I.J. Modern Education and Computer Science, **2016, 5, 64-70**
Published Online May 2016 in MECS (http://www.mecs-press.org/)
DOI: 10.5815/ijmecs.2016.05.08

A Novel Approach for Data Cleaning by Selecting the Optimal Data to Fill the Missing Values for Maintaining Reliable Data Warehouse

Raju Dara[1]
Research Scholar, Department of Computer Science and Engineering
Jawaharlal Nehru Technological University Kakinada, Andhra Pradesh, India
Email: rajurdara@gmail.com

Dr. Ch. Satyanarayana[2]
Professor, Department of Computer Science and Engineering
Jawaharlal Nehru Technological University Kakinada, Andhra Pradesh, India
Email: chsatyanarayana@yahoo.com

Dr. A. Govardhan[3]
Professor, Department of Computer Science and Engineering
Jawaharlal Nehru Technological University, Hyderabad. Telanagana, India
Email: govardhan_cse@yahoo.co.in

***Abstract*—**At present trillion of bytes of information is being created by projects particularly in web. To accomplish the best choice for business benefits, access to that information in a very much arranged and intuitive way is dependably a fantasy of business administrators and chiefs. Information warehouse is the main feasible arrangement that can bring the fantasy into reality. The upgrade of future attempts to settle on choices relies on upon the accessibility of right data that depends on nature of information basic. The quality information must be created by cleaning information preceding stacking into information distribution center following the information gathered from diverse sources will be grimy. Once the information have been pre-prepared and purified then it produces exact results on applying the information mining question. There are numerous cases where the data is sparse in nature. To get accurate results with sparse data is hard. In this paper the main goal is to fill the missing values in acquired data which is sparse in nature. Precisely caution must be taken to choose minimum number of text pieces to fill the holes for which we have used Jaccard Dissimilarity function for clustering the data which is frequent in nature.

***Index Terms*—**Apriori similarity function, Classification, Data Cleaning, Jaccard Dissimilarity function.

I. Introduction

Content grouping and arrangement has denoted a sensible significance in the field of information mining, manufactured neural systems, Bio-informatics [1]. It has been utilized differently as a part of different applications which include enhancing recovery productivity of data recovery frameworks, scanning huge report accumulations, sorting out the outcomes returned by a web index in light of client's question, creating scientific classification of web documents [2][3]. Bunching calculations might likewise be utilized as a part of recognizing the missing information or fields in social tables or database documents.

The primary issue with content documents is that the information is in the unstructured configuration in view of which a large portion of the presently existing database calculations don't make a difference for content bunching. The quality furthermore effectiveness of any information mining strategy is a component of commotion of the elements utilized for the procedure of bunching [5].

Dimensionality lessening is additionally one of the key issues in content bunching. The technique for dimensionality eliminating so as to lessen stop words, stemming words or utilizing systems has been managed in the past works [4].

The issue of making dimensionality lessening utilizing visit thing sets is gradually picking up significance from 2005. When we are with regular thing sets the closeness nature among content records is more related when contrasted with techniques expressed in [14]. Bunching is typically did utilizing two systems which can be named as: 1. Complete Strategy additionally called as Static Strategy, and 2. Incremental Strategy.

In the event of the incremental bunching approach [11], we might need to re-group the content records as and when another content document is considered for bunching. This incorporates discovering content record to bunch separation to put the new content document or content record into a current group or to shape another bunch.

The fundamental stride to be considered for outlining any bunching calculation is to first plan an appropriate similitude measure or utilizes any of the current closeness measures. This may be trailed by planning new bunching calculation or making utilization of any of such existing calculations. Using artificial neural networks, new features for instances will be built and the problem of intrusion detection will be mapped as a 10 feature problem, such feature creation and as features in new problem only have discrete values, in final classification decision tree will be used[12].

The idea of grouping utilizing least spreading over trees is composed by Charles Zhan in his compelling paper in 1971 over four decades back. The fig.1. beneath demonstrates the same disseminate plots of the illustrations of 2D information sets. This is still a test for specialists as no single bunching calculation has been created that are fit for recognizing same groups pretty much as people see them.

