A Practical Approach for Emails Multiclass Classification according to Senders using Naïve Bayers Technique

G. Girija Rani
RGM College of Engineering &Technology,
Nandyal-518501

M. Indra Sena Reddy
RGM College of Engineering &Technology,
Nandyal-518501

ABSTRACT
Emails are parts of everyday life. These messages have become increasingly important and widespread method of communication because of its time speed, where the amount of email messages received per day can range from tens for a regular user to thousands for companies. Everyone is overwhelmed with emails, including relational (structured) and non-relational (semi-structured or non-structured), quite a bit of which is repetitive, stale and of drastically differing quality. This large quantity is confounded. Not just spam messages are thought to be 'garbage', additionally undesirable messages (e.g. advertisements, lottery) individuals' waste a lot of time unknowingly by surfing them. So there is much need to categorization of Emails. Classification can help to meet lawful and administrative necessities for recovering particular data inside of a set time span, and this is frequently the inspiration driving implementing data classification. This paper aims at examining on ways doing supervised and unsupervised grouping of messages as per email content.

Keywords
Supervised, unsupervised, classification.

1. INTRODUCTION
Data mining (DM) [10] frequently alluded as learning disclosure in databases (KDD), is a procedure of nontrivial extraction of verifiable, already obscure and probability helpful data from an extensive volume of data [11][12]. DM is a multi-disciplinary approach comprising of database technology, high performance computing, machine learning, numerical mathematics, statistics and visualization. The Data Mining algorithms ought to be computationally possible for data investigation however takes low human intervention. As specified, DM can be performed by utilizing a few strategies [13]. Among those methods, classification [14] is exceptionally main stream and this procedure is in effect seriously utilized as a part of numerous genuine business applications now-a-days [15].

In general classification or categorization is

In the present scenario: A depiction of an example, $x \in X$, where X is the example dialect or instance space.

Problem is: How to depict text documents. To a set of classes: $C = \{c_1, c_2, \ldots, c_n\}$

To estimate: The category of x: $c(x) \in C$, where $c(x)$ is a *classification function* whose domain is X and whose range is C.

Classification Methods:

(1) Hand-Operated classification

- Used by Yahoo! (initially), look keen, about.com, ODP, PubMed.
- Very accurate when it will be finalized by specialist.
- Consistent if it is for few and problem is small.
- Difficult and cost more to expand.
- Automatic classification methods are needed for big problems.

(2) Automatic document classification

Hand-coded standard-based systems

- Done by CS dept.'s spam filter, Reuters, CIA, and so on.
- Companies give different "IDE" for writing such standards.
- E.g., relegate classification if report contains a given boolean mix of words.
- Standing queries: Commercial systems have complex query languages (everything in IR query languages + accumulators)
- Accuracy is often very high if a rule has been carefully refined over time by a subject expert.
- Building and maintaining these rules is expensive.

(3) Supervised learning of a document-label assignment function.

Many systems partly rely on machine learning (Autonomy, MSN, Verity, Enkata, Yahoo! …)

- Naive Bayes (simple, common method)
- k-Nearest Neighbors (simple, vigorous)
- Support-vector machines (new, more robust)
- … plus many other methods
- No free lunch: requires hand-classified training data
- But data can be built up (and refined) by amateurs.[6]

Email characterization falls into the text classification rule of machine learning. Email Classification can be Content based or Request based classification. Content based classification is classification based on specific subjects in a document determines the class to which the document belongs to. This paper aims on Content based classification. Spam email filtering is a standout amongst the most mainstream points in

text classification throughout the years. There are a few issues to categorize Emails and this is one which has come up:

- **Representation of documents**

For text classification, the most widely recognized mark (or signature) of Emails is words, sequences of words, part-of-speech tags, word clusters etc. [2]. In this paper a bag of words are used for representation.

- **Classifier selection**

Among the supervised learning classifiers Naïve Bayes Classifiers is chosen as it is simple and most common used classifier for text classification.

