International Journal of Applied Engineering Research ISSN 0973-4562 Volume 10, Number 14 (2015) pp 34603-34606

A Filter Based Feature Selection for Protein Sequence Classification over Hadoop

R Bhavani [1,*], G Sudha Sadasivam[2]
[1,*]Department of Computer science and Engineering, Government College of Technology, Coimbatore, TamilNadu, India.
[*]bhavanirajasekar@gmail.com
[2]Department of Computer science and Engineering, PSG College of Technology, Coimbatore, TamilNadu, India.
sudhasadhasivam@gmail.com

Abstract- Sequence mining is an important area in biological data mining. Protein sequence classification is about mining protein sequences and classifying them into different families based on their sequence patterns. It is helpful in predicting the structure and function of protein Feature selection is a crucial step in classification of protein sequences into existing super families. This paper proposes a novel feature selection algorithm which first transforms the protein sequences into feature vectors and reduces the size of the feature vector based on the apriori property. As the size of the protein sequences is large feature extraction is done parallel using Map Reduce programming over Hadoop framework. Experimental results show that the proposed method of feature selection reduces the features by 96% to 97% and also improves accuracy by 3% to 5%.

Keywords- filters, apriori property, sequence classification, correlation analysis, feature selection, Map Reduce.

1. Introduction

Data Mining is the process of extracting or mining useful and valuable knowledge from large amount of data. The database of biological data like protein sequences and DNA sequence etc., is increasing dramatically. Extracting useful knowledge from these biological data is an essential task in Bioinformatics.

The protein sequences contain twenty different amino acids. These sequences provide information regarding protein function, structure families and evolution information. Protein sequence classification is the most important technique to identify different functional groups or families based on their sequence patterns. A main issue of these a classification system is representing protein sequences, which largely determines the performance of classifiers. Feature extraction for protein sequence classification involves extraction of specific features from the sequences. The major feature construction methods existing in literature are amino acid composition, n-grams and motifs. Amino acid composition is where a protein is expressed as a vector of 20-dimensional space, in which its 20 components are defined by the composition of its 20 amino acids[1,2,3]. n-grams are the sequence of 'n' characters extracted from the larger sequence[4,5,6]. It is generated by sliding a window of n characters on the whole sequence. Many researchers have used combination of both amino acid composition and amino acid pairs as features for classification[7,8]. In protein sequences motifs are the short subsequences given an allowed number of mutations[9,10,11].

Due to high dimensions of the extracted features, there is a need for feature subset selection system that can accurately classify the novel protein sequences into existing super families. Principal Component Analysis (PCA) is a powerful mathematical technique used to reduce the dimensionality of the parameter space[12,13]. The 20-dimensional amino acid composition space is reduced to an orthogonal space with fewer dimensions, and the original base functions are converted into a set of orthogonal and normalized base functions with the combination of amino acid composition and PCA. Linear Discriminant Analysis(LDA) is applied to obtain the most important features for protein classification[14]. Hash kernels is used to map the high-dimensional input spaces into low-dimensional spaces for large scale classification[15].

As the size of the protein sequence database increases, the computation time also increases. Map Reduce programming model is used to analysis such large datasets in a parallel manner. It reduces the computation cost of processing the protein sequences[16].

This paper aims at selecting the most significant features from various kinds of protein super family sequences which leads to improved classification results. First N-gram descriptor for each protein super families are extracted in parallel. Second, apriori property is applied to the N-gram descriptor to obtain the most significant features of the particular super family. This is a filter based approach where significant features are selected based on the apriori property.

2. Proposed Approach

The flowchart of the proposed approach to classify the protein sequences into their particular super families is depicted in Figure 1. This approach starts with the extraction of N-grams from protein sequences and then obtaining the mean vector for each super family using Map Reduce programming model. Next apriori property is applied on the mean vector to obtain most frequent n-grams. Finally correlation analysis is applied to select most significant features from various super families. After selection of significant features, different classifiers have been used for classification and the performance of the proposed technique has been evaluated.

