Volume 6, Issue 8, August 2016 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Analysis of Cancer Prediction Using WNPMS-ARM Techniques

[1]K. Arutchelvan, [2]Dr. Pon Periasamy
[1] Programmer (SS), Department of Computer Science, Department of Pharmacy, Annamalai University, Tamilnadu, India
[2] Associate Professor, Department of Computer Science, Nehru Memorial College, Tamilnadu, India

Abstract— Data Mining is the process of extracting useful and nontrivial information from databases. Databases tend to be very big. As a consequence, fast and scalable data mining techniques are increasingly becoming more important. The system introduces four new data mining techniques which use binary representation of data, and take advantage of bit vectors to allow for fast computation and low memory requirements.Finding frequent item sets is a very important problem that can be solved in exponential time. In general, the algorithms that ¯no all the frequent item sets are not practical. Introduces a fast and approximate algorithm for finding the frequent item sets in quadratic time.In the same chapter,it is shown the algorithm works very well in practice.Finally, Introduces an original approximative technique for comput¬ing the distances between objects. This technique uses random hash functions andrelates collisions to distances.It is shown that the technique provides a very good approximation of distances and substantial time gains.

Keywords—Data mining, Classification, Association Rules mining, Clustering and VMM

I. INTRODUCTION

These approaches are basically physical memory-dependent. In other words, they focus on the time scalability issue of the problem and greatly rely on the available physical memory to mine the database. But, with the advancement of information and storage technology, business organizations currently have huge collections of data repositories that should be mined to extract useful knowledge. It is very unrealistic to assume that the data structures to deal with these huge databases can all fit into the primary memory. This arguments supported by the works of Goethals, Varanasi, and Buehrer , who investigated the memory requirement issues of some prominent ARM methods on some real-world datasets. It is evident from their work that even the standard physical memory available in modern computers is not sufficient enough to mine for association rules from large real-world data sets. The traditional trend is to rely on the default virtual memory manager (VMM) of the operating system (OS) to take care of the limited primary memory problem associated with large data structures. Typically, VMM stores the overloaded data into the secondary memory based on some reassumed memory usage criteria. However, this direct and unplanned use of virtual memory results in an unpredictable behavior or thrashing, as depicted by some of the works described in the literature. As result, there is a serious memory-related problem to be tackled shall we need ARM models capable of effectively mining large databases.

II. LITERATURE SUREVEY

In [MallikaRangasamy, 2009] developed a new algorithm called an Efficient Statistical Model based classification algorithm for cancer classification using very few genes from micro gene expression data. This model used classical statistical technique for the purpose of ranking the gene and 2 various classifiers used for gene selection and prediction. The projected method proves that which is capable of generating very high accuracy with the use of very few genes. This paper utilized a three-cancer dataset as Lymphoma; Liver and Leukemia. There are some missing values in these datasets that can be filled by the use of The K-Nearest neighbour (KNN) algorithm. Gene selection can be carried out with the help of ANOVA, Linear Discriminant Analysis (LDA) and SVM-OAA RBF Kernel. Linear Discriminant Analysis (LDA) used for the 2 class datasets such as Liver and Leukemia. Support Vector Machine-One-Against- All (SVM-OAA) and Linear Discriminant Analysis (LDA) is used as a classifier for performance evaluation. Datasets are randomly divided into two one for training and another part for testing and gene ranking that is ANOVA P-Values can be computed using one-way ANOVA. Top genes were selected from the ranked data and gene combination performed. The classifier is trained using all possible gene combinations and the classifier is validated using 5 fold or 10 fold cross validation methods. The best gene combination can be selected from the result of accuracy. Compared with the previous result obtained by ELM [5] SVM OAA attains best accuracy with the use of very few genes than LDA. The same classifier is used on Leukemia and Liver datasets for both the gene selection and classification that improves the strength of the model.

