Reducing the Annotation Burden in Text Classification

Krithara A.[a 1], Goutte C.[b 2], Amini M.-R.[c 3] and Renders J.-M.[a 4]
[a] *Xerox Research Centre Europe, 6 chemin de Maupertuis, F-38240 Meylan, FRANCE*
[b] *National Research Council Canada, Institute for Information Technology, Interactive Language Technologies Group, 101 St-Jean-Bosco Street, Gatineau, QC K1A 0R6, Quebec, CANADA*
[c] *Department of Computer Science, University of Paris VI, 8 rue de Capitaine Scott, 75015 Paris, FRANCE*

In this paper we describe a method which combines semi-supervised and active learning for the classification task. In particular, we propose a semi-supervised PLSA (Probabilistic Latent Semantic Analysis) algorithm [4] combined with a certainty-based active learning method, in order to classify text documents.

Keywords: Semi-Supervised Learning, Active Learning, PLSA, Machine Learning.

1 OVERVIEW

Active and semi-supervised learning have been used in various ways and different contexts to reduce the labeling effort in supervised machine learning. Both techniques aim at solving the same problem, but from different perspectives. We consider the pool-based active learning model, where the idea is essentially to select promising unlabeled examples from a given set in a sequential process in the sense that the corresponding target objects contribute to a more accurate prediction function.

Our approach follows the work of [4] who proposed a semi-supervised variant of the Probabilistic Latent Semantic Analysis (PLSA, [5]) model. This algorithm includes additional fake labels for unlabeled examples and extends the Expectation-Maximisation (EM) algorithm to uncover the relationships between components and labels (including the fake one). PLSA is based on a solid statistical framework which has been proved to be competitive compared to other existing semi-supervised techniques. In particular, it captures polysemy and synonymy [5] in natural language. The novelty of our work is to combine active learning and the semi-supervised PLSA model. The idea is to select the most ambiguous examples and to correctly annotate them, in order to augment the training data and to improve the performance of the semi-supervised learning module.

In supervised machine learning, labeled examples are required to learn a classification rule. However, the annotation of the data is, in many situations, a time-consuming and costly process, whereas big amounts of unlabeled data are often available. As it attempts to learn from both labelled and unlabelled data, semi-supervised learning has met with high interest. It may help reduce the labelling effort by combining labeled and unlabeled data. Various methods of semi-supervised learning have been explored and have been shown to yield promising results. For example, Nigam et al. [11] proposed an algorithm using the naive Bayes classifier and EM in order to learn from labeled and unlabeled data. Another approach, co-training, was proposed by [2]. Assuming that we have two views of the same data, co-training uses the examples with high classification score of the one view as a training set in the second view. Co-EM [10] combines co-training and the EM algorithm. Within the realm of Support Vector Machine, [6] applied a transductive inference approach to improve text categorisation using unlabelled data. A survey of more semi-supervised methods can be found in [16].

Active learning addresses the issue of the annotation burden from a different perspective. It tries to minimize the annotation cost by labeling as few examples as possible and focussing on the most useful examples. Among different types of active learning methods, *uncertainty-based methods* try to find the most ambiguous example to label. Measures of ambiguity include classification probability in naive Bayes [7], distance to a decision boundary in SVMs [13] or the disagreement between different committee members in Query by Committee [3]. Another type of active learning methods [12] tries to *reduce the future error*, i.e. seeks examples that minimize the expected generalization error. Baram et al. [1] proposed a combination of different active learners, choosing at each step the most convenient learner in each particular case.

By combining semi-supervised and active learning, we attempt to benefit from both approaches to addressing the annotation burden problem. The semi-supervised learning component improves the

[1] Anastasia Krithara: Anastasia.Krithara@xrce.xerox.com
[2] Cyril Goutte was with XRCE and is now with the National Research Council Canada, Email: Cyril.Goutte@nrc-cnrc.gc.ca
[3] Massih Reza Amini: Massih-Reza.Amini@lip6.fr
[4] Jean-Michel Render: Jean-Michel.Renders@xrce.xerox.com

classification rule and the measure of its confidence, while the active learning queries for labelling the most relevant and potentially useful examples. Previous work on the combination of semi-supervised and active learning includes [8], where EM is used with unlabeled data integrated into the active learning algorithm. Also [10] combined multi-view learning with active learning in Co-EMT. In [15], semi-supervised learning is combined with active learning and applied to content-based image retrieval. Zhu X. et al [17] have combined semi-supervised and active learning using Gaussian fields and harmonic functions.