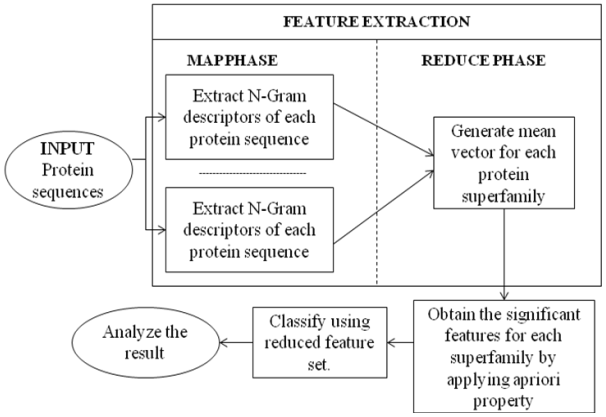

Fig. 1. Flowchart of the proposed approach

2.1 Feature Extraction

All protein sequences are represented by a combination of twenty amino acids. There is a need to represent these protein sequences in the form of a minimum number of numeric features. As the size of the protein sequences is large, Map Reduce programming model of Hadoop framework is used to extract the features from protein sequences. This programming model consists of two phases viz, Map phase and Reduce phase. In Map phase N-gram from the protein sequences are extracted and in Reduce phase the mean vector of each protein super family is obtained.

2.1.1 Map phase

The steps followed in Map phase are,

Map<*Key, Value*>

Key : *protein sequence number*

Value: *protein sequence*

(i) Replace each letter in protein sequence with six letter exchange group

(ii) Obtain the feature value of each n-gram using Eq. (1)

Output<*Key, Value*>

Key : *super family number*

Value: *feature vector*

First the six letter exchange group[17] is used to represent a protein sequence, where A={H,R,K}, B={D,E,N,Q}, C={C}, D={S,T,P,A,G}, E={M,I,L,V} and F={F,Y,W}. After replacement the sequences are transformed into a feature vector space using a feature encoding method. In this paper, a sequence encoding method based on the combination of different n-gram descriptors is utilized for the extraction of valuable features from a protein sequence. The main advantage of such replacement is that, if n=3 then the total number of k-grams extracted will be 258 ($6^1+6^2+6^3$) rather than 8420 ($20^1+20^2+20^3$). The feature value of each n-gram is given as

$$x = \frac{c}{len(S)-(n-1)} \tag{1}$$

where S is the sequence, len(S) is the length of the sequence, c is the number of occurrences of the n-gram and n is the size of the n-gram. For example suppose S=PEKTNEK. First after replacement the sequence will be S=DBADBBA. The value of the feature DB with respect to S is 2/(7-1) = 0.33.

3.1.2 Reduce phase

The steps followed in Map phase are,

Reduce<*Key,Value*>

Key : *superfamily number*

Value: *feature vector*

Find the average of the feature vector for each superfamily using Eq.(2)

Output<*Key,Value*>

Key : *superfamily number*

Value: *mean vector*

The Map phase transforms each protein sequence to a vector of $\sum_{k=1}^{n} 6^k$ features. The Reduce phase accepts the feature vectors from Map phase and calculates the mean vector of each protein superfamily. Let X^i denote the i[th] superfamily of sequences. X^i_n represents the m[th] sequence of the superfamily X^i, m=1,2,…,N_i, where N_i is the number of sequences in the i[th] superfamily. $X^i_m(j)$ is the feature vector representing the m[th] sequence of the i[th] superfamily with j features. For each superfamily, the mean vector is calculated as follows[18]

$$\overline{X}^i(j) = \frac{\sum_{m=1}^{N_i} X^i_m(j)}{N_i} \tag{2}$$

3.2 Feature Selection

Once the mean vector of the protein superfamily is obtained, feature reduction takes place by applying the apriori property[19] namely: "All subsets of a frequent itemset must also be frequent". Here the mean value of each vector is considered as the support count of it. There are two steps followed in this stage viz. join step and prune step.

Join step: To find L_n, a set of candidate n-grams is generated by joining $L_{n-1}$with itself. This set of candidates is denoted C_n.

