International Journal of Computer Applications Technology and Research
Volume 3– Issue 9, 595 - 599, 2014, ISSN: 2319–8656

# An Evaluation of Feature Selection Methods for Positive-Unlabeled Learning in Text Classification

Azam Kaboutari
Computer Department
Islamic Azad University,
Shabestar Branch
Shabestar, Iran

Jamshid Bagherzadeh
Computer Department
Urmia University
Urmia, Iran

Fatemeh Kheradmand
Biochemistry Department
Urmia University of Medical Sciences
Urmia, Iran

**Abstract**: Feature Selection is important in the processing of data in domains such as text because such data can be of very high dimension. Because in positive-unlabeled (PU) learning problems, there are no labeled negative data for training, we need unsupervised feature selection methods that do not use the class information in the training documents when selecting features for the classifier. There are few feature selection methods that are available for use in document classification with PU learning. In this paper we evaluate four unsupervised methods including, collection frequency (CF), document frequency (DF), collection frequency-inverse document frequency (CF-IDF) and term frequency-document frequency (TF-DF). We found DF most effective in our experiments.

**Keywords**: feature selection; unsupervised feature selection; positive-unlabeled learning; PU learning; document classification

## 1. INTRODUCTION

Feature selection for classification is the process of selecting a subset of relevant features among many input features and to remove any redundant or irrelevant one. The default in classifying text documents is to use terms as features. Feature selection reduces the dimensionality of the feature space, which leads to a reduction in computational burden. Furthermore, in some cases, classification can be more accurate in the reduced space. [12]

Many methods for feature selection have been presented. Most of these methods are supervised that use the class information in the training data when selecting features for the classifier. Hence, for supervised methods to be usable, a pre-classified set of documents must be available.

In recent years, a new type of learning problems has been raised due to the emergence of real-world problems that blurred traditional machine learning tasks division into supervised and unsupervised categories. These are partially supervised learning problems that do not need full supervision. One of these problems is the problem of learning from positive and unlabeled examples. This problem, called Positive-Unlabeled learning or PU learning [2], assumes two-class classification. However, the training data only has a small set of labeled positive examples and a large set of unlabeled examples, but no labeled negative examples. We suppose this problem in the context of text classification and Web page classification.

So, supervised feature selection methods cannot be applied for the feature selection of the PU learning problem when there are no available training data for the second class. However, there are few feature selection methods that are unsupervised and available for use in partially supervised learning problems. In Unsupervised feature selection methods, the training data does not need to be manually classified. All that is needed is a fixed set of documents the classifier is to be used on. Hence, these methods are handy for PU learning problem.

In Web and text retrieval applications, the PU learning problem occurs frequently, because most of the time the user is only interested in documents of a particular topic. In this application positive documents are usually available or collecting some from the Web or any other source is relatively easy. But Collecting negative training documents is especially delicate and arduous because (1) negative training examples must uniformly represent the universal set excluding the positive class and (2) manually collected negative training documents could be biased because of human's unintentional prejudice, which could be detrimental to classification accuracy [8]. PU learning eliminates the need for manually collecting negative training documents.

PU learns from a set of positive data as well as a collection of unlabeled data. Unlabeled data indicates random samples of the universal set for which the class of each sample is arbitrary and uncorrelated. Random sampling can be done in most databases, warehouses, and search engine databases (e.g., DMOZ[1]) or it can be done independently directly from the Internet. So the dimensions of feature space that contains the terms appearing in the training (positive and unlabeled) documents will be very high and need for effective methods for feature selection is essential.

In this paper we review some unsupervised feature selection methods and evaluate their performance on a number of PU learning techniques. We find that feature selection based on document frequency seems particularly promising for PU Learning.

In the next section we review some related works that focused on evaluation of feature selection methods for text classification. Section 3 provide an overview of PU learning and describe the PU learning techniques included in the evaluation. In section 4 we describe some unsupervised feature selection methods considered in the evaluation - the evaluation is presented in section 5. The paper concludes with a summary and some proposals for further research in section 6.

