Available online at www.isas.org.in/jisas
JOURNAL OF THE INDIAN SOCIETY OF AGRICULTURAL STATISTICS 72(1) 2018 61–69

Text Document Categorization using Machine Learning Algorithm in Agricultural Domain

Sreekumar Biswas[1] and Rajni Jain[2]
[1]*ICAR-Indian Agricultural Statistics Research Institute, New Delhi*
[2]*ICAR-National Institute of Agricultural Economics and Policy Research, New Delhi*

Received 11 November 2016; Revised 12 December 2017; Accepted 27 December 2017

SUMMARY

Research in the field of agriculture is increasing in such a way that it is getting very tedious job for the scholars to find out their intended research paper by accessing the journals available in our library. Document categorization, in the field of machine learning, is a field of study by which the job of classification does not need any human intervention. The task of classification is done automatically by the machine itself. In this work, a number of research titles has been classified using various machine learning algorithms for searching the best classifying algorithm for document categorization.

Keywords : Text categorization, Text mining, Machine learning, Receiver operating characteristic, Entropy, Classifiers, Data mining techniques, Learning algorithms.

1. INTRODUCTION

With the advancement of the research in information technology, the domain of the information available in digital format is increasing at an exponential rate. As a consequence, the exposure of documents to a researcher has increased; however, only a few of them are relevant for researchers. The agricultural domain is a vast area of information content. Being one of the biggest fields of research, the amount of information and data is increasing in a rapid manner. The information is available in textual form and hence, is unstructured. Due to the ever-growing amount of textual information, users are facing challenges like organizing, analyzing and searching large numbers of documents to get their desired result. Systems that automatically classify text documents into predefined thematic classes or detect clusters of documents with similar content, offer a promising approach to tackle this complexity. The objective of the experiment is to categorize the research articles into suitable classes or categories, so as to reduce the human labour and develop a way of automated categorization of texts. The rest of the paper is organized as follows. The second section describes text categorization with suitable example, section 3 describes some review of literature, section 4 describes the methodology and section 5 presents the framework. Section 6 presents the results of the experiments, followed by conclusion and future work.

Corresponding author: Sreekumar Biswas
E-mail address: sreekumar.iasri@gmail.com

2. TEXT CATEGORIZATION

Text Categorization (TC) refers to the task of classifying a set of text documents automatically into different categories from a predefined set of categories. Let (D_i, C_j) be a collection, $i = 1,2,\dots,n$ and $j = 1,2,\dots,m$. $D = \{D_1, D_2, \dots, D_n\}$ is the collection of documents and $C = \{C_1, C_2, \dots, C_m\}$ are the set of predefined categories. The task of text categorization is to assign values, either 1 or 0, yes or no, true or false, to each document in the set D, depending upon the situation whether the document D_i belongs to any of the C_j's or not, respectively. TC is an example of Machine Learning (ML) in the form of Natural Language Processing (NLP). Text categorization is illustrated here with the help of an example.

Example 1:

Let us consider that there are three different categories, namely Computer Science, Agriculture Engineering, Genetics and we are given three documents:

(a) An approach to feature selection based on Ontology.

(b) Evaluation of the Livelihood Impacts of a Micro-Irrigation Project.

(c) Exploring Interactions between Pathogens and the Drosophila Gut.

These three sentences belong to Computer Science, Agriculture Engineering, Genetics, which are denoted by *1, 2, 3,* respectively. To categorize these documents, all the features in each document and the respective class are denoted in a vector of the form <features>, class. Here <features> represents a set of features. Table 1 shows the extracted features from the three titles in the heading row. Further three rows in the table show the vector for each title in binary form. "1" under an attribute means the feature is present in a document and "0" means it is absent in the document.

Using the extracted vectors a text classifier can be built which can classify new documents. Let a new document, "A novel method incorporating gene ontology information for unsupervised clustering and feature selection" needs to be classified. To perform this classification the document also needs to be represented in vector form as shown in table 2.

Comparing the new vector (class is unknown) with the vectors (class is known) of table 2.1, we can classify the new document(s). The comparison is shown in Fig. 1. In this comparison, bold faced binary values are matching with the unknown vector to be classified. We observe that there are 19, 13, 13 matches with v1, v2, v3 vectors, respectively. Thus the new document represented by the vector v4 belongs to class 1, which is representing computer science.

