www.ijecs.in
International Journal Of Engineering And Computer Science ISSN: 2319-7242
Volume 4 Issue 11 Nov 2015, Page No. 15065-15070

Dynamic and intelligent classifier for efficient retrieval from web mining

M.Parvathi MCA, M.Phil[1], .S.Thabasu Kannan, M.TECh, Ph.D, MBA[2]

[1]Head, Dept. of Computer Applications,
Senthamarai College, Madurai 625 021
pranavparu2674@gmail.com

[2]Principal, Pannai College of Engg& Tech,
Sivagangai – 630 561,
thabasukannan@gmail.com

Abstract: In recent years the WWW has turned into one of the most important distribution channels for private, scientific and business information. The reason for this development is relatively low cost of publishing a website and more up-to-date view on a business for millions of users. As a result the WWW has been growing tremendously for the last five years. The Google recently reported that it is currently indexing over 7 billion text documents. The number of registered international top level domains has increased more than 9 times over the last 5 years.
The main aim of this paper is to retrieve the effective and efficient retrieval of required documents from various web pages of various websites. Here the efficient and effectiveness can be measured in terms of relevancy and similarity. For achieving more relevance during retrieving the required documents we can use some KDD techniques to extract specific knowledge from the WWW. For achieving relevancy and similarity, some classification methods have been used. For this purpose we have analyzed various classification methods in data mining to evaluate their own performance. The factors like precision and recall have been used to measure the performance and to calculate the accuracy and the number of documents retrieved during a particular period of time. To identify the level of relevancy and similarity the level of accuracy plays an important role. For this we have defined a threshold for comparison. If the evaluated accuracy is greater than the threshold then the similarity level is high. Otherwise the similarity level will be low. If the number of relevant document retrieved during a particular period of time is large the level of efficiency increases. Otherwise it will be low. For the purpose of testing we have taken 30,000 single HTML documents from 300 web sites. We have taken 4 existing classification techniques to compare the efficiency of newly developed classifier.

Keywords: Precision, Recall, Accuracy, Relevancy, Similarity, Classifier, Pruning, Parameter

1. Introduction

The WWW is currently the largest source of information that is available to a broad public. Most established approaches of web content mining are concerned with the efficient retrieval of specific HTML documents. To find specific information from vast amounts, there are several methods to retrieve interesting content from the WWW. Search engines download a large amount of webpages and index them with respect to the key words occurring within them. The search engine returns links to all documents containing these key terms, which can exceed several thousands.
The classification of websites is differ from the classification of single webpages. Sites may strongly vary in size, structure, used language and techniques. Many professional sites, especially in the non-English-speaking regions, are at least bilingual to provide international usability. Most page classification use only text documents in a single language which may prove insufficient when trying to handle whole sites. To download a site from the web, first to examine the homepage, then use a HTML-parser to determine the links to the other pages within a site. Then explore the corresponding webpages in the same way as the homepage. It is necessary to mark the pages already visited, since a webpage might be reachable by following more than one link.
The most common way to classify single HTML documents is to use naive Bayes classifiers, decision tree (DT), K-nearest neighbor (k-NN) or Support Vector Machines (SVM)on a feature space of terms. Here the quality of the results depends highly on the right choice of terms.

1. Previous study

a) Naive bayes approach to Website Classification

Naive Bayes classifiers are simple probabilistic classifiers based on applying Bayes' theorem with assumptions between the features. It is a supervised learning method and a statistical method for classification. It allows capturing uncertainty by determining probabilities of the outcomes. It calculates explicit probabilities for hypothesis and it is robust

to noise in input data. It is applied to decision making and inferential statistics that deals with probability inference. It is used the knowledge of prior events to predict future events. It is suited when the dimensionality of the inputs is high. It often performs better in many complex real world situations.