## 2. RELATED WORK

Previous feature selection studies for text domain consider the problem of selecting one set of features for multi-class classification. These problems are traditional classification

[1] http://www.dmoz.org/

problems that labeled examples for each class are available for use in training and often supervised methods are applied for feature selection.

For example a review of traditional feature selection methods used in text classification can be found in [14]. This study considered five feature selection metrics, including document frequency (DF), information gain (IG), mutual information (MI), $\chi^2$-test (CHI) and term strength (TS) and found that IG and CHI are most effective in their experiments.

Another work [6] presents an empirical comparison of twelve feature selection methods. In addition, a new feature selection method, called bi-normal separation, is shown to outperform other commonly known methods in some circumstances.

In other study [7], ten feature selection methods including a new feature selection method, called the GU metric were evaluated. The experiments were performed on the 20 Newsgroups data sets with the Naive Probabilistic Classifier. The results show that the GU metric obtained best F-score.

# 3. POSITIVE-UNLABELED LEARNING

One of the difficulties of supervised learning algorithms is that a large number of labeled examples are needed in order to learn accurately. In text classification, the labeling is typically performed manually by reading the documents, which is a time consuming task and can be very labor intensive. Partially supervised learning problems such as PU learning do not need full supervision, and therefore are able to reduce the labeling effort.

PU learning is a collection of techniques for training binary classifier on positive and unlabeled examples only. Traditional binary classifiers for text or Web pages require laborious preprocessing to collect positive and negative training examples.

In PU learning [2], two sets of examples are available for training: the positive set P and an unlabeled set U, which is assumed to contain both positive and negative examples, but without these being labeled as such. The aim is to build an accurate binary classifier without the need to collect negative examples.

Two kinds of approaches have been suggested to build PU classifiers: the two-step approach and the direct approach. The two-step approach as its name indicates consists of two steps: (1) extracting some reliable negative (RN) documents from the unlabeled set, (2) Constructing a set of classifiers by using a classification algorithm iteratively and then selecting a good classifier from the set. These approaches include S-EM [3], PEBL [8], Roc-SVM [10] and CR-SVM [11]. Direct approaches such as biased-SVM [4] and Probability Estimation [5] also are offered to solve the problem.

## 3.1 Techniques for Step 1

In two-step approaches five techniques proposed for step 1:

### *3.1.1 Spy*

It randomly samples small percentage of positive documents from P and put them in U to act as “spies”. Thus new sets Ps and Us are made respectively. Then runs the naïve Bayesian (NB) algorithm using the set Ps as positive and the set Us as negative. The NB classifier is then applied to assign each document d in Us a probabilistic class label Pr(+1|d). It uses the probabilistic labels of the spies to decide which documents are most likely to be negative. S-EM [3] uses Spy technique.

### *3.1.2 Cosine-Rocchio*

It first extracts a set of potential negatives PN from U by computing similarities of the unlabeled documents in U with the positive documents in P using the cosine measure. To extract the final reliable negatives, the algorithm applies the Rocchio classification method to build a classifier f using P and PN. Those documents in U that are classified as negatives by f are regarded as the final reliable negatives and stored in set RN. This method is used in [11].

### *3.1.3 1DNF*

It first find the set of words W that occur in the positive documents more frequently than in the unlabeled set, then extract those documents from unlabeled set that do not contain any word in W. These documents form the reliable negative documents. This method is employed in PEBL [8].

### *3.1.4 Naïve Bayesian*

It runs the naïve Bayesian (NB) algorithm using the set P as positive and the set U as negative. The NB classifier is then applied to classify each document in U. Those documents that are classified as negative documents denoted by RN. This method is employed in [4].

### *3.1.5 Rocchio*

The algorithm is the same as that in previous technique except that NB is replaced with Rocchio. This method is used in Roc-SVM [10].

