\hat{Pr}(w_i|1) = \frac{1 + N(w_i, PD)}{Card(V) + N(PD)} \quad (3)$$

We now give formulas which are needed in the next section. We can write the following equation:

$$Pr(w_i) = Pr(w_i|0)Pr(0) + Pr(w_i|1)Pr(1) \quad (4)$$

where $Pr(w_i)$ is the probability that the generator creates w_i and $Pr(1)$ is the probability that the generator creates a word in a positive document. Let us suppose that we are given a set $D = PD \cup ND$ of labeled documents. An estimate of $Pr(w_i)$ is $\frac{N(w_i,D)}{N(D)}$. An estimate of $Pr(1)$ is $\frac{N(PD)}{N(D)}$. But, under the assumption that the lengths of documents are independent of the class, another estimate of $Pr(1)$ is $\hat{P}(1) = \frac{Card(PD)}{Card(D)}$.

Table 1: Naive Bayes from labeled documents (NB)

Given a set D of labeled documents, the naive Bayes classifier classifies a document d consisting of n words $(w_1, \ldots, w_n)$ – with possibly multiple occurrences of a word w – as a member of the class $$\mathsf{NB}(d) = \underset{c\in\{0,1\}}{\text{argmax}}\, \hat{P}(c) \prod_{i=1}^{i=n} \hat{Pr}(w_i|c) \quad (5)$$ where the class probability estimates are calculated according to Equations 1 and the word probability estimates are calculated according to Equations 2 and 3.

2.2 Naive Bayes from positive and unlabeled examples

In the present section, training data consist of a set PD of positive documents together with a set UD of unlabeled documents. The key point is to compute sufficiently accurate probability estimates in Equation 5 from positive and unlabeled data only. We assume that an estimate $\hat{P}(1)$ of the positive class probability $P(1)$ is given to the learner. Then, an estimate of the negative class probability is setting $\hat{P}(0)$ to $1 - \hat{P}(1)$. The key step is estimating the word probabilities.

Estimating Word Probabilities

Let us consider that we are given an estimate $\hat{P}(1)$ of the positive class probability $P(1)$, a set PD of positive documents together with a set UD of unlabeled documents.

The positive word probability estimates are calculated using Equation 3 with the input set PD of positive documents.

For the negative word probabilities, from Equation 4, we derive the following equation:

$$Pr(w_i|0) = \frac{Pr(w_i) - Pr(w_i|1) \times Pr(1)}{1 - Pr(1)} \quad (6)$$

We use this equation in order to derive the negative word probability estimates. In Equation 6, positive class probabilities are estimated with Equation 3. We now give formulas for the estimates of $Pr(w_i)$ and $Pr(1)$.

Estimate of $Pr(w_i)$. Assuming that the set of unlabeled documents is generated according to the underlying generative model, probability $Pr(w_i)$ is estimated on the set of unlabeled documents by:

$$\hat{Pr}(w_i) = \frac{N(w_i, UD)}{N(UD)} \quad (7)$$

Estimate of $Pr(1)$. We will consider two different estimates for $Pr(1)$. First, under the assumption that the lengths of documents are independent of the class, positive and negative documents have the same average length and $\hat{Pr}(1)$ could be set to $\hat{P}(1)$.

Second, we have seen that, given a set $D = PD \cup ND$ of labeled documents, an estimate of $Pr(1)$ is $\frac{N(PD)}{N(D)}$. We can deduce the following equation:

$$\hat{Pr}(1) = \frac{N(PD)}{Card(PD)} \times \frac{Card(PD)}{Card(D)} \times \frac{Card(D)}{N(D)}$$

In the case where an estimate of $P(1)$ and a set PD of positive documents together with a set UD of unlabeled documents are given to the learner, the first term $\frac{N(PD)}{Card(PD)}$ in the previous equation can be calculated with the input set PD; the second term corresponds to $\hat{P}(1)$ which is given as input to the learner; and, assuming that unlabeled documents are generated according to the underlying probabilistic model, the third term can be estimated over the set UD of unlabeled examples. This leads to the following estimate for $Pr(1)$:

$$\hat{Pr}(1) = \frac{N(PD)}{Card(PD)} \times \hat{P}(1) \times \frac{Card(UD)}{N(UD)}$$

When the sets PD and UD are quite small, it may be possible that our estimate for $Pr(1)$ is greater than 1. Thus, we bound our estimate:

$$\hat{Pr}(1) = \min\left\{ \frac{N(PD)}{Card(PD)} \times \hat{P}(1) \times \frac{Card(UD)}{N(UD)} \; ; \; \frac{1+\hat{P}(1)}{2} \right\} \quad (8)$$

Equations 3, 7 and 8 provide estimates for word probabilities appearing in Equation 6.

