A Fuzzy Semi-Supervised Support Vector Machines Approach to Hypertext Categorization

Houda Benbrahim[1] and Max Bramer[2]

Abstract Hypertext/text domains are characterized by several tens or hundreds of thousands of features. This represents a challenge for supervised learning algorithms which have to learn accurate classifiers using a small set of available training examples. In this paper, a fuzzy semi-supervised support vector machines (FSS-SVM) algorithm is proposed. It tries to overcome the need for a large labelled training set. For this, it uses both labelled and unlabelled data for training. It also modulates the effect of the unlabelled data in the learning process. Empirical evaluations with two real-world hypertext datasets showed that, by additionally using unlabelled data, FSS-SVM requires less labelled training data than its supervised version, support vector machines, to achieve the same level of classification performance. Also, the incorporated fuzzy membership values of the unlabelled training patterns in the learning process have positively influenced the classification performance in comparison with its crisp variant.

1 Introduction

In the last two decades, supervised learning algorithms have been extensively studied to produce text classifiers from a set of training documents. The field is considered to be mature as an acceptable high classification effectiveness plateau has been reached [1]. It has become difficult to detect statistically significant differences in overall performance among several of the better systems even though they are based on different technologies.
However, to achieve these good results, a large number of labelled documents is needed. This coincides with the conclusions from computational learning theory that state that the number of training examples should be at least a multiple of the number of features if reasonable results are sought [2]. Often, several thousand features are used to represent texts, and this leads to a need for thousands of labelled training documents. Unfortunately, obtaining this large set is a difficult task. Labelling is usually done using human expertise, which is tedious,

[1] Dr. Houda Benbrahim
University of Portsmouth, School of Computing, PO1 3HE, UK. email: houda.benbrahim@port.ac.uk
[2] Prof. Max Bramer
University of Portsmouth, School of Computing, PO1 3HE, UK. email: max.bramer@port.ac.uk

expensive, time consuming and error prone. On the other hand, unlabelled documents are often readily available in large quantities, and one might prefer to use unsupervised learning algorithms (restricted here to clustering). Yet, learning solely from unlabelled documents cannot be used to classify new documents into predefined classes because knowledge about classes is missing. In this case, semi-supervised learning comes to the rescue as it lies in between supervised and unsupervised learning approaches. It takes advantage of the strengths of both learning paradigms, i.e. it learns accurate classifiers and exploits the unlabelled data, and discards their major drawbacks, i.e. their need for a large labelled training set and their inability to identify the classes.

The principal question that may arise in semi-supervised learning is how to combine labelled and unlabelled data in the learning system. In order to benefit from unlabelled data in a supervised learning model, a learner must augment unlabelled examples by class labels in some way. However, fully using this newly labelled and originally unlabelled set of training documents in the supervised learning process may harm the performance of the resulting classifier.

Classifying the unlabeled data using any classifier is error prone. Consequently, the newly labelled data imputed in the training set might be noisy, and this usually harms the performance of the learning algorithm as its performance might decrease with noisy training data. A possible solution to this problem is to modulate the influence of the originally unlabelled data in the supervised training phase. This might be achieved by introducing fuzzy memberships to unlabelled documents. In this case, a fuzzy membership value is associated with each document such that different documents can have different effects in the learning of the classifier.

In this paper, a Fuzzy Semi-Supervised Support Vector Machine approach is proposed for hypertext categorization.

Many researchers have studied semi-supervised support vector machines, which attempt to maximize the margin on both labelled and unlabelled data, by assigning unlabelled data to appropriate classes such that the resulting margin is the maximum. Earlier works include Transductive support vector machine (TSVM) first introduced by [3], which uses the unlabelled test set in the training stage. The problem with TSVM is that its training is more difficult. [4] uses an iterative method with one SVM training on each step, while mixed integer programming was used in S3VM [5]. [6] formulated the problem as a concave minimization problem which is solved by a successive linear approximation algorithm and produced V3SVM and CV3SVM.

SVM is sensitive to noise and outliers in the training dataset [7]. To solve this problem, one approach is to do some processing on the training data to remove noise or outliers, and use the remaining set to learn the decision function [8]. Among the other approaches is the introduction of fuzzy memberships to data points such that different data points can have different effects in the learning of the separating hyperplane. Few fuzzy support vector machine approaches exist that treat noise and outliers as less important and let these points have lower membership values [9, 10].

