BVG Trust

J. Sci. Trans. Environ. Technov., 2016, 10(2) : 92-97

Scientific Transactions in Environment and Technovation

Efficient optimized similarity cluster search in high dimensional spaces

https://doi.org/10.56343/STET.116.010.002.009
http://stetjournals.com

T. Tamilselvi[1], V. Geetha[1] and M.V. Srinath[2]

[1]Department of Computer Science, S.T.E.T. Women's College, Sundarakkottai, Mannargudi, Thiruvarur Dt., Tamilnadu, India.
[2]Department of Master of Computer Application, S.T.E.T. Women's College, Sundarakkottai, Mannargudi, Thiruvarur Dt., Tamilnadu, India.

Abstract

The recent evolution in Social Networking Services (SNSs) data mining server such as Facebook and Twitter are getting more popular and analyzing social network data has become one of the most important issues in various areas. Among those analysis jobs, community detection from social network data gains much attention from academia and industry since it has many real-world applications such as friend recommendation and target marketing.

This proposed technique EOSCS (Efficient Optimized Similarity Cluster Search) in High Dimensional Spaces to detect the better community structure in big data mining. Community detection is to partition the set of network nodes into multiple groups such that the nodes within a group are connected densely, but connections between groups that are presented in the vertex. In first probe the path between every pair of nodes with trivial and non trivial to predecessor nodes, then to calculate each pair of nodes in "weight between's" and the every pair are interlinked. The minimized path length of interlink nodes verified by time and data weight. This proposed techniques delete the edges with maximum nodes count by which node more information they allocate by rank. The experiment results show the shortest map when compared to the existing ones.

Keywords: Clustering, Filter, Graph Based Clustering

INTRODUCTION

The feature subset selection is an effective way for reducing dimensionality, removing irrelevant data, increasing learning accuracy and improving result comprehensibility. Many feature subset selection methods have been proposed and studied for machine learning applications. They can be divided into four broad categories: the Embedded, Wrapper, Filter, and Hybrid approache as described by Das, (2001). The embedded methods incorporate feature selection as a part of the training process and are usually specific to given learning algorithms, and therefore may be more efficient than the other three categories.

Traditional machine learning algorithms like decision trees or artificial neural networks are examples of embedded approaches. The wrapper methods use the predictive accuracy of a predetermined learning algorithm to determine the goodness of the selected subsets, the accuracy of the learning algorithms is usually high. However, the generality of the selected features is limited and the computational complexity is large. The filter methods are independent of learning algorithms, with good generality. Their computational complexity is low, but the accuracy of the learning algorithms is not guaranteed. The hybrid methods are a combination of filter and wrapper methods by using a filter method to reduce search space that will be considered by the subsequent wrapper.

It mainly focuses on combining filter and wrapper methods to achieve the best possible performance with a particular learning algorithm with similar time complexity of the filter methods. The wrapper methods are computationally expensive and tend to over fit on small training sets. The filter methods, in addition to their generality, are usually a good choice when the number of features is very large. Thus, we will focus on the filter method in this paper. With respect to the filter feature selection methods, the application of cluster analysis has been demonstrated to be more effective than traditional feature selection algorithms and hence are applied in the distributional clustering of words to reduce the dimensionality of text data. In cluster analysis, graph theoretic methods have been well studied and used in many applications (Demsar,2006). The results have, sometimes the best agreement with human performance. The general graph theoretic clustering is simple: Compute a neighborhood graph of instances, then delete any edge in the graph that is much longer / shorter (according to some criterion) than its neighbors. The result is a forest and each tree in the forest represents a cluster. In this study, apply the graph theoretic clustering methods to features.

*Corresponding Author :
email: *t.tamil.selvi190@gmail.com*

In particular, it adopts the Minimum Spanning Tree (MST) based clustering algorithms, because it do not assume that data points are grouped around centers or separated by a regular geometric curve and have been widely used in practice. Based on the MST method, this propose a attribute based Fast clustering based feature Selection algorithm EOSCS (Hall, 1999).The EOSCS algorithm works in two steps. In the first step, features are divided into clusters by using feedback verification clustering methods. In the second step, the most representative feature that is strongly related to target classes is selected from each cluster to form the final subset of features.