Smoothing Word Probabilities

Using Equation 7, estimates for negative word probabilities $\hat{Pr}(w_i|0)$ given by Equation 6 can be rewritten:

$$\frac{N(w_i, UD) - \hat{Pr}(w_i|1) \times \hat{Pr}(1) \times N(UD)}{(1 - \hat{Pr}(1)) \times N(UD)}$$

The estimates $\hat{Pr}(w_i|0)$ can be negative. Thus, we set the negative values to 0 and normalize our estimates such that they sum to 1. Let Z be the normalizing factor defined by

$$Z = \sum_{w_i \in V | Pr(w_i|0) > 0} Pr(w_i|0)$$

Using the Laplace smoothing method, estimates for negative word probabilities $\hat{Pr}(w_i|0)$ are given by:

$$\frac{1 + \max\{R(w_i); 0\} \times \frac{1}{Z}}{Card(V) + (1 - \hat{Pr}(1)) \times N(UD)} \quad (9)$$

where $R(w_i)$ is set to $N(w_i, UD) - \hat{Pr}(w_i|1) \times \hat{Pr}(1) \times N(UD)$, $\hat{Pr}(w_i|1)$ is calculated according to Equation 3, and $\hat{Pr}(1)$ is either set to $\hat{P}(1)$ or is calculated according to Equation 8.

Table 2: Naive Bayes from positive and unlabeled examples (PNB)

Given an estimate $\hat{P}(1)$ of the positive class probability $P(1)$, a set PD of positive documents together with a set UD of unlabeled documents, the positive naive Bayes classifier classifies a document d consisting of n words $(w_1, \ldots, w_n)$ as a member of the class $$\mathsf{PNB}(d) = \underset{c \in \{0,1\}}{\operatorname{argmax}} \hat{P}(c) \prod_{i=1}^{i=n} \hat{Pr}(w_i|c) \quad (10)$$ where the class probability estimate $\hat{P}(0)$ is set to $1 - \hat{P}(1)$, the word probability estimates are calculated according to Equation 3 for the positive class and according to Equation 9 for the negative class.

3 Experimental results

We consider the WebKB Course dataset[1], a collection of 1051 web pages collected from computer science departments at four universities. The binary classification problem is to identify web pages that are course home pages. The class course is designed as the positive class in our setting. In the WebKB dataset, 22% of the web pages are positive. We consider the *full-text view* which consists of the words that occur on the web page. The vocabulary is the set of words in the input data sets; no stoplist is used and no stemming is

[1]available at `http://www-2.cs.cmu.edu/afs/cs/project/theo-4/text-learning/www/datasets.html`

Table 3: results for PNB on the WebKB Course dataset when varying the number of unlabeled documents

p is set to 20		p is set to 50	
u	error	u	error
20	27.155	50	15.265
30	16.597	100	8.010
40	12.000	120	7.485
50	10.353	130	7.298
60	8.611	140	7.265
70	8.698	150	7.611
80	8.922	160	7.576
100	9.586	170	7.668
150	13.365	180	7.693
200	16.048	200	8.239

performed. We give experimental results for our algorithm PNB when varying the number of unlabeled documents and when using different estimates for $Pr(1)$. Then, we conduct experiments to compare PNB and NB while varying the number of labeled documents. In a last set of experiments, we compare error rates when giving as input different values for the positive class probability.

3.1 Varying the number of unlabeled documents

We use the algorithm PNB where $Pr(1)$ is estimated using Equation 8. We set the input $\hat{P}(1)$ to 0.22. We consider two values for the number p of positive documents : 20 and 50. We let vary the number u of unlabeled documents. For each value of p and u, 200 experiments are conducted. Error rates are estimated on an hold-out test set and error rates are averaged over these 200 experiments.

Experimental results (see Table 3) show that the error decreases and reaches a minimal value. We note that when the number of unlabeled documents becomes too large, performance of PNB may be poor. For a given number of positive documents, the optimal value for the number of unlabeled documents is not known. In the following, we assume that estimates will be done on a set of unlabeled documents containing approximately $Card(PD)$ positive documents. Consequently, we set the number of unlabeled documents to $Card(PD)/\hat{P}(1)$ where PD is the set of positive documents and $\hat{P}(1)$ the estimate of the positive class probability. Results given in Table 3 show that this choice is not optimal from an experimental point of view on the WebKB Course dataset.

3.2 Estimating $Pr(1)$

We compare three variants of PNB depending on how the estimate of $Pr(1)$ is calculated. PNB takes as input $\hat{P}(1) = 0.22$ together with randomly drawn sets PD and UD such that $Card(UD) = Card(PD)/\hat{P}(1)$. In the first variant, $Pr(1)$ is estimated using Equation 8. In the second one, $\hat{Pr}(1)$ is set to $\hat{P}(1)$, i.e. it is supposed that the knowledge of the average length of positive documents is negligible in the classification decision. In the third one, $Pr(1)$ is estimated on the whole WebKB Course dataset of 1051 web pages and we set $\hat{Pr}(1)$ to 0.28[2].

