A Literature Survey on Contrast Data Mining

Dinkal Shah[1], **Narendra Limbad**[2]

[1, 2]Department of Computer Engineering, Gujarat Technological University, Gujarat, India

Abstract: ***Data mining refers to extracting or "mining" knowledge from large amount of data. Data mining also called KDD (Knowledge Discovery). There are some types of classification techniques : Decision Tree, Naïve Bayesian, Neural Network, Contrast Data Mining. Contrast Data mining is a new approach in data mining. Various types of contrast pattern are described. Contrast data mining is the mining of patterns and models contrasting two or more classes or conditions. The ability to distinguish, differentiate and contrast between different data sets is a key objective in data mining. Emerging patterns are sets of items whose frequency changes significantly from one dataset to another. HGEP strategy has higher accuracy than the NEP strategy.***

Keywords: Contrast data Mining, EP, NEP, HGEP.

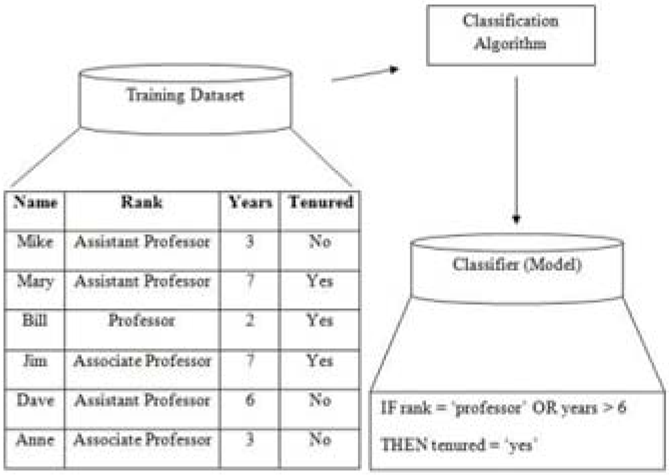
1. Introduction

1.1 Classification

Classification is supervised learning method use labelled training data, in which class label of each training data is known in advance and new data is classified based on the training set is known as supervised learning [1].

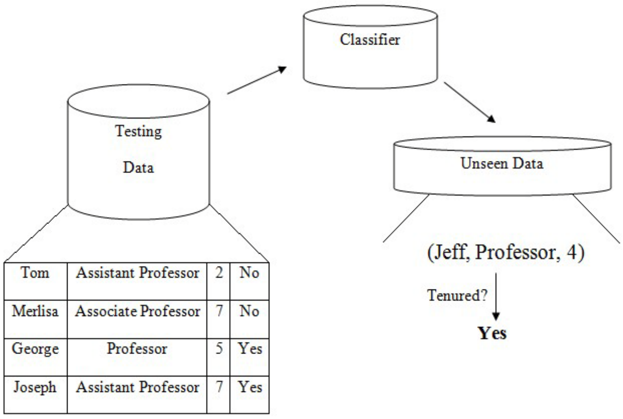
1.2 Classification Process

Classification is a method for classifying the rule in between predictor attribute and class label attribute.



Name	Rank	Years	Tenured
Mike	Assistant Professor	3	No
Mary	Assistant Professor	7	Yes
Bill	Professor	2	Yes
Jim	Associate Professor	7	Yes
Dave	Assistant Professor	6	No
Anne	Associate Professor	3	No

Figure 1: Model Construction (Learning)



Tom	Assistant Professor	2	No
Merlisa	Associate Professor	7	No
George	Professor	5	Yes
Joseph	Assistant Professor	7	Yes

Figure 2: Model Usage (Classification)

In first step, data are known and with the help of these data we create the classification rule and in second step test the data and directly got the output. Some of the applications of classification are Routing in telecommunication networking, travelling, schedule etc.

1.3 Different approaches of classification

Here, let's have a brief look at various classification approaches such as Decision Tree Induction, Naïve Bayesian Classification, Artificial Neural Network, Contrast Data Mining.

1. Decision Tree

Decision tree induction is the learning of decision trees from class – labelled training tuples. A decision tree is a flowchart like tree structure, where each internal node (non leaf node) denotes a test on an attribute, each branch represents an outcome of the test, and each leaf node (or terminal node) holds a class label [1]. The topmost node in a tree is the root node. It is constructed top down tree structure with the help of a greedy algorithm. The attribute that has highest in formation gain is selected. The selected attribute become root node. A branch is created for every value of the attribute. The same process of the algorithm is used recursively on each branch to form a decision tree once an attribute has appeared in a node, it is not considered again in any of the node's decedents. After completion of the tree, every path from the root to leaf node becomes a rule. The leaf represents the class label of the rule[1].