Features in different clusters are relatively independent; the clustering based strategy of FAST has a high probability of producing a subset of useful and independent features. The proposed feature subset selection algorithm EOSCS was tested upon 35 publicly available image, microarray, and text data sets. The experimental results show that, compared with other five different types of feature subset selection algorithms, the proposed algorithm not only reduces the number of features, but also improves the performances of the four well known different types of classifiers.

Data Collection

Finding scholarly information on the World Wide Web can be very frustrating. There is no way to search through a large selection of only scholarly sites with the current Web search tools. The existing search tools provide search algorithms that shift through millions of Web pages with no way to limit the search to a category of Web sites. Nobody seems to know how to do any automatic filtering for quality of Web sites. However, librarians have been doing quality filtering of materials for many years, but no one seems conscious of the standards carefully developed by information professionals over the past century (Friedman, 1940).

In the print world, the academic library performs this filtering function by providing patrons with a subset of print works pertaining to academia. This selection role is filled by library staff members using either explicit or tacit criteria to select individual works. Some sites, such as the Internet Public Library (http://www.ipl.org), attempt to select scholarly sites. However, because of the rapid introduction of new documents on the World Wide Web, a human cannot keep up and the resource is quickly outdated.

In order to handle the vast number of documents on the Web, an automated selection system is needed. First, the criteria used by academic librarians to select print works will be examined. These criteria can be translated into equivalent criteria for Web pages. A Web robot can then be designed to determine these criteria for a page. After creating a training set of examined Web pages with their selection decisions, data mining techniques can be used to create a classification model that will be a quality filter for Web pages.

Most of the existing works are motivated by a commonly performed task in the biomedical domain (Yang *et al*., 2007; Apache Hadoop, 2013) that of constructing a systematic review. Authors of systematic reviews seek to identify as much as possible of the relevant literature in connection with some aspect of medical practice, typically a highly specific clinical question. The review's authors assess, select, and synthesize the evidence contained in a set of identified documents, to provide a "best currently known" summary of knowledge and practice in that field.

The collections used as the source material are already large, and continue to grow. For example, as at end of 2009, MEDLINE, the largest of the available collections, contained more than 19 million entries, with more than 700,000 citations having been added during the year. To construct each systematic review, a complex Boolean search query is used to retrieve a set of possibly relevant documents (typically in the order of one to three thousand) which are then comprehensively triaged by multiple assessors. (Dean and Ghemawat, 2008; Chaiken *et al*., 2008)

Recently, hierarchical clustering Cohen *et al*. (2009) has been adopted in word selection in the context of text classification. Distributional clustering has been used to cluster words into groups based either on their participation in particular grammatical relations with other words by Pereira or on the distribution of class labels associated with each word.

As distributional clustering of words is agglomerative in nature, and result in sub-optimal word clusters and high computational cost, it shows a new information-theoretic divisive algorithm for word clustering and applied it to text classification. It proposed to cluster features using a special metric of Barthelemy distance, and then makes use of the dendrogram of the resulting cluster hierarchy to choose the most relevant attributes. Unfortunately, the cluster evaluation measure based on Barthelemy distance does not identify a feature subset that allows the classifiers to improve their original performance accuracy. Furthermore, even compared with other feature selection methods, the obtained accuracy is lower.

CFD Method

Conditional Functional Dependencies (CFDs) were recently introduced for data cleaning. They extend standard Functional Dependencies (FDs) by enforcing patterns of semantically related constants. CFDs have been proven more effective than FDs in detecting and

repairing inconsistencies (dirtiness) of data and are expected to be adopted by data cleaning tools that currently employ standard FDs for surveys on data cleaning tools.

However, CFD-based cleaning methods to be effective in practice, it is necessary to have techniques in place that can automatically discover or learn CFDs from sample data, to be used as data cleaning rules. Indeed, it is often unrealistic to rely solely on human experts to design CFDs via an expensive and long manual process. As indicated in cleaning-rule discovery is critical to commercial data quality tools.

This practical concern highlights the need for studying the discovery problem for CFDs; given a sample instance r of a relation schema R, it is to find a canonical cover of all CFDs that hold on r, i.e., a set of CFDs that is logically equivalent to the set of all CFDs that hold on r. To reduce redundancy, each CFD in the canonical cover should be minimal, i.e., nontrivial and left-reduced (for nontrivial and CFDs).

