An Improved Expectation Maximization based Semi-Supervised Email Classification using Naïve Bayes and K- Nearest Neighbor

Hiral Padhiyar
CE Department,
Uka Tarsadiya University

Purvi Rekh
CE Department,
SCET, Surat

ABSTRACT
With the development of Internet and the emergence of a large number of text resources, the automatic text classification has become a research hotspot. Emails is one of the fastest and cheapest communication ways that today it has became the part of communication means of millions of people. It has become a part of everyday life for millions of people, changing the way we work and collaborate. The large percentage of the total traffic over the internet is the email. Email data is also growing rapidly, creating needs for automated analysis. In many security informatics applications it is important to detect deceptive communication in email. In the iterative process in the standard EM-based semi-supervised learning, there are two steps: firstly, use the current classifier constructed in the previous iteration to predict the labels of all unlabeled samples; then, reconstruct a new classifier based on the new training samples set. In this work, an EM based Semi-Supervised Learning algorithm using Naïve Bayesian is proposed in which unlabeled documents are divided into two parts, reliable and misclassified. An Ensemble technique is used to add only reliable unlabeled documents to the training set. Also preprocessing of unlabelled documents is performed before learning process of Naïve Bayesian and K-NN classifiers during first step of EM to reduce time of preprocessing, so with this proposed work accuracy of classifier will be increased and execution time will be decreased.

Keywords
Email Classification, Naïve Bayes, K-NN, SSL.

1. INTRODUCTION
Emails is one of the fastest and cheapest communication ways that today it has became the part of communication means of millions of people. Text (such as E-mail) classification is a process of assigning an electronic document to one or more categories based on its content. Semi-Supervised learning (SSL) is a machine learning method which is combination of Supervised and Unsupervised Learning. Main limitation of Supervised Learning is that large amount of training data are required to get good accuracy. In many real-world applications, it is often fairly expensive to collect many labeled samples since labels are manually assigned by experienced analysts. In contrast, lots of unlabeled samples can be easily collected. As a consequence, semi-supervised learning has become a very hot topic in machine learning and data mining, which combines small amount of labeled samples with large amount of unlabeled samples to improve learning performance. So, main task in Semi-Supervised learning is to label all unlabeled documents using available labeled documents to increase the size of training set and eventually increasing accuracy of classification.

Expectation Maximization is a class of iterative algorithms for maximum likelihood estimation in problems with incomplete data. This process provides a way that incorporates unlabeled data into supervised learning, and experiments show that using unlabeled data can reduce classification error. There are two steps in the iterative process in the standard EM-based semi-supervised learning: firstly, use the current classifier constructed in the previous iteration to predict the labels of all unlabeled samples; then, reconstruct a new classifier based on the new training samples set, which is composed of labeled samples and all unlabeled samples (with the predicted labels).

2. THEORETICAL BACKGROUND AND RELATED WORK
Classification is the process of finding a model (or function) that describes and distinguishes data classes or concepts, for the purpose of being able to use the model to predict the class of objects whose class label is unknown. The derived model is based on the analysis of a set of training data (i.e., data objects whose class label is known). Whereas classification predicts categorical (discrete, unordered) labels, prediction models continuous-valued functions. That is, it is used to predict missing or unavailable *numerical data values* rather than class labels.

2.1 Classification using Naïve Bayes:
A naive Bayes classifier is a simple probabilistic classifier based on applying Bayes theorem with strong (naive) independence assumptions. A more descriptive term for the underlying probability model would be "independent feature model".

Bayes theorem:

Let X be a data tuple. In Bayesian terms, X is considered "evidence." As usual, it is described by measurements made on a set of n attributes. Let H be some hypothesis, such as that the data tuple X belongs to a specified class C. For classification problems, to determine $P(H|X)$, the probability that the hypothesis H holds given the "evidence" or observed data tuple X. In other words, we are looking for the probability that tuple X belongs to class C, given that we know the attribute description of X. where P $(H|X)$ is the posterior probability, or *a posteriori probability*, of H conditioned on X. In contrast, $P(H)$ is the prior probability, or *a priori probability*, of H.