Table 1. Vector representation of training data

An	Approach	to	feature	Selection	based	on	Ontology	Evaluation	of	the	Livelihood	Impacts	a	Microirrigation	Project	Exploring	Interactions	between	Pathogens	and	the	Drosophila	Gut	Class
1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	2
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	3

Table 2. Vector representation of test data

An	Approach	to	feature	selection	based	on	Ontology	Evaluation	of	the	Livelihood	Impacts	a	Microirrigation	Project	Exploring	Interactions	between	Pathogens	and	the	Drosophila	Gut	Class
0	0	0	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	

<0, 0, 0, 1, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0>*UNKNOWN*

<1, 1, 1, **1, 1**, 1, 1, **1**, **0**, **0**, **0**, **0**, **0**, **0**, **0**, **0**, **0**, **0**, **0**, **0**, **0**, **0**, **0**, **0**, **0**>*1* (19 matches)

<**0**, **0**, **0**, 0, 0, **0**, **0**, 0, 1, 1, 1, 1, 1, 1, 1, 1, **0**, **0**, **0**, **0**, **0**, **0**, **0**, **0**>*2* (13 matches)

<**0**, **0**, **0**, 0, 0, **0**, **0**, 0, **0**, **0**, **0**, **0**, **0**, **0**, **0**, **0**, 1, 1, 1, 1, 1, 1, 1, 1 >*3* (13 matches)

Fig. 1. Comparison of Vectors

Besides document classification, there are number of other uses of TC. Next section presents literature review regarding applications of TC.

3. REVIEW OF LITERATURE

In the late 80's, a set of experts were used to build a classifier manually by making rules. It was a huge time consuming task as they need to study the characteristics for the given categories. The late 90's witnessed the huge growth of digital information and the manual work of TC became laborious. ML techniques were then adopted to solve the problems. TC by the application of ML was proven to be a huge success. The most common application of TC is to classify news according to their genre, e.g. into sports, politics etc. The application is extended to classification of web documents into directories, such as Yahoo Directory (Ceci and Malerba, 2007). There are some other areas where TC is proved to be very useful. For product categorization, Kuroiwa *et al.* (2007) showed the recommendation of books to the customers of Amazon, using TC. Felcher *et al.* (2001) showed the automatic organization of products on large scale on-line shopping portals. Pang and Lee (2004), Zhu and Zhang (2006) showed the automatic organization of user product reviews for the new users willing to purchase. Text categorization has been found useful for E-mail filtering also. Siefkes *et al.* (2004), Bratko *et al.* (2006), Bickel *et al.* (2007), Kosmopoulos *et al.* (2008)detected the spam e-mails from important e-mails using TC. Detection of authorship is an important area of research for publishing industry. Holmes and Richard (1995) applied multivariate techniques and genetic algorithms that can be useful in fraud detection in authorship. Sebastiani (2001), Zhang (2004), Kotsiantis (2006) showed the application of TC for classification of digital documents. However, no literature is available to the best of our knowledge which describes the auto classification of agricultural research. In this paper an attempt has been made to categorize titles of agricultural research articles as per pre-defined labels.

4. METHODOLOGY

Algorithms are the important components for building any document classifier. To build a classifier, data is divided into two parts namely training and test data. Training part is used by the algorithm to build a classifier and test part is used to evaluate the generated classifier. In this section various potential algorithms for TC and evaluation techniques for evaluating the classifiers are discussed.

4.1 Algorithms:

4.1.1. ZeroR: The simplest classifier among all the ML classifiers to be discussed later on this paper is the ZeroR algorithm. This algorithm depends on the target ignoring all predictor values. ZeroR classifier directly predicts the majority class. ZeroR has no capacity to predict, but it is useful for determining a baseline performance as a benchmark for other classification methods. At first, the algorithm constructs a frequency table for the target and selects its most frequent value. As found in the training set, this algorithm assigns each attribute with the most common class.