The discovery problem is, however, highly nontrivial. It is already hard for traditional FDs since, among other things, a canonical cover of FDs discovered from a relation r is inherently exponential in the arity of the schema of r, i.e., the number of attributes in R. Since CFD discovery subsumes FD discovery, the exponential complexity carries over to CFD discovery. Moreover, CFD discovery requires mining of semantic patterns with constants, a challenge that was not encountered when discovering FDs, as illustrated by the example below.

Data Archive using OLAP

The discovery problem has been studied for FDs for two decades in the previous research papers for database design, data archiving, OLAP (Online Analytical Processing), and data mining. It was first investigated in miner papers, which shows that the problem is inherently exponential in the arity of the schema R of sample data r. One of the best-known methods for FD discovery is TANE, a level wise algorithm that searches an attribute-set containment lattice and derives FDs with k þ 1 attributes from sets of k attributes, with pruning based on FDs generated in previous levels. TANE takes linear time in the size of input sample r, and works well when the arity jRj is not very large. The algorithms of CFDs follow a similar level wise approach.

However, the level wise algorithms may take exponential time in jRj even if the output is not exponential in jRj. In light of this, another algorithm, referred to as Fast FD , explores the connection between FD discovery and the problem of finding minimal covers of hyper graphs, and employs the depth-first strategy to search minimal covers. It takes (almost) linear time in the size of the output, i.e., in the size of the FD cover.

PRUNING DATA USING PRE COMPUTATION TECHNIQUE

To reduce similarity computation effort, the notion of pruning was introduced by (Buckley and Lewit et al., 1985) for term-at-a-time processing for document-at-a-time processing. These authors reasoned that a system that correctly identifies the top *r* documents is no less useful than one that completely scores the whole collection. In the case of the term-at-a time approach of Buckley and Lewit, whole query terms might be dropped as a result of pruning and when they are, both processing time and disk transfer time can be saved.

On the other hand, in the case of the document-at-a-time approach of Turtle and Flood, some pointers might be dispensed with after just a cursory amount of processing, saving overall processing costs but all inverted lists must be fetched. Moffat *et al*. (2001) described a mechanism for inserting additional information called "skips" into document-sorted compressed inverted lists in order to support a term-at-a-time processing strategy that they called CONTINUE. Skips are forward pointers within a compressed inverted list, and allow unnecessary sections to be passed over with minimal effort, and then decoding resumed. The other key aspect of the CONTINUE approach is the notion of OR-mode and AND-mode processing of index pointers.

If a pointer to some document is processed in OR-mode, then it has the authority to nominate this document as being a potential answer, and have it considered by subsequent processing steps, even if no other query terms appear in it. Every document that is eventually scored and ranked must have been nominated by a pointer processed in OR-mode. On the other hand, pointers processed in AND-mode are permitted to boost the scores of previously nominated documents, but are not allowed of themselves to nominate documents. If all of the index pointers corresponding to some document are processed in AND-mode, then that document will not be scored, and will not be considered as a candidate answer.

FUZZY SET BASED TOP CLUSTERING

Fuzzy systems are designed to provide customer support through a range of different technologies and Information Retrieval (IR) tools play a fundamental role in this activity. Efficiency and effectiveness in data retrieval are crucial for the overall problem solution process but they depend on the infrastructure data are stored into and the correspondent abstraction model.

The abstraction associated with an object should capture all its peculiarities into an easily manageable representation but deciding which the "relevant"

features of an object are complex and uncertainty makes this task even harder. Focusing on Information Retrieval system, implementation issues are critical both for the overall performance of the system and the accuracy of the retrieved information. Customers usually provide data with different degrees of confidence depending on how that information has been collected.

Current Information Retrieval tools do not explicitly model the uncertainty associated to information but they "mix" the measure of relevance associated to information with the relative measure of confidence. They don't even manage the feedback provided by users about the accuracy and usefulness of the retrieved solutions. The explicit management of relevance and confidence on information, integrated with an adaptively process is the key factor to improve the retrieval precision of a help desk system.

