International Journal of Data Mining Techniques and Applications
Volume: 03 Issue: 01 June 2014, Page No. 24- 29
ISSN: 2278-2419

Performance Analysis of Selected Classifiers in User Profiling

ManasRanjan Patra[1],V.Mohan Patro[2]
Department of Computer Science, Berhampur University,Berhampur-760007,Odisha, India
Email- mrpatra12@gmail.com , vmpatro@gmail.com

Abstract-User profiles can serve as indicators of personal preferences which can be effectively used while providing personalized services. Building user files which can capture accurate information of individuals has been a daunting task. Several attempts have been made by researchers to extract information from different data sources to build user profiles on different application domains. Towards this end, in this paper we employ different classification algorithmsto create accurate user profiles based on information gathered from demographic data. The aim of this work is to analyze the performance of five most effective classification methods, namely Bayesian Network(BN), Naïve Bayesian(NB), Naives Bayes Updateable(NBU), J48, and Decision Table(DT). Our simulation results show that, in general, the J48has the highest classification accuracy performance with the lowest error rate. On the other hand, it is found that Naïve Bayesian and Naives Bayes Updateable classifiers have the lowest time requirement to build the classification model.

Keywords–Classifiers; data mining; Confusion matrix; evaluation

I. INTRODUCTION

A user profile refers to the explicit digital representation of a person's identity. A user profile can also be considered as the computer representation of a user model. User profiles describe user interests, characteristics, behavior and preferences. A user profile is a set of information about a given user in a given context and on a specific period of time [1]. User profiling is the practice of gathering, organizing and interpreting the user profile information.It refers to construction of a profile through the extraction from a set of data. User profiles can be found on recommender systems, or dynamic websites (such as online social networking sites or bulletin boards). Behind every instance of personalization, there is a profile that stores the user preferences, context of use and other information that can be used to deliver a user experience tailored to the individual needs and preferences[2].Recommendation systems are widely adopted in e-commerce businesses for helping customers locate products they like to purchase and the major challenge for these systems is bridging the gap between the physical characteristics of data with the users' perceptions [3]. In order to address this challenge, employing user profiles to improve accuracy become essential. Targeting customers, based on inaccurate user profile data will not be as effective as targeting customers based on accurate user profile data. Thus, companies perform preprocessing on user profile data as part of effort to maintain the accuracy of their user profile data[4].Yet another application which can benefit from user profiling is E-governance wherein personalized online services can be provided to citizens. User profiling would help e-Gov service systems communicate effectively and efficiently with their users. With the provision of user profiles, the systems can recognize the user every time he/she logs on to the system and the user does notneed to enter the same personal information again. This will make users' interactions with the system easier and faster[5].Several techniques are being applied to build user profiles which can be effectively used in different application domains to provide customized services. Among others classification is one of the widely used techniques which have been successfully applied to classify users based on information gathered from different sources. However, the performance of these classification techniques greatly vary in terms of how accurately they classify the data for building accurate user profiles.

In [6], Alicia *et al.* compared the performance of NB, J48 and DT showing the correctly classified instances rate, time taken to build the model, area under curve etc. In [7], Cufoglu*et al.* compared Naïve Bayesian Tree (NBTree), Sequential Minimal Optimization (SMO), Naïve Bayesian(NB), Instance-Based Learner(IB1), J48 (a version of C4.5), Classification and Regression Tree (SimpleCART) and Iterative Dichotomiser Tree (ID3) classifiers in the personalizationapplications.In [6],using an automobile data set, the authors have obtained only upto 73% of correctly classified instances rate whereas in our case using a user profile data set we could achieve more than 95%. For both the data sets the time taken to build the model is the highest in case of Decision Table classifier in comparison to other classifiers used in the experimentation.

From [7] and our observation it is clear that there is no substantial increase in the time taken to build the model when the number of instances is within 1000. However, the time taken to build the model is always in increasing order when the number of instances is in thousands.In [8], Panda *et al.* compared the performance of NB, Id3 and J48 algorithms for network intrusion detection. According to the authors, NB performs better than ID3 and J48 with respect to overall classification accuracy. However, the authors added that Decision Trees (ID3 and J48) are robust in detecting new intrusive behaviours in comparison to NB.In our work, we have applied all the classifiers available in the WEKA(Waikato Environment for Knowledge Analysis) workbench[9] on the user profile dataset and observed that 20 classifiers resulted in classification accuracy above 98%. Out of these 20 classifiers we have selected the top 5 classifiers, namely, Bayesian Network (BN), Naïve Bayesian (NB), Naives Bayes

International Journal of Data Mining Techniques and Applications
Volume: 03 Issue: 01 June 2014, Page No. 24- 29
ISSN: 2278-2419

Updateable (NBU), J48 (Java implementation of C4.5), and Decision Table (DT) for further analysis.