Prune step: C_n is a superset of L_n, that is, its members may or may not be frequent, but all of the frequent n-grams are included in C_n. Any (n−1)-gram that is not frequent cannot be a subset of a frequent n-gram. Hence, if any (n−1)-subset of a candidate n-gram is not in L_{n-1}, then the candidate cannot be frequent either and so can be removed from C_n. At the end of this step, candidates having support count no less than the minimum support count are frequent and therefore belong to L_n. In this work n=3 is considered.

4. Experimental Results and Analysis

In the experiments, two datasets were considered. Dataset1 contains six enzyme classes viz Oxidoreductases (EC 1.-.-.-), Transferases (EC 2.-.-.-), Hydrolases (EC 3.-.-.-), Lyases (EC 4.-.-.-), Isomerases (EC 5.-.-.-), Ligases (EC 6.-.-.-) and its filesize is 136MB. Dataset2 contains proteomes of Anopheles gambiae, Bos Taurus, Canis lupus familiaris, Danio rerio, Drosophila melanogaster, Equus caballus, Felis catus, Gallus gallus, Homo sapiens, Macaca Fascicularis, Mus musculus, Oryctolagus cuniculus, Ovis aries, Saccharomyces Cerevisiae,

Xenopus tropicalis were considered. The filesize of the second dataset is 310MB. Experiments were carried out in a Hadoop cluster of 8 machines with each having configuration of 1 GB RAM, 75 GB Hard disk drive and Intel core 2 duo processor.

Features are extracted from the protein sequences and then feature selection method was applied to reduce the size of the feature vector. From Table 1 it is observed that without using the feature selection technique, the feature size would be bigger (8420 if N=3) and this would ultimately decrease the classification accuracy. Using the proposed approach the number of features was reduced by 96% to 97%.

Table 1. Comparison on Number of features.

Dataset	Feature Representation	Number of features for (n=3)
Dataset1	N-gram	8420
	Proposed	250
Dataset1	N-gram	8420
	Proposed	287

Classification using K-Nearest Neighbor, Naive Bayes and Decision Tree classifiers were done using Weka data mining tool. A tenfold cross validation model was used in order to assess the results of the experiment. The performances measures[21] used to validate the experimental results are as follows.

$$Percision = \frac{\sum_{i=1}^{l} \frac{TP_i}{TP_i + FP_i}}{l} . \quad (3)$$

$$\text{Re}call = \frac{\sum_{i=1}^{l} \frac{TP_i}{TP_i + FN_i}}{l} . \quad (4)$$

$$Accuracy = \frac{\sum_{i=1}^{l} \frac{TP_i + TN_i}{P_i + N_i}}{l} . \quad (5)$$

where TP_i stands for true positive of $class_i$, TN_i stands for true negative of $class_i$, FP_i stands for false positive of $class_i$, FN_i stands for false negative of $class_i$, Table 2 shows the precision and recall of various classifiers for dataset1 and dataset2. Figure. 2 and 3 gives the comparison of the accuracy of the various classifiers using different feature selection techniques for dataset1 and dataset2 respectively. It is observed that K-nearest neighbor classifier offered a better performance as compared to other classifiers. On an average the proposed feature selection method yields 3% to 5% better accuracy.

Table 2. Performance Measure for Dataset1 and Dataset 2.

Performance Measure	Feature Representation	Dataset1			Dataset2		
		K-NN[a]	NB[b]	DT[c]	K-NN[a]	NB[b]	DT[c]
Percision	N-gram	0.82	0.75	0.77	0.81	0.75	0.79
	Proposed	0.85	0.78	0.82	0.88	0.76	0.84
Recall	N-gram	0.83	0.73	0.76	0.82	0.74	0.74
	Proposed	0.84	0.79	0.82	0.85	0.76	0.80
Note: [a]K-Nearest Neighbour classifier, [b]Naive Bayes classifier, [c]Decision Tree classifier							

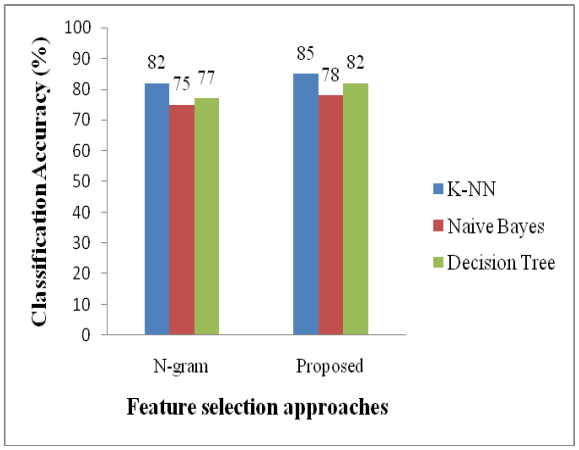

Fig. 2. Accuracy comparison of feature selection approaches for dataset1.