## 3.2 Techniques for Step 2

If the reliable negative set RN is sufficiently large and contains mostly negative documents, a learning algorithm such as SVM using P and RN used in this step and it works very well. But if a very small set of negative documents identified in step 1, then running a learning algorithm will not be able to build a good classifier, rather a learning algorithm iteratively till it converges or some stopping criterion is met. For iteratively learning approach two techniques proposed, which are based on EM and SVM respectively.

### *3.2.1 EM-NB*

This method is based on naïve Bayesian classification (NB) and the EM algorithm. The Expectation-Maximization (EM) algorithm is an iterative algorithm for maximum likelihood estimation in problems with missing data [1]. The EM algorithm consists of two steps, the Expectation step that fills in the missing data, and the Maximization step that estimates parameters. Estimating parameters leads to the next iteration of the algorithm. EM converges when its parameters stabilize. In this case the documents in Q (= U−RN) regarded as having missing class. First, a NB classifier f is constructed from set P as positive and set RN as negative. Then EM iteratively runs and in Expectation step, uses f to assign a probabilistic class labels to each document in Q. In the Maximization step a new NB classifier f is learned from P, RN and Q. The classifier f from the last iteration is the result. This method is used in [3].

### *3.2.2 SVM Based*

In this method, SVM is run iteratively using P, RN and Q (= U-RN). In each iteration, a new SVM classifier f is constructed from set P as positive and set RN as negative, and then f is applied to classify the documents in Q. The set of documents in Q that are classified as negative is removed from Q and added to RN. The iteration stops when no document in Q is classified as negative. The final classifier is the result. This method, called I-SVM is used in [8]. In the other similar method that is used in [10] and [4], after iterative

SVM converges, either the first or the last classifier selected as the final classifier. The method, called SVM-IS.

# 4. FEATURE SELECTION METHODS

There are two main categories of feature selection methods: filters and wrappers. In filter methods feature scoring metrics are used on each feature for measure feature relevance and ranking features. Wrapper methods perform a search algorithm like greedy hill-climbing over the space of all feature subsets, repeatedly calling the same induction algorithm that is later used for building the classifier, as a subroutine to evaluate subsets of features. Where filter methods evaluate each feature independently, wrappers evaluate feature sets as a whole, which would avoid redundant features and lead to better results. However, wrapper methods are often impractical and very computationally intensive for large datasets, and are also more prone to overfitting, so filter methods are more commonly used.

Unsupervised feature selection methods [9] are methods that do not use the class information in the training data when selecting features for the classifier. It means that the training data does not need to be manually pre-classified. All that is needed is a fixed set of documents from the collection the classifier is to be used on. Hence, these methods are handy if there is no pre-classified training data available, and if there is no time to create such data. So these methods are suitable for PU learning. However, pre-classified documents are of course needed for evaluation of the classifier's performance.

In the current study we choose four unsupervised filter methods for feature selection in PU Learning:

## 4.1 Collection Frequency (CF)

The collection frequency [9] of a feature is the total number of instances of the feature in the collection, in our case in P ∪U. It does not look at which documents or categories the feature occurs in, it is simply a count.

## 4.2 Document Frequency (DF)

One of the simplest methods of vocabulary reduction and vector dimensionality reduction is the document frequency [12]. The document frequency of a feature is the number of documents containing a feature in the training set, in our case in P ∪U.

## 4.3 Collection Frequency-Inverse Document Frequency (CF-IDF)

The CF-IDF [9] is computed by weighting the collection frequency values by the inverse document frequency for feature:

$$CF{-}IDF\ (w) = CF\ (w) \times \log_2 (N / DF\ (w)) \qquad (1)$$

Where w denoted feature and N is the total number of documents in the training data, in our case N= |P ∪U|.