2. Naive Bayesian

It is an approach that learns from probabilistic knowledge. It is a statistical classification technique based on Bayes theorem. It has been used to generate impressive results and is easy to program and fast to train. It calculates the probabilities of a given sample belonging to different classes [1]. Naïve Bayes is a special kind of Bayesian network that has been commonly used for data classification. Its predictive performance is comparable with other commonly used classifiers such as CN2. Bayes classifier learns the class conditional probabilities of each attribute from supervised training data with the help of Bayes' theorem. A test sample is then assigned a class that has the highest probability.

3. Neural Network

It is learn the classification rules by layered graph with output of one node feeding into one or many other nodes in the next layer. It consists of an interconnected group of artificial neurons and processes information using a connectionist approach to computation.

4. Contrast Data Mining

Contrasting is one of the most basic types of analysis and is used by all types of people. It is routinely employed to help us understand the world and to better deal with the problems and challenges we face. Contrast data mining is one of the classification approaches which involve the mining of patterns and models contrasting two or more classes/ conditions.

2. Survey on Contrast Data Mining

a)Contrast Data Mining

Contrasting involves the comparison of one set/kind/class of objects against another set/kind/class. Usually, we contrast given classes of objects in order to identify the differences that exist between them. These differences can provide useful in sights on how, and perhaps also why, the objects differ. The ensuing understanding gained from the how and why can then help guide us on how to use different objects in an appropriate way.

Contrast data mining is the mining of patterns and models contrasting two or more classes/conditions. Before the age of computers, techniques for contrasting sets of objects were based on traditional statistical methods, such as comparison of the respective means of the features of the objects in the two sets, or comparison of the respective distributions of attribute values. These approaches can be limited, since it may be difficult to use them for identifying specific patterns in the data that offer novel and actionable insights.

b)Emerging Pattern

Given two or more data sets contrast patterns are patterns that describe significant differences between the given datasets. A pattern is considered as describing differences between the two data sets if some statistics (e.g., support or risk ratio) for with respect to each of the datasets are highly different.

We often refer to the dataset/class where a pattern has the highest frequency as it is home data set/class. Many names have been used to describe contrast patterns, including emerging patterns, contrast sets, group differences, patterns characterizing change, classification rules and discriminating patterns [2].Contrast data mining can also be applied to many types of data, including vector data, transaction data, sequence data, graph data, image data and data cubes [2]. Emerging Patterns are those whose frequencies change significantly from one data set to another. They represent strong contrast knowledge. It is the new type of knowledge pattern that describes significant changes (differences or trends) between two classes of data. An Emerging Pattern is an item set whose support in one set of data differs from its support in another[2].

Following are the various types of Emerging Patterns which are proposed till now:

1. ρ-Emerging Patterns (ρ-EP) [4]

Given two different classes of datasets D1 and D2 , the growth rate of an item set X from D1 to D2 is defined as

$$GR(x)=\begin{cases}0, & \text{if } \text{supp}_1(x)=0 \ \&\text{supp}_2(x)=0\\ \infty, & \text{if} \text{supp}_1(x)=0 \ \&\text{supp}_2(x)>0\\ \dfrac{\sup_2(x)}{\sup_1(x)}, & \text{otherwise}\end{cases}$$

Emerging Patterns are those item sets with large growth rates from D1 to D2.
Given a growth rate threshold ρ>1,an item set X is said to be a ρ-Emerging Pattern (ρ -EP or simply EP) from a background dataset D1 to a target dataset D2 if GR(X)>=ρ . When D1 is clear from the context, an EP X from D1 to D2 is simply called an EP of D2 or an EP in D2.The support of X in D2,supp2 (X),denoted as supp(X), is called the support of the EP. The background data set D1 is also referred to as the negative class, and the target dataset D2 as the positive class. An EP with high support in its home class and low support in the contrasting class can be seen as a strong signal indicating the class of a test instance containing it. The strength of such a signal is expressed by its supports in both classes and its growth rate. [4]

2. Jumping Emerging Patterns (JEP) [4]

The strength of an EP X is defined as

$$strength(x)=\frac{GR(x)}{GR(x)+1}*\sup(x)$$

A Jumping Emerging Patterns (JEP) is a special type of Emerging Pattern and also a special type of discriminate rule.
A Jumping Emerging Pattern (JEP)from a background dataset D1 to a target dataset D2 is defined as an Emerging Pattern from D1 to D2 with the growth rate of ∞. Note that for a JEP X, strength(X)=supp(X).