In [Wang, X., and Gotoh, et al, 2009] screened high-class discriminative power and gene pairs utilized to create simple prediction models. These prediction models were used in single genes or gene pairs based on the soft computing approach and rough set theory for selecting single genes. The simple prediction models were applied to four these data sets such as CNS tumor, colon tumor, lung cancer and DLBCL.A rule base pipeline was used as a ruse based method to

construct cancer predictors. Feature selection used an attribute depended degree from rough set theory and rule classifier was created by the use of selected genes. Using the attribute depended degree, some single genes or gene pairs can be detected. The algorithm was applied to the central nervous system (CNS) tumor, colon tumor, lung cancer, and diffuse large B-cell lymphoma (DLBCL) from Kent Ridge Bio-medical Dataset. Single genes are founded through the use of high-class discriminative power. Gene pair or a single gene builds four decision rules, which are used to execute prediction of cancer. The classifiers C4.5 and Naive Bayes are used to predict performance of the gene sets. The C4.5 and Naive Bayes result is compared with FCBF, CFSSF and ReliefF .The efficiency of this method can be validated with the use of Leave-one-out cross-validation (LOOCV). Cancer prediction using soft computing produces better results than the previously published results.

III. PROPOSED SYSTEM

3.1 Approximate Sorting

In our previous work described in, it was shown that by sorting the ordered transactions in a string like fashion, we can improve the spatial locality of the FP tree so that closely related nodes are placed close by in memory. The exact sorting of large databases are costly and instead of having an external sorting algorithm such as Sort-Merge, we can achieve similar performance by adopting an approximate hash sorting algorithm. The two approximate sorting algorithms that arrange the transactions of the database in blocks based on the arithmetic or geometric distribution of the frequencies of items. The sorting algorithms ensure that each transaction of blockiprecedes all the transactions of blocki+1according to the sorting order. Here, each block of transactions corresponds to a physical file. Without the approximate sorting, the initial prefix-tree construction process takes significantly longer time when the prefix-tree grows out of the available physical memory, resulting in frequent page faults. However, we will show later in Section V-D1 that we can avoid the need for an approximate sorting algorithm; hence, one more costly database scan is needed if we construct the prefix-tree in chunks.

3.2 Reorganizing Data, I/O Conscious FP-Tree:

Spatial locality of FP-growth can be improved by reorganizing the FP-tree in such a way that nodes of the FP-tree are reorganized in a depth-first manner. A cache conscious prefix-tree, which is a reorganized version of the original FP-tree. First, a contiguous block of memory equivalent to the size of the FP-tree is allocated. Next, the original FP-tree is traversed in a depth-first manner and nodes of the FP-tree are copied sequentially into this new allocated block to ensure better spatial locality among nodes that are in the same prefix path.