Highlights in a dataset need not be numerical dependably. They may be in any organization for instance literary, double, pictures. A suitable comparability measure should be considered for removing critical conclusions. The extricated elements may be notwithstanding being time arrangement.

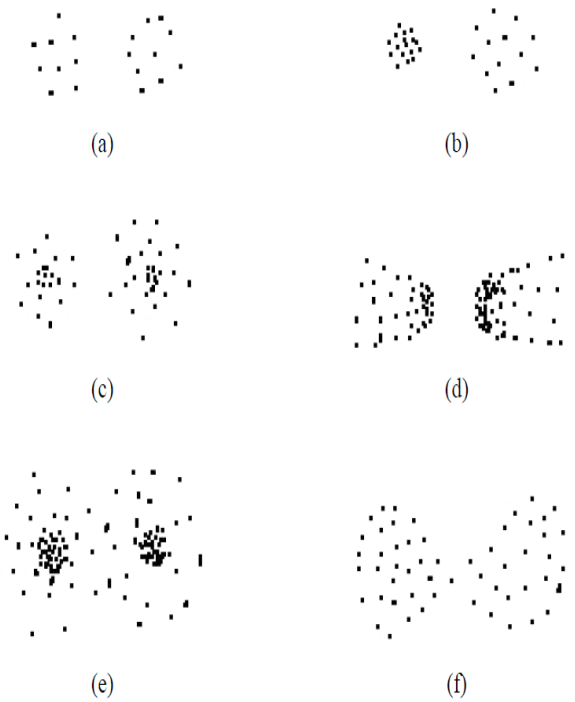

Fig.1. Sample speck examples of two bunches of differing densities and division - recreated from [Zahn, 1971].

A. Back Ground

Efficacy and enactment of text clustering algorithms can be enhanced by feature selection methods , authors offered an innovative feature selection method known as "Term Contribution (TC)" and applied a comparative study on a variety of feature selection methods for the clustering of text including Document Frequency (DF), Term Strength (TS), Entropy-based (En), Information Gain (IG), etc., eventually, outlined an "Iterative Feature Selection (IF)" methodology, which states the unattainability of label problem by effective supervised feature selection method for the selection of features iteratively to perform clustering [10]. Scholars proposed several induction principles and models wherein the corresponding optimization delinquent can be only almost solved by an even larger number of algorithms, thence, comparing clustering algorithms, must take into account a careful understanding of the inductive principles involved [13]. Initially derived a global criterion for feature clustering, subsequently presented a fast, divisive algorithm that monotonically decreases this objective function value, thus converging to a local minimum, presented detailed experimental results using Naive Bayes and Support Vector Machines on the 20 Newsgroups data set and a 3-level hierarchy of HTML documents collected from Dmoz Open Directory [15]. Authors presented a local search procedure called as first-variation, which refines a given clustering by incrementally moving data points between clusters, so that it achieves a higher objective function value. An enhancement of first variation allows a chain of such moves in a Kernighan-Lin fashion and leads to a better local maximum. Combining the enhanced first-variation with spherical k-means yields a powerful "ping-pong" strategy that often qualitatively improves k-means clustering and is computationally efficient [16].

Suggested a formal perspective on the difficulty in finding a unification, in the form of an impossibility theorem: for a set of three simple properties, it has been shown that there is no clustering function satisfying all three, relaxations of these properties expose some of the interesting (and unavoidable) trade-offs at work in well-studied clustering techniques such as single-linkage, sum-of-pairs, k-means, and k-median [18]. Generally, text representation inludes two tasks: indexing and weighting. This paper has comparatively studied TF*IDF, LSI and multi-word for text representation [19]. A new method called Maximum Capturing is proposed for document clustering, Maximum Capturing includes two procedures such as one is constructing document clusters and the next is assigning cluster topics, Proposed a normalization process based on frequency sensitive competitive learning for Maximum Capturing to merge cluster candidates into predefined number of clusters and finally the experiments are carried out to evaluate the proposed method in comparison with CFWS, CMS, FTC and FIHC methods [21].