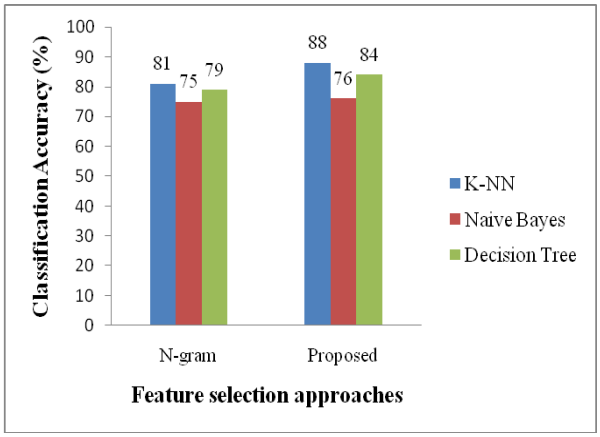

Fig. 3. Accuracy comparison of feature selection approaches for dataset2

Table 3 gives the computation time required for extracting k-grams from protein sequences. The speedup of the feature extraction algorithm is given by

$$Speedup = \frac{T_1}{T_p} \quad (6)$$

where p is the number of processors, T_1 is the execution time on one processor and T_p is the execution time of p processors. From the graph in Figure. 4. it can be seen that parallel implementation of feature extraction algorithm using Map Reduce paradigm exhibits good scalability for larger file size. It is also observed that, for p=2, the scalability is not achieved because the mapper runs on the slave and the reducer runs on the master, which is same as p=1 where the mapper and reducer run on the same machine.

Table 3. Computation Time in seconds (for k-gram extraction)

Dataset	Size in MB	Number of Processors			
		1	2	3	4
Dataset1	136	320	320	130	130
Dataset2	310	810	810	275	172

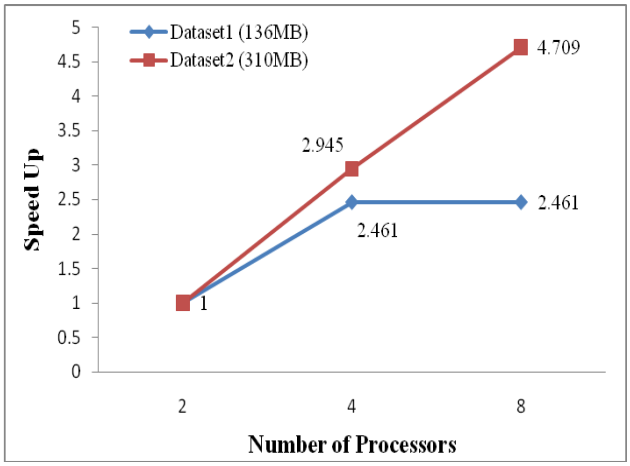

Fig. 4. Speedup of the proposed feature extraction algorithm

5. Conclusion

In this paper, a novel filter based approach for feature selection from protein sequences using Map reduce programming model is proposed. N-gram descriptors of protein sequences were first extracted. The irrelevant and redundant features are then removed using apriori property The computation cost of the large protein sequence dataset is minimized due to parallel processing of protein sequences over Hadoop clusters. Classification using K-Nearest Neighbor, Naive Bayes and Decision Tree confirm the performance of the proposed filter based feature selection approach.

References

[1] Nishikawa, Ken, Yasushi Kubota and Tatsuo, 1983, "Classification of proteins into groups based on amino acid composition and other characters I Angular distribution", J Biochem, 94l(3), pp.981-995.