II. CLASSIFICATION TECHNIQUES

Bayesian networks are probability based and are used for reasoning and decision making under uncertainty, and heavily rely on Bayes' rule which is defined as follows [10];

- Assume A_i attributes where i= 1,2,3,…,n, and which take values a_i where i= 1,2,3,…,n.
- Assume C as class label and $E = (a_1, a_2,.. a_n)$ as unclassified test instance.
- E is classified into class Cwith the maximum posterior class probability$P(C \mid E)$,

$$P(C \mid E) = \arg\max_c P(C)P(E \mid C)$$

Bayesian Networks can represent uncertain attribute dependencies; however it has been proven that learning optimalBayesian network is NP (Non-deterministic Polynomial) hard [11]. Naïve Bayesian Classifier is one of the Bayesian Classifier techniques whichis also known as the state-of-the-art of Bayesian Classifiers. In many works it has been proven that Naïve Bayesian classifiers are one of the most computationally efficient, effective and simple algorithms for machine learning and data mining applications [12][13][14]. Naïve Bayesian classifiers assume that all attributes within the same class are independent, given the class label. Based on this assumption, the Bayesian rule has been modified as follows to define the Naïve Bayesian rule [10];

$P(C|E) = \arg \max_c P(C) \prod_{i=1}^{n} P(A_i| C)$Naïve Bayesian classifiers are normally used for interactive applications. However, because of its Naïve conditional in dependence assumption, optimal accuracy cannot be achieved.Naïve Bayes Updateable is the updateable version of Naïve Bayes in WEKA (Waikato Environment for Knowledge Analysis) work bench. This classifier uses a default precision of 0.1 for numeric attributes when build Classifier (a method in java code) is called with zero training instances.J48 is an open source Java implementation of the C4.5 algorithm in the WEKA data mining tool. C4.5 builds decision trees from a set of training data in the same way as ID3, using the concept of information entropy[15].

The training data is a set $S = s_1, s_2, \ldots$ of already classified samples. Each sample s_i consists of a p-dimensional vector $(x_{1,i}, x_{2,i}, \ldots, x_{p,i})$ wherex_j represent attributes or features of the sample, as well as the class in which s_i falls.Decision table is based on logical relationships just as the truth table. It is a tool that helps us to look at the combination of both completeness and inconsistency of conditions[6]. Decision tables, like decision trees or neural nets, are classification models used for prediction. They are induced by machine learning algorithms. A decision table consists of a hierarchical table in which each entry in a higher level table is decomposed into values of a pair of additional attributes to form another table. The structure is similar to dimensional stacking.

III. EXPERIMENTAL SETUP

A. Data Set

For our experimentation we have used Census Income data set[16]which has 15 different attributes, namely, Age, Work-class, Final-weight, Education, Education-num, Marital-status, Occupation, Relationship, Race, Sex, Native-country, capital-gain, capital-loss, Hours-per-week & Income. This data set has been used by different authors for different purposes. It has been used to classify if the income of a person is greater than 50K based on several census parameters, such as age, education, marital status.

B. Kappa Statistics (KS)

Kappa is a chance-corrected measure of agreement between the classification and the true classes. Kappa statistic is used to access the accuracy of any particular measuring cases. It is used to distinguish between the reliability of the data collected and their validity. The value of kappa is less than or equal to 1 where the value of 1 indicates perfect agreement.

C. Cross-Validation

Cross validation calculates the accuracy of the model by separating the data into two different populations - a training set and a testing set. The model is created from the training set and its accuracy is measured based on how well it classifies the testing set. This testing process is continued k times to complete the k-fold cross validation procedure. We have used 10-fold cross-validation in which the available data are randomly divided into 10 disjoint subsets of approximately equal size. One of the subsets is then used as the test set and the remaining 9 sets are used for building the classifier. This test set is then used to estimate the accuracy. This is done repeatedly 10 times so that each subset is used as a test subset once. The accuracy estimate is then the mean of the estimates for each of the classifiers.