We will first describe the semi-supervised variant of PLSA (section 2). As this method yields a good estimate of the confidence of the classification rule, we extend it to handle pool-based active learning (section 3). In experiments carried out on binary classification problems using the 20 newsgroup dataset, we show that we may benefit from both the semi-supervised PLSA and the active selection of examples (section 4). In section 5, we discuss the implications of this work and future directions.

2 SEMI-SUPERVISED PLSA

Let us consider a classification problem where we have both labeled and unlabeled data. Let us assume we have *K* classes and our data consist of *l* labeled examples $L = \left(x_i, y_i\right), i \in \left\{1,\mathrm{K}\ ,l\right\}$ and *u* unlabeled examples $U = x_{l+j},\ j \in \left\{1,\mathrm{K}\ ,u\right\}$. We denote by *N=l+u* the total number of examples in the training set. We also assume that the dataset is a collection of documents $D = \left(d_1,\mathrm{K}\ ,d_{N_d}\right)$, containing words from the vocabulary $W = \left(w_1,\mathrm{K}\ ,w_{N_w}\right)$. Each example *x* is a co-occurrence of a word *w* and a document *d*, which we denote by $x = \left(w,d\right)$. We model our data using a mixture model, under the assumption that *d* and *w* are independent, conditionally to a component of the mixture (i.e. within one class/component, all documents have similar content):

$$P\left(w,d\right) = P\left(d\right)\sum_c P\left(w\,|\,c\right)P\left(c\,|\,d\right) \qquad (1)$$

where *c* is the number of components of the model. PLSA associates a latent context variable with each word occurrence, which explicitly accounts for polysemy. It aims to discover something about the meaning behind the words and the topic of the document.

In the semi-supervised learning method we adopt [4], we place a "fake" label on all unlabeled data. The motivation is to try to solve the problem of unlabeled components, i.e. the components which contain only unlabeled data (which is very likely to happen whenever *l*<<*u*). If we do not use the "fake" label, arbitrary class probabilities will be assigned to these components, which will lead to arbitrary decision during the classification. So, all labeled data will keep their real labels and all the unlabeled data get a new label *z=0*. In other words, $\forall 1 \le i \le l, z_i = y_i$ and $\forall (l+1) \le i \le N, z_i = 0$. Note that the label is in fact uniquely determined by the document, i.e. all examples from the same document *d* have the same label *z* (a possibly *y*). From (1), taking into account the label *z* in the same manner, we have:

$$P(w,d,z) = P(d)\sum_c P(w\,|\,c)P(c\,|\,d)P(z\,|\,c) \qquad (2)$$

As mentioned in the previous section, we use a semi-supervised version of the PLSA algorithm. Let us assume for simplicity that we have *K* components (i.e. we assume one component per class) and *K+1* classes including the new "fake" label. The likelihood of model (2) given our dataset is:

$$L_1 = \sum_{i \in L \cup U} \log P(w_i, d_i, z_i) \qquad (3)$$

We can combine the co-occurrences for the same pair *(w,d)* , i.e. when a word *w* appears more than once in a document *d*, by introducing a new notation $n(w,d)$ which represents the number of occurrences of the word *w* in the document *d*. The likelihood becomes:

$$L_2 = \sum_d \sum_w n(w,d) \log P(w,d,z(d)) \qquad (4)$$

where *z(d)* is the label of document *d* (*z(d)*=0 for all unlabeled documents, as explained previously). The above likelihood can be split in two terms, corresponding to the labeled and the unlabeled documents, respectively:

$$L_3 = \sum_{d \in L} \sum_{w} n(w,d) P(d = z(d)) \log P(w \mid c = z(d)) P(d \mid c = z(d)) P(z(d) \mid c = z(d)) + \sum_{d \in U} \sum_{w} n(w,d) \log P(d) \sum_{c} P(w \mid c) P(c \mid d) P(z = 0 \mid c) \quad (5)$$