This paper deals with a proposed Fuzzy-Semi-Supervised Support Vector machine framework. It is introduced in two steps. First, we describe the concept of semi-supervised clustering guided by labelled data. Then, we define how unlabelled data is partially incorporated into the learning process of the support vector machines model. Several experiments will be conducted to provide empirical evidence about (i) the effect of the number of labelled training documents in the fuzzy semi-supervised support vector machines learning process, and (ii) the effect of the number of unlabelled training documents in the fuzzy semi-supervised support vector machines learning process.
Fuzzy semi-supervised support vector machines approach is described in section 2. Section 3 presents experiments and results, comparing different classification algorithms. Section 4 concludes the paper.

2 Fuzzy Semi-Supervised Support Vector Machines Approach

Semi-supervised learning is halfway between supervised and unsupervised learning. In addition to unlabelled data, the algorithm is also provided with labelled data. In this case, the data set X can be divided into two parts: set $X_L = \{x_1, \ldots, x_L\}$, for which labels $Y_L = \{y_1, \ldots, y_L\}$ are provided, and a set $X_u = \{x_1, \ldots, x_u\}$ where the labels are not known. The objective of semi-supervised learning is to benefit from both supervised and unsupervised learning when combining labelled and unlabelled data.
The open question that may arise is how to take advantage of the unlabelled data to build a classifier. There are many approaches to this problem. The one adopted in this work is to train a classifier based on labelled data as well as unlabelled data. Typically, the unlabelled data is clustered then labelled, and then the augmented labelled data is used to train the final classifier. Two key issues in this approach are (i) how to impute labels to unlabelled data and (ii) how to use the augmented labelled data to train the classifier.
The semi-supervised task in this paper can be formulated as follows: As a first step, a clustering algorithm (unsupervised learning) can be applied to discover groups in the unlabelled data; in this case, a c-means clustering algorithm [11] might be used. However, determining a suitable number of clusters and generating a suitable starting solution is a challenge for clustering algorithms. To overcome this dilemma, labelled data can be used in the unsupervised learning step. Therefore, a semi-supervised c-means algorithm [12] is applied. It also allows labelling the discovered clusters/groups. As a second step, a model is learned based on a supervised learning algorithm namely support vector machines trained by the whole set of labelled data and the newly labelled unlabelled data.
In the crisp support vector machines approach, each training pattern has the same weight/importance in deciding about the optimal hyperplane. In this paper, and in this proposed FSS-SVM algorithm, the originally unlabelled data along with their imputed class labels in addition to the labelled data are used as a training set.

However, classical SVM learning is sensitive to noisy data because of the inherent "over-fitting" problem. This may increase the classification error [7, 13], and in order to decrease the effect of this possible noise that might originate from the unlabelled training sample, each training pattern is assigned a membership value, that corresponds to its weight in SS-FCM, to modulate the effect of the training data on the learning process of SVM. FSS-SVM also maximizes the margin of separation and minimizes the classification error so that a good generalization can be achieved. To reach that objective, FSS-SVM models the effect of unlabelled data incorporated in the training set.

FSS-SVM

The proposed fuzzy semi-supervised support vector machines algorithm works as follow:

- Let X be the set of training examples. X is divided into two parts: set $X_L = \{x_1, \ldots, x_L\}$, for which labels $Y_l = \{y_1, \ldots, y_L\}$ are provided, and a set $X_u = \{x_1, \ldots, x_u\}$ where the labels are not known.

- SSFCM is used to impute the class labels of the unlabelled data set. Each unlabelled example x_j is assigned to class $y_j^u = \arg\max_{i \in \{1,\ldots,c\}} u_{ij}^u, \forall j \in \{1,\ldots,n_u\}$ with membership value μ_{ij}.

- The set $X_L = \{(x_1, y_1), \ldots (x_L, y_L)\}$ of labelled patterns, and a set of $X_u = \{(x_1, y_1, \mu_1), \ldots (x_u, y_u, \mu_u)\}$ of unlabelled patterns with their corresponding imputed class label and fuzzy membership value in that class are used as a training set for FSS-SVM.