D. Confusion Matrix

A confusion matrix (*also* known as a contingency table or an error matrix) is a table layout that allows visualization of the performance of a supervised learning algorithm. Following confusion matrix is for 2 class values of an attribute.

Table-1

		Predicted Class	
		C_1	C_2
Actual Class	C_1	True positive	False negative
	C_2	False positive	True negative

C_1 – particular class C_2 – different class

True positive (TP) - The number of instances correctly classified as C_1

International Journal of Data Mining Techniques and Applications
Volume: 03 Issue: 01 June 2014, Page No. 24- 29
ISSN: 2278-2419

True negative (TN) - The number of instances correctly classified as C_2

False positive (FP) - The number of instances incorrectly classified as C_1 (actually C_2)False negative (FN) - The number of instances incorrectly classified as C_2 (actually C_1)

P = Actual positive = TP + FN

P^1 = Predicted positive =TP + FP

N = Actual negative = FP + TN

N^1 = Predicted negative =FN + TN

TP rate = Sensitivity = TP / P

TN rate = Specificity = TN / N

FP rate = selectivity =1 – TN rate = FP / N

Precision = TP / P^1

TP rate = % of correctly classified instances

Accuracy = (TP + TN) / (P + N)

= TP / P * P / (P + N) + TN / N * N / (P + N)

=Sensitivity * P / (P + N) + Specificity * N / (P + N)

Here, it can be noted that false negative cases are risky. For example, in case of medical diagnosis, if a cancerous patient is incorrectly predicted as non-cancerous patient, it can be a major mistake. Similarly, in case of student examination results a student who has passed an examination can be incorrectly indicated as fail. Therefore, it is highly essential that false negatives should be minimized as far as possible.Further, we define two risk measures for a classifier, namely, high risk rate and low risk rate which are defined as follows:High risk rate is defined as the ratio of false negative to predicted negative (i.e., FN / N^1) and Low risk rate is defined as the ratio of false positive to predicted positive (i.e., FP / P^1).This can be clear from the following instance of confusion matrix, where the class attribute "result" is having 4 class values, i.e. FAIL, PASS, 2^{nd} and 1^{st}.

TABLE 2- Instance Of Confusion Matrix

a=FAIL	b=PASS	c = 2ND	d = 1ST	classified as
2556	0	0	0	**a = FAIL**
2	17274	1398	47	**b = PASS**
0	590	5847	850	**c = 2ND**
0	0	896	3794	**d = 1ST**

This is an instance of result got from classifier Naïve Bayes in WEKA work bench for a real examination system having 14 attributes, out of which "result" is one attribute. This data file is having 33254 instances. Here it is observed that 2 of "pass" students are predicted as "fail". This is the case of false negative, which is of high risk.

IV.SIMULATION AND EVALUATION

We compare the results of five different classifiers (BN,NB,NBU, J48 and DT). Simulations were conducted using demographic profile data provided by UCI's census dataset. The data set consists of 15 attributes. Several rounds of simulation were carried out using different instances of data, namely,1000, 2000, 3000 & 4000 on a Intel(R) Core(TM)2 Duo machine with 1.83GHz, and 1.99GB RAM capacity. All simulations were performed in the WEKA (Waikato Environment for Knowledge Analysis) machine learning platform that provide a workbench which consists of collection of implemented popular learning schemes that can be used for practical data mining and machine learning works. We chose 10 fold cross-validations as a test mode where 10pairs of training sets and testing sets are created. The selected classification algorithms were run on thesame training data sets and were tested on the same testingsets to obtain the classification accuracy. The performance comparison under different cases is presented below.

A. Calculation for time to build the model

It takes some timeto build the model for a classifier, which depends on number of instances and the classifier as depicted in figures 1 and 2.TABLE II. Time Taken To Build The Model Using Different Instances Of The User Profile Dataset

Table-3

Time analysis		Classifiers				
		BN	*NB*	*NBU*	*J48*	*DT*
Instances	*1000*	0.08	0.02	0.02	0.09	0.56
	2000	0.08	0.02	0.02	0.09	1.05
	3000	0.08	0.02	0.02	0.16	1.41
	4000	0.08	**0.03**	**0.03**	0.19	1.77