3. Essential Jumping Emerging Patterns (EJEP) [4]

EJEPs are defined as minimal item sets whose supports in one data class are zero but in another are above a given support threshold ξ. Given ξ >0 as a minimum support threshold, an Essential Jumping Emerging Pattern(EJEP) from D1 to D2, is an item set X that satisfies the following

Conditions:
1. suppD1(X) =0 and sup D2(X) >ξ, and
2. Any proper subset of X does not satisfy condition 1.

When D1 is clear from context, an EJEP X from D1 to D2 is simply called an EJEP of D2. The support of X inD2, suppD2(X), is called the support of the EJEP, denoted as supp(X). It is obvious that EJEPs also have infinite growth rates, which indicates they have strong predictive power. Their JEPs from D1to D2 are the item sets whose supports in D1 are zero but in D2 is non-zero. In condition1, we further require the supports in D2 to be above a minimum support threshold ξ, which makes an EJEP cover at least a certain number of instances in a training dataset. Condition2 shows that any proper subset of an EJEP is not an EJEP anymore, which means EJEPs are the shortest JEP.[4]A JEP, by definition, is not necessarily the shortest. A shorter JEP means fewer items (attributes).

If we can useless attributes to distinguish two data classes, adding more attributes will not contribute to classification, and even worse, bring noise when classifying by aggregating JEPs. Supersets of EJEPs are not useful in classification because of the following reason. Let E1 and E2 be two different item sets satisfying condition1, and E1E2. E1covers more (at least equal) instances of the training dataset than E2, because supp(E1) > sup(E2).

4. Chi-Emerging Patterns (Chi-EP) [4]

We say that an item set, X ,is a Chi Emerging Pattern (Chi EP), if all the following conditions about X are true:

a) Supp(x) >= ξ, where ξis a minimum support threshold;
b) GR(x) >=ρ, where ρ is a minimum growth rate threshold;
c) It has larger growth rate than its subsets;
d) It is highly correlated according to common statistical measures such as chi-square value. Length-1 item sets, that satisfy the above three conditions, pass chi-square test directly.

5. Noise Tolerant Emerging Patterns (NEP) [6]

According to different types of the training data, the strategies of the EPs can be divided into two categories, i.e., the EPs with the infinite growth rate and the EPs with the finite growth rate. The EJEP strategy only cares about those item sets with the infinite growth rate. It ignores those patterns which have very large growth rates, although not infinite, i.e., the so called “noise”. However, the real-world data always contains noises and the NEP strategy considers noises and provides higher accuracy than the EJEP strategy. EJEPs allow noise tolerance in dataset D2. However, real-world data always contains noises in both dataset D1 and dataset D2. Both JEPs and EJEPs cannot capture those useful patterns whose support in dataset D1 is very small but not strictly zero; that is, they appear only several times due to random noises. Therefore the Noise-tolerant EPs were proposed.

6. High Growth-Rate Emerging Patterns (HGEP) [6]

Although the NEP strategy takes noise patterns into consideration, it still will miss some item sets with a large growth rate, which may result in the low accuracy. Therefore, in this paper, we propose a High Growth-rate EP (HGEP) strategy to improve the disadvantage of the NEP strategy. To provide EPs with the high growth rate(GR), we take an item set X which satisfies the following condition into consideration: GR(proper subset(X))<GR(X). If an item set X satisfies the above condition, we keep the item set which has longer length and a higher growth rate than those of its subsets. High Growth Emerging Pattern (HGEP), which can improve the accuracy of a classifier [6] . An item set X is an HGEP for dataset D2 from dataset D1todatasetD2, if X satisfies one of the following two conditions: where δ1 and δ2 are the support thresholds of the dataset D1and D2.

Condition 1:

1.1 $0 < suppD1(X) \leqq \delta1$ and $suppD2(X) \geqq \delta2$,

where $\delta1 << \delta2$

1.2 GR (proper subset(X))<GR(X).

Condition 2:

2.1 $suppD1(X)=0$ and $suppD2(X)>=\delta2$.

