A FAST CLUSTERING BASED FLEXIBLE AND ACCURATE MOTIF DETECTOR TECHNIQUE FOR HIGH DIMENSIONAL DATA

[1]N.Deepika, [2]R.Saravana Kumar
[1] PG Scholar, Department of Information Technology, Jayam College of Engineering and Technology,Dharmapuri.
[2]Assistant Professor, Department of Information Technology, Jayam College of Engineering and Technology, Dharmapuri.
[1] deepirajan@gmail.com, [2]saravanakumar.surya@gmail.com

Abstract: As DNA samples are taking as datasets to analyse data effectively with a novel motif mining algorithm called Flexible and Accurate Motif detector (FLAME) technique that uses a concurrent traversal of two suffix trees to efficiently explore the space of all motifs. We present an algorithm that uses FLAME as a building block and can mine combinations of simple approximate motifs under relaxed constraints. The approach we take in FLAME explores the space of all possible models. In order to carry out this exploration in an efficient way, we first construct two suffix trees: a suffix tree on the actual data set that contains counts in each node (called the data suffix tree), and a suffix tree on the set of all possible model strings (called the model suffix tree). To get effective and accurate motif detection.

Keywords: Flame, Motif, Datasets, Suffix tree, K-Nearest Neighbours.

1. INTRODUCTION

With the aim of choosing a subset of good features with respect to the target concepts, 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 approaches. 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 limit 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 wrapper methods are computationally expensive and tend to overfit 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. 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.

In cluster analysis, graph-theoretic methods have been well studied and used in many applications. Their 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 our study, we apply graph theoretic clustering methods to features. In particular, we adopt the minimum spanning tree (MST) based clustering algorithms, because they do not assume that data points are grouped around centers or separated by a regular geometric curve and have been widely used in practice. FLAME travels via the data suffix tree and model suffix tree once it attains its target without wasting time by traveling to the end of the trees, stops traversing at that point where it attains its target

2. RELATED WORK

Feature subset selection can be viewed as the process of identifying and removing as many irrelevant and redundant features as possible. This is because:

- irrelevant features do not contribute to the predictive accuracy, and
- redundant features do not redound to getting a better predictor for that they provide mostly information which is already present in other feature(s). Of the many feature subset selection algorithms, some can effectively eliminate irrelevant features but fail to handle redundant features yet some of others can eliminate the irrelevant while taking care of the redundant features.

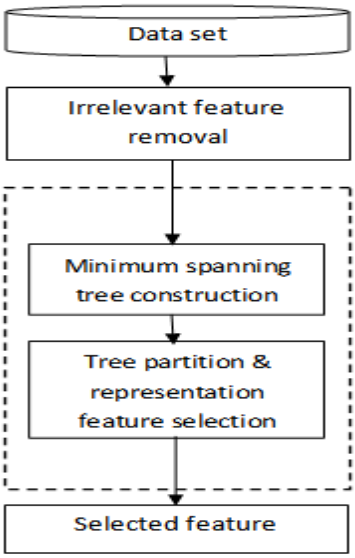


Figure1: Framework of the existing fast clustering

FAST algorithm falls into the second group. Traditionally, feature subset selection research has focused on searching for relevant features. A well-known example is Relief, which weighs each feature according to its ability to discriminate instances under different targets based on distance-based criteria function. Our proposed FLAME technique, Feature subset collection viewed as the process of classifying and removing as many unrelated and completed features as possible. This is because unrelated features do not donate the analytical correctness and terminated features do not redound to getting a better analysis that they provide mostly information which is already present in other feature.

The many feature subset selection algorithms some effectively remove unrelated features but fail to handle terminated features some of others can remove the unrelated taking care of the completed features. ACO and Apriori algorithm falls into the second group. The feature subset selection exploration has intensive on searching for related features. A recognized example is Release weighs each feature allowing its capability to differentiate occurrences under different targets based on distance-based measures function. The unsuccessful at removing terminated features as two analytical but highly correlated features are likely both to be decidedly weighted. The allowing this method to work with unruly and unfinished data sets and to deal with multiclass problems but still recognize redundant features. The approach we take in FLAME explores the space of all possible models. In order to carry out this exploration in an efficient way, we first construct two suffix trees: a suffix tree on the actual data set that contains counts in each node (called the data suffix tree), and a suffix tree on the set of all possible model strings (called the model suffix tree).

2. FEATURE SUBSET SELECTION ALGORITHM

The FLAME algorithm is mainly divided into three steps:

3.1 Extraction of the structure information from the dataset:

Construct a neighborhood graph to connect each object to its K-Nearest Neighbors (KNN); Estimate a density for each object based on its proximities to its KNN;

Objects are classified into 3 types: Cluster Supporting Object (CSO): object with density higher than all its neighbors; Cluster Outliers: object with density lower than all its neighbors, and lower than a predefined threshold; the rest.

