An unsupervised, fast correlation-based filter for feature selection for data clustering

Part Pramokchon and Punpiti Piamsa-nga

Department of Computer Engineering, Faculty of Engineering, Kasetsart University, Jatujak, Bangkok, 10900, THAILAND
{g4785036, pp}@ku.ac.th

Abstract. Feature selection is an important method to provide both efficiency and effectiveness for high-dimension data clustering. However, most feature selection methods require prior knowledge such as class-label information to train the clustering module, where its performance depends on training data and types of learning machine. This paper presents a feature selection algorithm that does not require supervised feature assessment. We analyze relevance and redundancy among features and effectiveness to each target class to build a correlation-based filter. Compared to feature sets selected by existing methods, the experimental results show that performance of a feature set selected by the proposed method is comparably equal and better when it is tested on the RCV1v2 corpus and Isolet data set, respectively. However, our technique is simpler and faster and it is independent to types of learning machine.

Keywords: feature selection, unsupervised learning, clustering, filter-based method, correlation, similarity, redundancy

1 Introduction

Data clustering is an automatic process for grouping unlabeled data into a number of similar groups, called clusters, by assessment of their contents. It is an unsupervised process by nature [1-2]. Most data clustering methods typically employ feature vector models for representing data instants [3-5]. However, this representation will suffer when applying with high-dimension and sparse data, known as the curse of dimensionality. For example in text document clustering, a document is represented by a feature vector known as Bags of words (BOW) that generally uses all words in data collection as features [1,6-8]. Most words in documents are usually redundant and irrelevant to the clustering. Hence, determining clusters in this data space is not only computationally expensive but it also degrades the learning performance [2-3, 8-10].

In order to eliminate irrelevancy and redundancy, we have to select a smaller number of most relevant features to the targeted clusters [2, 11-13]. Feature selection methods can be classified as filter-based approaches, wrapper-based and hybrid approach [2, 12, 14]. Filter-based methods are to select features by "usefulness" of each individual feature for clustering. The usefulness is determined by assessment of feature's characteristics using certain statistical criteria, without any learning process [1,

T. Herawan et al. (eds.), *Proceedings of the First International Conference on Advanced Data and Information Engineering (DaEng-2013)*, Lecture Notes in Electrical Engineering 285, DOI: 10.1007/978-981-4585-18-7_10,

3, 7-8, 11, 15]. On the other hand, wrapper-based methods employ a chosen learning algorithm to determine an optimal subset of features. Even though they outperform filter-based methods, they are in general more computationally expensive. The selected feature subset is usually overfit to a chosen learning algorithm [2, 11-12, 14]. Several authors have proposed hybrid methods, which take advantages of both filter and wrapper methods [1-2, 7, 12]. However, the wrapper and hybrid approach is usually not suitable when availability of time or computing resources is a constraint. Thus, the filter model is still preferable in terms of computational time [2, 7, 11-12, 14].

There are two types of filter-based feature selections, supervised and unsupervised method. For supervised methods, class label, which information to identify class of a data entity, must be provided. Some supervised feature selection methods have been successfully used in text classification [3, 16]. However, if class label is not available, many research projects introduce some unsupervised feature selection such as Document Frequency (DF) [3, 15-16], Term Contribution (TC) [1, 6, 15, 17], Term Variance (TV) [6-7, 17] and Mean Median (MM) [7]. Some researchers proposed to reduce redundancy and irrelevance for the feature selection algorithm, such as [14]; however, class label is required and it cannot be used for data clustering directly.

In this article, we propose an unsupervised filter-base feature selection for data clustering. The method is unsupervised method and it is independent to type of learning machines. Furthermore, we integrate concept of evaluating feature redundancies into the proposed algorithm. The redundancy assessment is a feature similarity measurement based on the geometric view of features. [7] Unlike classical filter-based methods, which have to predefine a threshold by expertise or empirical experiments to pick up top-ranked features, our proposed method uses a coefficient of confident to select relevant features, rather than finding a new threshold for every new set of data. The performance of the algorithm is evaluated on a corpus dataset for high-dimension data analysis, namely the RCV1v2 dataset [18] and Isolet dataset [19]. Compared by F1, Average Accuracy and Rand statistics, the experiment on RCV1v2 dataset shows that clustering accuracy of the proposed algorithm significantly is comparably equal to the baseline filter-based method. The result also shows that proposed method outperforms the baseline on the Isolet dataset.