As our abilities to collect and store various types of datasets are continually increasing, the demands for advanced techniques and tools to understand and make use of these large data keep growing. No single existing field is capable of satisfying the needs. Data Mining and Knowledge Discovery (DMKD), which utilizes methods, techniques, and tools from diverse disciplines, emerged in last decade to solve this problem [22].

Data received at the data warehouse from external sources usually contains various kinds of errors, e.g.

Spelling mistakes, inconsistent conventions across data sources, and/or Missing fields, Contradicting data, Cryptic data, Noisy values, Data Integration problems, Reused primary keys, Non unique identifiers, inappropriate use of address lines, Violation of business rules etc. The Decision Tree Induction Algorithm is used to fill the Missing Values in different data sources and also provided solutions to clean Dummy Values, Cryptic Values, and Contradicting data [24].

Scientists view bunching and characterization in a particular sense. Bunching is seen as an unsupervised learning process where no former or some or Apriori data exists on the classes of the information. Characterization then again is seen as a managed learning process which has pre-arranged preparing information. CART handles both categorical and continuous attributes to build a decision tree, which handles missing values and it uses Gini Index as an attribute selection measure to build a decision tree, the CART uses cost complexity pruning to remove the unreliable branches fromthe decision tree to improve the accuracy [25]. This paper focuses on bunching as opposed to arrangement.

In the section-2 a brief review on data cleaning task is explained, in section-3 we discussed the problem with missing data in a text, in section-4 a solution for the above mentioned problem is defined and in section-5 a brief note on experimentation and its results are presented followed by conclusion.

II. Review on Data Cleaning Task

Digital Fundamentals is the fundamental course for Computer Engineering as well as a few other Engineering degrees. Enrolments have been over 150 since 2008.

In this section a brief review on data cleaning task [6] [7] [8] is presented. It is divided into 6 steps as shown below.

- Data acquisition and metadata

Metadata is "information about information". Metadata is foundation data, which depicts the substance, quality, condition, and other suitable attributes of the information. Metadata can be sorted out into a few levels going from a basic posting of fundamental data about accessible information to itemized documentation around a singular information set or even individual components in a dataset. Metadata is especially required in GIS in light of the fact that data about spatial, topical and worldly substance is in numerical structure, in this manner pointless without substance code depiction. At the point when map information are in a computerized structure, Metadata is just as critical, however its improvement and upkeep regularly require a more cognizant exertion on the part of information makers and resulting clients who might adjust the information to suit their specific needs [9].

- Fill in missing values
  a. Ignore the tuple: usually done when class label is missing.
  b. Use the attribute mean (or majority nominal value) to fill in the missing value.
  c. Use the attribute mean (or majority nominal value) for all samples belonging to the same class.
  d. Predict the missing value by using a learning algorithm: consider the attribute with the missing value as a dependent (class) variable and run a learning algorithm (usually Bayes or decision tree) to predict the missing value.
- Combined date format
- Changing nominal to numeric
- Identify outliers and smooth out noisy data
  a. Binning
     i. Sort the characteristic values and parcel them into containers (see "Unsupervised discretization" beneath);
     ii. Then smooth by canister means, container middle, or receptacle limits.
  b. Clustering: bunch values in groups and afterward identify and uproot anomalies (programmed or manual)
  c. Regression: smooth by fitting the information into relapse capacities.
- Correct inconsistent data
  a. Use domain knowledge or expert decision

III. Why Missing Data is a Problem

In data cleaning process, after acquiring required data we often encounter with cases where the data is sparse [23]. As the traditional statistical algorithms assume that all entities in the assumed method are evaluated. To acquire accurate result of analysis we have to fill the missing data but care must be taken to fill the holes with minimal data. To achieve the same we bunch a given arrangement of content documents in order to make content records of comparable nature fall into one group and those of disparate nature into other arrangement of bunches. The way of does not bunch nor the quantity of groups is not known ahead of time and must be found.

IV. Proposed Method of Filling Values

We take Jaccard dissimilarity coefficient for finding similarity between any two text files, which can be extended to find similarity between any two sets of clusters by considering Jaccard dissimilarity measure. The main idea is to apply frequent pattern finding algorithm such as Apriori to the text files, and further reduce the dimensionality of the input text files as against to the earlier method of just eliminating the stop words and stemming words. In addition we follow the dynamic programming approach using a tabular method.