2.2 Any proper subset of X does not satisfy Condition.

Relationships between Various EPs:

They have the following properties [6]:

$EP \supseteq JEP \supseteq EJEP$

$NEP \supseteq EJEP$ & $HGEP \supseteq EJEP$

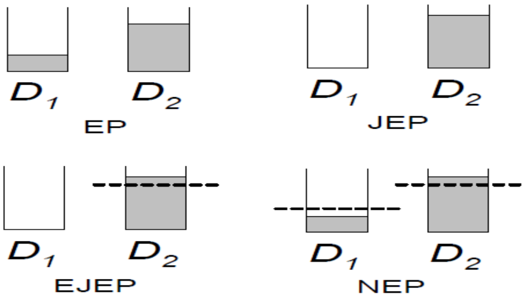


Figure 3: The various Typesd of EPs [6]

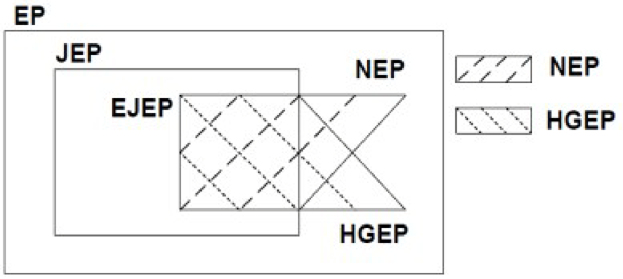


Figure 4: The illustration of all kind of EPs[6]

3. Contrast Pattern Tree Structure (CP-Tree)

Inspired by the FP- tree [3], the CP-tree data structure is used for EP mining for the first time [2].ACP-tree registers the counts in both the positive and negative class. An example CP-tree is illustrated in Figure 5 from dataset Table 1. It is also referred as CP-tree.

Table 1: An example data set with two classes[2]

ID	Class Label	Itemsets	Itemsets(ordered by)
1	D1	{a, c, d, e}	{e, a, c, d}
2	D1	{a}	{a}
3	D1	{b, e}	{e, b}
4	D1	{b, c, d, e}	{e, b, c, d}
5	D2	{a, b}	{a, b}
6	D2	{c, e}	{e, c}
7	D2	{a, b, c, d}	{a, b, c, d}
8	D2	{d, e}	{e, d}

Because every training instance is sorted by its support ratio (the order is denoted as a) Between both classes when interesting into the CP-tree, item with the high ratio, which are more likely to appear in an SJEP, are closer to the root. The map from a path in the CP-tree to an item set is a one-to-one mapping [2].Using the predefined order Á, we can produce the complete set of paths (item sets) systematically through depth-first searches of the CP-tree.

Unlike the FP-growth algorithm that performs frequent pattern mining from leaf to root and must create many conditional FP-trees during the process, the CP- tree based algorithm searches the CP-tree depth first from root and performs a powerful technique, node merge, along with the search [2]. The CP-tree based algorithm can discover SJEPs of both D1 and D2 from the CP-tree at the same time–a “single-scan” algorithm. Previous EP mining methods such as the border-based algorithms and cons EP Miner have to call the corresponding algorithm twice using D1 and D2 as targeted at a set separately [4]. Unlike those two approaches, we do not need to construct one CP-tree for miningSJEPsofD1, and construct another CP-tree for mining SJEPs of D2.

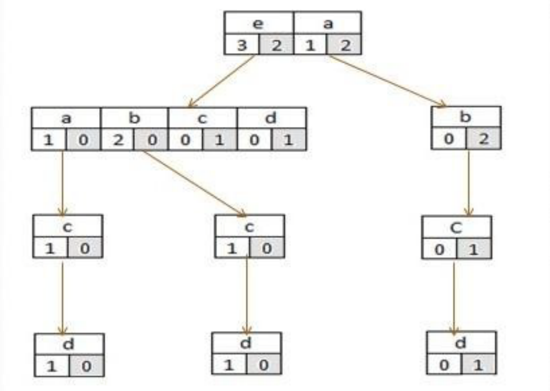


Figure 5: Theoriginal CP-treeoftheTable1dataset[2]

The algorithm constructs tree structures to target the likely distribution of JEPs. In the CP- tree of the example data set the root R= {e (3, 2), a (1,2)}. This means that:

For those instances beginning with e (instances with ID=1, 3,4, 6 and 8),‘e’ appears 3 times in the positive class and 2 times in the negative. For those instances beginning with a (instances with ID=2, 5 and 7), ‘a’ appears 1 time in the positive class and 2 times in the negative. Note that ‘a’ also appear in some instances beginning with e, e.g., the instance with ID =1.