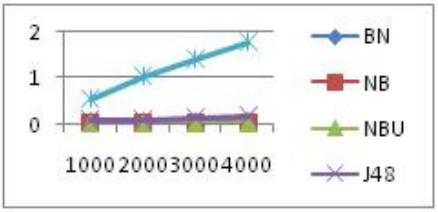

Figure-1

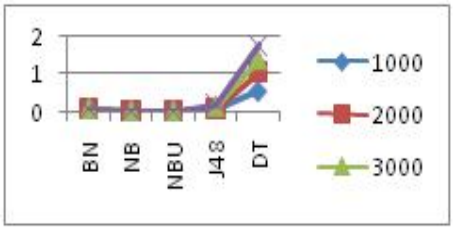

Figure-2

Figure 1&2 - Time taken in seconds to build the model vs. number of instances in user profile datasetAccording to Table-II, time analysis shows that DT takes maximum time to build the model, whereas NB and NBU take minimum time. Increasing the data instances gradually we found that the increase in model building time is much higher in case of DT in comparison to BN, NB, NBU, and J48.

International Journal of Data Mining Techniques and Applications
Volume: 03 Issue: 01 June 2014, Page No. 24- 29
ISSN: 2278-2419

B. Calculation of CCI rate

Once a model is built for the classifier, the rate of correctly classified instances (CCI) can be computed which isthe ratio of correctly classified instances to the total number of instances in the data set.

TABLE-4. Correctly Classified Instances (Cci) Rate Of The Classifiers For Different Instances Of User Profile Dataset

CCI rate in %		Classifiers				
		BN	**NB**	**NBU**	**J48**	**DT**
Instances	1000	95.9	97.1	97.1	99.8	99.8
	2000	98.6	98.65	98.65	99.9	99.9
	3000	98.77	99.03	99.03	100	100
	4000	99.03	99.15	99.15	**100**	**100**

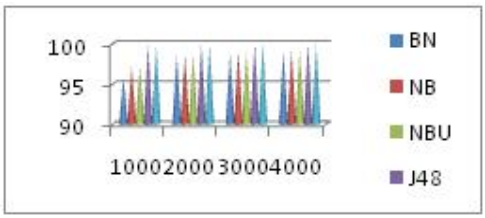

Figure-3

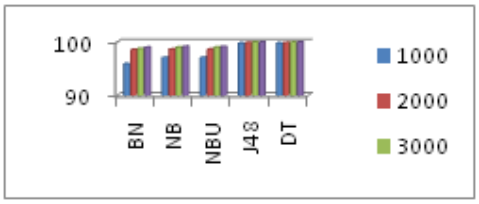

Figure-4

Fig 3 & 4 - Correctly Classified Instances rate (in %) of five classifiers for different instancesAs depicted in Table-III, theCorrectly Classified Instances rate or TP rate in cases of J48 and DT are higher whereas it is the lowest for BN. However, for NB and NBU it has the same values, which is also the case for J48 and DT. It is further observed that increasing the data instances the rate is also increasing in all classifiers as presented in figures 3 and 4.

C. Calculation of KS, MAE and RAE

TABLE -5 .Ks, Mae & Rae For 4th Data Set Having 4000 Instances

classifiers	KS	MAE	RAE
BN	0.9879	0.005	4.9409
NB	0.9894	0.0018	1.7679
NBU	0.9894	0.0018	1.7679
J48	**1**	**0**	**0**
DT	1	0.0066	6.5745

Table-5 provides the values for Kappa Statistics (KS), Mean Absolute Error (MAE) and Relative Absolute Error (RAE) for all the five classifiers. It is clear that J48 gives accurate values as KS is 1, MAE is 0 and RAE is 0. Next to it is DT.

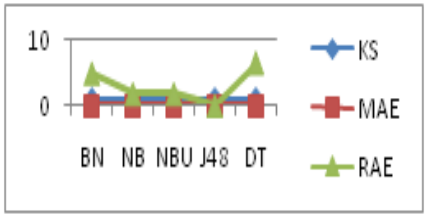

Fig 5 for Table IV

TABLE 6-A. Ks Of 3 Data Sets For 5 Classifiers

		KS for Instances		
		1000	***2000***	***3000***
Classifiers	***BN***	0.9488	0.9825	0.9847
	NB	0.9639	0.9831	0.988
	NBU	0.9639	0.9831	0.988
	J48	0.9975	0.9987	1
	DT	0.9975	0.9987	1