The similarity measure used is a function of frequent patterns instead of just normal words that appear in the text files. The concept of frequent patterns is considered as the text files with the similar frequent patterns are more related compared to the negative case.

Let

P = total frequent patterns in both files,
C = common frequent patterns in both the files.
N= number of text files
F=frequent item sets
T=text files

Then Jaccard Dissimilarity measure is given by the ratio defined by $J = \frac{(P-C)}{P}$.

Using above dissimilarity function, for each pair of text files form the dissimilarity table. Choose minimum value each time for deciding clusters to fill the missing values because the minimum value means that text files are most similar.

The proposed algorithm is as follows:

Algorithm: Text_Cluster (N, F, T)

1. Check if the input file is in .txt or .doc or .docx format. If not, convert it in to proper format.
2. For all text files T of the form .doc or .txt do
   2.1. Eliminate Stop words followed by Stemming words from each text file.
   2.2. Apply any exist , ing or newly designed algorithm to find the frequent item sets such as Apriori to further reduce the dimensionality.
   2.3. Define feature size of the problem instance equal to count of all the frequent patterns in step-1b.
3. Let f_1, f_2, $f_3 \ldots f_n$ be frequent item sets from each text file obtained after step1 and F be the corresponding feature set. Form a feature set consisting of unique frequent item sets from each file.

$$F = \{f_1, f_2, f_3, \ldots\ldots\ldots, f_n\} \quad (1)$$

4. Form a Dissimilarity Matrix of the order N X N with each row and column corresponding to each of the N text files.
5. For each text file in input file set do
   5.1. For each text file in input file set do
      5.1.1. Fill the corresponding cell value with computed dissimilarity measure between two text files for only upper triangular matrix elements.
6. At each step
   6.1. Find the cell with minimum value containing this value in the matrix.
   6.2. Group each such pairs to form the clusters.
   6.3. If there exist common text file for any two clusters, merge such clusters into a single new cluster.
7. Repeat Step-4 until no file pair exists to be clustered.
8. Output the set of clusters finally obtained after step 5.
9. Identify topics and Label the clusters by considering candidate entries.

The fundamental thought to apply pattern finding calculation Apriori to the content documents, and advance decreased the dimensionality of the data content records as against to the prior strategy for simply dispensing with the stop words and stemming words. Further dynamic programming approach was used in a tabular method to fill the holes in text.

V. Experimentation and Results

For experimentation we have considered the text file sets with the frequent pattern which are obtained after applying any one of the frequent pattern mining algorithm as in table 1.

The feature set is given by Feature Set = {Euclidean, Manhattan, Cluster, Classification, Mining, Sugar, Coffee, Football, Cricket, Tennis}

Now consider only the element of matrix with the minimum value as shown in the table 2., and fig.2

Table 1. Text files and Corresponding Frequent Item Sets

Text Files	Frequent Item Sets
Text File 1	{ Euclidean, Manhattan, Cluster}
Text File 2	{Classification , Mining, Cluster}
Text File 3	{ Classification, Manhattan, Mining, Cluster}
Text File 4	{Euclidean, Manhattan, Mining, Cluster}
Text File 5	{Sugar , Coffee}
Text File 6	{Euclidean, Manhattan, Mining}
Text File 7	{Football, Cricket, Tennis}

Table 2. Similarity Matrix Obtained By Jaccard Similarity Measure over Frequent Item Sets Obtained From Each Text File

	F1	F2	F3	F4	F5	F6	F7	F8	F9
F1	0	0.84	0.714	0.57	1.0	0.67	1.0	1.0	0.67
F2	0	0	0.57	0.714	1.0	0.833	1.0	1.0	0.67
F3	0	0	0	0.625	1.0	0.714	1.0	1.0	0.57
F4	0	0	0	0	1.0	0.57	1.0	1.0	0.57
F5	0	0	0	0	0	1.0	1.0	1.0	1.0
F6	0	0	0	0	0	0	1.0	1.0	0.67
F7	0	0	0	0	0	0	0	0.6	1.0
F8	0	0	0	0	0	0	0	0	1.0
F9	0	0	0	0	0	0	0	0	0