The simplest way to classify websites is to apply established techniques of webpage classification to the homepage of a website. This approach is simple and efficient with the assumption that the homepage contains the information to identify the complete website. We just generate a single feature vector, counting the frequency of terms over all webpages of the whole site, i.e. we represent a website as a single *super-page*. The advantage is that it is not much more complex than the classification of single pages. The *drawback* of this approach is very sensitive to the right selection of key terms. Structural features like the occurrence of frame tags lose most of their significance. Another problem is the loss of local context. Keywords appearing anywhere within the site are aggregated to build up a bag-of-words view of the whole website. This classifier achieved insufficient accuracy in most experiments.

Here reading the first *n* pages of a website do not yield a good accuracy. First, the topology of a website is a matter of individual design and therefore tends to be very heterogeneous. Many sites contain large amounts of pages providing only structure but no content. Another important aspect is how much content is provided on a single page. The same amount of information could be spread over several pages or be contained in one large webpage.

b) Classification using Page Classes

We map the webpage to a page class. A page class represents a certain type of webpage that is likely to appear in a certain type of website. Since the terms only influence the page class of a webpage, the local context is preserved. To determine the page class, we use text-classification. Since there is always a classification error for each page, the probability that the complete graph is correctly labeled is rather low. But the average number of correctly labeled nodes is about the mean classification accuracy of the page classification. Based on the labeled pages of a website, we propose the following representations of a website:

Each page class defines a dimension of the feature space. For each page class, the feature values represent the number of pages within the site. This representation does not exploit the link structure of the site, but it considers a website as a set of labeled webpages. In other words, we treat a website as a multi-instance object and use a webpage classifier to condense this set into a single feature vector.

Algorithm SVM

candidateSV = { closest pair from opposite classes } while there are violating points do

Find a violator candidateSV = candidateSV S violator if any αp < 0 due to addition of c to S then candidateSV = candidateSV / p repeat till all such points are pruned end if end while

A. Find the closest pair of points in kernel space requires n/2 kernel computations where n= the total no of data points. But, in case we use a distance preserving kernel like the exponential kernel the nearest neighbors in the feature space are the same as the nearest neighbors in the kernel space. Hence we need not perform any costly kernel evaluations for the initialization step.

B. Adding a point to the Support vector set:Given a set S which contains only support vectors and to add another support vector c to S.

Website trees

A Decision Tree (DT) is a classification scheme which generates a tree and a set of rules, representing the model of different classes, from a given dataset. Here each internal node denotes a test on an attribute, each branch represents an outcome of the test and leaf nodes represent the classes or class distributions. We can easily derive the rules corresponding to the tree by traversing each leaf of the tree starting from the node. It may be noted that many different leaves of the tree may refer to the same class labels, but each leaf refers to a different rule.

DT does not require any parameter setting from the user and thus are especially suited for exploratory knowledge discovery. It is relatively fast and the accuracy is superior to other classification methods. They provide a clear indication of which fields are most important for prediction or classification. The entropy is a measure of the uncertainty associated with a random variable. As uncertainty and or randomness increases for a result set so does the entropy. The value of entropy lies between 0 – 1.

$$Entropy(D) = \sum_{i=1}^{c} -pi \log 2(pi)$$

$$Gain(D, A) = Entropy(D) - \sum_{j=1}^{v} {|Dj|}/{|D|} Entropy(Dj)$$

Algorithm

Create a node N; if tuples in D belongs the same class C then returns N as a leaf node labeled with the class C; if attribute list is empty then return N as a leaf node labeled with the majority class in D; if splitting attribute is discrete-valued and multi-way splits allowed then for each outcome j of splitting criterion let Dj be the set of data tuples in D satisfying outcome j; if Dj is empty then attach a leaf labeled with the majority class in D to node N; else attach the node returned by Generate decision tree end for return N;

To capture the essence of the link structure within a site, we represent it as a labeled tree. The idea is that the structure of most sites is more hierarchic than network-like. Sites begin with a unique root node provided by the homepage and commonly have directory-pages that offer an overview of the topics and the links leading to them. Furthermore, in most sites the information in the area around the homepage is very general and gets more and more specific with increasing distance. For building website trees, we use the minimum number of links as a measure of distance between two pages of a site. To construct a website tree the minimal paths from the homepage to every page in the website are joined. Therefore, we perform a BFS through the graph of a website and ignore the links to pages we already visited. If two paths of equal length leading to the same webpage, the path occurring first is chosen.

c) Classification without Page Classes

For the classification of an unknown object, a basic k-NN classifier performs a k-NN query on the training database and

predicts the most frequent class in the result set. The key to the effectiveness of k-NN classification is an intuitive distance function. Since the content of each single page can be represented by frequencies, a whole website is represented by a multi-instance object. The *sum of minimum distances* most adequately reflects the similarity between two websites.