In simple terms, a naive Bayes classifier assumes that the presence or absence of a particular feature is unrelated to the presence or absence of any other feature, given the class variable. For example, a fruit may be considered to be an apple if it is red, round, and about 3" in diameter. A naive

Bayes classifier considers each of these features to contribute independently to the probability that this fruit is an apple, regardless of the presence or absence of the other features. For some types of probability models, naive Bayes classifiers can be trained very efficiently in a supervised learning setting. In many practical applications, parameter estimation for naïve Bayes models uses the method of maximum likelihood; in other words, one can work with the naive Bayes model without believing in Bayesian probability or using any Bayesian methods.

An advantage of Naive Bayes is that it only requires a small amount of training data to estimate the parameters (means and variances of the variables) necessary for classification. Because independent variables are assumed, only the variances of the variables for each class need to be determined and not the entire covariance matrix.

For E-mail classification like to check mail is spam or not, In [8] Naive Bayes classifier examines all of the instance vectors from both classes. It calculates the prior class probabilities as the proportion of all instances that are spam (Pr [spam]), and non-spam (Pr [nonspam]). Then (assuming binary attributes) it estimates four conditional probabilities for each attribute: Pr [true|spam], Pr [false|spam], Pr [true|notspam], and Pr [false|notspam]. These estimates are calculated based on the proportion of instances of the matching class that have the matching value for that attribute. To classify an instance of unknown class, the "naïve" version of Bayes's rule is used to estimate first the probability of the instance belonging to the spam class, and then the probability of it belonging to the not-spam class. Then it normalizes the first to the sum of both to produce a spam confidence score between 0.0 and 1.0.

2.2 Text classification using K- nearest neighbor [8]

The k-nearest neighbor (K-NN) classifier is considered an example-based classifier, that means that the training documents are used for comparison rather than an explicit category representation, such as the category profiles used by other classifiers. As such, there is no real training phase. When a new document needs to be categorized, the k most similar documents (neighbors) are found and if a large enough proportion of them have been assigned to a certain category, the new document is also assigned to this category, otherwise not. Additionally, finding the nearest neighbors can be quickened using traditional indexing methods. To decide whether a message is legitimate or not, we look at the class of the messages that are closest to it. The comparison between the vectors is a real time process. This is the idea of the k nearest neighbor algorithm:

Stage1.

Training Store the training messages.

Stage2.

Filtering Given a message x, determine its k nearest Neighbors among the messages in the training set. If there are more spam's among these neighbors, classify given message as spam. Otherwise classify it as legitimate mail.

2.3 Working of classic Semi-Supervised EM algorithm [3]

Here, documents are classified based on the combination of Expectation-Maximization (EM) and a naive Bayes classifier. The algorithm first trains a classifier using the available labeled documents, and probabilistically labels the unlabeled documents. The iterative process in the standard EM-based semi-supervised learning includes two steps: firstly, use the classifier constructed in previous iteration to classify all unlabeled samples; then, train a new classifier based on the reconstructed training set, which is composed of labeled samples and all unlabeled samples.

Table 1 shows the classic EM based algorithm and Table 2 shows comparison of semi-supervised algorithms proposed by different researchers:

Table 1. Classic EM Based Algorithm

Pre-processing of D^l and D^u

Inputs : Collections D^l of labeled documents and D^u of unlabeled documents.

Method :

Build an initial naive Bayes classifier, $\Theta\hat{}$, from the labeled documents, D^u , only. Use maximum a posteriori parameter estimation to find $\Theta\hat{}$ argmax $P(D/\Theta)P(\Theta)$

- Loop while classifier parameters improve, as measured by the change in L_c (Θ/ D, z) c
- **(E-step)**

  Use the current classifier, $\Theta\hat{}$, to estimate component membership of each unlabeled document, *i.e.*, the probability that each mixture component (and class) generated each document, $P(c_j / d_i ; \Theta\hat{})$.
- **(M-step)**

  Re-estimate the classifier, $\Theta\hat{}$, given the estimated component membership of each document. Use maximum a posteriori parameter estimation to find $\Theta\hat{} = \text{argmax}_\Theta P(D / \Theta)P(\Theta)$

Output:

A classifier, $\Theta\hat{}$, which takes an unlabeled document and predicts a class label.