4.1.2. OneR: OneR is the abbreviation for "One Rule". It is a simple classifier, but possesses the capability of better prediction than ZeroR. It generates one rule for each of the predictors in the data set, and then, it selects the rule with the smallest total error as its "one rule". For creating a rule for each predictors, a frequency table must be created for each predictors against the target. This algorithm produces rules that are very simple for the humans to interpret. One drawback of OneR is that it does not generate any score or probability. But like the ZeroR, it can also be used for determining a baseline performance as a benchmark for other classification methods.

4.1.3. J4.8 Decision Tree: J4.8 is a java implementation of the C4.5 algorithm and is a version of a decision tree algorithm. The core algorithm for building decision trees (predictive ML model) was referred as ID3 by J.R. Quinlan (1993). It employs divide and conquer approach (Hunt, Marin and Stone, 1996) through the space of possible branches with no backtracking. ID3 uses Entropy and Information Gain to construct a decision tree. The decision tree decides the value of the target of a new test dataset based on various attribute values of the training data set.

4.1.4. Naïve Bayes: This algorithm assumes that the effect of an attribute value on a given class is independent of the values of the other attributes and is based on Bayes' theorem of probability. The assumption is based on class conditional independence. This model very is easy to build. As a result of these assumptions, the computation of the Bayesian classification becomes more efficient. Let's see how this algorithm works. First, the data set is converted into a frequency table using statistical techniques. In the second step, a likelihood table is formed by finding out the probabilities. Finally, by using the Naïve Bayesian equation, the posterior probability for each class is calculated. The class that have the highest posterior probability is the prediction.

$$p(C|x) = \frac{p(x|C)\,.p(C)}{p(x)}$$

Where, $p(C|x)$ is the posterior probability of the class given the probability of the predictor, $p(x|C)$ is the likelihood probability of the predictor given the probability of the class, $p(C)$ is the prior probability of the class, and $p(x)$ is the prior probability of the predictor. The Naïve Bayes approach arrives at the correct classification until the correct category is more probable than the others.

4.1.5. K-Nearest Neighbor: KNN comes under the non-parametric lazy group of algorithms. This algorithm is non-parametric because it does not take into account the distribution of the data. It is called lazy because it does not use the training data points for generalization, for this reason, the training phase is very fast. This algorithm that stores all available cases and classifies new cases based on a similarity measure, for example, distance functions etc. This algorithm is used to test the degree of similarity amidst documents and k training data and to store the information about the classification data, by this it can determine the class of the given test documents. A given test data is classified by a majority vote of its neighbors, with the data points being assigned to the class, surrounded by its most common k nearest neighbors measured by a distance function. There are various distance functions such as Euclidian and Manhattan. These two distance measures are valid to use only if the variables are continuous. If the encountered variables are categorical, then Hamming distance measure should be used.

4.1.6. Random Forest: Random Forest (RF) is a collection of simple decision trees which are decisive enough for classifying a data set. The response from each trees are taken into consideration for the final outcome of the forest. RF (supervised learning technique) was introduced by Leo Breiman and Adele Cutler (2006).It applies the supervised learning method by which the information regarding the training data is recorded and used to build the model. Now the derived model from the training data can then be used to classify test instances.

4.1.7. Support Vector Machine: Support Vector Machine is a supervised machine learning algorithm, used classification or regression problems where the training dataset trains SVM model about the classes so that the model can easily classify any new testing dataset. SVMclassifies the dataset into different classes by finding a hyper plane. The hyper plane separates the dataset into different classes. It is very much possible to find more than one hyper plane that divides the data into different classes, the selection among all these hyper planes is done by maximizing the distance between the support vectors and the hype rplane. The hyper plane that has maximum distance, gets selected.

4.1.8. Multilayer Perceptron: A multilayer perceptron (MLP) is a computational model that have been developed by the inspiration from biological neural network. It is also called Artificial Neural Network (ANN). ANNs have already created a lot of excitement in the machine learning community in the field of speech recognition, computer vision and text processing. The basic unit for this computation is called node or the neuron. Each node receives input from other nodes and the

inputs are then assigned some weights. Then the node applies function to the weighted inputs and produces the output. The function applied is non-linear and called the activation function. There are several activation function such as sigmoid, tanh and Rectified Linear Unit (ReLU). An MLP consists of three or more layers. An input layer, one or more hidden layer, and an output layer.