2. PROCEDURE

2.1 Dataset

An extensive corpus of certifiable email messages from Enron employees. [1] The Enron corpus was made open amid the lawful examination concerning the Enron Corporation. This dataset, along with a careful clarification of its source, is accessible at http://www-2.cs.cmu.edu/~enron/. In the crude Enron corpus, there is a sum of 619,446 messages belonging to its 158 employees. Since email categorization is very reliant on nature of people, distinctive individuals from diverse foundations may have distinctive way of organizing messages that they receive. Along these lines, it is picked messages from workers in the same organization, so synthesis of their messages is comparative. Among every one of the messages in this dataset, randomly picked over 2000 emails as the dataset.

2.2 Representation

From the dataset next step is to collect bag of words and count the most occurrences. In this supervised classification is used.

Step 1: Here an environment is created for reading emails by using VB.NET.

Step 2: Among those emails which are of not more than four lines are considered for classification.

Step 3: From them a bag of words are taken.

Step 4: Next MS-Excel is used for placing bag of words labeling in the following way

Categorizing:

The messages are placed into the following five categories:

1. From nearest and dearest.
2. Working community.
3. Co-workers.
4. Publications.
5. Notifications from other organizations.

Step 5: **Stemming:**
Stemming is performed, to the words collected form the dataset of the emails so that phrases with the same meaning be the same.

Step 6: Token list count and modifying list:
From the dataset all tokens are collected and placed along with their counts in excel sheet and modified this list based on least frequently used ones are omitted. The main task is to have the feature vector in a form of matrix.

This matrix contains

- Initially Tokens from each email one by one read are placed according to their categories in separate excel sheets.
- After getting enough samples of each category separate sheets are made to one.
- Separate the samples into 10 gatherings. Without fail, put 9 of them into training sets, and the staying one as testing example.
- The training and testing samples can be used when ever necessary.

2.3 Classification

In this paper Naïve Bayes classification is used for doing classification.

3. RESULTS AND DISCUSSION

The different set of calculations and final results of implementation of the proposed work are displayed. By using Naive Bayes and feature selection. From the training phase matrix the Table1. Which contains prior probabilities for five different categorizes (classes).

Table 1: The Prior probability of each class.

P(C1)	9/37=0.2432
P(C2)	9/37=0.2432
P(C3)	8/37=0.2162
P(C4)	5/37=0.1351
P(C5)	6/37=0.1621

To Compute P(X/Ci) for i=1, 2, 3, 4, 5. The computed conditional probabilities are shown in Table2. [8].

Table2: Class labeled training tuples conditional probabilities.

	C1	C2	C3	C4	C5
L1	3/9	1/9	4/8	5/5	3/6
L2	4/9	1/9	0/8	0/5	0/6
L3	5/9	5/9	0/8	0/5	0/6
L4	4/9	3/9	0/8	0/5	0/6
L5	3/9	6/9	1/8	2/5	2/6
L6	7/9	0/9	0/8	0/5	1/6
L7	3/9	0/9	0/8	0/5	0/6
L8	3/9	0/9	0/8	0/5	0/6
L9	6/9	7/9	0/8	0/5	0/6
L10	0/9	6/9	0/8	0/5	0/6
L11	0/9	6/9	0/8	0/5	0/6
L12	0/9	5/9	0/8	0/5	1/6
L13	6/9	5/9	1/8	0/5	2/6
L14	0/9	0/9	4/8	0/5	0/6
L15	0/9	0/9	6/8	0/5	1/6
L16	0/9	0/9	6/8	0/5	1/6
L17	1/9	0/9	5/8	0/5	0/6
L18	0/9	0/9	0/8	4/5	0/6
L19	1/9	1/9	0/8	3/5	1/6
L20	1/9	0/9	0/8	3/5	2/6
L21	0/9	0/9	0/8	3/5	0/6
L22	0/9	0/9	0/8	0/5	4/6
L23	6/9	6/9	5/8	3/5	3/6