3.2 Local/Neighborhood approximation of fuzzy memberships:

Initialization of fuzzy membership: Each CSO is assigned with fixed and full membership to itself to represent one cluster; all outliers are assigned with fixed and full membership to the outlier group; The rest are assigned with equal memberships to all clusters and the outlier group;

Then the fuzzy memberships of all type 3 objects are updated by a converging iterative procedure called Local/Neighborhood Approximation of Fuzzy Memberships, in which the fuzzy membership of each object is updated by a linear combination of the fuzzy memberships of its nearest neighbors.

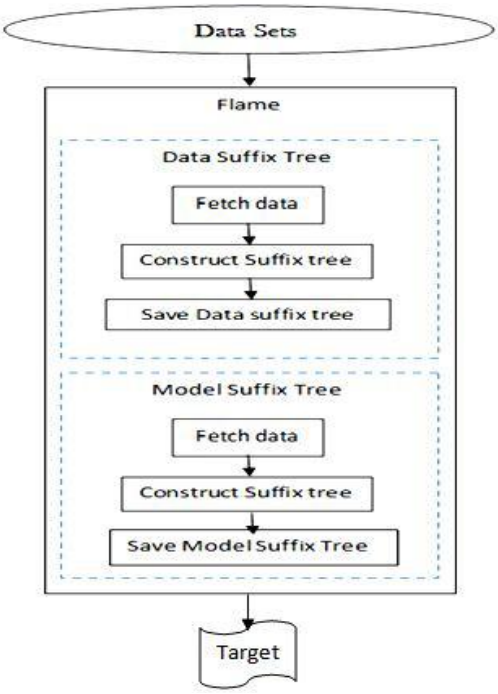


Figure2: Framework of the proposed Flame technique

3.3 Cluster construction from fuzzy memberships in two possible ways:

One-to-one object-cluster assignment, to assign each object to the cluster in which it has the highest membership; one to- multiple object-clusters assignment, to assign each object to the cluster in which it has a membership higher than a threshold.

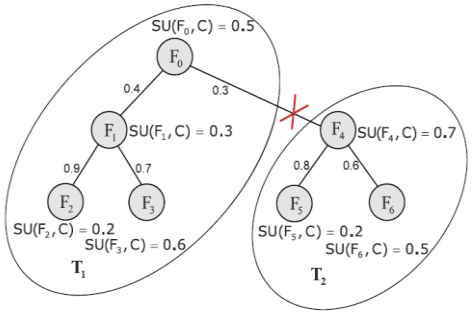
4. SYSTEM MODULES

4.1 Dataset Processing:

In this module the datasets are being loaded from system to the application. Mainly here we prefer to upload the DNA data to the system. DNA data are basically large in real time, so finding the patterns among this data set are highly expensive task in terms of system speed, accuracy and size. Association instructions display attributes that occur recurrently collected in a given dataset. These relationships are not based on essential properties of the data themselves but rather based on occurrence of the data items.

4.2 Data Suffix Tree:

In this module the data suffix tree has been generated. A suffix tree on the actual data set that contains counts in each node called the data suffix tree. The data suffix tree helps us quickly compute the support of a model string. Recall that a suffix tree with counts is merely a suffix tree in which every node contains the number of leaves in the sub tree rooted at that node. In other words, every node contains the number of occurrences of the string corresponding to that node. Substantial calculation power and storage capacity of cloud computing systems allow experts to organize computation and data concentrated requests without organization asset where large application datasets can be stored in the cloud. However, due to the datasets should be intentionally stored in order to reduce the overall application cost.



According to the above definitions, feature subset selection can be the process that identifies and retains the strong Relevance features and selects R-Features from feature clusters. The behind heuristics are that

- Irrelevant features have no/weak correlation with target concept.
- Redundant features are assembled in a cluster and a representative feature can be taken out of the cluster.

4.3 Model Suffix Tree:

The next step after constructing the Data suffix Tree is constructing the model suffix tree. Since the second suffix tree (built on all possible model strings) can be extremely large, FLAME does not actually construct this suffix tree. Rather, it algorithmically generates portions of this tree as and when needed. FLAME then explores the model space by traversing this (conceptual) model suffix tree. Using the suffix tree on the data set, FLAME computes support at various nodes in the model space and prunes away large portions of the model space that are guaranteed
not to produce any results under the model. This careful pruning, ensures that FLAME does not waste any time

exploring models that do not have enough support. The FLAME algorithm simply stops when it has finished traversing the model suffix tree and outputs the modelstrings that had sufficient support.

4.4 FLAME (Flexible and Accurate Motif detector):

It starts by traversing the nodes of the model space in depth-first order. At each node in the model suffix tree, the subroutine Evaluate_Support is called to compute the list of matches and the new support. This routine uses the match list from the parent node to speed up the computation. The routine Expand_Matches ensures that the number of mismatches to the model string does not exceed d. At any node, if FLAME discovers that the support is lower than k, it prunes away that sub tree in the model suffix tree, and continues its traversal. If it finds a model of length L with the required support, it simply outputs the result.