4.2 Evaluation:

There are various strategies for the evaluation of classifiers. Accuracy, specificity, sensitivity, F-score, k-fold cross validation etc. this study is concerned with finding the best algorithm that can classify agricultural articles. The classification should be significantly better than the classification performed by the other algorithms. For this, paired t-test and ROC analysis has been performed to evaluate the performance of the algorithms.

4.2.1. The t-test: While dealing with more than two algorithms, it is not convenient to use the accuracy to estimate the better performing algorithm. It should be clear that one algorithm is statistically better than the others. For this purpose, the corrected resampled t-test is performed over the results given by the categorization task. It is nothing but the Student's t-test with some modifications in it. The Student's t-statistic is given by:

$$t = \frac{\bar{d}}{\sqrt{\sigma_d^2 / k}}$$

Where, the difference between the means found for the cross validation, is the number of folds of cross validation, follows Student's t distribution with degrees of freedom. Under this test, the null hypothesis will be. The corrected t-test is preferred over the uncorrected one because the uncorrected test always assumes that the samples are independent. However, the way cross validation works, the samples are not independent. As a result, a very high type I error is produced, that is, it gives a very high false positive rate. So, for the corrected resampled t-test, it is assumed that the repeated holdout method is used rather than cross validation and it is repeated times on the different splits of the same dataset. The accuracy estimates of the learning algorithms are recorded accordingly. Suppose that every time, is the size of the training data and is the size of test data. The test statistic of the corrected resampled t-test is given by:

$$t = \frac{\bar{d}}{\sqrt{\left(\frac{1}{k} + \frac{n_2}{n_1}\right)\sigma_d^2}}$$

All the notation being the same as mentioned earlier. After getting the value of, it should be compared with the table value for the specified confidence interval for Student's distribution and interpret the result accordingly.

4.2.2. Receiver Operating Characteristic: Receiver operating characteristics (ROC) graphs are useful for organizing classifiers and visualizing their performance. It is used where the learner is trying to select samples of test instances that have a high proportion of positives. The ROC curves determines the performance of a classifier without considering the class distribution. In an ROC curve, the number of true positives (TP) are plotted on the y-axis and the number of false positives (FP) are plotted on the x-axis. Each point on the ROC curve represents a TP-rate/FP-rate pair corresponding to a particular decision threshold. To evaluate the Machine learning algorithms, the ROC system is applied. Area under ROC ranges from 0.5 to 1. It is clear that different ROCs will be produced for different algorithms. Among all the classifiers, the one, whose ROC is much closer to the y-axis, is considered the best performing algorithm. In other words, the classifier, whose area under ROC tends towards 0.5 is the worst performing one and the classifier, whose area under ROC tends towards 1 is the best performing one.

5. EXPERIMENTAL FRAMEWORK

In this section, brief description about data and the framework of the experiment has been discussed.

5.1 Data:

The data for conducting the experiment was collected from Prof. M. S. Swaminathan Library, Indian Agricultural Research Institute, New Delhi. The data set is initially organized under two fields, namely Title and Class. Title refers to name of the research article and class refers to the theme under which research article is categorized. There are two classes in the data set namely AI and SS. The data is prepared in the arff format which is supported in WEKA (Witten and Frank, 2005). Vectorization is done on this data set for further for text categorization experiment which is illustrated in the next sub-section. The data is summarized in the following table:

Table 3. Description of data

Name of the file	Data_Ag.arff
Number of records	48
Source	Compiled on the basis of research articles selected from Prof. M. S.Swaminathan Library, Indian Agricultural research Institute
No of classes	2 (AI-Artificial Intelligence, SS-Social Science)
Attributes	Title of the paper, 2. Class
Type of attributes	String

5.2 Algorithm for Document Categorization:

Fig. 2 shows the framework for document categorization.