TABLE 6-B. Mae Of 3 Data Sets For 5 Classifiers

		MAE for Instances		
		1000	***2000***	***3000***
classifiers	***BN***	0.0122	0.0074	0.0059
	NB	0.0049	0.0027	0.002
	NBU	0.0049	0.0027	0.002
	J48	0.0002	0.0001	0
	DT	0.0189	0.0113	0.0084

TABLE 6-C.Rae Of 3 Data Sets For 5 Classifiers

		RAE for Instances		
		1000	***2000***	***3000***
classifiers	***BN***	12.076	7.4191	5.8911
	NB	4.8186	2.7255	1.9934
	NBU	4.8186	2.7255	1.9934
	J48	0.2428	0.1249	0
	DT	18.8063	11.2926	8.3107

It is evident from Table-V and figures 6, 7 and 8 that the values of KS tend to 1 as the number of instances increases, whereas the error values approach 0 in case of MAE and RAE.

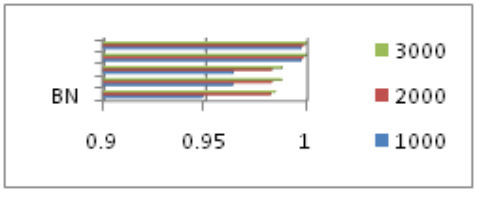

Fig 6 - Kappa Statistics for classifiers

International Journal of Data Mining Techniques and Applications
Volume: 03 Issue: 01 June 2014, Page No. 24- 29
ISSN: 2278-2419

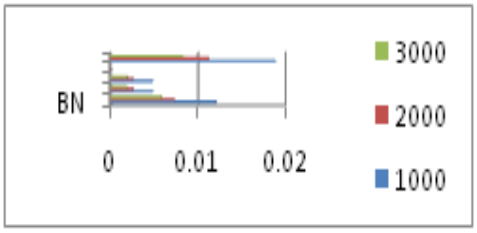

Fig 7 - Mean Absolute Errors in classifiers

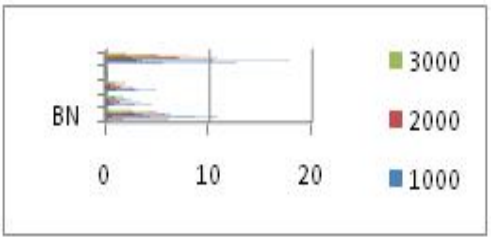

Fig 8 - Relative Absolute Errors in classifiers

D. Comparison of performances with the complete data set

Here, we have used the complete data set for our experimentation after removing some irrelevant attributes and compute time to build the model, correctly classified instances (CCI), KS, MAE and RAE. Next, attributes like capital-gain, which is less relevant to education (class attribute) has been removed and values are calculated again which shows improvement in the results. In this way other 2 attributes (capital-loss & race) are removed. Table-VI presents the values for the complete data set (CDS) with all attributes, and table VII presents the values obtained in case of a reduced data set (RDS), after eliminating 3 attributes from CDS. Further, a comparison of values obtained for CDS and RDS is depicted in figures 9 and 10 which clearly indicate an improvement when irrelevant attributes are removed from the data set.

TABLE 7 – VALUES OF CDS FOR 4 CLASSIFIERS

Classi-fiers	Time	CCI %	KS	MAE	RAE
BN	1.73	99.8177	0.9977	0.0012	1.2107
NB	0.5	99.4728	0.9935	0.0012	1.2248
J48	2.05	100	1	0	0
DT	16.99	100	1	0.0011	1.048

TABLE 8– VALUES OF RDS FOR 4 CLASSIFIERS

Classi-fiers	Time	CCI	KS	MAE	RAE
BN	0.94	99.874	0.9984	0.0011	1.0531
NB	0.31	99.9536	0.9994	0.0006	0.5495
J48	1.67	100	1	0	0
DT	12.75	100	1	0.0011	1.048