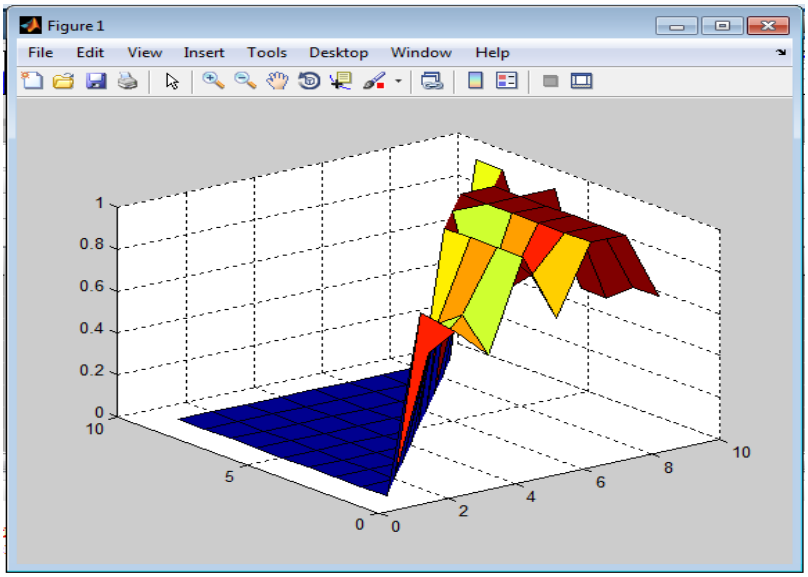

Fig.2. Three dimensional shaded surfaces obtained by using Jaccard Similarity Measure over Frequent Item Sets Obtained from each Text File

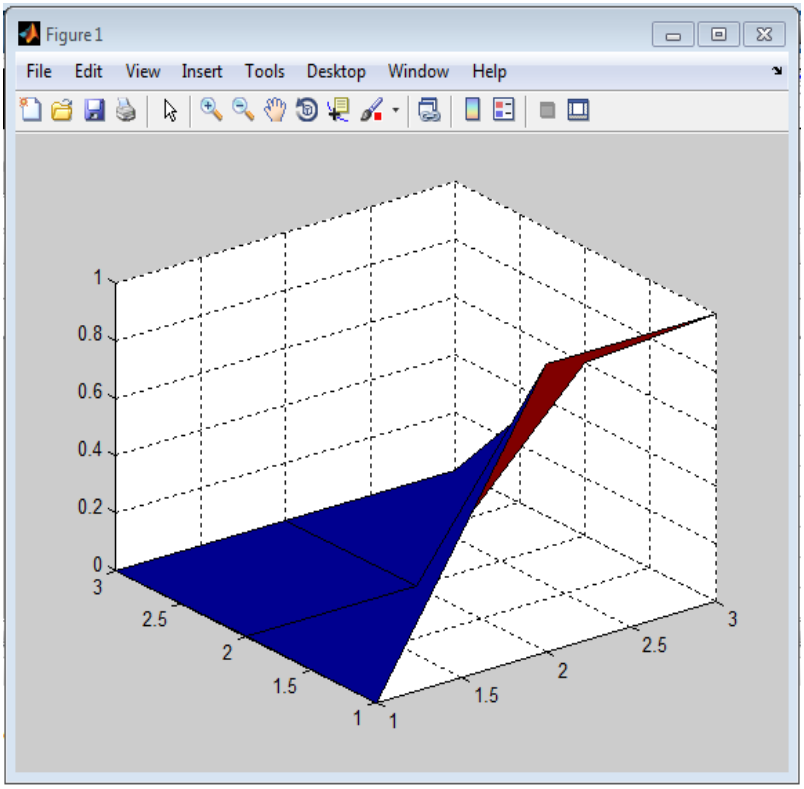

Fig.3. Three dimensional shaded surfaces obtained after applying step 2

Step1: Locate the first least esteem from the lattice and target just those cells having this quality to shape starting group. Here the cell values [1, 4], [2, 3], [3, 9], [4, 6], [4, 9] have minimum value of 0.57. Hence [1, 2, 3, 4, 6, 9] forms one cluster.