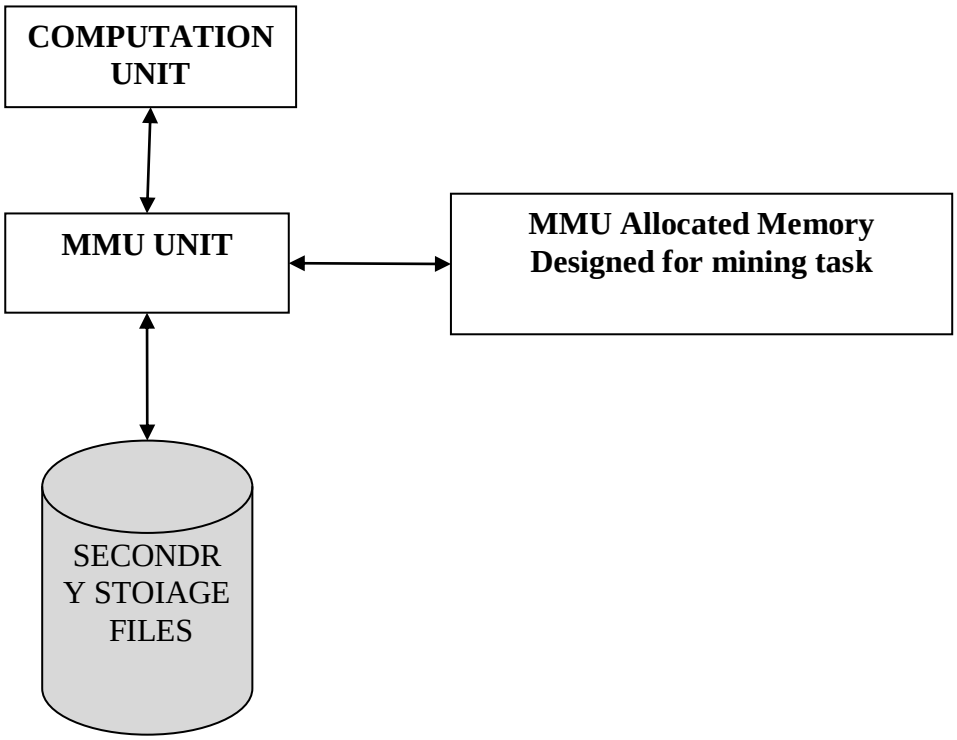


Figure 1 flow diagram

3.3 Reorganizing Computation, Page Blocking:

We know that FP-growth exhibits poor temporal locality as it needs to read the same FP-tree multiple times for each frequent item in the header table of the FP-tree; once, to collect the counts to find the conditional pattern bases and second time, to build the conditional FP-tree from the collected pattern bases. If the VMM system stores part of the tree in secondary memory, this approach may result in constant page faults which can cause the system to slow down immensely; this problem is known as thrashing. In order to get around this problem, the works descried in and have suggested the idea of page blocking which rearranges the computational part of the FP growth in such a way that all the

computations of a particular block of the I/O-conscious FP-tree is exhausted in one-fellsweep. As a result, if the VMM system needs to load this block from the secondary storage to main memory, it needs to do so only once; and this reduces the page faults dramatically. The idea, as suggested in and is presented in Algorithm. As the mining algorithm needs to traverse the tree multiple times in bottom-up fashion, it can traverse this cache conscious prefix-tree to achieve improved performance due to better spatial locality of nodes resulting from this rearranging. The memory block in the bottom of the figure now accommodates the nodes in depthfirst manner instead of having them in the order of their creation time have adopted the same strategy to improve the spatial locality when the data structure corresponding to the FP-tree goes out of main memory.