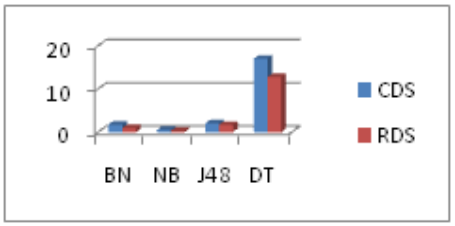

Fig 9 - Time analysis of classifiers for CDS vs RDS

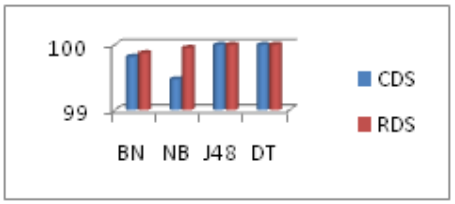

Fig 10 - CCI analysis of classifiers for CDS vs RDS

V.CONCLUSION

In this work, we evaluated the performance of five classifiers namely, Bayesian Network, Naïve Bayesian, Naives Bayes Updateable, J48, and Decision Table in terms of their accuracy. Simulations were performed in the WEKA machine learning platform. The objective of the work was to find the best classification algorithm that has high classification accuracy while building user profiles. According to the simulation results J48 classifier performs much better on user related information with least errors. Furthermore, Naïve Bayesian and Naives Bayes Updateable have taken least time to build the model. This indicates that J48 classification algorithm should be favored over Bayesian Network, Naïve Bayesian, Naives Bayes Updateable, and Decision Table classifiers in the personalization applications where classification accuracy performance is important.

REFERENCES

[1] Teixeira, C., Pinto, J.S., Martins, J.A., "User Profiles in Corporate Scenarios", IEEE, 2004, pp. 614-619.

[2] Bartolomeo, G.,Kovacikova, T., "User Profile Management in Next Generation Networks", AICT' 09. Fifth Advanced International Conference onTelecommunications, 2009, pp. 129-135.

[3] Damian Fijałkowski, RadosławZatoka, "An architecture of a Web recommender system using social network user profiles for e-commerce", IEEE, 2011, pp. 287–290.

[4] Sung Hyuk Park, Sang Pil Han, Soon Young Huh, Hojin Lee, "Preprocessing Uncertain User Profile Data: Inferring User's Actual Age from Ages of the User's Neighbors", ICDE '09, IEEE 25th International Conference onData Engineering, 2009, pp. 1619-1024.

[5] Malak Al-hassan, Haiyan Lu, Jie Lu, "A Framework for Delivering Personalized e-Government Services from a Citizen-Centric Approach", Proceedings of iiWAS2009, Kuala Lumpur, Malaysia, 2009, pp. 436-440.

[6] Alicia Y.C. Tang, NurHananiAzami, Norfaezah Osman, "Application of Data Mining Techniques in Customer Relationship Management for An Automobile Company", Proceedings of the 5th International Conference onIT & Multimedia at UNITEN (ICIMU 2011) Malaysia, IEEE, 2011.

[7] Cufoglu A., Lohi M.,Madani K., "A Comparative Study of Selected Classifiers with Classification Accuracy in User Profiling", Seventh International Conference on Machine Learning and Applications, IEEE, 2009, pp. 787-791.

[8] Panda M., Patra M. R., "A comparative study of data mining algorithms for network intrusion detection", 1st International Conference on Emerging Trends in Engineering and Technology,IEEE, 2008, pp. 504-507.
[9] Wekahomepage.Availableat:http://www.cs.waikato.ac.nz/ml/weka/accessed on 24/12/11.
[10] Jensen F.V., "Introduction to Bayesian Networks", Denmark, Hugin Expert A/S, 1993.
[11] Jiang L.,Guo Y., "Learning lazy Naïve Bayesian classifier for ranking", Proceedings of the 17th IEEE International Conference on Tools with Artificial Intelligence (ICTAI'05), IEEE, 2005, Page(s) 5pp.
[12] Wang Z., I. Geoffrey Webb, "Comparison of lazy Bayesian rule and tree-augmented Bayesian learning", Paper presented at the IEEE Conference on Data Mining, ICDM 2002, Page(s): 490 – 497.
[13] Shi Z., Huang Y., Zhang S., "Fisher score based naive Bayesian classifier". Paper appears in ICNN&B International Conference on Neural Networks and Brain, IEEE 2005, pp. 1616-1621.
[14] Santafe G., Lozano J.A., Larranaga P., "Bayesian model averaging of Naive Bayes for clustering", IEEE Transactions on Systems, Man, and Cybernetics, 2006, Vol. 36, No. 5, pp. 1149 – 1161.
[15] http://en.wikipedia.org/wiki/C4.5_algorithmaccessed on 21/03/13.
[16] Asuncion A. and Newman D.J. (2007) UCI Machine Learning Repository Irvine, CA: University of California, School of Information and Computer Science. [Online] Available from: http://www.ics.uci.edu/~mlearn/MLRepository.htmlaccessed on 11/02/13.