The first part of (5) can be maximised analytically, but the second part, due to the presence of the sum under the log, requires an iterative optimisation such as the EM algorithm. As shown in [4], the steps of the EM algorithm can be depicted as:

The E-step equation is (for all values of *w* and *d*):

$$\pi_c(w,d) = P(c \mid w, d, z(d)) = \frac{P(c \mid d) P(w \mid c) P(z(d) \mid c)}{\sum_{c'} P(c' \mid d) P(w \mid c') P(z(d) \mid c')} \quad (6)$$

The M-step equations are:

$$P(w \mid c) \propto \sum_{d} n(w,d) \pi_c(w,d) \quad (7)$$

$$P(c \mid d) \propto \sum_{w} n(w,d) \pi_c(w,d) \quad (8)$$

$$P(z \mid c) \propto \sum_{d, z(d)=z} \sum_{w} n(w,d) \pi_c(w,d) \quad (9)$$

In order to avoid mixing examples with different labels in the same component, we assume that each component *c* may only generate examples from one class $(z = z_c)$ or unlabelled examples $(z = 0)$. This means that $\forall c, \forall z, P(z \mid c) = 0$ if $z \notin \{0; z_c\}$.

Once the model parameters are obtained (as a fixed point of EM), for each example x, we want to distribute the probability obtained for the “fake” label *z*, with the ``true" labels:

$P(y \mid x) \propto P(z = y \mid x) + \lambda P(y \mid z = 0) P(z = 0 \mid x)$

In our experiments, $P(y \mid z = 0) = 1/2$ and $\lambda = 0.005$.

3 ACTIVE LEARNING

We consider the pool-based active learning model, where the essential idea is to select promising unlabeled examples in a sequential process. By `promising` we mean that the knowledge of the corresponding label will contribute to a more accurate prediction function. Combining semi-supervised and active learning appears to be particularly beneficial in reducing the annotation burden for the following reasons:

1. It constitutes an efficient way of solving the exploitation/exploration problem: semi-supervised learning is more focused on exploitation, while active learning is more dedicated to exploration. Semi-supervised learning alone may lead to poor performance in the case of very scarce initial annotation. It strongly suffers from poorly represented classes, while being very sensitive to noise and potentially instability. On the other hand, active learning alone may spend too much time querying useless examples, as it can not exploit the information given by the unlabeled data

2. In the same vein, it may alleviate the data imbalance problem due to each method separately. Semi-supervised learning tends to over-weight easy-to-classify examples that will dominate the process, while active learning has the opposite strategy, resulting in exploring more deeply the hard-to-classify examples [14].

3. Semi-supervised PLSA is able to provide a more motivated estimation of the confidence score associated to the class prediction for each example, taking into account the whole data set, including the unlabelled data. As a consequence, active learning based on these better confidence scores can be expected to be more efficient.

In order to implement active learning on top of the semi-supervised PLSA algorithm described earlier, we choose to annotate the most ambiguous document. Ambiguity is measured as the entropy of the posterior of a document, $P(c|d)$. In the case of binary classification this is simplified in choosing the example for which the probability is closest to 0.5. The selected document is annotated and added to the labelled dataset. The classification rule is then updated using the semi-supervised PLSA algorithm.

4 EXPERIMENTS

The first experiments we have performed on the 20 newsgroups dataset give promising results. We consider binary classification by selecting pairs of newsgroups from the data. In particular, we have chosen the following three pairs of newsgroups:

- (Easy) rec.sport.baseball (994) vs. rec.sport.hockey (999)
- (Moderate) comp.sys.ibm.pc.hardware (982) vs. comp.sys.mac.hardware (961)
- (Hard) talk.religion.misc (628) vs. alt.atheism (799)

They correspond to easy, moderate and hard problems respectively. In all three cases, we use the 80% for the data as a training set (labeled and unlabeled) and we use the rest 20% as a test set, in order to estimate the accuracy of our method. We start with only 1 labeled example in each category and the rest of the examples are unlabeled. We run our method, which will choose the next example to label among the unlabeled ones, pick the label from the reference annotation and update the model accordingly. The procedure continues and other examples are sequentially labelled as requested by the active learning step. In our experiments, we query 50 examples.