5. EXPERIMENTAL PROCEDURE

The FLAME algorithm is mainly divided into three steps: Extraction of the structure information from the dataset: Construct a neighborhood graph to connect each object to its K-Nearest Neighbors (KNN); Estimate a density for each object based on its proximities to its KNN. Objects are classified into 3 types: Cluster Supporting Object (CSO): object with density higher than all its neighbors; Cluster Outliers: object with density lower than all its neighbors, and lower than a predefined threshold; the rest. Local/Neighborhood approximation of fuzzy memberships: Initialization of fuzzy membership: Each CSO is assigned with fixed and full membership to itself to represent one cluster; All outliers are assigned with fixed and full membership to the outlier group; The rest are assigned with equal memberships to all clusters and the outlier group; Then the fuzzy memberships of all type 3 objects are updated by a converging iterative procedure called Local/Neighborhood Approximation of Fuzzy Memberships, in which the fuzzy membership of each object is updated by a linear combination of the fuzzy memberships of its nearest neighbors. Cluster construction from fuzzy memberships in two possible ways: One-to-one object-cluster assignment, to assign each object to the cluster in which it has the highest membership; One-to-multiple object-clusters assignment, to assign each object to the cluster in which it has a membership higher than a threshold.

Average Subset and Time, we 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, we obtain M×N classification Accuracy for each feature selection algorithm and each data set. Average these Accuracy, we obtain mean accuracy of each classification algorithm under each feature selection algorithm and each data set.

6. CONCLUSION

In this paper, we have presented the difficult of feature selection for the high dimensional data clustering. This is a difficult problem because the pounded truth class markers that can guide the selection are unattainable in clustering. Besides the data may have a large number of structures and the irrelevant ones can ruin the clustering. In this we recommend a novel feature allowance scheme for a clustering principle in which the heaviness for each feature is a measure of its influence to the clustering task. A novel motif mining algorithm called FLAME that uses a concurrent traversal of two suffix trees to efficiently explore the space of all motifs. It is also accurate, as it always finds the pattern if it exists. Accordingly we give a well-defined objective function which can be clearly solved in an iterative technique. Investigational results expression the effectiveness of the suggested process.

REFERENCES

[1] Almuallim H. and Dietterich T.G., Algorithms forIdentifying RelevantFeatures, In Proceedings of the 9th Canadian Conference on AI, pp 38-45,1992.

[2] Almuallim H. and Dietterich T.G., Learning boolean concepts in the presence of many irrelevant features, Artificial Intelligence, 69(1-2), pp 279-305, 1994.

[3] Arauzo-Azofra A., Benitez J.M. and Castro J.L.,A feature set measure based on relief, In Proceedings of the fifth international conference on Recent Advances in Soft Computing, pp 104- 109, 2004.

[4] Baker L.D. and McCallum A.K., Distributional clustering of words for text classification, In Proceedings of the 21st Annual international ACM SIGIR Conference on Research and Development in information Retrieval, pp 96-103, 1998.

[5] Butterworth R., Piatetsky-Shapiro G. and Simovici D.A., On Feature Selection through Clustering, In Proceedings of the Fifth IEEE international Conference on Data Mining, pp 581- 584, 2005.

[6] Cardie, C., Using decision trees to improve casebased learning, In Proceedings of Tenth International Conference on Machine Learning, pp 25-32,1993.

[7] Chanda P., Cho Y., Zhang A. and Ramanathan M., Mining of Attribute Interactions Using Information Theoretic Metrics, In Proceedings of IEEE international Conference on Data Mining Workshops, pp 350-355, 2009.

[8] Chikhi S. and Benhammada S., ReliefMSS: a variation on a feature ranking ReliefF algorithm. Int. J. Bus. Intell. Data Min. 4(3/4), pp 375-390, 2009.

[9] Guyon I. and Elisseeff A., An introduction to variable and feature selection, Journal of Machine Learning Research, 3, pp 1157-1182, 2003.

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

[11] J. Biesiada and W. Duch, "Featuresmn Election for High-Dimensional data a Pearson Redundancy Based Filter," Advances in Soft Computing, vol. 45, pp. 242-249, 2008.

[12] S. Das, "Filters, Wrappers and a Boosting Based Hybrid for Feature Selection," Proc. 18th Int'l Conf. Machine Learning, pp. 74- 81, 2001.

[13] M. Dash, H. Liu, and H. Motoda, "Consistency Based Feature Selection," Proc. Fourth Pacific Asia Conf. Knowledge Discovery and Data Mining, pp. 98-109, 2000.

[14] D.A. Bell and H. Wang, "A Formalism for Relevance and Its Application in Feature Subset Selection," Machine Learning, vol. 41, no. 2, pp.175-195, 2000.

[15] M. Dash and H. Liu, "Feature Selection for Classification," Intelligent Data Analysis, vol. 1,no. 3, pp. 131-156, 1997.

[16] W. Cohen, "Fast Effective Rule Induction," Proc. 12th Int'l Conf. Machine Learning (ICML '95), pp. 115-123, 1995.

[17] R. Battiti, "Using Mutual Information for Selecting Features in Supervised Neural Net Learning," IEEE Trans. Neural Networks, vol. 5, no. 4, pp. 537-550, July 1994.