Method:

i. Find k nearest neighbors of d in the existing dataset on the basis of distance or similarity measure.
ii. Determine which class c is the class to which most of those k known samples belong.
iii. Assign the new sample d to the class c.

A new instance is classified as the most frequent class of its k nearest neighbors. It contains very simple and intuitive idea and it is very easy to implement. k-NN is based on instance based learning (IBL), case based reasoning (CBR) and lazy learning. Threshold for the most frequent class in the neighbor is the usual guaranty.

Distance usually relates to all the attributes and assumes all of them have the same effects on distance. The similarity metrics do not consider the relation of attributes which result in inaccurate distance and then impact on classification precision. Wrong classification due to presence of many irrelevant attributes is often termed as the *curse of dimensionality*

The class of each labeled instance is compared with the label assigned by a k-NN obtained with all instances. If both labels coincide, the instance is maintained in the file, otherwise it is eliminated. If the true class and the class predicted by the k-NN are the same the instance is not selected, otherwise the instance is selected. The method depends on the storage ordering.

2. New method

In our new method, it downloads only a small part of a website, which still achieves high classification accuracy. This method performs incremental classification and stops downloading additional pages when an area around the homepage is visited that is likely to be a good representation of the purpose of a website. Additional pages are found by following the links on the homepage. So the new system use ranking algorithms to ensure the list of results starts with the most relevant pages. Here the classification of websites which can be employed to maintain web directories automatically, increasing the recall of this established method for searching the web. A breadth-first traversal seems to be a promising approach. Since this traversal strategy orders the pages with respect to their distance to the homepage, the more general and therefore more important pages are visited first. The key to efficient classification is to prune certain sub-trees in the graph of the website. Thus a node can only be ignored when every path leading to it is pruned. The main advantage of the new method is that it requires a small amount of training data to estimate the parameters. Our new method is based on the following propositions:

Case 1: The membership of a complete path in some site class depends on the pages closest to the homepage. General information about the class of a website is most likely placed within a few links from the homepage. If a new topic follows, it appears in the context of the former topic.

Case 2:There are a whole sub-tree and the path leading to it does not show any clear class membership at all. A tree could always become highly specific after the next node. But after a reasonable length of the path, the probability that the meaning of the sub-tree is of general nature is significantly decreasing. So the strength of the class information has to be measured by the length of the path.

To exploit these propositions, it is necessary to measure the degree of class membership for a path and its impact on site classification. Here the ability of a website classifier to incrementally calculate the class membership is very useful. So we applied pruning to the methods of both cases that performed best.

The conditional probabilities yield the information about the degree that a path supports a site class. Since the focus lies on the importance of a path for site classification, the actual class or classes it supports are not relevant. To quantify the importance for the complete site, we use the variance of the conditional probabilities over the set of all website classes. Since the variance is a measure for the heterogeneity of the given values, it mirrors the ability of a path to distinguish between the different site classes. If variance is low then the path is treated similar by the model of any class. The first requirement is not as easy to fulfill, but is provided with high probability after a certain length of the path is reached.

An additional webpage that is more likely to be found in a site class different from the currently predicted class will most likely decrease the weights. Thus after a few nodes on a path *s* a decreasing value of weights indicates a changing topic. Now our first proposition can be applied, i.e. the path can be pruned.

With increasing lengths, it is more and more unlikely that an additional factor can increase the variance strongly enough. Due to the required growth of variance and the decreasing influence of the additional factor, most paths are cut off after a certain length. This corresponds to the requirement made by our second case that a path will not provide general information about the class of the website after a certain length. Hence we avoid reading large sub-trees without any impact on site classification. This method not only increases the efficiency of website classification, but it can also improve the classification accuracy. By providing an effective heuristic to disregard areas that are relevant information, the classifiers will offer better accuracy. This rule tries to cut off misleading areas from the website tree and thus can reduce the processing time and also increase the classification accuracy.