## 4.4 Term Frequency-Document Frequency (TF-DF)

In [13], a method based on the term frequency combined with the document frequency is presented. They call it Term Frequency-Document Frequency, and prove it better than DF measure. TF-DF for feature w is computed as follows:

$$TF{-}DF\ (w) = (n_0 \times n_1 + c\ (n_0 \times n_2 + n_1 \times n_2)) \quad (2)$$

Where c≥1 is a constant, n0 is the number of documents in the training data without the feature; n1 is the number of documents where the feature occurs exactly once, n2 is the number of documents where the feature occurs twice or more. As the value of c increases, we give more weight for multiple occurrences of a term. The authors of [13] use c=10 in their experiments, and we follow this decision in our experiments.

# 5. EVALUATION

## 5.1 Data Set

In our experiments the universal set is the Internet. We used DMOZ, which is a free open directory of the Web containing millions of Web pages, to collect random samples of Internet pages as unlabeled set U. To construct an unbiased sample of the Internet, a random sampling of a search engine database such as DMOZ is sufficient [8]. We randomly selected 5,700 pages from DMOZ to collect unbiased unlabeled data. We also manually collected 539 Web page about diabetes as positive set P to construct a classifier for classify diabetes and non diabetes Web pages. For evaluating the classifier, we manually collected 2500 non-diabetes pages and 600 diabetes page. (We collected negative data just for evaluating the classifier we construct.)

## 5.2 Performance Measure

We report the result with F-score, a good performance measure for binary classification. F-score is the harmonic mean of precision and recall. Precision is defined as number of correct positive predictions division by number of positive predictions. Recall is defined as number of correct positive predictions division by number of positive data.

## 5.3 Experimental Results

We now present the experimental results. We extracted features from normal text of the content of Web pages, and then we perform stopwording, lowercasing and stemming. Finally we get a set of about 176,000 words. We used four methods which is discussed briefly in Section IV in our evaluation and create a ranked list of features, and returns the i highest ranked features as selected features, which i is in {200, 400, 600, 1000, 2000, 3000, 5000, 10000}.

As discussed in Section III, we studied 5 techniques for Step 1 and 3 techniques for Step 2 (EM-NB, I-SVM and SVM-IS). Clearly, each technique for first step can be combined with each technique for second step. In this paper, we will empirically evaluate only the 5 possible combinations of methods of Step 1 and Step 2 that available in the LPU[2], a text learning or classification system, which learns from a set of positive documents and a set of unlabeled documents. These combinations are S-SVM which is Spy combined with SVM-IS, Roc-SVM is Rocchio combined with SVM-IS, Roc-EM is Rocchio+EM-NB, NB-SVM is Naïve Bayesian+ SVM-IS and NB-EM is Naïve Bayesian+ EM-NB.

In our experiments each document is represented by a vector of selected features, using a bag-of-words representation and term frequency (TF) weighting method which the value of each feature in each document is the number of times (frequency count) that the feature (word) appeared in the document. When running SVM in Step 2, the feature counts are automatically converted to normalized tf-idf values by LPU. The F-score for 5 combinations of methods of Step 1 and Step 2 are shown in Figure 1 to 5. In each combination we perform an evaluation of 4 feature selection methods.

[2] http://www.cs.uic.edu/~liub/LPU/LPU-download.html

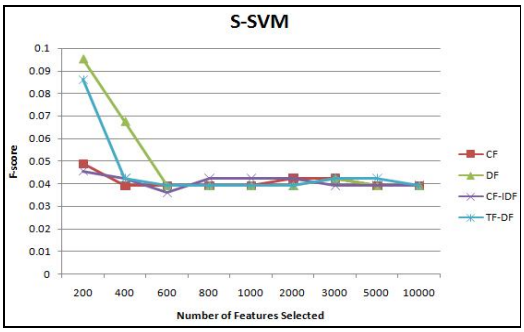


Figure 1. Results of LPU (Spy in Step 1 and SVM-IS in step 2) using 4 feature selection methods.