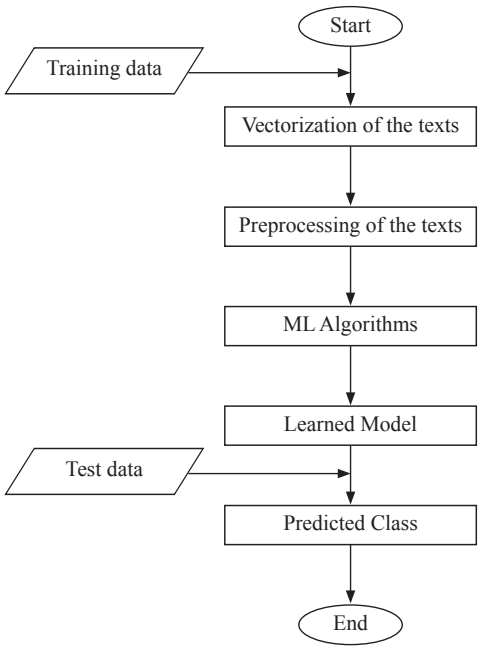

Fig. 2. Flowchart of the proposed work

After the input of training data first step is vectorization. In vectorization, string of texts is represented by a vector. After creating the vector, preprocessing of the vectorized data is done. Data preprocessing consists of stemming and stop-word removal. Stemming is the process of merging words that are different grammatical forms of the same word. Thus, stemming reduces the words to their roots. For example, connection, connections, connective, connected, connecting, all reduce to connect after stemming. Stop-words are those words that appear very frequently in a document. For example, a, an, the, on, of, or, was, were, etc. are the stop words. For classifying any document, it is necessary to remove these words from the document, as these words affect the efficiency of classification. For pre-processing, in this paper, string to word vector filter has been used using weka (Witten and Frank, 2005). After pre-processing of the documents, ML algorithms (section 3) are applied on the processed documents to obtain the corresponding classification model for TC. Now, test data set is used to evaluate the accuracy of the learned model.

6. RESULTS

The experiment was conducted over two classes, namely Social Sciences and Artificial Intelligence. Each of the classes contained a number of titles (strings) from various research papers. For performing TC, the data was randomly split into five different training and test sets and the procedure was repeated on each of the training and test data. The results of the text categorization experiment are presented in table 4.

Following observations are made from table 4:

i. Test set accuracy is less than training set accuracy. This happens because some documents some documents do not match with the existing documents in the training set.

ii. Algorithms ZeroR, OneR LibSVM, J48, and KNN are unsuitable for text categorization (Table 4).

iii. Algorithms NB, RF and MLP are observed with more than 80% accuracy of test data set.

Table 4. Comparison of training data and test data accuracy as obtained using various algorithms to perform text categorization

Iteration	ZeroR		OneR		LibSVM		J48		IBK		MLP		RF		NB	
	Train	Test	Train	Test	Train	Test	Train	Test	Train	Test	Train	Test	Train	Test	Train	Test
1	53.6	37.5	53.6	37.5	57.1	37.5	96.4	37.5	100	62.5	100	68.8	100	75	100	87.5
2	50	50	50	50	96.4	75	82.1	68.6	96.4	56.3	96.4	87.5	96.4	87.5	96.4	93.8
3	60.7	31.3	60.7	30.3	60.7	31.3	89.3	75	100	56.3	100	81.3	100	81.3	100	87.5
4	50	50	50	50	100	75	92.9	75	100	81.3	100	87.5	100	87.5	100	93.8
5	53.6	43.8	58.7	56.3	56.6	43.8	92.9	56.2	96.4	62.5	96.4	75	96.4	68.8	96.4	81.3
Average	53.58	42.52	54.6	44.82	74.16	52.52	90.72	62.46	98.56	63.78	98.56	80.02	98.56	80.02	98.56	88.78

iv. NB is observed as the best algorithm with 89% test accuracy. Thus, further evaluation can be done among MLP, RF and NB using NB as the base line algorithm.

v. Comparison of F-measure of the algorithms is done using results from 10*10 cross validation and paired t-test. ZeroR is used as baseline algorithm. F-measure is defined as twice the product of precision and recall divided by the sum of precision and recall. Null hypothesis is that the given algorithm performs better than ZeroR. The corresponding result is shown in fig. 3. The results show that Random Forest, Multilayer perceptron and Naïve Bayes performs significantly better than ZeroR.

vi. Comparison among MLP, RF and NB using t-test confirms that NB is the best among these three (Fig. 4).

vii. ROC curve of ZeroR (base line), NB, RF and MLP are presented in fig. 5. Area under ROC for ZeroR, RF, MLP and NB are 0.5, 0.9547, 0.8946 and 0.8926, respectively. As area represents the efficiency of the algorithm, hence it can be concluded that NB outperforms the other algorithms.