3. Performance evaluation

a. Evaluation of Website Classifiers

Here our classifiers were tested on two scenarios. First, we will focus on the case that page classes and corresponding training pages are available. We compared the accuracy of the introduced classifiers and examined the performance of the introduced pruning method. In the second part of our evaluation we compared the classification accuracy and examined the classification time of k-NN classifiers.

We have found the following during comparison of various documents retrieval from three different service oriented organizations web sites. From fig1 we got the following:

The level of up and down for health care service oriented organizations is very meager.

DOI: 10.18535/Ijecs/v4i11.34

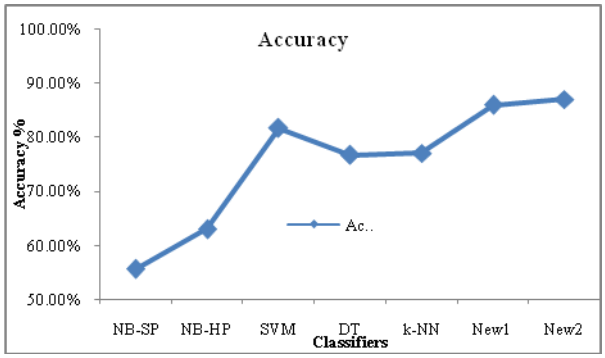

Fig 3: Provider wise accuracy

The curve for education service organization is uneven, because the precision for SVM and our new method is high and other existing methods are very low. With respect to internet service organization, the classifier except naïve Bayes is more or less same. The precision level for our new system lies on same.

From fig2 we got the following: for our new method the level of recall is more or less same for all types of service oriented organizations. With respect to health care service organizations, the recall level for naïve Bayes is too low. In support vector machine, the level is same for health care service oriented organizations and education service organization. For our new system the recall for internet service organization and education service organization are more or less nearer.

b. Experiments using Webpage classes

For the purpose of testing we have taken 30,000 single HTML documents from 300 web sites. The distribution of the website classes was: 112, *IT service providers*, 90, *Academic service providers* and 98, Healthcare service providers. To make the experiments reproducible, the downloaded information was stored locally. To classify the pages into the page classes, we labeled about 2% of the pages in the test-bed and obtained a classification accuracy of about 72% using naive Bayes on the manually labeled pages. The remaining 98% of the pages were labeled by the naive Bayes classifier based upon this training set.

Since the super-page approach provided only an accuracy of about 55%, it seems not to be well-suited for website classification. Webpage classification of the homepage using naive Bayes performed similarly bad by achieving only a classification accuracy of 63%, which underlines the assumption that more pages than the homepage are necessary for accurate website classification.

The best method using the complete website turned out to be the new method–pages closest to home page which yielded 10.3% more classification accuracy than the decision tree classifier. It also clearly outperformed the SVM by 5.3%. As a comparison the new method–pages closest to home page, applying the introduced method, increased the accuracy by 1% to 87% by reading only 57% of the data. To compare the methods using page classes with those ones that do not, we additionally applied the k-NN which is the best performing type of this direction to this test-bed. Though the k-NN offered reasonable results as well, it was outperformed by Pruned–pages closest to home page by about 9.9%.

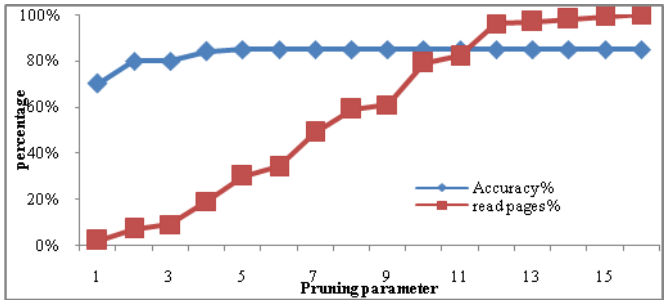

Fig 4: Effect of the pruning parameter on the classifier accuracy and the % of downloaded webpages.