PROPOSED MODEL

Feature selection process is the vital one in the architecture of data retrieval process in web mining. It involves identifying a subset of the most useful features that produces compatible results as the original entire set of features. This proposed algorithm EOSCS is evaluating from both the efficiency and effectiveness points of view. While the efficiency concerns the time required to find the multiple attribute based feature selection, the effectiveness is related to the quality of the mechanism designed to perform the feature selection.

Based on the proposed idea, attribute based fast clustering-based feature selection algorithm EOSCS is proposed and going to experiments with different parameter set. The **EOSCS** algorithm works in three steps. It exploits the concept of edge betweeness to divide a network into multiple communities. Though it is being widely used, it has limitations in supporting large-scale networks since it needs to calculate the shortest path between every pair of nodes in a network. In this technique develop a parallel version of the Group Node (GN) algorithm to support large-scale networks. This proposed technique, which we call Shortest Path Betweenness of MapReduce Algorithm EOSCS that utilizes the MapReduce model (Zeng *et al*., 2012). This algorithm consists of four major stages, and all operations are executed in parallel. It also suggest an approximation technique to further speed up community detection processes.

The **EOSCS** algorithm works in three steps. In the first step, features are divided into clusters by using graph-theoretic clustering methods. In the second step, the most representative feature that is strongly related to target cluster classes is selected from each cluster to form an attribute based classes. Features in different clusters are relatively either dependent or independent, the clustering based strategy of EOSCS has a high probability of producing a subset of useful and independent features. To ensure the efficiency of FAST, we adopt the efficient Minimum-Spanning Tree (MST) clustering method (Cohen *et al.,* 1995). The efficiency and effectiveness of the EOSCS algorithm are evaluated through an empirical study. The third step is feature selected data are verified with the true database server that are driven from the attribute based cluster of classes. So this system shows the better performance than the existing FAST, Fast Correlation Based Filter (FCBF) and Binary Tree (Btree) based systems (Fleuret,2004) . It also tested in the most popular open-source platform for MapReduce, and then conducted performance tests for SPB-MRA on Amazon EC2 instances. The results showed that elapsed time decreases almost linearly as the number of reducers increases and the approximation technique introduces negligible errors.

SIMILARITY CLUSTER FORMATION IN MINING SERVERS

In this module, we extract the similar document from the data set based on the given Boolean query. The similar document is extracted using Trem Frequency-Inverse Document Frequency (TF-IDF) values. Compute the similarity score for the given query and the data set. Get the highest similarity score document.

WEIGHT COMPUTATION USING GN

In this module, shows compute the score (weight age) for each nodes in a data set from various database servers. The recursive nature of EOSCS queries makes it necessary to calculate the scores on lower levels in the query tree first. One obvious possibility would be to try and add processing logic to each query node as it acts on its clauses. But optimizations such as max-score could only be employed at the query root node, as a threshold is only available for the overall query score. Instead, It follow a holistic approach and prefer to be able to calculate the document score given a set of query terms $S \subseteq T$ present in a document, no matter where they appear in the query tree.

SHORTEST DISTANCE EDGE DETECTION

To provide early termination of each node weight scoring, It also propose the use of term independent score bounds that represent the maximum attainable score for a given number of terms. A lookup table of score bounds M_r is created, indexed by r that is consulted to check, if it is possible for a candidate document containing r of the terms to achieve a score greater than the current entry threshold. That is, for each (r = 1. . . n) we seek to determine,

$$M_r = \max\{CalcScore(S,B) \mid S \text{ d"T and } \mid S \mid = r\}$$

The number of possible term combinations that could occur in nodes is $O(\binom{n}{r})$ for each r, which is $O(2^n)$ in total, and a demanding computation. However, the

scoring functions only depend on the clause scores, that is, the overall score of each sub-tree, meaning that the problem can be broken down into subparts, solved for each sub-tree separately, and then aggregated. The simplest (sub) tree consists of one term, for which the solution is trivial. For a particular operator node with clauses C, let n_c denote the number of terms in the sub-tree of clause c ε C. A table with **r = 0,...,**$\sum_{c \in C} n_c$ possible terms present is then computed and to compute each M_r, all possibilities to decompose r into a sum over the clauses r ¼ P c2C rc $\gamma = \sum_{c \in C} r_c$ have to be considered.