Here, D^l = Set of Labeled documets , D^u = Set of Unlabeled documents

N = Number of Labeled Documents, U = Number of Unlabeled Documents

P = Number of Features (words), i = Number of iterations

Table 2. Comparison of Algorithms proposed by Researchers

Criteria	Reference Papers					
	[1]	[2]	[3]	[5]	[6]	[7]
Dataset Used	E-mails/ benchmark spam filtering corpora (PU1 & LINGSPAM)	Public spam e-mail dataset	Chinese Short documents of different categories.	TREC07p corpus	DARPA 1999 dataset	E-mails
Distribution of Dataset uniform	Yes	Yes	Yes	NS	NS	NS
Training, Testing Split	1 set for testing and 1 for training.	NS	NS	NS	NS	NS
Parameters compared for Accuracy	NS	Accuracy for diff algo on parameter AUC & F1	Time of iteration Vs. macro F1	True positive rate Vs. false positive rate	Classification accuracy Vs. number of labeled alerts	NS
Measures of evaluation used	Accuracy Measure f1	Accuracy Measure F1	Macro F1	ROC Curve	Classification Accuracy	AUC and Measure F
Method used for initial distribution of EM	NOT SSL	NOT SSL	NB	NB	NB	NS
Feature Selection method used	IG and TFV(term frequency variance)	NS	MI(Mutual Information)	TF-IDF	NS	Feature fusion
Uses more than one classifier	Yes	No	Yes	Yes	Yes	Yes

3. LIMITATION OF CURRENT ALGORITHM

There is a problem in the process of reconstructing the training set in classic EM based algorithm, some unlabeled samples are misclassified by the current classifier because the initial labeled samples are not enough, and the performance of the classifier is not well. These misclassified samples are considered directly as training samples, and used to construct a new classifier. This process affects the performance and accuracy of classification.

4. PROPOSED ALGORITHM

The heading of a section should be in Times New Roman 12-point bold in all-capitals flush left with an additional 6-points of white space above the section head. Sections and subsequent sub- sections should be numbered and flush left. For a section head and a subsection head together (such as Section 3 and subsection 3.1), use no additional space above the subsection head.

4.1 Features of a New System

To solve this problem, an improved EM-based semi-supervised learning method is proposed in which in the iterative process, unlabeled samples are divided into two parts, reliable and misclassified; then a new training samples set is reconstructed by adding the reliable part only to the labeled samples set. The partition of unlabeled samples is implemented by fusion NB and K-NN.

Following table shows reason why Naïve Bayesian is widely used for E-mail Classification and why TF-IDF feature selector is used. It gives better performance than other classifier if we consider both Time and Accuracy factors. Also following results of simulation shows accuracy of K-NN is greater than all classifier but it takes more time. Here Simulation is done in Rapid Miner Tool.

Table 3. Comparison of time & accuracy for different classifiers and different feature selection method

	TF/Execution Time	**TF-IDF/ Execution Time**
Naïve bayes	96.25% / 5 s	93.75% / 4 s
SVM	87.5% / 5 s	81.25% / 6 s
K-NN	86.25% / 3 s	93.80% / 2 s
DT	>30 min	>30 min

4.2 Design of proposed work

Following are the steps of proposed work:

1) Divide Documents into training and Test documents.
2) Perform preprocessing on documents
3) Apply feature Selection process
4) Apply proposed algorithm of SSL and make learn classifier
5) Apply learned classifier on test documents to find their category.

4.3 Algorithm of proposed system

Below table 4 shows the algorithm of proposed system.

Table 4. Algorithm of Proposed Work

N = Number of Labeled Documents, U = Number of Unlabeled Documents p = Number of Features,
i = Number of iterations, M = No of Reliable documents

Input:

Labeled training set $L=\{d1,d2,...,d_m\}$; (Preprocessed) Unlabeled training set $U=\{u1,u2,\ldots,u_n\}$; (Not preprocessed) Test set $T=\{t1,t2,\ldots,t_p\}$;

Document Preprocessing Steps of Unlabeled data:

1) Tokenization: Space is used as tokenizer.

2) Remove Stop words: Stop words mean extremely common words, such as 'the', 'and', 'of', 'can', 'we' which are considered useless are removed from the documents.

3) Word Stemming: Porter stemmer [10] is used for Stemming.

4) Feature Selection: TF-IDF is used as feature selection method.