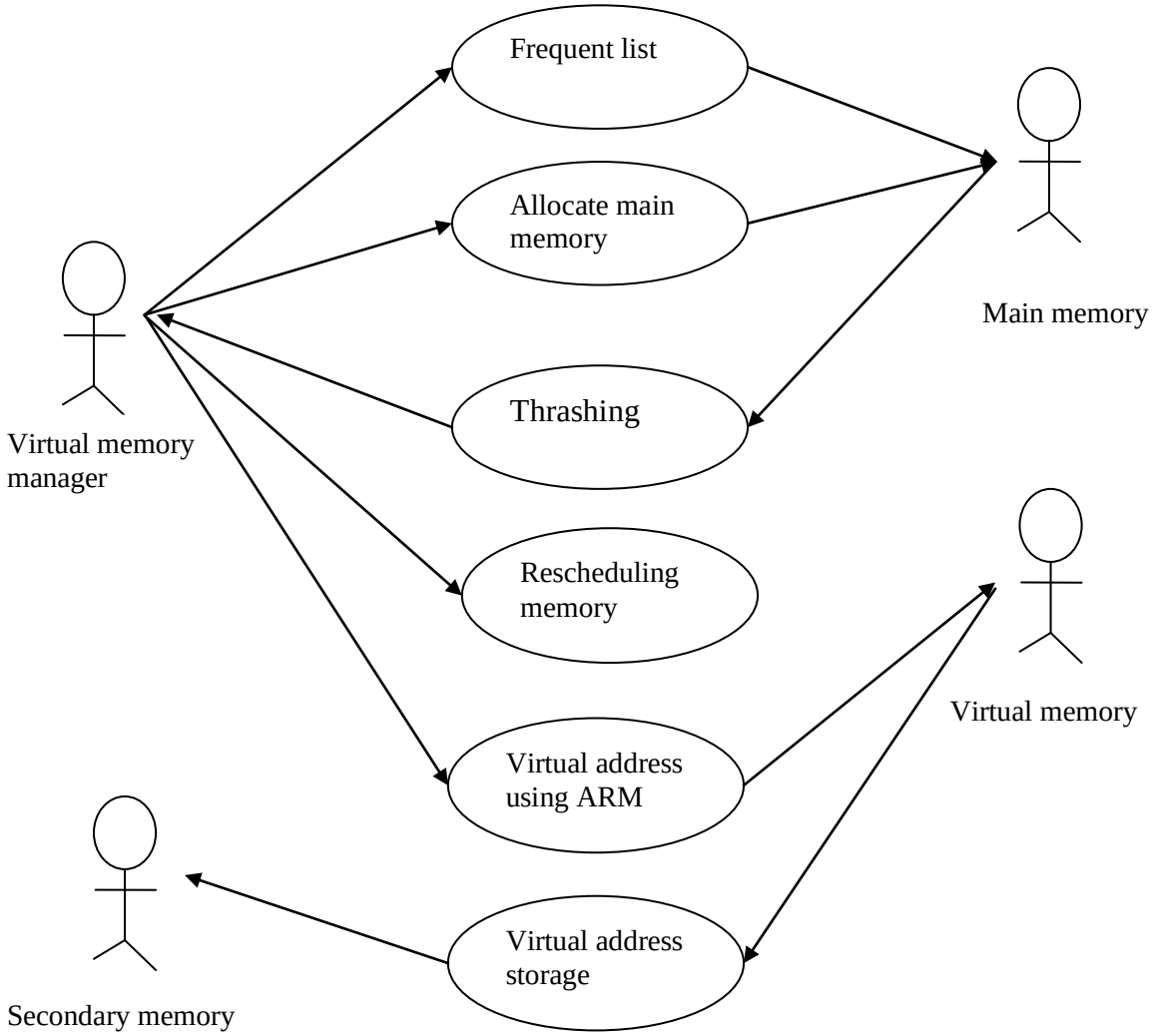


Figure 2 WNPMS-ARM Architecture

3.4 Algorithm

Algorithm 1 Blocked fp-Growth

FP-Growth works in a divide and conquer way. It requires two scans on the database. FP-Growth rest computes a list of frequent items sorted by frequency in descending order(F-List) during its rest database scan. In its second scan, the database is compressed into a FP-tree. Then FP-Growth starts to mine the FP-tree for each item whose support is larger than » by recursively building its conditional FP-tree. The algorithm performs mining recursively on FP-tree. The problem of ¯ending frequent item sets is converted to search-in and constructing trees recursively.

1: procedure BLOCKEDFP-GROWTH (Top, H, s) _ Initial prefix-tree is Top, header table is H, and minimum support threshold is, s
2: for each block, BT of the prefix- tree, Top do
3: for each frequent item, I in H do
4: Follow the node-links of I (starting from H)
to traverse the nodes in BT, and collect counts
for the conditional pattern bases of I that
meet the minimum support criterion.
5: end for
6: end for
7: for each frequent item, I in Do
8: Aggregate conditional pattern base counts that

are collected from each block, BT to create,
gag-count-list
9: end for
10: for each block, BT of the prefix-tree, Top do
11: for each frequent item, I in H do
12: if all the entries of gag-count-listare _< s
13: Build the conditional prefix-tree Top
is
14: end if
15: end for
16: end for
17: for each frequent item, I in Do
18: if Top
isexists then
19: FP-Growth Top
is, Http
is
, s _ Http
is
is the
header table for Top
is
20: end if
21: end for
22: end procedure

Algorithm 2 Bounded-Growth

1: procedure BOUNDEDFP-GROWTH (*T-Id, Item-Id, s*) _
Initial tree Id is *T-Id*, and minimum support
threshold is *s*
2: CollectCountsFor*T-Id*()
3: BuildCFTsFor*T-Id*()
4: for each frequent item *I* do
5: if the tree with id *treeIdi*exists then
6: BoundedFP-Growth*treeIdi*, *s*
7: update CLT, ILT, DTLT
8: end if
9: end for10: end procedure