OPTIMIZED MAP REDUCING USING EOSCS

In this module, it gets the top k nodes for the give correlated clustering query by Gates *et al*., (2009). Our objective is to construct a query sequence q1, q2... qv of EOSCS return data queries that can be submitted to the database, retrieve as few data as possible and still contain all the documents that would be in the top-k results.

SYSTEM ARCHITECTURE

PROPOSED ALGORITHM – EOSCS

Inputs:D (F1, F2……..Fm, C) - the given data set

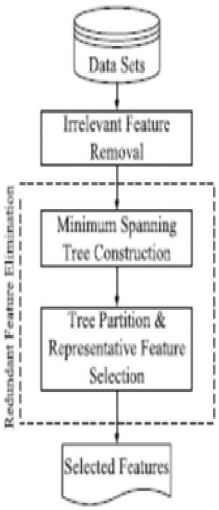


Fig.1. Shows architecture of EOSCS

è the T- Relevance threshold.

Output:S- selected feature subset.

/ / = = = = **part 1:**Irrelevant Feature Removal = = = =

1. Fori=1 to mdo

2. T- Relavance = SU (Fi,C)

3. if T- Relavance >θ then

4. S = SU{ Fi };

/ / = = = = **part 2:**minimum spanning tree construction = = = =

5. G = NULL; / / G is a complete graph

6. For each pair of features{F'i,F'j} ⊂ S do

7. F – Correlation = SU (F'i, F'j)

8. Add F'i and / or F'j to G with F – Correlation as the weight of

The corresponding edge;

9. Min Span Tree = prime (G); / / Using prime algorithm to generate the Minimum spanning tree

/ / = = = = **part 3:**Tree partition and Representative feature selection = = = =

10. Forest = min Span Tree

11. For each edge Eij ε Forest **do**

12. if

SU(F'i, F'j) < SU (F'i, C)∧ SU (F'i, F'j) < SU (F'i, C) **then**

13. Forest = Forest – Eij

14. S = Ø

15. For each tree Ti ε Forest do

16. F^j_R = argmaxF'_kε T_iSU (F'_K, C)

17. S = S U{ F^j_R};

18. Returns S

RESULTS AND DISCUSSION

This section presents the experimental results in terms of the proportion of selected features, the time to obtain the feature subset, the classification accuracy, and the Win / Draw / Loss record. In the experiment, for each feature subset selection algorithm, we obtain M×N feature subsets and the corresponding runtime Time with each data set. Average Subset and Time,obtain the number of selected features further the proportion of selected features and the corresponding runtime for each feature selection algorithm on each data set.

For each classification algorithm, this system EOSCS obtain M×N classification Accuracy for each feature selection algorithm and each data set. Average this Accuracy, it obtains mean accuracy of each classification algorithm under each feature selection algorithm and each data set. The procedure experimental process comparative results show the graph.

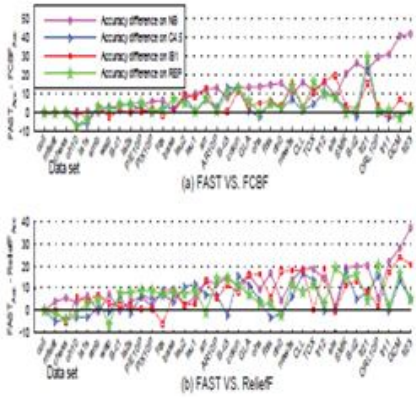
For each of the four classification algorithms, although the θ values where the best classification accuracies are obtained are different for different data sets, the value of 0.2 is commonly accepted because the corresponding classification accuracies are among the best or nearly the best ones.

When determining the value of θ, besides classification accuracy, the proportion of the selected features should be taken into account as well. This is because improper

proportion of the selected features results in a large number of features is retained and further affects the classification efficiency.

CONCLUSION

This technique EOSCS had been done successful implementation in high dimensional database servers.



Graph 1. Shows the accuracy differences between ABFAST and comparative algorithms.