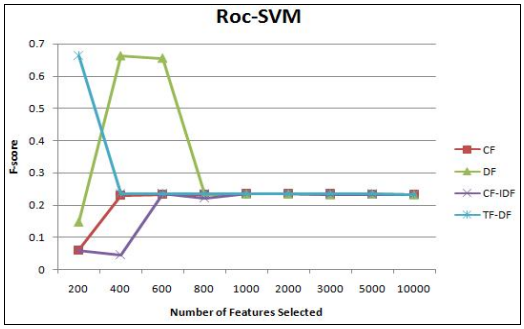


Figure 2. Results of LPU (Rocchio in Step 1 and SVM-IS in step 2) using 4 feature selection methods.

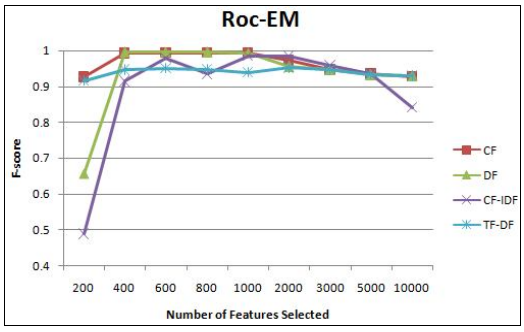


Figure 3. Results of LPU (Rocchio in Step 1 and EM-NB in step 2) using 4 feature selection methods

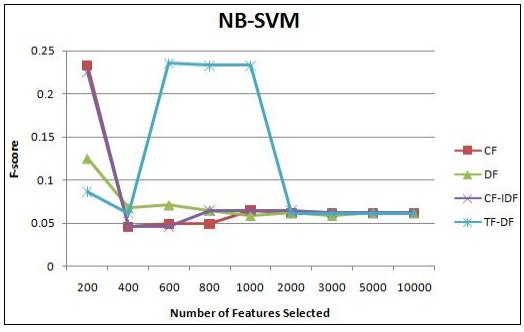


Figure 4. Results of LPU (Naïve Bayesian in Step 1 and SVM-IS in step 2) using 4 feature selection methods.

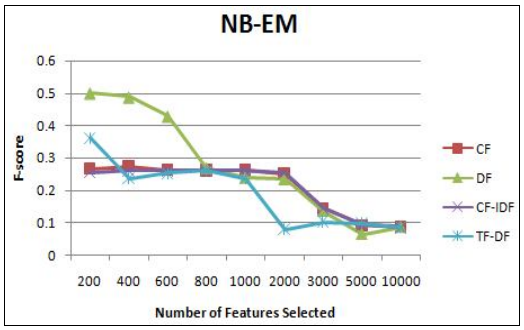


Figure 5. Results of LPU (Naïve Bayesian in Step 1 and EM-NB in step 2) using 4 feature selection methods

As Figure 1 shows, very poor results are obtained using feature selection methods in S-SVM which Spy is used in Step 1 and SVM-IS is used in Step 2. Since we obtain better results in other combinations that SVM-IS is used in Step 2, we conduct that Spy in not good technique for Step 1 in our experiments.

Figure 2 shows that when using Rocchio technique in Step 1, better results can be achieved using all feature selection methods. In this case, DF method in average is better than other feature selection methods.

Figure 3 shows the best results we have obtained in our experiments. As can be seen in Figure 3, when number of feature is 400 and more, all 4 feature selection methods can achieve good results, but CF method results in average is better than others. Figure 3 also shows that how using EM-NB instead of SVM-IS in Step 2 can improve results of all feature selection methods significantly.

Figure 4 shows results of 4 feature selection methods when Naïve Bayesian is used for Step 1 and SVM-IS for Step 2. In this case also we have obtained poor results. Best result in average is obtained from TF-DF method that is 0.122. When using EM-NB instead of SVM-IS in Step 2, results are improved. These results are shown in Figure 5. In this case, with increasing the dimension of feature space, the results are worse. Best result in average is obtained from DF method.

The average results of 4 feature selection methods in each combination of techniques of Step 1 and Step 2 are shown in Table 1. Last column indicate the method that achieved best result among other methods.