Step 2: Locate the following least esteem from the above table 2, which is 0.6 here and focus on those cells as they shape the best applicant arrangements as shown in table 3 and fig.3.

Table 3. Similarity matrix obtained after step 2

	F5	F7	F8
F5	0	1.0	1.0
F7	0	0	0.6
F8	0	0	0

Hence [7, 8] from one cluster

Step 3: Since the only left out file is 5 it is considered as a separate cluster. So the clusters formed finally are as shown below

Cluster 1: {1, 2, 3, 4, 6, 9}
Cluster 2: {7, 8}
Cluster 3: {5}

VI. Conclusion

The proposed calculation has the information as likeness lattice and yield being set of groups. We utilize the idea of successive examples to decrease the further dimensional of the content documents and afterward apply the Jaccard divergence measure. To accomplish more space proficiency, the utilization of existing incessant example mining calculation may be supplanted by any recently composed calculation which is more effective. We don't predefine the number of groups and content documents being fit to one of those bunches or classes at long last. The future extent of this work can be reached out to group utilizing classifiers, for example, SVM and applying fluffy rationale for naming the bunches framed.

References

[1] R.Agrawal, R.Srikant. "Fast algorithms for mining association rules" in the Proceedings of 20th International Conference on Very Large Data Bases, VLDB 1215,Pg. 487-499

[2] R.Agrawal, T.Imielinski, A. Swami. "Mining association rules between sets of items in large databases" in the proceedings of ACM SIGMOD Conference on managing data, 22(2), pg.207-216.

[3] R.Agrawal, R.Srikant. "Mining Sequential Patterns" in the proceedings of 11th International Conference on Data Engineering 1995.

[4] R.Agrawal, C.Faloutsos, A. Swami. "Efficient similarity search in sequence databases", Foundations of Data knowledge and Engineering, pg.69-84

[5] R Agrawal, JC Shafer. " Parallel mining of association rules". in the IEEE Transactions on Knowledge and Data Engineering, 1996, Vol8(6), pg.962-969.

[6] J Shafer, R Agrawal, M Mehta." SPRINT: A scalable parallel classifier for data mining, Proc. 1996 Int. Conf. Very Large Data Bases, 544-555

[7] M Mehta, R Agrawal, J Rissanen," SLIQ: A fast scalable classifier for data mining", Advances in Database Technology,1996, pg.18-32

[8] Narendra, Patrenahalli M. "A Branch and Bound Algorithm for Feature Subset Selection", IEEE Transactions on computers, Vol26 (9), 1977.

[9] Ari Frank, Dan Geiger, Zohar Yakhin. " A Distance-Based Branch and Bound Feature Selection Algorithm, pg.241-248., UAI2003

[10] Tao Liu, Shengping Liu, Zheng Chen. "An Evaluation on Feature Selection for Text Clustering", Proceedings of the 12th International Conference on Machine Learning (ICML-2003), Washington DC, 2003.

[11] Jung-Yi Jiang, Ren-Jia Liou, and Shie-Jue Lee, "Fuzzy Similarity-based Feature Clustering for Document Classification", proceedings of International conference on Information Technology and applications in Outlying Islands, pg.477-81, 2009.

[12] SAEED Khazaee, ALI Bozorgmehr, "A New Hybrid Classification Method for Condensing of Large Datasets: A Case Study in the Field of Intrusion Detection", MECS-I. J. Modern Education and Computer Science, April 2015, 4, 32-41, DOI: 10.5815/ijmecs.2015.04.04, (http://www.mecs-press.org/).

[13] Vladimir Estivill-Castro."Why so many clustering algorithms". ACM SIGKDD Explorations Newsletter, pg.65-72, 2002.

[14] C. Agarwal et.al. "A survey of text clustering algorithms", Text book on Mining Text data, Springer Publications, 2012.

[15] Inderjit S. Dhillon, Subramanyam Mallela, Rahul Kumar. "Enhanced Word Clustering for Hierarchical Text Classification, proceedings of ACM. KDD 2002.