- The optimal hyperplane problem can be regarded as the solution to:

$$\min \tfrac{1}{2}\|w\|^2 + C\left[\sum_{i=1}^{L}\xi_i + \sum_{j=1}^{u}\mu_j\xi_j^*\right]$$

Subject to:
$$y_i\left[\langle w, x_i\rangle + b\right] \geq 1 - \xi_i, i = 1,\ldots,L$$
$$y_j\left[\langle w, x_j\rangle + b\right] \geq 1 - \xi_j^*, j = 1,\ldots,u$$
$$\xi_i \geq 0, i = 1,\ldots,L$$
$$\xi_j^* \geq 0, j = 1,\ldots,u$$

Since ξ_i is the measure of error of a pattern x_i in the SVM learning process, the term $\mu_i\xi_i$ is then the measure of error with different weighting. The smaller the value μ_i , the smaller the effect of ξ_i , which means that the corresponding x_i is treated as less important.

Hence the solution is:

$$\lambda^* = \arg\min_{\lambda} \tfrac{1}{2} \sum_{i=1}^{L+u} \sum_{j=1}^{L+u} \lambda_i \lambda_j y_i y_j \left\langle x_i, x_j \right\rangle - \sum_{i=1}^{L+u} \lambda_k$$

With constraints:

$$0 \le \lambda_i \le C, i = 1,..., L$$

$$0 \le \lambda_i \le \mu_i C, i = 1,..., u$$

$$\sum_{j=1}^{L+u} \lambda_j y_j = 0$$

3. Experiments

In this section, several experiments have been conducted to provide empirical evidence that learning from both labelled and unlabelled data through our proposed fuzzy semi-supervised support vector machines approach outperforms the traditional crisp supervised SVM learning algorithm which learns only from labelled data.
Mainly, we will check in those experiments:

- The effect of the number of labelled training documents in the fuzzy semi-supervised support vector machines learning process.
- The effect of the number of unlabelled training documents in the fuzzy semi-supervised support vector machines learning process.

3.1 Datasets

BankSearch [14] and Web->KB (www.cs.cmu.edu/~webkb/) hypertext datasets were used to evaluate the performance of the new classifier. However, we do not have available unlabelled data related to these datasets. For this reason, 30% of the available data was held aside and used as unlabelled data.

3.2 The classification task

The classification problem for both datasets is a single-label-per-document multiclass case, which means that the classifiers must decide between several categories, and each document is assigned to exactly one category. However, all the classification tasks were mapped into their equivalent binary classification problems. The one against all method was used to split the n-class classification problem into n-binary problems.

3.3 Document presentation

The pre-processing step for documents in both datasets comprises the following. The content of HTML pages, along with their corresponding extra information extracted. Each document representation is enhanced by its title + link anchor + meta data + similar neighbour [15]. However, when using labelled and unlabelled data for learning a classifier, we have to specify how the unlabelled data will participate in the different steps of the hypertext representation, namely, indexation, feature reduction and vocabulary generation.
For the indexation phase, all indexes occurring in both labelled and unlabelled documents are taken into consideration; this is to enrich the vocabulary of the dataset in case there are a small number of labelled documents.
Dimensionality reduction can also be applied when dealing with labelled and unlabelled documents. However, some restrictions are posed. For example, the information gain feature selection technique cannot be used in this case as it requires that the class label be known. To be able to use it anyway, the measure can be restricted to labelled documents, this leads to loss of information related to unlabelled data. Moreover, this class-dependent feature selection tends to be statistically unreliable as we are assuming that the labelled documents are scarce.
Hence, for feature reduction, we apply only stop word removal, stemming, and elimination of words that occurs at most once in the training dataset. Then all the remaining indexes are used to build the dictionary.

3.4 Evaluation procedure

Two different evaluation procedures were carried out for the two datasets.
For WEB->KB dataset, a 4-fold leave-one-university-out-cross-validation was used. That is for each experiment, we combined the examples of three universities to learn a classifier which was then tested on the data of the fourth university.
For the BankSearch dataset, the holdout method is used. The dataset is randomly split into 70% training and 30% testing and repeated 30 times.
Micro-averaged F1 and accuracy measures were used to evaluate the classifiers.

3.5 The effect of the number of labelled training documents

Figures 1 and 2 show the classification F1 measure of the fuzzy semi-supervised support vector machines (FSS-SVM) on the two hypertext datasets when the number of labelled training documents is varied, and the number of unlabelled training documents is kept fixed (30% from each class). The results are contrasted with the learning results of SVM (which learns from only the labelled training documents), SSFCM and SS-SVM.