The algorithm involves (i) removing irrelevant features, (ii) constructing a minimum spanning tree from relative ones, and (iii) partitioning the MST and selecting representative features by verify it with the true data verification. Based on the proposed idea, attribute based fast clustering-based feature selection algorithm (EOSCS) is proposed and going to experiments with different parameter set. The EOSCS algorithm works in three steps. It exploits the concept of edge between to divide a network into multiple communities. Though it is being widely used, it has limitations in supporting large-scale networks since it needs to calculate the shortest path between every pair of nodes in a network. In this technique develop a parallel version of the GN algorithm to support large-scale networks. This proposed technique, which we call Shortest Path between of Map Reduce Algorithm EOSCS that utilizes the Map Reduce model.

FUTURE ENHANCMENT

In future, this model for feature selection from the high dimensional database systems will have been implemented and tested with the different set of parameters. From the analysis the above technique performs very well on the microarray database servers. The reason lies in both the characteristics of the data set itself and the property of the proposed algorithm. For the purpose of exploring the relationship between feature selection algorithms with high intensity of data volume, in which algorithms are more suitable for which types of data, it ranks the six feature selection algorithms according to the classification accuracy of a given classifier on a specific type of data after the feature selection algorithms are performed.

REFERENCES

Apache Hadoop, http:/ /hadoop.apache.org/accessed:August 1,2013.

Buckley,C. and Lewit, A.F. 1985. Optimization of Inverted Vector Searches In: *Processdings of the 8th Annuval International ACM SIGIR Conference on Research and Development in Information Retrieval,* P.97-110. https://doi.org/10.1145/253495.253515

Chaiken, F., Jenkins, B.,Larson P.A, Ramsey, B., Shakib, D., Weaver, S., and Zhou, J. 2008. Scope: Easy and efficient parallel processing of massive data sets, *Proceedings of the VLDB Endowment,* P.1265–1276. https://doi.org/10.14778/1454159.1454166

Cohen, J., Dolan, B., Dunlap, M., Hellerstein, J. M. and Welton, C. 2009. Mad skills: new analysis practices for big data, *Proceedings of the VLDB Endowment,* P. 1481–1492. https://doi.org/10.14778/1687553.1687576

Das, S. 2001. Filters, wrappers and a boosting based hybrid for feature Selection, In:*Proceedings of, the Eighteenth International Conference on Machine Learning*, P. 74-81.

Dean, J. and Ghemawat, S. 2008. Map Reduce: simplified data processing on large clusters *Communications of the ACM*, 51(1): 107–113. https://doi.org/10.1145/1327452.1327492

Demsar , J. 2006. Statistical comparison of classifiers over multiple data sets, *J.Mach. Learn.,* P.1-30.

Fleuret, F. 2004. Fast binary feature selection with conditional mutual Information,*Journal of Machine Learning Research*, 5,(2004), P. 1531-1555.

Friedman, M. 1940. A comparision of alternative tests of the siginifigance for the problem of ranking, *Ann. Math. Statist.,*11:86-92. https://doi.org/10.1214/aoms/1177731944

Gates, A. F., Natkovich, O., Chopra, S., Kamath, P., Narayanamurthy, S. M., Olston, C., Reed, B., Srinivasan, S., and Srivastava, U. 2009. Building a high-level dataflow system on top of Map-Reduce: the Pig experience, *Proceedings of the VLDB Endowment*, P. 1414–1425. https://doi.org/10.14778/1687553.1687568

Hall, M.A. 1999. *Correlation-Based Feature Subset Selection for Machine Learning*, Ph.D. dissertation Waikato, New Zealand: Univ.Waikato.

Moffat, J., Liu, H. and Motoda,H. 2001. Consistency based hybrid for feature Selection In: *Proceedings of the Eighteenth International Conference on Knowledge Discovery and Data mining*, P.74-81.

Yang, H. E., Dasdan, A., Hsiao, R. L.and Parker, D. S. 2007. Map-Reduce-Merge: simplified relational data processing on large clusters, In: *Proceedings of 2007 ACM SIGMOD International Conference on Management of Data*, P. 1029–1040. https://doi.org/10.1145/1247480.1247602

Zeng, Z. F., Wu, B. and Zhang, T. T. 2012. A multi-source message passing model to improve the parallelism efficiency of graph mining on MapReduce, In: *Proceedings of 2012 IEEE International Parallel and Distributed Processing Symposium Workshops & PhD Forum (IPDPSW)*, P. 2019–2025.PMid:23705370 https://doi.org/10.1109/IPDPSW.2012.251