Algorithm 3 CollectCountsFor*T-Id*

1: procedure COLLECTCOUNTSFOR*T-Id*
2: ConsultMTLT to check whether *T-Id* exists in memory
3: if the tree with id *T-Id* exists in *M.B* then
4: make this tree active
5: load Header table *H* if it is already not in *M.B*
6: for each frequent item *I* on *D*o
7: generate a unique tree id *treeIdi*for *I* collect
counts from the conditional pattern bases of *I* by
8: traversing the memory based FP-tree for *T-Id*
from*H*
9: update CLT
10: end for
11: end if
12: consult DTLT to identify blocks of the tree with
Id *T-Id*
13: for each block *b* of the prefix-tree corresponding to
tree, *T-Id*do
14: load this block *b* into FB
15: for each freq. item *I* in *ILT* corresponding to the tree *T-Id*do
16: generate a unique tree id *treeIdi*for *I* if not
already generated load the node-Ids of *I* from
ILT to traverse the prefix-tree nodes
17: in FB, and collect counts for the conditional
pattern bases of *I* that meet the minimum

support criterion.
18: update CLT
20: end for21: end procedure

Algorithm 4 BuildCFTsFor*T-Id*

1: procedure BUILDCFTsFOR*T-Id*
2: Consult MTLT to check whether *T-Id* exists in memory
3: if the tree with id *T-Id* still exists in *M.B* then
4: make this tree active
5: load Header table *H* if it is already not in *M.B*
6: for each frequent item *I* on *D*o
7: build the FP-tree for *treeIdi*from the conditional memory-based FP-tree of *T-Id* following the *node-link* pointers starting from*H* _During the tree building phase, the
MMU may decide to save some of the existing trees or parts of the current tree on disc
8: update ILT and DTLT
9: if the tree with id *T-Id* does not exist in *M.B* anymore then
10: Mark the newly added blocks just saved on disc corresponding to this partial tree;
11: *lastTree= I*;
12: break;
13: end if
14: end for
15: end if
16: consult DTLT to identify blocks of the tree with id *T-Id*
17: for each block *b* (other than the marked blocks) of the prefix-tree corresponding to tree Id, *T-Id*do
18: load this block *b* into FB
19: for each freq. item *I* in *ILT* corres. to the tree *T-Id*do build the FP-tree for *treeIdi*from the conditional
20: pattern bases of *I* by traversing the discbased prefix-tree of *T-Id* in *b* following the
node-Ids loaded from ILT _During the tree building phase, the MMU may decide to save some of the on disc
21: update ILT and DTLT
22: end for
23: end for
24: for each marked block *b* of the prefix-tree corres. To tree, *T-Id*do
25: State load this block *b* into FB
26: for each freq. item *I* (*I* >*lastTree*) in *ILT* corres. to the tree *T-Id*do
27: build the FP-tree for *treeIdi*from the conditional pattern bases of *I* by traversing the
disc-based prefix-tree of *T-Id* in *b* following the node-Ids loaded from ILT
28: update ILT and DTLT
29: end for
30: end for

3.5: End Procedure

The system approach could use only a bounded portion of the primary memory and this gives the opportunity to assign other parts of the main memory to other tasks with different priority. In other words, we propose a specialized memory management system which caters to the needs of the ARM model in such a way that the proposed data structure is constructed in the available allocated primary memory first. If at any point the structure grows out of the allocated memory quota, it is forced to be partially saved on secondary memory. The secondary memory version of the structure is accessed in a block-by-block basis so that both the spatial and temporal localities of the I/O access are optimized. Thus, the proposed framework takes control of the virtual memory access and hence manages the required virtual memory in an optimal way to the best benefit of the mining process to be served. Several clever data structures are used to facilitate these optimizations.

The system approach is a platform independent framework capable of running ARM on a wide range of computer systems regardless of the specifications. Thesis possible because we have successfully decoupled the dependency on the VMM that is otherwise exhibited by the traditional approaches. This way, depending on the system load, we may limit the amount of memory to be used by the ARM model another parts of the main memory may be assigned to other tasks with different priorities. In other words, we propose a specialized memory management system which caters to the needs of the ARM model in such a way that the proposed data structures are constructed in the available allocated primary memory firsthand they are forced to be saved on the secondary memory in spatially conscious way if at any point, any structure grows beyond the allocated quota.