Now we explain the reason behind the need for calling merge tree(T1, T2) during the mining process. One item set can contribute to the counts of a number of patterns by 1.

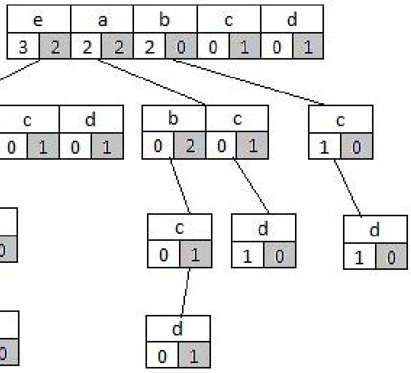


Figure 6: Modified CP-Tree after merging nodes

Note that $e \prec a \prec b \prec c \prec d$. Consider the leftmost branch of the CP-tree shown in Figure 5, which corresponds to the first instance of the dataset shown in Table, {e, a, c, d}. It can contribute to 15 patterns. We can partition these item sets into three groups: [4]

- Group 1 includes the prefix of {e, a, c, d}: {e}, {e, a},{e, a, c}, {e, a, c, d}.
- Group 2 include the prefix of some suffix of {e, a, c, d}: {a}, {a, c} and {a, c, d} (these 3 patterns are {a, c, d}'s prefix); {c} and {c, d} (these 2 patterns are {c, d}'s prefix); {d} (it is {dg}'s prefix);
- Group 3 includes 5 item sets: {e, a, d}, {e, c}, {e, c, d}, {e, d} and {a, d}.

When {e, a, c, d} is inserted into CP-tree initially, only the counts of {e}, {e, a}, {e, a, c}, {e, a, c, d} are registered correctly. Note that they are all the prefixes of {e, a, c, d}, which belong to the Group 1. Let R denote the root of CP-tree, T2 = R, and T1 = R.next[i], where R.items[i] = e. [4] After calling merge tree(T1; T2), that is, merging the sub tree R. next[i] with R, <a, c, d> is merged to the paths from the root. So the counts of {a}, {a, c} and {a, c, d} ({a, c, d}'s prefix) will become correct when merging recursively. Similarly, the counts of {d} and {d, e} ({d, e}'s prefix), and the counts of {e} ({e}'s prefix) will be correct later on. These itemsets belongs to the Group 2. For those item sets in the Group 3, we use {e, d} as an example to illustrate the idea. Not only {e, a, c, d}, but also {e, b, c, d} and {e, d} contribute the count of {e, d}. By merging the sub trees of R.e.a, R.e.b and R.e.c into R.e , we accumulate all the counts of d, hence deriving the correct count for {e, d}. [4]The left sub-tree is merged with right sub-tree in final tree. Here, the counts will be summed together. Basically, the process merges all the nodes of ST into corresponding parts of R.[5] Let then examine both counts of R.e(3 : 2) and find that item set {e} is not an EJEP. We perform depth-first search on the CP-tree. After calling merge(M;N), where M is N.a's sub tree, we examine N and and that item set {e, a}(1 : 0) (1 and 0 are two counts in D1 and D2, respectively) is not an EJEP. Because both counts are smaller than the threshold, we do not go down this branch further looking for EJEPs. Instead, we turn to the next item in N and merge N. b's sub tree with N. The search is done recursively and the complete set of EJEPs can be generated, that is {e, b} (2 : 0), {a, b}(0 : 2) and {e, c, d}(2 : 0).

The study of performance of the HGEP strategy under the input datasets from the UCI Machine Learning Repository: (www.ics.uci.edu/~mlearn/MLRepository.html)

The comparison of the accuracy between our HGEP strategy and the NEP strategy is shown in Table 1. It is observed that HGEP strategy provides the higher accuracy than the NEP strategy or the accuracy of the HGEP strategy is still very close to that of the NEP strategy. Therefore, the average accuracy of our HGEP strategy is still better than that of the NEP strategy. In the following cases, the random noises were added to three datasets and observe how the accuracy is affected by the percentage of the increasing noises between HGEP strategy and the NEP strategy.