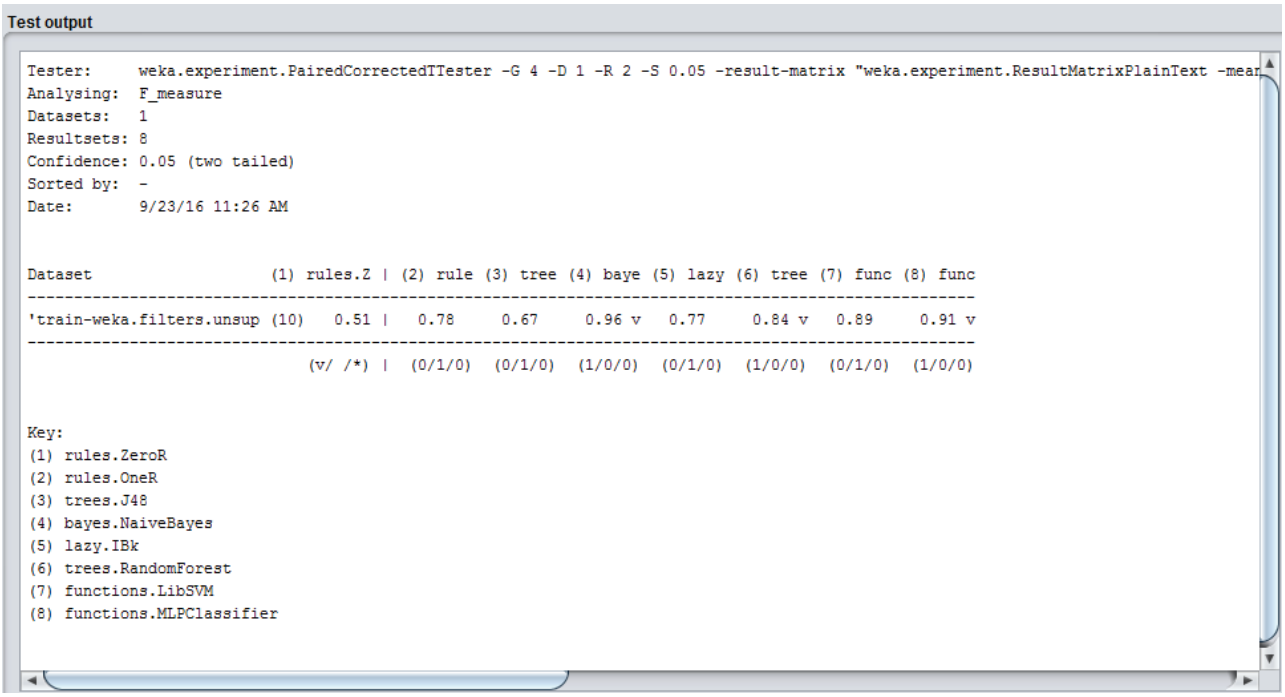

Fig. 3. Comparing performance of algorithms based on F-Measure using paired t-test