From fig4 we acquired the following: since the recall and precision for our newly developed system are closer to each other. Its accuracy level is also increased. The accuracy level of support vector machine is also more or less same as our new system. The accuracy level is same for pruning parameter >=4. The number of documents retrieved per second is meager from the pruning parameter =11 and there is a tremendous increase from 5-11.

The second set of experiments demonstrates the effects of the new method when applied to the most promising approach in this scenario, the new method–pages closest to home page. In the above figure for values of 5 to 15 the achieved accuracy exceeds 86%, which is the accuracy when reading all webpages. Since the accuracy did not react very sensitive to varying values for parameter after a reasonable value was reached, it is relatively easy to make a choice that favors accuracy and/or efficiency.

c. Evaluation without explicit Page Classes

In our test-bed, we chose 6 different website classes and built an additional Healthcare providers class from a randomly chosen mixture of other Yahoo classes. Our training database consisted of 86 websites for the category" education providers" and between 12 & 47 example sites for the 6 classes. The total number of sites was 234, comprising a total of about 18,000 single webpages. In this test-bed, no page classes were provided to label single webpages within a website. The first set of experiments tested precision and recall for each of the 6 website classes only for the two-class case.

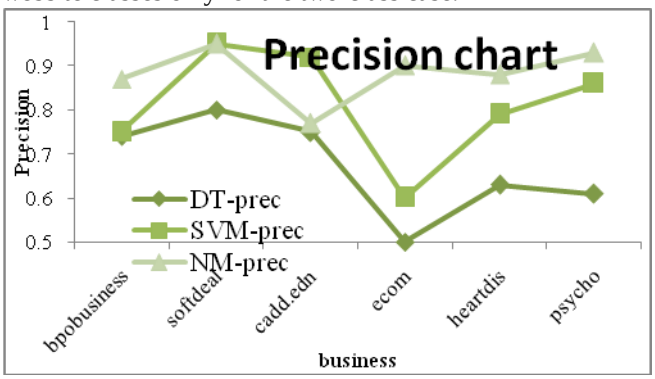

Fig 5: Classifier wise precision

DOI: 10.18535/Ijecs/v4i11.34

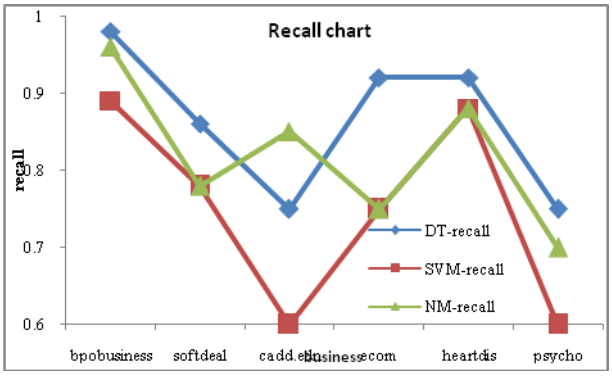

Fig 6: Classifier wise recall

From fig5 we got the following: the precision for education service organization retrieval is very low with respect to decision tree and support vector machine is low as compared with any other retrieval. For our newly developed system, the level of precision is high for all retrieval except caddied. Overall precision and recall for our new method is comparatively higher than any other existing methods. Hence the accuracy level is also high.

A second set of experiments investigated the ability of the above three classification methods to handle more than one class by giving the complete training set to the classifier. The results displayed the ability of the basic k-NN classifier to provide good precision and recall without using page classes. The k-NN-classifier achieves a better trade-off between precision and recall in most of the cases. The k-NN provided very good accuracy and outperformed the other two classifiers. The accuracy is used to measure because it is the most common quality measure for classification problems distinguishing more than two classes.