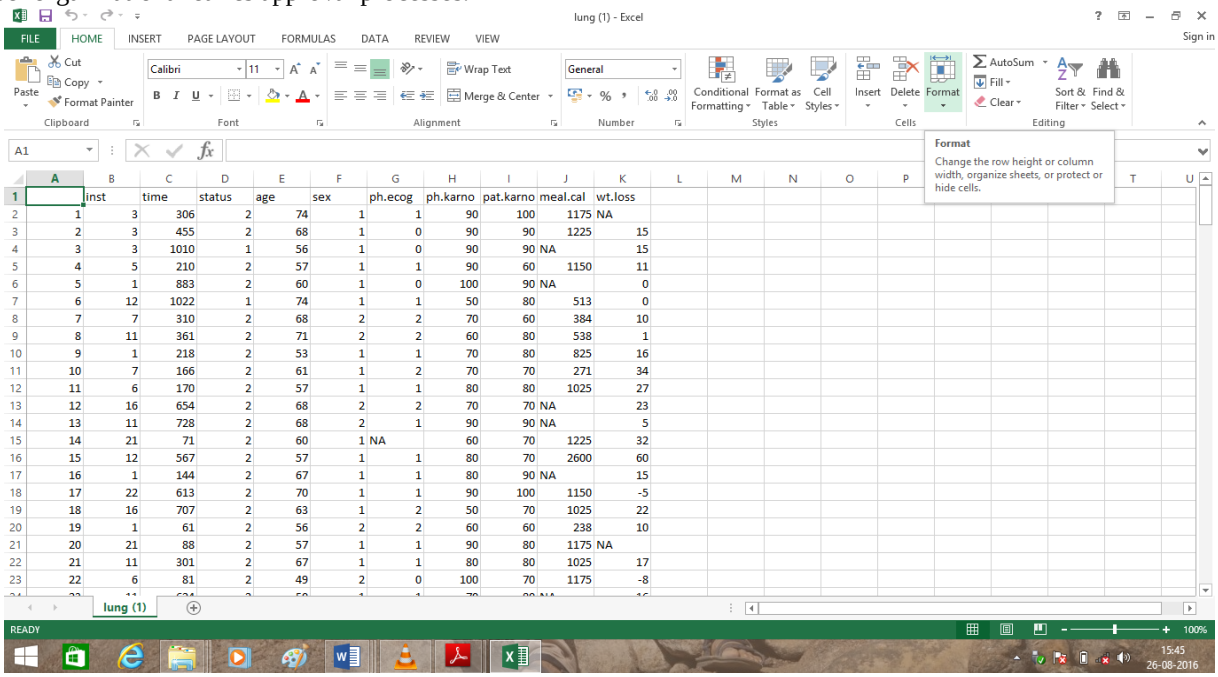
The memory management unit accesses the secondary memory version of the data structures in block-by-block basis with a prescribed guideline so that both spatial and temporal localities of the I/O access are optimized. The system do not sacrifice much in the multitasking capability of the working CPU because it does not exhaust the whole available main memory. The initial results presented in this paper exhibit the superiority of the proposed framework compared to the original FP-growth, which is one of the most attractive ARM algorithms described in the literature.

IV. EXPERIMENTAL RESULTS

The research study used samples from UCI repository. These samples were Wisconsin Breast Cancer Dataset (WBC) original, Wisconsin Breast Cancer Diagnosis (WDBC) and Wisconsin Breast Cancer prognosis (WPBC). WBC contained 699 records with each record having 9 features plus the class attributes. WDBC contained 569 records with each record having 32 features plus the class attribute while WPBC contained 198 records with each record having 34 features plus the class attribute and lung cancer 228 records.