Note: L1-Please, L2-Let, L3-Me, L4-Know, L5-Thanks, L6-Attach, L7-Curve, L8-Need, L9-Congrats, L10-Hope, L11-

Relation, L12-Happy, L13-To, L14-Win, L15-Click, L16-Subscribe, L17-E.mail, L18-Aggrement, L19-Any, L20-Questions, L21-Sign, L22-Day, L23-You. C1= Co-workers, C2= From nearest and dearest, C3= Working community, C4= Publications, C5= Notifications from other organizations. From the above Table2 it can be seen that some values are of *Zero*. This situation is called Zero Probability. This cancels the effects of all the other (posteriori) probabilities (on Ci) involved in the product. So this is avoided and by making use of the technique Laplacian correction or Laplace estimator for correcting the problem. Using the above probabilities, it is obtained for the tuple X1="Please, let, me, know, thanks." for each category the following Table 3.

P(X1/C1)	0.0121
P(X1/C2)	0.0009
P(X1/C3)	0.00006
P(X1/C4)	0.0005
P(X1/C5)	0.0002

Similarly for X2= "Please, let, me, know, to, hope, you, relation"

P(X2/C1)	0.0001
P(X2/C2)	0.0003
P(X2/C3)	0.0000
P(X2/C4)	0.0000
P(X2/C5)	0.0000

Similarly for X3= "You, Subscribe, win, email."

P(X3/C1)	0.0009
P(X3/C2)	0.0003
P(X3/C3)	0.1708
P(X3/C4)	0.0009
P(X3/C5)	0.0006

Similarly for X4= "Thanks, you, day, questions, please"

P(X4/C1)	0.00224
P(X4/C2)	0.00005
P(X4/C3)	0.0006
P(X4/C4)	0.0370
P(X4/C5)	0.01851

Similarly for X5= "Attach, question, you, day"

P(X5/C1)	0.0112
P(X5/C2)	0.0003
P(X5/C3)	0.0004
P(X5/C4)	0.0067
P(X5/C5)	0.0185

To find the class Ci, that maximizes P(X/Ci) P(Ci), to compute

	P(Ci/X1)=P(X1/Ci)*P(Ci)
i=1	0.0121*0.2432=**0.0029**
i=2	0.0009*0.2432=0.0002
i=3	0.0000*0.2162=0
i=4	0.0005*0.1351=0.00006
i=5	0.0002*0.1621=0.00003

Therefore, the Naïve Bayesian classifier predicts C1 for tuple X1.

	P(Ci/X2)=P(X2/Ci)*P(Ci)
i=1	0.0009*0.2432=0.0002
i=2	0.0003*0.2432=0.0000
i=3	0.1708*0.2162=**0.0369**
i=4	0.0009*0.1351=0.0001
i=5	0.0006*0.1621=0.0000

Therefore, the Naïve Bayesian classifier predicts C3 for tuple X2.

	P(Ci/X3)=P(X3/Ci)*P(Ci)
i=1	0.0001*0.2432=0.00002
i=2	0.0003*0.2432=**0.00007**
i=3	0.0000*0.2162=0.0000
i=4	0.0000*0.1351=0.0000
i=5	0.0000*0.1621=0.0000

Therefore, the Naïve Bayesian classifier predicts C2 for tuple X3.

	P(Ci/X4)=P(X4/Ci)*P(Ci)
i=1	0.0022*0.2432=0.0005
i=2	0.0000*0.2432=0.0000
i=3	0.0006*0.2162=0.0001
i=4	0.0370*0.1351=**0.0049**
i=5	0.0185*0.1621=0.0030

Therefore, the Naïve Bayesian classifier predicts C4 for tuple X4.