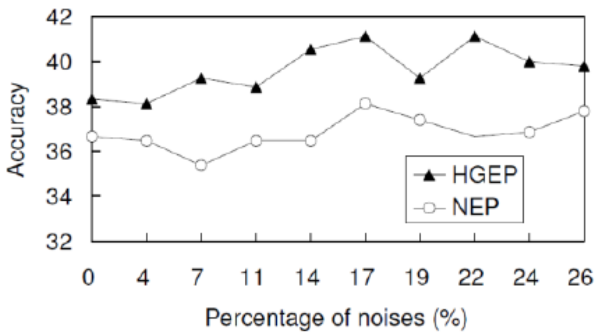


Figure 7: The effect of increasing noises on dataset *diabetes*

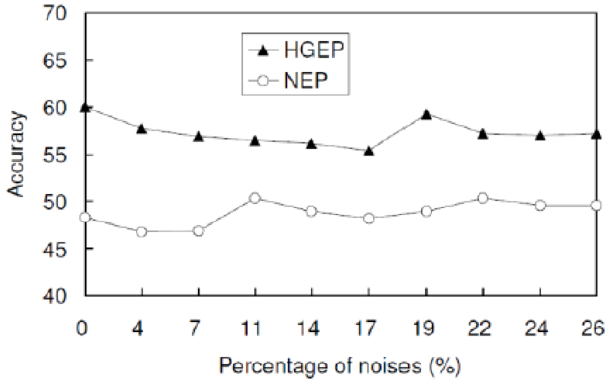


Figure 8: The effect of increasing noises on dataset *mux6.*

In Fig. 7 and Fig. 8, the comparison is given, when the *diabetes* and *mux6* as the input datasets, respectively. From both figures, it is clear that the accuracy of our HGEP strategy is better than that of the NEP strategy.

4. Conclusion and Future Work

This paper provides study of Emerging Patterns in the field of data mining and Knowledge Discovery in Databases (KDD). Specifically, it has investigated the following problems: (1) how to define various kinds of Emerging Patterns that provide insightful knowledge and are useful for classification; (2) how to mine those useful Emerging Patterns. Based on the comparison with the NEP strategy by using several real microarray datasets, we have shown that the accuracy of our HGEP strategy is higher than that of the NEP strategy. Survey on improve the quality of rule and build the accurate classifier.

References

[1] J. Han and M. Kamber, *"Data Mining: Concepts and Techniques"*, 2nd ed., Morgan Kaufmann Publishers, 2006.

[2] KotagiriRamamohanarao, James Bailey and Hongjian Fan, *"Efficient Mining of Contrast Patterns and Their Applications to Classification"*, IEEE Society, ICISIP 2005, pp. 39-47.

[3] KotagiriRamamohanarao, James Bailey and Guozhu Dong, *"Contrast Data Mining: Methods and Applications"*, ICDM 2007

[4] Hongjian FAN, *"Efficient Mining of Interesting Emerging Patterns and Their Effective Use in Classification"*, The University of Melbourne, 2004

[5] Guozhu Dong, James Bailey, *"Overview of Contrast Data Mining as a Field and Preview of an Upcoming Book"*, IEEE Society, ICDMW 2011

[6] Ye-In Chang, Zih-Siang Chen, and Tsung-Bin Yang ,*"A High Growth-Rate Emerging Pattern for Data Classification in Microarray Databases"*, Lecture Notes on Information Theory Vol. 1, No. 1, March 2013

[7] KotagiriRamamohanarao, Thomas Manoukian and James Bailey, *"Fast Algorithms for Mining Emerging Patterns"*, Springer 2002

[8] KotagiriRamamohanarao and James Bailey, *"Discovery of Emerging Patterns and Their Use in Classification"*, Springer, 2003

[9] Arnaud Soulet, Bruno Cr´emilleux and Marc Plantevit, *"Summarizing Contrasts by Recursive Pattern Mining"*, ICDMW 2011

[10] Raj Kumar, Dr. Rajesh Verma, *"Classification Algorithms for Data Mining: A Survey"*, IJIET 2012

[11] Masaharu Yoshioka, *"Analyzing Multiple News Sites by Contrasting Articles"*, IEEE 2008

[12] Liang Wang, Yizhou Wang and Debin Zhao, *"Building Emerging Pattern (EP) Random Forest for Recognization"*,IEEE Society, ISIP 2010

[13] KotagiriRamamohanarao, Qun Sun and Xiuzhen Zhang, *"Noise Tolerance of EP-Based Classifiers"*, Springer 2003

[14] UCI Machine learning Repository, http://archive.ics.uci.edu/ml/datasets.html