4.1 Data Collection

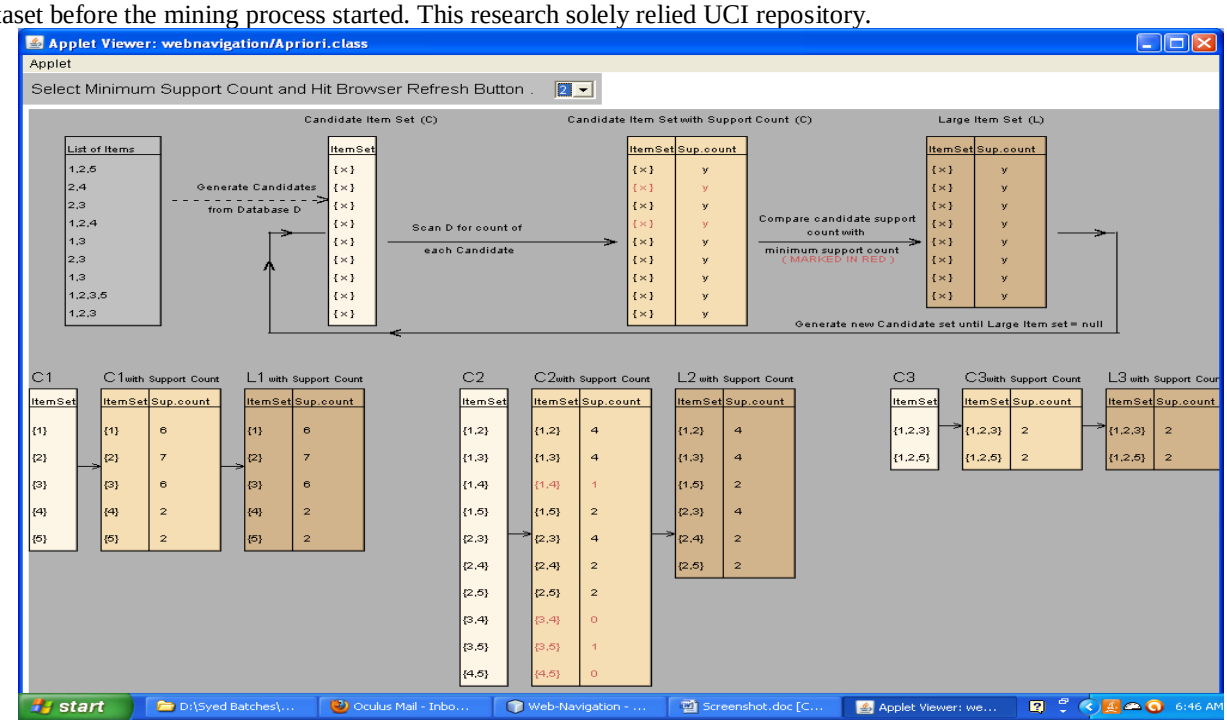
A high quality data was required for realizing best results, it was important therefore that its acquisition be highly reliant on the quality of the data collection process. The study relied on the utilization of UCI online databases available publicly for research purposes. Data in these databases were collected from clinical environment, and have undergone proper organizational ethics approval processes.



	inst	time	status	age	sex	ph.ecog	ph.karno	pat.karno	meal.cal	wt.loss
1	3	306	2	74	1	1	90	100	1175	NA
2	3	455	2	68	1	0	90	90	1225	15
3	3	1010	1	56	1	0	90	90	NA	15
4	5	210	2	57	1	1	90	60	1150	11
5	1	883	2	60	1	0	100	90	NA	0
6	12	1022	1	74	1	1	50	80	513	0
7	7	310	2	68	2	2	70	60	384	10
8	11	361	2	71	2	2	60	80	538	1
9	1	218	2	53	1	1	70	80	825	16
10	7	166	2	61	1	2	70	70	271	34
11	6	170	2	57	1	1	80	80	1025	27
12	16	654	2	68	2	2	70	70	NA	23
13	11	728	2	68	2	1	90	90	NA	5
14	21	71	2	60	1	NA	60	70	1225	32
15	12	567	2	57	1	1	80	70	2600	60
16	1	144	2	67	1	1	80	90	NA	15
17	22	613	2	70	1	1	90	100	1150	-5
18	16	707	2	63	1	2	50	70	1025	22
19	1	61	2	56	2	2	60	60	238	10
20	21	88	2	57	1	1	90	80	1175	NA
21	11	301	2	67	1	1	80	80	1025	17
22	6	81	2	49	2	0	100	70	1175	-8

Figure. 3 lung cancer dataset.

Feature selection was important in this research since it required pattern recognition, statistics, and data mining. The aim behind feature selection was to select a subset of record variables by ignoring features that possessed little or less importance. For example, a physician can make a decision based on some features i.e. whether a dangerous surgery is necessary for treatment or not. The study used feature selection methods to minimize the number of features in the dataset before the mining process started. This research solely relied UCI repository.