Experimental results (see Figure 1) show that a better estimate of $Pr(1)$ slightly increases the accuracy of PNB classifiers. PNB classifiers where $\hat{Pr}(1)$ is set to $\hat{P}(1)$ perform better than PNB classifiers where $\hat{Pr}(1)$ is calculated using Equation 8 when the train set is small. Indeed the variance of the estimation of $Pr(1)$ is high when only a small number of documents are available. But, when there are enough documents (20 positive documents), the accuracy of PNB classifiers where $\hat{Pr}(1)$ is calculated using Equation 8 is close to the accuracy of PNB classifiers where $Pr(1)$ is estimated on the whole WebKB Dataset.

3.3 A comparison between NB and PNB

For a given number p, we compare: NB classifiers obtained from p labeled documents; PNB classifiers obtained with input $\hat{P}(1) = 0.22$, p

[2]Note that under the assumption that the length of documents is independent of the class, $Card(PD)/Card(D)$ and $N(PD)/N(D)$ are unbiased estimates of $Pr(1)$. On the WebKB Course dataset, we find respectively 0.22 and 0.28 which suggests that this assumption could be not correct.

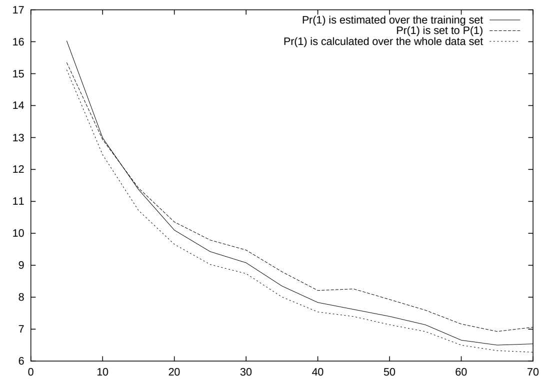

Figure 1: Comparison of PNB with three different estimates of $Pr(1)$. Error rates are averaged over 200 experiments

positive documents and $N \simeq p \times 1/0.22$ unlabeled documents; NB classifiers obtained from N labeled documents. We use algorithm PNB where $\hat{Pr}(1)$ is estimated using Equation 8. For each value p and each algorithm, 200 experiments are conducted. Error rates are estimated on an hold-out test set and are averaged over the 200 experiments. Error rates are given together with standard deviation.

Experimental results (see Table 4 and Figure 2) show that PNB classifiers outperform NB classifiers obtained from p labeled documents. These experimental results are quite promising showing that p positive examples completed with unlabeled examples have a higher value than p labeled examples, at least for small values of p.

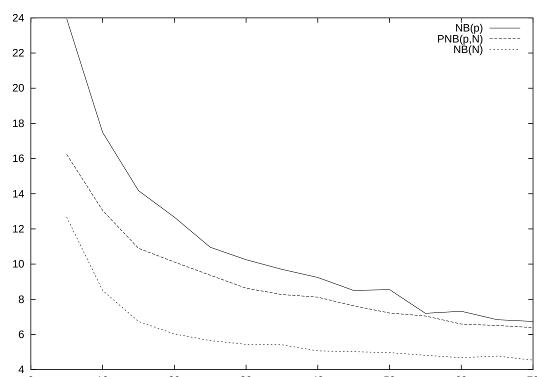

Figure 2: Comparison of NB_p, $\mathsf{PNB}_{p,N}$ and NB_N.

Table 4: A comparison between NB and PNB.

p	N	NB_p	$\mathsf{PNB}_{p,N}$	NB_N
5	22	23.95(12.4)	16.24(12.67)	12.67(4.72)
10	45	17.49(7.00)	13.05(4.68)	8.50(3.56)
15	68	14.18(5.55)	10.90(4.13)	6.74(2.40)
20	91	12.67(4.72)	10.12(3.70)	6.03(1.95)
25	114	10.96(4.26)	9.37(3.39)	5.65(1.79)
30	137	10.25(4.51)	8.63(2.95)	5.44(1.64)
35	159	9.70(4.26)	8.27(2.74)	5.41(1.58)
40	182	9.24(4.22)	8.12(2.61)	5.07(1.45)
45	205	8.50(3.56)	7.63(2.52)	5.02(1.49)
50	228	8.55(3.73)	7.22(2.39)	4.97(1.38)
55	251	7.20(2.97)	7.05(2.12)	4.81(1.42)
60	274	7.32(3.18)	6.59(1.83)	4.68(1.35)
65	297	6.84(2.45)	6.51(1.94)	4.77(1.37)
70	319	6.74(2.40)	6.39(1.95)	4.54(1.29)