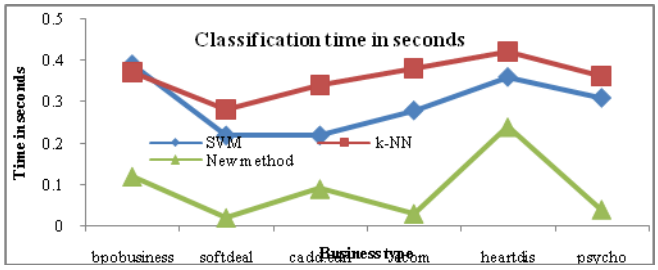

Fig 7: Classification time in sec per website

The results clearly show that the basic k-NN classifier takes a considerable amount of time for classification. The k-NN performed pretty well compared to the new method–pages closest to home page and offered a speed-up of about 100 compared to the basic k-NN approach. This enormous speed up is due to the small average number of centroids (about 180 per centroid set) and the use of incremental classification considering only few pages of a website (about 20) for very accurate classification. In general, the k-NN classifier offered a very good accuracy depends on retrieval time.

A third experiment investigated the effects of the parameter setting of clustering algorithm used to derive the centroid sets. For the heart dis example the below figure shows the dependency of accuracy and classification time on the k of neighbors needed to define a core point and the radius. The shape of the graph indicates that the influence of the radius is very stable within the interval from 0 to 0.5 which is half of the possible target interval of the cosine coefficient. On the other hand, the influence of the number of neighbors k shows an obvious decrease of accuracy for k = 3 and no significant efficiency gain for k < 2. Therefore, setting k = 2 and = 0.4 offered a good trade-off between classification time and accuracy.

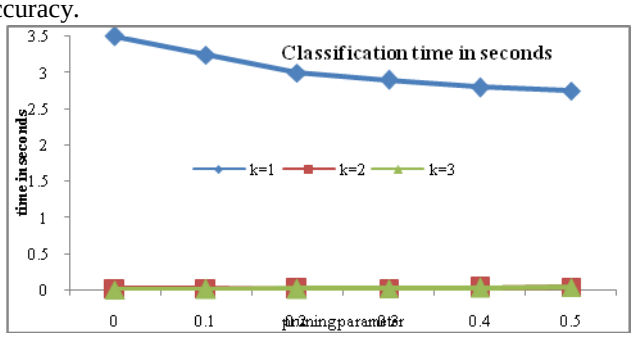

Figure 8: Classification time depending on the parameter

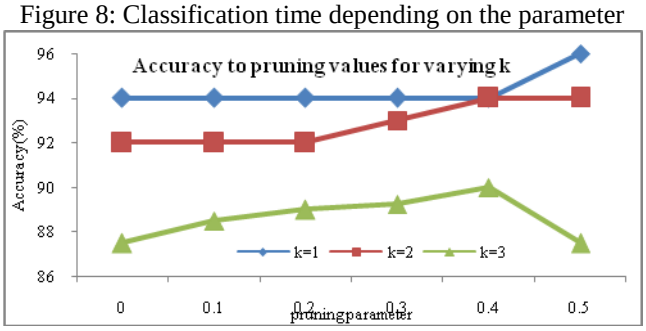

Figure 9: Accuracy depending on the parameter

4. CONCLUSION

To conclude, the simple methods of website classification like the homepage and the super-page approach were not suitable to achieve reliable website classification. For the scenario that page classes and corresponding training pages can be provided, the new method–pages closest to home page performed best. Since this approach does not employ the link structure like other DT, it treats websites as multi-instance objects, i.e. sets of feature vectors. For the scenario without page classes, the k-NN outperformed all other classifiers and demonstrated classification times that are suitable for real-world applications. The new method is capable to reduce the classification time and to increase the accuracy.

Due to lack of time, we have taken only four existing classifier to compare the performance of the newly developed classifier. In near future, we may extend the same paper by comparing the performance of 10 classifiers to get more effectiveness and efficiency in terms of relevancy and similarity. And also we have taken 1,000 samples from three different types of service provider's web sites. It may be extended to get more efficiency and effectiveness by increasing the samples to 2,000 from 6 or more different types of organizations web sites.