V. CONCLUSION

In this paper, a new approach WNPMS-ARM for diagnosing breast and lung cancer was put into test. The system was used to minimize the number of features. The reduced numbers of features were then applied as the new dataset to WNPMS-ARM. Further, *K-NN* and the distance functions were computed. The process iterated until it found the most suitable feature values that satisfied classification accuracy. The resulted computed indicated that ideally, No single features selection method best satisfies all datasets and learning algorithms.

REFERENCE

[1] N. Revathy And R. Amalraj, "Accurate Cancer Classification Using Expressions Of Very Few Genes", International Journal Of Computer Applications, Vol.14, No.4, 2010.

[2] Ritu Chauhan "Data clustering method for Discovering clusters in spatial cancer databases" International Journal of Computer Applications (0975-8887) Volume 10-No.6, November 2010.

[3] Rajashree Dash "A hybridized K-means clustering approach for high dimensional dataset" International Journal of Engineering, Science and Technology Vol. 2, No. 2, 2010, pp. 59-66.

[4] G. Rajkumar " Intelligent Pattern Mining and Data Clustering for Pattern Cluster Analysis using Cancer Data" International journal of Engineering Science and Technology Vol. 2(12), 2010, ISSN: 7459-7469

[5] SantanuGhorai, Anirban Mukherjee, SanghamitraSengupta, And Pranab K. Dutta, " Cancer Classification From Gene Expression Data By NPPC Ensemble", IEEE/Acm Transactions On Computational Biology And Bioinformatics, Vol. 8, No. 3, May/June 2011.

[6] K.Kalaivani "Childhood Cancer-a Hospital based study using Decision Tree Techniques" Journal of Computer Science 7(12): 1819-1823, 2011 ISSN: 1549-3636

[7] Shyi-Ching Liang, Yen-Chun Lee and Pei-Chiang Lee , "The Application of Ant Colony Optimization to the Classification Rule Problem", 2011 IEEE International Conference on Granular Computing.

[8] Arezoo Modiri and Kamran Kiasaleh," Permittivity Estimation for Breast Cancer Detection Using Particle Swarm Optimization Algorithm", 33rd Annual International Conference of the IEEE EMBS Boston, Massachusetts USA, August 30 - September 3, 2011

[9] Yao Liu and Yuk Ying Chung, "Mining Cancer data with Discrete Particle Swarm Optimization and Rule Pruning", IEEE 2011.

[10] HninWintKhaing," Data Mining based Fragmentation and Prediction of Medical Data", IEEE 2011.

[11] M. H. Mehta," Hybrid Genetic Algorithm with PSO Effect for Combinatorial Optimisation Problems", International Journal of Advanced Computer Research (IJACR), Volume-2 Number-4 Issue-6 December- 2012.

[12] Priyanka Dhasal, Shiv Shakti Shrivastava, Hitesh Gupta, Parmalik Kumar, "An Optimized Feature Selection for Image Classification Based on SVMACO", International Journal of Advanced Computer Research (IJACR) ,Volume-2 Number-3 Issue-5 September-2012.

[13] AmogRajenderan," Data Preparation for Web Mining A survey", International Journal of Advanced Computer Research (IJACR),Volume-2 Number-4 Issue-6 December-2012.

[14] PragatiShrivastava,Hitesh Gupta, "A Review of Density-Based clustering in Spatial Data", International Journal of Advanced Computer Research (IJACR) ,Volume-2 Number-3 Issue-5 September-2012.

[15] Boris Milovic "Prediction and Decision Making in Health Care using Data Mining" International Journal of Public Health Science Vol. 1, No. 2, December 2012, pp. 69-78 ISSN: 2252-8806

[16] S M Halawani "A study of digital mammograms by using clustering algorithms" Journal of Scientific & Industrial Research Vol. 71, September 2012, pp. 594-600.

[17] Charles Edeki "Comparative Study of Data Mining and Statistical Learning Techniques for Prediction of Cancer Survivability" Mediterranean journal of Social Sciences Vol 3 (14) November 2012, ISSN: 2039-9340.

[18] NeelamadhabPadhy "The Survey of Data Mining Applications and Feature Scope" Asian Journal of Computer Science and Information Technology 2:4(2012) 68-77 ISSN 2249-5126.