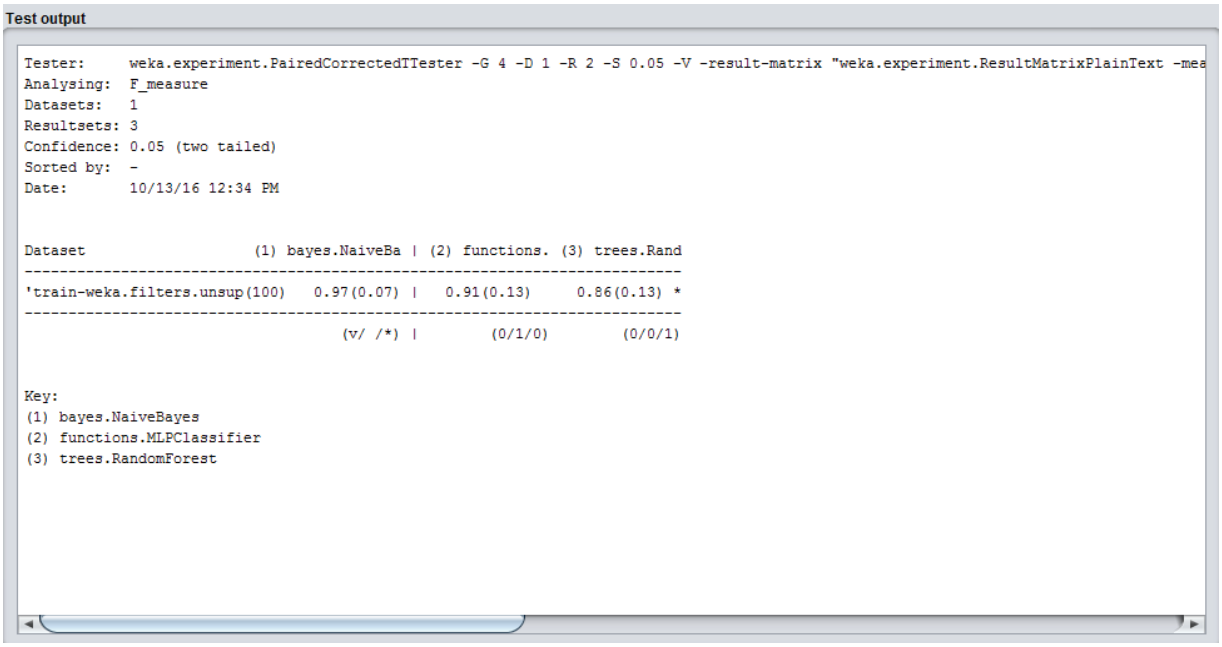

Fig. 4. Comparison of Naïve Bayes, Random Forest and Multilayer Perceptron using paired t-test