REFERENCES

[1] S. Vaithyanathan, J. Mao, and B. Dom. "Hierarchical Bayes for Text Classification". In Proc. Int. Workshop on Text and Web Mining, Melbourne, Australia, pages 36–43, 2012.

[2] J. Wang and J.D. Zucker, Solving Multiple-Instance Problem: A Lazy Learning Approach". In Proc. 17th Int. Conf. on Machine Learning (ICML'12), Stanford, CA, USA, pages1119–1125, 2012.

[3] J. Wang, H. Zeng, Z. Chen, H. Lu, L. Tao, and W. Ma. "Re CoM: reinforcement clustering of multi-type interrelated

data objects". In Proc. 26th ACM SIGIR Conf. on Research andDevelopment in Information Retrieval, Toronto Canada, pages 274–281, 2013.

[4] R. Agrawal, C. Faloutsos, and A. Swami. "Efficient Similarity Search in Sequence Databases". In Proc. 4th. Int.Conf. on Foundations of Data Organization and Algorithms, Evanston, ILL, USA, Lecture Notes in Computer Science (LNCS), Springer, pages 730: 69–84, 2013.

[5] S. Berchtold and H.-P. Kriegel. "S3: Similarity Search in CAD Database Systems". In Proc. ACM SIGMOD Int. Conf. on Management of Data (SIGMOD'97), Tucson, AZ, USA, pages 564–567, 2011.

[6] S. Brecheisen, H.-P. Kriegel, P. Kroger, M. Pfeifle, and M. Schubert."Using Sets of Feature Vectors for Similarity Search onVoxelized CAD Objects". In Proc. ACM SIGMOD Int. Conf.on Management of Data, San Diego, CA, USA),pages 587–598, 2013.

[7] C.J.C. Burges. "A Tutorial on Support Vector Machines forPattern Recognition". Data Mining and Knowledge Discovery,2(2):121–167, 2010.

[8] N. Cristianini and J. Shawe-Taylor. "An introduction to supportvector machines and other kernel-based learning methods".Cambridge University Press, 2012.

[9] J. Gehrke, R. Ramakrishnan, and V. Ganti. "RainForest -A Framework for Fast Decision TreeConstruction of LargeDatasets". In Proc. 24th Int. Conf. on Very Large Data Bases, New York, NY, USA, pages 416–427, 2011.

[10] E.-H. Han and G. Karypis. "Centroid-Based Document Classification:Analysis and Experimental Results". In Proc. 4thEuropean Conf. on Principles of Data Mining and KnowledgeDiscovery, Lyon, France, Lecture Notes in ComputerScience (LNCS), Springer, pages 1910: 424–431, 2012.

[11] H.-P. Kriegel, P. Kr¨oger, A. Pryakhin, and M. Schubert. "UsingSupport Vector Machines for Classifying Large Sets of Multi instance Objects". In Proc. SIAM Int. Conf. on Data Mining(SDM'2004), Lake Buena Vista, FL, USA, pages 102–113,2014.

Author Profile

Ms. M. Parvathi has been working as Professor and Head in Computer Applications Department, Senthamarai College, Madurai. She received her M.Phil degree in Computer Science from Manonmaniam Sundaranar University, Thirunelveli. She registered her Doctorate at Mother Teresa University. Her current research interests focus on the area of Data Mining. She has published a Data Mining survey paper in IJLTET. Her research work is guided by Dr. S. Thabasu Kannan, Principal, Pannai College of Engineering and Technology

Prof.Dr.S.Thabasu Kannan has been working as Professor and Principal in Pannai College of Engineering and Technology, Sivagangai and rendered his valuable services for more than two decades in various executive positions. He has published more than 50 research level papers in various refereed International/National level journals/proceedings. He has authored 11 text/reference books on the information technology. He has received 11 awards in appreciation of his excellence in the field of research/education. He has visited 5 countries to present his research papers/articles in various foreign universities. He has been acting as consultant for training activities at Meenakshi Trust, Madurai. His area of interest is Big data applications for bioinformatics applications. Under his guidance 8 Ph.d scholars pursuing and more than 150 M.Phil scholars were awarded. His several research papers have been cited in various citations.