SS-SVM is a simple version of a semi-supervised SVM. The originally unlabelled data is classified using SSFCM algorithm. Then, each pattern is crisply assigned to the class that corresponds to the highest value in the resulting membership matrix.
The horizontal axes indicate the number of labelled training documents. For instance, a total of 11 training documents for the BankSearch dataset correspond to 1 (one) document per class and a total of 40 training documents correspond to10 documents per class for the Web->KB dataset. The vertical axes indicate the F1 measure on the test sets.
In all experiments, the fuzzy semi-supervised support machine performs better than its supervised version when the number of labelled training documents is small, i.e. FSS-SVM can achieve a specific level of classification accuracy with much less labelled training data. For example, with only 550 labelled training examples for the BankSearch dataset (50 documents per class), FSS-SVM reaches 0.65 F1 measure classification, while the traditional SVM classifier achieves only 0.5. For the same labelled training set size, F1 measure of SS-SVM is 0.46 and 0.55 for SSFCM. In other words, to reach 0.65 classification F1 measure, for example, SVM requires about 1100 and FSS-SVM only 550 labelled training documents.
Similarly, for the WebKB dataset, the performance increase is smaller but substantial; this may be because of the small size of the unlabelled data. For instance, for 80 labelled training examples (20 documents per class), SVM obtains 0.29 F1 measure and FSS-SVM 0.59, reducing classification error by 0.3. For the same number of labelled documents, SS-SVM achieves 0.36 F1 measure and SSFCM 0.43.
For both datasets, FSS-SVM is superior to SVM when the amount of labelled training data is small. The performance gain achieved by the semi-supervised learners decreases as the number of labelled training documents increases. The reason for this is that more accurate classifiers can be learned from the labelled data alone. As the accuracy obtained through plain supervised learning approaches a dataset-specific plateau, we barely benefit from incorporating unlabelled documents through semi-supervised learning.

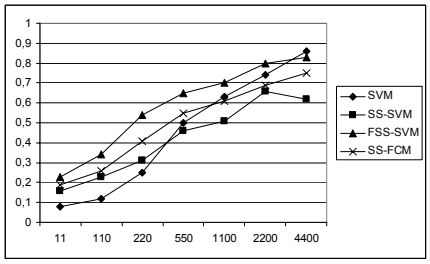

Figure 1: Classifiers F1 measure with different numbers of labelled data used for training for BankSearch dataset

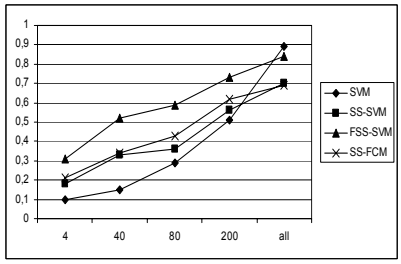

Figure 2: Classifiers F1 measure for different numbers of labelled data for training for WEB->KB dataset

In fact, note that the accuracy of FSS-SVM also degrades when the number of labelled training documents is very large. For instance, with 4400 labelled training examples (400 documents per class) on the BankSearch dataset, classification F1 measure decreases from 0.86 to 0.81.

To summarize, the results for the fuzzy semi-supervised support vector machines classifier show that the benefit we may achieve from the use of unlabelled documents strongly depends on the number of labelled training documents. The learning performance increases as the number of labelled training documents increases. So, when the number of labelled training documents is small, the learning algorithm needs more help. Therefore, the learner benefits from the additional unlabelled documents even though their imputed class labels are uncertain. However, it seems that the fuzzy hyperplane margin that modulates the influence of the imputed labelled data enhances the classifier's performance. SSSVM performance degrades in some cases in comparison with that of SVM as more unlabelled documents are incorporated in the training set. This might be explained by the fact that the imputed labels of the unlabelled data tend to be incorrect as they are predicted by SSFCM, and therefore may not be correctly classified.

3.6 The effect of the number of unlabelled training documents

In the previous set of experiments, we have shown that the extent to which we may benefit from unlabelled documents depends on the number of labelled training documents available. Obviously, this benefit will also depend on the number of unlabelled documents. The results below examine the effect of the unlabelled set size on the classifier's performance. Figures 3 and 4 show the classification F1 measure of FSS-SVM with different numbers of labelled training documents on the BankSearch and WEB->KB datasets when the number of unlabelled documents is varied (10%, 20% or 30% of the available unlabelled data). In all cases, adding unlabelled data often helps learning more effective classifiers. Generally, performance gain increases as the amount of labelled data decreases. Also, performance gain increases with the number of unlabelled documents until it reaches a plateau.