[16] Inderjit S. Dhillon,Yuqiang Guan, J. Kogan, "Iterative Clustering of High Dimensional Text Data Augmented by Local Search", Proceedings of the Second IEEE International Conference on Data Mining, pages 131-138, Maebishi, Japan, December 2002.

[17] http://www.cs.utexas.edu/users/dml/

[18] Jon Kleinberg,"An Impossibility Theorem for Clustering", NIPS 15, pg.446-53, 2002.

[19] Chu, Xu, Ihab F. Ilyas, and Paolo Papotti, "Holistic data cleaning: Putting violations into context", Data Engineering (ICDE), 2013 IEEE 29th International Conference on. IEEE, 2013.

[20] Wen Zhanga,Taketoshi Yoshida, Xijin Tang "A comparative study of TF*IDF, LSI and multi-words for text classification" Expert Systems with Applications 38 (2011) 2758–2765.

[21] Wen Zhanga,, Taketoshi Yoshida, Xijin Tang, Qing Wang. "Text clustering using frequent item sets", Knowledge-Based Systems, Volume 23.Pg.379–388, 2010

[22] Yi Peng, Gang Kou. A Descriptive frame work for the field of data mining and knowledge discovery, Intr. Journal of Information Technology and Decision making, Volume 7, No.4, 2008, Pg.639- 682.

[23] Hellerstein, Joseph M, "Quantitative data cleaning for large databases", United Nations Economic Commission for Europe (UNECE) (2008).

[24] Raju Dara and Dr. Ch. Satyanarayana, "A Robust Approach for Data Cleaning used by Decision Tree Induction Method in the Enterprise Data Warehouse", International Journal on Computational Science & Applications (IJCSA) Vol.5, No.4, August 2015.

[25] T.Miranda Lakshmi, A.Martin, R.Mumtaj Begum, Dr.V.Prasanna Venkatesan, "An Analysis on Performance of Decision Tree Algorithms using Student's Qualitative Data", MECS-I. J. Modern Education and Computer Science, June 2013, 5, 18-27, DOI:10.5815/ijmecs.2013.05.03, (http://www.mecs-press.org/).

Authors' Profiles

Mr. Raju Dara is a Research Scholar of Department of Computer Science and Engineering, Jawaharlal Nehru Technological University, Kakinada. He has 8 years of teaching experiences for Graduate and Post Graduate engineering courses. His current research interests are Data Warehousing, Image Processing. He published 8 research papers in international journals and 3 research papers in international conferences.

Dr. Ch. Satyanarayana is a Professo r in Department of Computer science and Engineering at Jawaharlal Nehru Technological University Kakinada. He completed B. Tech and M.Tech in computer science and engineering from Andhra University, Visakha Patnam, Andhra Pradesh. He was awarded his Doctoral degree in 2008 from J.N.T. University, Hyderabad. He has 15 years of experience. His areas of interest are Image Processing, Databases, Pattern Recognition and Network Security. He published more than 21 research papers in international journals and more than 100 research papers in international conferences. He has guided 15 Research scholars are working on different areas like Image Processing, Speech Recognition, and Pattern Recognition. He guided more than 78 M.Tech Projects, 56 MCA Projects, and 36 B.Tech Projects.

Dr.A.Govardhan is a Professor in Department of Computer science and Engineering and held many prestigious positions including principal of University college of Engineering (JNTUH), Director of Evaluations, Director of school of Information Technology, etc., at Jawaharlal Nehru Technological University, Hyderabad, Telangana State. He has 20 years of experience. His areas of interest are Data Warehousing & Mining, Image Processing, and Databases. He did B.E. (CSE) from Osmania University College of Engineering, Hyderabad in 1992, M.Tech from Jawaharlal Nehru University (JNU), New Delhi in 1994 and Ph.D from Jawaharlal Nehru Technological University, Hyderabad in 2003. He has 2 Monographs by Lambert Academic Publishing, Germany and Published in USA. He has guided 54 Ph.D theses, 125 M.Tech projects and he has published 350 research papers at International/National Journals/Conferences including *IEEE, ACM, Springer and Elsevier*.