Output: Classifier *NBC*;

Method:

Step 1:

1.1 Construct a NB classifier *NBC* with *L*. Set a new incremental correct subset of reliable data $R=\emptyset$;

1.2 Construct K-NN classifier *K-NNC* with *L*;

Step 2:

Run the following *E-step* and *M-step* circularly until classifier *NBC* is convergent:

2.1 (E-step)

For each ui *U-R* do

StepE1: Classify each sample u_i in *U* with classifier *NBC* and obtain the label *Cu1*;

StepE2: Classify each sample u_i in *U* with classifier *K-NNC* and obtain the label *Cu2*;

StepE3: If *Cu1=Cu2* then $L=L+\{<u_i, Cu1>\}$, $R=R+\{<u_i, Cu1>\}$, or set $i=i+1$ and go to *StepE1*;

2.2 (M-step)

Step M1: Reconstruct a NB classifier *NBC* with *L*;

Step M2: Reconstruct a K-NN classifier *K-NNC* with *L*;

Step 3:

Output the classifier *NBC*.

Step 4:

Classify each sample *ti* in *T* with classifier *NBC* and get its *Macro-F1*;

Following are the additional steps for improvement:

1) Two additional pre-processing steps are performed which are:

- Word Stemming using Porter Stemmer and s
- Feature Selection using TF-IDF

2) In E-step, K-NN classifier is used along with NB classifier to cross check class label assigned by NB as some misclassified documents may also be added to labeled dataset by NB classifier when initial labeled document set is limited.

3) In E-step, pre-processing of Unlabeled documents is performed in parallel with learning of Naïve Bayesian classifier and K-NN classifier to reduce time complexity.

5. EXPECTED OUTPUT

Accuracy: As from simulation of table 3, K-NN gives equal or somewhat better performance than NB even training documents are less, so improvement in accuracy is expected in proposed algorithm as K-NN is used along with NB to prevent adding misclassified documents in labeled training set.

6. CONCLUSION AND FUTURE EXTENSION

Semi-Supervised Learning with EM can be effectively used for improving performance of Classification when limited numbers of labeled documents are available for training. To solve problem of misclassified unlabeled documents added in labeled documents set of in each iteration of classic algorithm of EM, improvements are proposed in this research work, which include ensemble learning using k-NN and NB to include only reliable labeled documents to training set in each iteration to increase accuracy. By this proposed work not only accuracy of classification will be increased but also execution time will be reduced.

Future extension includes implementing proposed system to increase the accuracy of classification when initial labeled documents are less. And another goal to test effect of different feature selection and preprocessing Techniques on Classification using SSL.

7. REFERENCES

[1] S. Appavu and R.Rajaram, "*Learning to classifying threaten email*", 2008 IEEE.

[2] Lei SHI, Qiang WANG "Spam *e-mail classification using Decesion tree Ensemble*", 2012.

[3] Xinghua Fan and Houfeng Ma, "An *improved EM-based Semi-supervised learning method*", 2009 IEEE.

[4] Xiaojin Zhu, "Semi-*Supervised Learning Literature Survey*", Computer Sciences TR 1530, University of Wisconsin – Madison, 2005.

[5] Jun-ming Xu, Giorgio Fumera, Fabio Roli and Zhi-Hua Zhou "*Training SpamAssassin with Active Semi-supervised Learning*", CEAS 2009.

[6] Haibin Mei and Minghua zhang, "*A semi supervised IDS alert classification model based on alert context*", ICCSEE 2013.

[7] Ye Tian, Gary M. Weiss and Qiang Ma, "*A semi-supervised approach for web spam detection using combinatorial feature-fusion*", 2007.

[8] Vinod Patidar, Divakar Singh, "*A Survey on Machine Learning Methods in Spam Filtering*", International Journal of Advanced Research in Computer Science and Software Engineering, Page(s): 964-972, October 2013

[9] Jalili, S., Bitarafan, "*Increase the efficiency of text categorization based on the improved feature selection method*", 2006.

[10] MohammadReza FeiziDerakhshi and Nayer TalebiBeyrami, "*The Feature Selection and Dimensionality Reduction Methods for Email Classification*", Journal of Basic and Applied Scientific Research , 633-636, 2013.

[11] Xiaojin Zhu, "Semi-*Supervised Learning Literature Survey*", Computer Sciences TR 1530, University of Wisconsin – Madison, 2005.