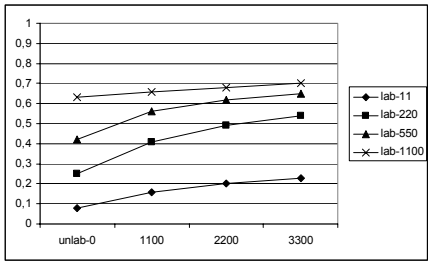

Figure 3: FSS-SVM F1 measure with different numbers of labelled training documents and different numbers of unlabelled training documents for BankSearch dataset.

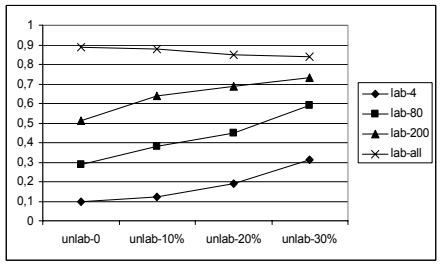

Figure 4: FSS-SVM F1 measure with different numbers of labelled training data and different numbers of unlabelled training data for WEB->KB dataset.

4. Conclusion

In this paper, we have presented a fuzzy semi-supervised support vector machines learning approach to hypertext categorization. This is learning from labelled and unlabelled documents. This is a crucial issue when hand labelling documents is expensive, but unlabelled documents are readily available in large quantities, as is often the case for text classification tasks. The following summarizes the results of the empirical evaluation:

• FSS-SVM can be used to learn accurate classifiers from a large set of unlabelled data in addition to a small set of labelled training documents. It also outperforms its supervised version (SVM). In other words, FSS-SVM requires less labelled training data to achieve the same level of classification effectiveness.

References

[1] Liere, R. and P. Tadepalli (1996). "The use of active learning in text categorization." Proceedings of the AAAI Symposium on Machine Learning in Information Access.

[2] Lewis, D. D. (1992). "Feature selection and feature extraction for text categorization." Proceedings of the workshop on Speech and Natural Language: 212-217.

[3] Vapnik, V. N. (1998). Statistical learning theory, Wiley New York.

[4] Joachims, T. (1999). "Transductive inference for text classification using support vector machines." Proceedings of the Sixteenth International Conference on Machine Learning: 200-209.

[5] Bennett, K. and A. Demiriz (1998). "Semi-supervised support vector machines." Advances in Neural Information Processing Systems **11**: 368–374.

[6] Fung, G. and O. Mangasarian (1999). "semi-supervised support vector machines for unlabeled data classification." (Technical Report 99-05). Data mining Institute, University of Wisconsin at Madison, Madison, WI.

[7] Zhang, X. (1999). "Using class-center vectors to build support vector machines." Neural Networks for Signal Processing IX, 1999. Proceedings of the 1999 IEEE Signal Processing Society Workshop: 3-11.

[8] Cao, L. J., H. P. Lee, et al. (2003). "Modified support vector novelty detector using training data with outliers." Pattern Recognition Letters **24**(14): 2479-2487.

[9] Lin, C. F. and S. D. Wang (2002). "Fuzzy support vector machines." IEEE Transactions on Neural Networks **13**(2): 464-471.

[10] Sheng-de Wang, C. L. (2003). "Training algorithms for fuzzy support vector machines with noisy data." Neural Networks for Signal Processing, 2003. NNSP'03. 2003 IEEE 13th Workshop on: 517-526.

[11] Bezdek, J. C. (1981). Pattern Recognition with Fuzzy Objective Function Algorithms, Kluwer Academic Publishers Norwell, MA, USA.

[12] Bensaid, A. M., L. O. Hall, et al. (1996). "Partially supervised clustering for image segmentation." Pattern Recognition 29(5), 859–871.

[13] Guyon, I., N. Matic, et al. (1996). "Discovering informative patterns and data cleaning." Advances in knowledge discovery and data mining table of contents: 181-203.

[14] Sinka, M. P. and D. W. Corne (2002). "A large benchmark dataset for web document clustering." Soft Computing Systems: Design, Management and Applications **87**: 881-890

[15] Benbrahim, H. and M. Bramer (2004). "Neighbourhood Exploitation in Hypertext Categorization." In Proceedings of the Twenty-fourth SGAI International Conference on Innovative Techniques and Applications of Artificial Intelligence, Cambridge, December 2004, pp. 258-268. ISBN 1-85233-907-1