**Table 1. Comparison of feature selection methods.**

| Methods | CF | DF | CF-IDF | TF-DF | Best |
|---|---|---|---|---|---|
| **S-SVM** | 0.041 | 0.049 | 0.041 | 0.045 | DF |
| **Roc-SVM** | 0.214 | 0.319 | 0.192 | 0.282 | DF |
| **Roc-EM** | 0.964 | 0.933 | 0.891 | 0.94 | CF |
| **NB-SVM** | 0.076 | 0.07 | 0.077 | 0.122 | TF-DF |
| **NB-EM** | 0.212 | 0.271 | 0.208 | 0.191 | DF |

## 6. CONCLUSIONS

In this paper, we discussed the 4 unsupervised methods for feature selection in learning a classifier from positive and unlabeled documents using the two-step strategy. An evaluation of 5 combinations of techniques of Step 1 and Step

2 that available in the LPU system was conducted to compare the performance of each feature selection method in each combination, which enables us to draw some important conclusions. Our results show that in general Document Frequency method outperforms other methods in most case. Also we found that best combination for LPU in our experiments is R-EM, which is Rocchio, combined with EM-NB. In this combination best results are obtained by the Collection Frequency method.

In our future studies, we plan to evaluate other combinations for Step 1 and Step 2 and other unsupervised feature selection methods for Positive-Unlabeled Learning.

## 7. REFERENCES

[1] A. Dempster, N. Laird and D. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," Journal of the Royal Statistical Society. Series B (Methodological), 1977, 39(1): p. 1-38.

[2] B. Liu and W. Lee, "Partially supervised learning", In "Web data mining", 2nd ed., Springer Berlin Heidelberg, 2011, pp. 171-208.

[3] B. Liu, W. Lee, P. Yu and X. Li, "Partially supervised classification of text documents," In Proceedings of International Conference on Machine Learning(ICML-2002), 2002.

[4] B. Liu, Y. Dai, X. Li, W. Lee and Ph. Yu, "Building text classifiers using positive and unlabeled examples," In Proceedings of IEEE International Conference on Data Mining (ICDM-2003), 2003.

[5] C. Elkan and K. Noto, "Learning classifiers from only positive and unlabeled data," In Proceedings of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD-2008), 2008.

[6] G. Forman, "An extensive empirical study of feature selection metrics for text classification," The Journal of Machine Learning Research, 3, 3/1/2003.

[7] G. Uchyigit, "Experimental evaluation of feature selection methods for text classification," Fuzzy Systems and Knowledge Discovery (FSKD), 2012 9th International Conference on , vol., no., pp.1294,1298, 29-31 May 2012.

[8] H. Yu, J. Han and K. Chang, "PEBL: Web page classification without negative examples", Knowledge and Data Engineering, IEEE Transactions on , vol.16, no.1, pp. 70- 81, Jan. 2004.

[9] Ø. Garnes, "Feature selection for text categorisation," Master's thesis, Norwegian University of Science and Technology, 2009.

[10] X. Li and B. Liu. "Learning to classify texts using positive and unlabeled data". In Proceedings of International Joint Conference on Artificial Intelligence (IJCAI-2003), 2003.

[11] X. Li, B. Liu and S. Ng, "Negative Training Data can be Harmful to Text Classification," In Proceedings of Conference on Empirical Methods in Natural Language Processing (EMNLP-2010), 2010.

[12] X. Qi and B. Davison, "Web page classification: Features and algorithms," ACM Comput. Surv., 41(2):1–31, 2009.

[13] Y. Xu, B. Wang, J. Li and H. Jing, "An extended document frequency metric for feature selection in text categorization," Proceedings of the 4th Asia information retrieval conference on Information retrieval technology, January 15-18, 2008, Harbin, China.

[14] Y. Yang, J. Pedersen, "A comparative study on feature selection in text categorization". In Proceedings of the Fourteenth International Conference on Machine Learning (ICML). Morgan Kaufmann, San Francisco, CA, 412–420,1997.