3.4 Giving an estimate of the positive class probability

We use the algorithm PNB where $Pr(1)$ is estimated using Equation 8. We consider two values for the number p of positive documents : 20 and 50. An estimate of the positive class probability on the whole WebKB Dataset is $\hat{P}(1) = 0.22$. We let vary the estimate for the positive class probability. PNB takes as input $\hat{P}(1)$ together with randomly drawn sets PD and UD such that $Card(UD) = Card(PD)/\hat{P}(1)$. $\hat{P}(1)$ takes value from 0.12 to 0.38 by step 0.02. For each value of $\hat{P}(1)$, 200 experiments are conducted. Error rates are estimated on an hold-out test set and error rates are averaged over these 200 experiments.

Experimental results are given in Table 5. They show that sufficiently accurate classifiers are obtained with rough estimates of $P(1)$. For instance, an estimate of $P(1)$ could be chosen between 0.2 and 0.3.

4 Conclusion

We have shown that text classification from positive and unlabeled data is feasible and that positive documents and labeled documents may have a comparable value as soon as the former are completed with enough un-

Table 5: PNB classifiers with different input values for $\hat{P}(1)$.

	p is set to 20	p is set to 50
$\hat{P}(1)$	error	error
0.12	16.74	13.47
0.14	15.12	11.37
0.16	13.77	9.99
0.18	11.93	8.88
0.20	10.76	8.00
0.22	10.60	7.22
0.24	9.66	7.18
0.26	9.25	6.71
0.28	9.96	6.78
0.30	10.21	7.23
0.32	11.29	8.18
0.34	12.41	9.22
0.36	12.69	9.70
0.38	13.74	11.26

labeled documents. As in the semi-supervised framework, unlabeled data are supposed to be freely available, the experimental results are promising but we need to apply our algorithms to other data sets. Following [7], it would be interesting to design algorithms from positive and unlabeled documents when the positive class probability is not given as input to the learner. Also, we intend to adapt the co-training setting from Blum and Mitchell [1] to the framework of learning from positive and unlabeled documents.

Acknowledgements

This research was partially supported by: "CPER 2000-2006, Contrat de Plan état - région Nord/Pas-de-Calais: axe TACT, projet TIC"; fonds européens FEDER "TIC - Fouille Intelligente de données - Traitement Intelligent des Connaissances" OBJ 2-phasing out - 2001/3 - 4.1 - n 3; "projet DATADIAB - ACI télémédecine et technologies pour la santé".

References

[1] A. Blum and T. Mitchell. Combining labeled and unlabeled data with co-training. In *Proc. 11th Annu. Conf. on Comput. Learning Theory*, pages 92 – 100, 1998.

[2] F. DeComité, F. Denis, R. Gilleron, and F. Letouzey. Positive and unlabeled examples help learning. In *Proc. 10th International Conference on Algorithmic Learning Theory*, pages 219 – 230, 1999.

[3] P. Domingos and M. Pazzani. On the optimality of the simple bayesian classifier using zero-one loss. *Machine Learning*, 29:103 – 130, 1997.

[4] T. Hofmann. Text categorization with labeled and unlabeled data: A generative model approach. In *Working Notes for NIPS 99 Workshop on Using Unlabeled Data for Supervised Learning*, 1999.

[5] Thorsten Joachims. Transductive inference for text classification using support vector machines. In *Proc. 16th International Conference on Machine Learning*, pages 200 – 209, 1999.

[6] M. Kearns. Efficient noise-tolerant learning from statistical queries. In *Proc. 25th ACM Symposium on the Theory of Computing*, pages 392 – 401, 1993.

[7] F. Letouzey, F. Denis, and R. Gilleron. Learning from positive and unlabeled examples. In *Proc. 11th International Conference on Algorithmic Learning Theory*, pages 71 – 85, 2000.

[8] D. D. Lewis. Naive (bayes) at forty: the independence assumption in information retrieval. In *Proc. 10th European Conference on Machine Learning*, pages 4 – 15, 1998.

[9] D. D. Lewis and M. Ringuette. A comparison of two learning algorithms for text categorization. In *Proc. 3rd Annual Symposium on Document Analysis and Information Retrieval*, pages 81 – 93, 1994.

[10] Kamal Nigam and Rayid Ghani. Analyzing the applicability and effectiveness of co-training. In *Proc. 9th International Conference on Information and Knowledge Management*, pages 86 – 93, 2000.

[11] Kamal Nigam, Andrew K. McCallum, Sebastian Thrun, and Tom M. Mitchell. Text classification from labeled and unlabeled documents using EM. *Machine Learning*, 39(2/3):103 – 134, 2000.