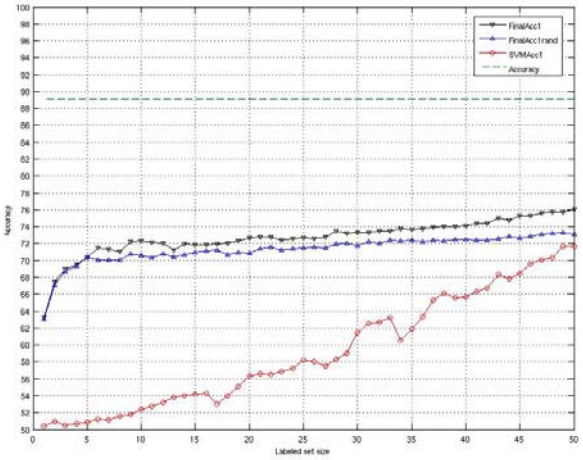

Fig. 1 Comparison of the active semi-supervised PLSA algorithm (top) with the semi-supervised PLSA querying random examples (middle) and SVM querying the examples which are closest to the margin (bottom), for the first pair of newsgroups (hockey vs. baseball). The horizontal line is the upper bound on the performance.

We compare our results with two other methods. In the first one, we use the semi-supervised PLSA, iteratively by querying randomly selected example instead of using active learning. This algorithm will help us see if active learning improves our results. The second alternative is a SVM model where the active learning step consists of querying the label of the example which is closest to the separating hyperplane. In addition, we compute an upper bound on the performance by using all the annotation on the training set. This gives an idea of how much of the potential performance we get using up to 50 annotated examples.

For each model, we perform 20 iterations with different initialization each time (different initial labeled examples). Performance is measured in classification accuracy and is displayed in figures 1 and 2. We can see that the active semi-supervised model gains about 10% in accuracy. The performance of SVM at the beginning is much worst. This is mostly due to the fact that it starts with only two labeled examples, so it's very difficult to learn a good model. The SVM gains more as active learning progresses. Still it is somewhat below the active semi-supervised PLSA model after 50 queries.

Note also that the harder the problem, the bigger the difference between the accuracy of the active, semi-supervised PLSA and the semi-supervised PLSA asking random queries, which shows the importance of the active learning method.

5 CONCLUSION

In this paper we have presented a combination of semi-supervised and active learning applied on the PLSA algorithm. We argue that the proposed technique addresses the issue of annotation cost in supervised learning in two ways: semi-supervised learning helps leverage unlabeled examples to improve categorization accuracy, while active learning helps annotators by choosing the potentially most informative examples. Our algorithm can be extended in order to take into account different labeling costs. In that way, we will be able to choose a document to label by its trade-off of ambiguity and annotation cost (for example, it is harder to annotate a long document than a short one). Also different active learning techniques can be used in addition

with the current proposition.

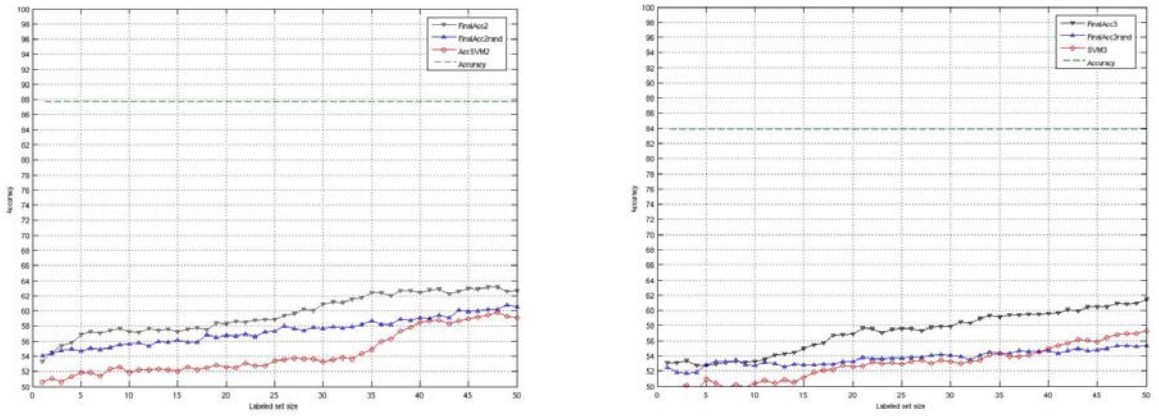

Fig. 2 Results for the other two pairs of newsgroups (pc vs. mac on the left and religion vs. atheism on the right).