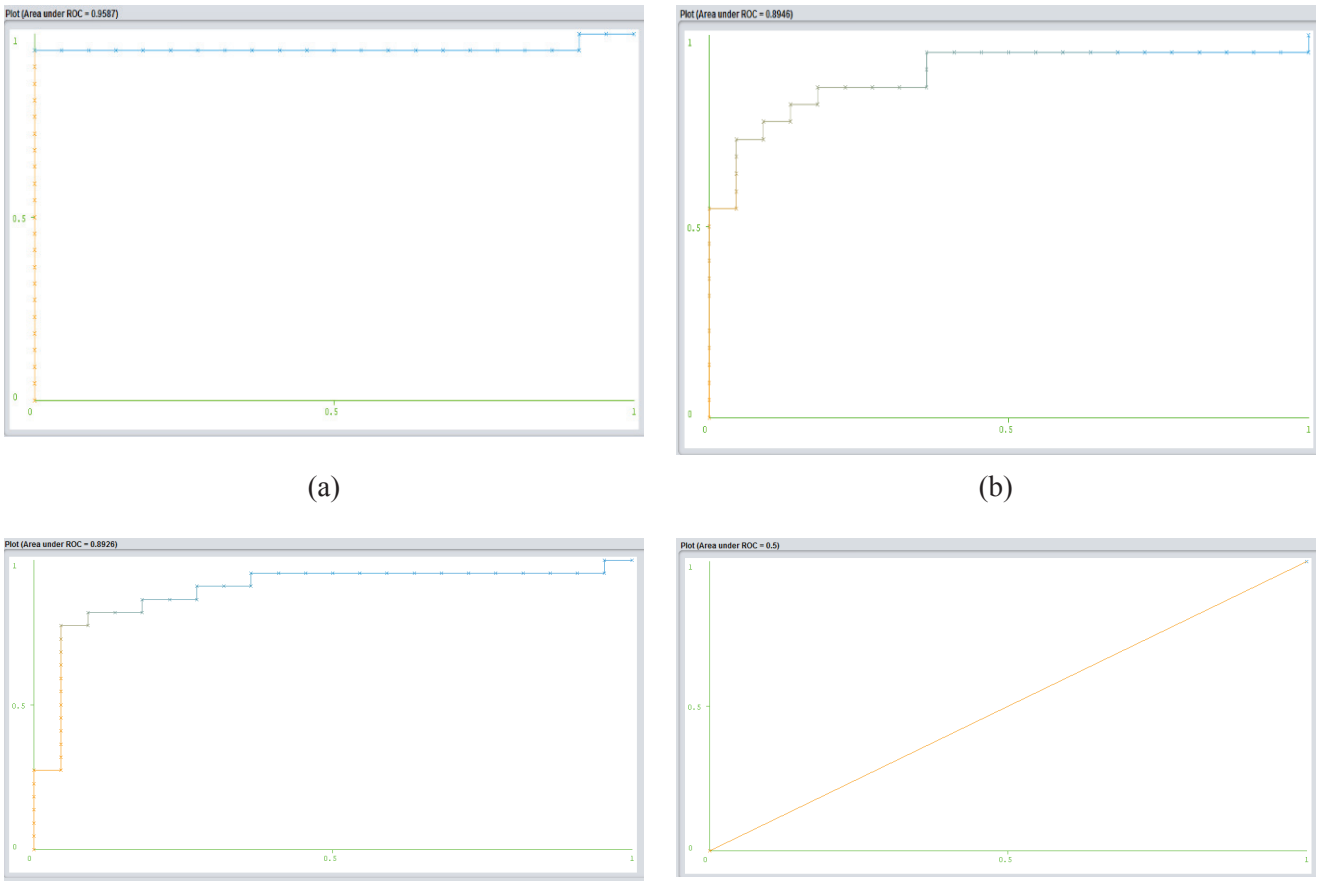

Fig. 5. Roc curves of (a) Naïve Bayes, (b) Random Forest, (c) Multilayer Perceptron, (d) ZeroR

7. CONCLUSION AND FUTURE WORK

This paper presented a framework of text categorization using various machine learning algorithms in agricultural context. In the experiment, ZeroR, OneR, J4.8 Decision Tree, Naïve Bayes, KNN, Random forest, Support Vector Machine (SVM), and Multilevel Perceptron based algorithms have been used for TC. We have compared performance of the algorithm(s) using F-measure and ROC analysis. Experimental results indicated that Naïve Bayes outperforms all the other algorithms in terms of all the measures of performance evaluation. In this paper, titles have been used to categorize documents. However, in future, bigger data set that is abstracts or full texts will be used for TC.

REFERENCES

Bickel, S., and Tobias S. (2007). "Dirichlet-enhanced spam filtering based on biased samples." Advances in neural information processing systems.

Ceci, M. and Malerba, D. (2007). "Classifying web documents in a hierarchy of categories: a comprehensive study." *J. Int. Inf. Syst.* 28.1: 37-78.

Felcher, E. M., Malaviya, P. and McGill, A. L. (2001) "The role of taxonomic and goal□derived product categorization in, within, and across category judgments." *Psychology & Marketing* 18.8: 865-887.

Holmes, D. I., and Richard S. F. (1995). "The Federalist revisited: New directions in authorship attribution." *Literary and Linguistic computing* 10.2: 111-127.

Kotsiantis, S. B., Zaharakis, I. D., and Pintelas, P. E. (2006). "Machine learning: a review of classification and combining techniques." *Artificial Intelligence Review* 26.3: 159-190.

Kuroiwa, T, and Bhalla, S. (2007). "Dynamic personalization for book recommendation system using web services and virtual library enhancements." *Comp. Inf. Tech.*, 2007. CIT 2007. 7th IEEE International Conference on. IEEE.

Pang, B, and Lee, L. (2004). "A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts." Proceedings of the 42nd annual meeting on Association for Computational Linguistics. Association for Computational Linguistics.

Sebastiani, F. (2001). "Organizing and using digital libraries by automated text categorization." Proceedings of the AI* IA workshop on artificial intelligence for cultural heritage and digital libraries.

Spink, A., Khopkar, Y. Shah, P., Debnath, S. (2003) "Search engine personalization: An exploratory study." First Monday 8.7.

Witten, I. H., Frank, E. (2005). Data Mining: Practical machine learning tools and techniques. Morgan Kaufmann.

Zhang, Y. (2004). "Using bayesian priors to combine classifiers for adaptive filtering." Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval. ACM.

Zhu, F., and Zhang, X. (2006). "The influence of online consumer reviews on the demand for experience goods: The case of video games." ICIS 2006 Proceedings: 25.