	P(Ci/X5)=P(X5/Ci)*P(Ci)
i=1	0.0112*0.2432=0.0027
i=2	0.0003*0.2432=0.00007
i=3	0.0004*0.2162=0.00008
i=4	0.0067*0.1351=0.0009
i=5	0.0185*0.1621=**0.0029**

Therefore, the Naïve Bayesian classifier predicts C5 for tuple X5.

From the tables it is inferred that it is estimating accurately the right class and the values are also highlighted.

4. FUTURE WORK

After implementing the Naïve Bayes Classifier in this work still performance can be improved with different algorithms, different signatures, combination of algorithms and more accurate datasets.

5. CONCLUSION

In summary, though a set of different supervised and unsupervised learning methods are there for text classifications problems, Naïve Bayes Classification guarantees

- Works very fast compared to other methods, with low storage requirements.
- Robust to Irrelevant Features.
- Most suitable in the situations where equal weightage is given to all features
- As Naïve Bayes depends on Bayes theorem if the features that are assumed are dependent then this cannot be the right solution.
- A good classifier for text classification.

- New thoughts in feature vector selections, experimenting with different classifiers, more accurate dataset, and also with these varieties of combinations still performance can be improved.

6. ACKNOWLEDGEMENT

The author would like to express sincere thanks to the University Grants Commission (UGC), Government of India for providing financial support in doing this minor research project (MRF-4579/14(SERO/UGC).

7. REFERENCES

[1] Bryan Klimt, Yiming Yang. *Introducing the Enron Corpus.*

[2] Ron Bekkerman. *Automatic Categorization of Email into Folders: Benchmark Experiments on Enron and SRI Corpora*. 2004.

[3] "A SURVEY OF TEXT CLASSIFICATION ALGORITHMS" chapter 6 by Charu C. Aggarwal

[4] "Emails classification by data mining techniques" by Mohammed A. Naser, Athar H. Mohammed *Department of Computer, College of Sciences for Women, University of Babylon.*

[5] "Data Mining: concepts and Techniques" book by Jiawei Han and Micheline Kamber.

[6] Text Classification: The Naïve Bayes algorithm Adapted from Lectures *by* Prabhakar Raghavan (Yahoo and Stanford) and Christopher Manning (Stanford).

[7] http://en.wikipedia.org/wiki/Naive_bayes.

[8] "Is Naïve Bayes a Good Classifier for Document Classification?" by S.L. Ting, W.H. Ip, Albert H.C. Tsang, International Journal of Software Engineering and Its Applications Vol. 5, No. 3, July, 2011.

[9] "An Improved Naive Bayes Text Classification Algorithm In Chinese Information Processing" by Lingling Yuan1 Jiaozuo, P. R. China, 14-15,August 2010, pp. 267-269 ISBN 978-952-5726-10-7.

[10] "Enhanced Classification Accuracy on Naive Bayes Data Mining Models" by Md. Faisal Kabir & Chowdhury Mofizur Rahman, Alamgir Hossain, Keshav Dahal, International Journal of Computer Applications (0975 – 8887) Volume 28– No.3, August 2011.

[11] Agarwal, R., Imielinski, T. and Swami, A. (1993) 'Database Mining: A Performance Perspective', IEEE: Special issue on Learning and Discovery in Knowledge Based Databases, pp. 914-925.

[12] Agarawal, R. and R. Srikant, (1994) 'Fast algorithms for mining association rules', Proceedings of the 20th International Conference on Very Large Data Bases, San Francisco, CA., USA., pp: 487-499. [13] Maindonald, J. H. (1999) 'New approaches to using scientific data statistics, data mining and related technologies in research and research training' Occasional Paper 98/2, The Graduate School, Australian National University.

[13] Quinlan, J. (1986), "Induction of Decision Trees," Machine Learning, vol. 1, pp.81-106.

[14] Berson, A., Smith, S. J. and Thearling, K. (1999) Building Data Mining Applications for CRM McGraw-Hill.