[2] Genfa, Zhou, Xu Xinhua and Zhang Chun Ting, 1992, "A weighting method for predicting protein structural class from amino acid composition", Eur J Biochem, 210(3), pp.747-749.

[3] Zhang, Chun-Ting and Kuo-Chen Chou, 1992, "An optimization approach to predicting protein structural class from amino acid composition", Protein Sci, 1(3), pp.401-408.

[4] Leslie C. S., Eskin E. and Noble W. S., 2002, "The spectrum kernel: A string kernel for SVM protein classification". Proc Pacific symposium on Biocomputing, pp.566-575.

[5] Yang Yang, Bao-Liang Lu and Wen-Yun Yang, 2008, "Classification Of Protein Sequences Based On Word Segmentation Methods", Proc Asia-Pacific Bioinformatics Conf, pp.177-186.

[6] Emanuelsson Olof, Henrik Nielsen, Soren Brunak and Gunnarvon Heijne, 2000 "Predicting subcellular localization of proteins based on their N-terminal amino acid sequence", J Mol Biol, 300(4), pp.1005-1016.

[7] Luo, Rui yan, Zhi ping Feng and Jia kun Liu, 2002, "Prediction of protein structural class by amino acid and polypeptide composition", Eur J Biochem, 269(17), pp.4219-4225.

[8] Park, Keun-Joon and Minoru Kanehisa, 2003, "Prediction of protein subcellular locations by support vector machines using compositions of amino acids and amino acid pairs", Bioinformatics, 19(13), pp.1656-1663.

[9] Blekas, Konstantinos, Dimitrios I. Fotiadis and Aristidis Likas, 2005, "Motif-based protein sequence classification using neural networks", J Comput Biol, 12(1), pp.64-82.

[10] Kunik V., Solan Z., Edelman S., Ruppin E. and Horn D., 2005, "Motif extraction and protein classification". Proc IEEE Conf Computational Systems Bioinformatics, pp.80-85.

[11] Saidi, Rabie, Mondher Maddouri and Engelbert Mephu Nguifo, 2010, "Protein sequences classification by means of feature extraction with substitution matrices". BMC Bioinformatics, 11:175.

[12] Du, Qi-Shi, Zhi-Qin Jiang, Wen-Zhang He, Da-Peng Li and Kou-Chen Chou, 2006, "Amino acid principal component analysis (AAPCA) and its applications in protein structural class prediction", J Biomol Struct Dyn, 23(6), pp.635-640.

[13] Wang, Bo. and Michael A. Kennedy, 2014, "Principal components analysis of protein sequence clusters". J Struct funct Genomics, 15(1), pp.1-11.

[14] Dutta, Subhajit, Probal Chaudhuri and Anil K Ghosh, 2014, "Linear Discriminant Analysis of Character Sequences Using Occurrences of Words", Statistica Sinica, 24, pp.493-514.

[15] Caragea C, Silvescu A and Mitra P, 2011, "Protein sequence classification using feature hashing", Proc IEEE Conf Bioinformatics and Biomedicine, pp.538 – 543.

[16] Jeffrey Dean and Sanjay Ghemawat, 2004, "Mapreduce: Simplified data processing on large clusters", Technical report, Google Inc.,

[17] Wang, Jason Tsong-Li, Qicheng Ma, Dennis Shasha and Cathy H. Wu, 2001, "New techniques for extracting features from protein sequences". IBM Systems Journal, 40(2), pp.426-441.

[18] Iqbal, Muhammad Javed, Ibrahima Faye, Brahim Belhaouari Samir and Abas Md Said, 2014, "Efficient Feature Selection and Classification of Protein Sequence Data in Bioinformatics", The Scientific World Journal, Vol 2014, Article ID 173869, 12 pages.

[19] Jiawei Han, Micheline Kamber and Jian Pei, "Data Mining: Concepts and Techniques", 3rd ed, San Francisco: Morgan Kaufmann Publishers Inc.; 2012.

[20] Sokolova, Marina and Guy Lapalme, 2009, "A systematic analysis of performance measures for classification tasks", Information Processing and Management, 45(4), pp.427-437.