Acknowledgment

This work was supported in part by the IST Programme of the European Community, under the PASCAL Network of Excellence, IST-2002-506778. This publication only reflects the authors' views.

REFERENCES

[1] Baram Y., R. El-Yaniv, and K. Luz. Online choice of active learning algorithms, 2003.
[2] Blum A. and Mitchell T. Combining labeled and unlabeled data with co-training. In COLT: Proceedings of the Workshop on Computational Learning Theory, Morgan Kaufmann Publishers, pages 92–100, 1998.
[3] Freund Y., H. Seung S., Shamir E., and Tishby N. Selective sampling using the query by committee algorithm. Machine Learning, 28(2-3):133–168, 1997.
[4] Gaussier E. and Goutte C. Learning from partially labelled data – with confidence. In Proceedings of Learning with Partially Classified Training Data - ICML'05 workshop, 2005.
[5] Hofmann T. Probabilistic latent semantic indexing. In Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval, pages 50–57, 1999.
[6] Joachims T. Transductive inference for text classification using support vector machines. In Ivan Bratko and Saso Dzeroski, editors, Proceedings of ICML-99, 16th International Conference on Machine Learning, pages 200– 209, Bled, SL, 1999. Morgan Kaufmann Publishers, San Francisco, US.
[7] Lewis D. D. and William A. Gale. A sequential algorithm for training ext classifiers. In W. Bruce Croft and Cornelis J. van Rijsbergen, editors, Proceedings of SIGIR-94, 17th ACM International Conference on Research and Development in Information Retrieval, pages 3–12, Dublin, IE, 1994. Springer Verlag, Heidelberg, DE.
[8] McCallum A. and Nigam K. Employing EM and pool-based active learning for text classification. In Proceedings of the Fifteenth International Conference on Machine Learning (ICML'98), pages 350–358, 1998.
[9]Muslea I., Minton S., and Knoblock C. Active + semi-supervised learning = robust multi-view learning. In Proceedings of the Nineteenth International Conference on Machine Learning (ICML'02), pages 435–442, 2002.
[10] Nigam K. and Ghani R. Analyzing the effectiveness and applicability of co-training. In CIKM, pages 86–93, 2000.
[11] Nigam K., McCallum A. K., Thrun S., and Mitchell T. Text classification from labeled and unlabeled documents Using EM. Machine Learning, 39(2/3):103–134, 2000.
[12] Roy N. and McCallum A. Toward optimal active learning through sampling estimation of error reduction. In Proc. 18th International Conf. on Machine Learning, pages 441–448. Morgan Kaufmann, San Francisco, CA, 2001.
[13] Tong S. and Koller D. Support vector machine active learning with applications to text classification. In Pat Langley, editor, Proceedings of ICML-00, 17th International Conference on Machine Learning, pages 999–1006, Stanford, US, 2000. Morgan Kaufmann Publishers, San Francisco, US.
[14] Tur, Hakkani-Tr, and R. E. Schapire. Combining active and semi-supervised learning for spoken language understanding. Speech Communication, 2(45):171–186, February 2005.
[15] Zhou Z.-H., Chen K.-Z., and Jiang Y. Exploiting unlabeled data in content based image retrieval. In Proceedings of European Conference on Machine Learning (ECML'04), pages 525–536, 2004.
[16] Zhu X. Semi-supervised learning literature survey. Technical Report 1530, Computer Sciences, University of Wisconsin-Madison, 2005. http://www.cs.wisc.edu/jerryzhu/pub/ssl survey.pdf.
[17] Zhu X., J. Lafferty, and Z. Ghahramani. Combining active learning and semi-supervised learning using Gaussian fields and harmonic functions. In Proceedings of The Continuum from Labeled to Unlabeled data- ICML'03 Workshop, 2003.