The rest of the paper is organized as follows. Background of feature selection is summarized in Section 2. Section 3 presents our proposed feature selection algorithm and describes its characteristics. Experiments are explained and results are discussed in Section 4. Section 5 concludes this work.

2 Features and Feature Redundancy

Notations used in this paper are as follows: $F = \{f_1, \dots, f_{|F|}\}$ be a set of original distinct features; $D = \{d_1, \dots, d_{|D|}\}$ be a training dataset; A training sample d_j is represented as a feature weight vector, $\overrightarrow{d_j} = [w(f_1, d_j), \dots, w(f_{|F|}, d_j)]$. This feature weight $w(f_i, d_j)$ quantifies the importance of the feature f_i for describing semantic

content of d_j. For example, text clustering prefers feature weight such as the Term Frequency or the Term Frequency Inverse Document Frequency [1-2, 6, 8-9, 15, 18].

Feature redundancy can be represented in terms of feature correlation. It is widely accepted that the features are redundant to each other if their values are completely correlated [7, 11, 14]. The traditional filter-based feature selection is incapable of removing redundant features because redundant features likely have similar rankings. As long as features are deemed relevant to the class, they will all be selected even though many of them are highly correlated to each other. For high-dimensional data which may contain a large number of redundant features, this approach may produce results far from optimal. Many research projects address some similarity/correlation/redundancy measures that have been used for feature selection, such as correlation coefficient [7, 14, 20], symmetrical uncertainty [7, 14] and absolute cosine [7]. In [7] argue that using angles to measure similarity is better suitable for high-dimensional sparse data. Therefore, absolute cosine is used as geometric view in our proposed algorithm.

3 Proposed method

The proposed method composed of two approaches: measuring feature relevance by feature score to keep highly relevance features; and measuring feature redundancy by feature correlation to identify redundant features. Then, we define a policy to eliminate less important features. The algorithm is listed in Algorithm 1 (Unsupervised fast correlation-based filter algorithm.). In Algorithm 1, scores of each feature are computed and then used it for sorting (lines 2 and 3); each feature will be compared with others in order to find relevancy (lines 4-12); feature redundancy is measured by computing feature similarity [7] (line 8); each feature is then considered to be removed from the output (lines 13-17); and after looping for every feature, the result, which is a feature set that each has high relevance and low similarity among themselves, is returned (line 20).,

Algorithm 1 Unsupervised Fast Correlation-Based Filter Algorithm

Input : Original feature set (F)
Training data (D)
Threshold parameter (α)
Output : Optimal feature subset (Opt)
1: for each feature f_i in F do **//** Compute score all feature
2: $s(f_i) \leftarrow Score(f_i, D)$
3: $ST \leftarrow SortedFeatureByScoreDes(s(f_i), F)$ // set of sorting feature in F by score descending
4: $f_j \leftarrow GetFirstElement(ST)$
5: do begin
6: $F' \leftarrow GetAllNextFeature(f_j, ST)$
7: for each feature f_i in F'

```
8:      sim(f_i, f_j) ← ComputeSimilarity(f_i, f_j)
9:    thres_redundant(f_j) ← ComputeRedudantThreshold(α)
10:   thres_remove(f_j) ← ComputeRemoveThreshold(α)
11:   ST' ← SortFeatureByScoreAs(s(f_i), F') //set of sorting fea-
ture in F' by score ascending
12:   cr ← CumulativeRelevance(ST')
13:   for each feature f_i in ST'   // redundant identify
and remove
14:     if (sim(f_i, f_j) > thres_redundant)
15:       if (cr − s(f_i) > thres_remove)
16:         remove f_i from ST
17:   f_j ← GetNextElement(f_j, ST)
18: end until (f_j = NULL)
19: Opt = ST
20: return Opt
```

In practical, many feature selection methods suffer the problem of selecting appropriate thresholds for both redundant feature identification and redundancy elimination. Thus, we proposed the statistics based method to compute threshold to identify redundant feature of feature *fj*, is defined as

$$Thres_{redundant} = \overline{sim(f_j)} + Z_{1-\frac{\alpha}{2}} \cdot \sigma_{f_j} \quad (1)$$

where α is a confidence coefficient, $Z_{1-\frac{\alpha}{2}}$ is the $1-\frac{\alpha}{2}$ quantize of the $N(0,1)$, $\overline{sim(f_j)}$ and $\sigma_{f_i}$are average and standard deviation, respectively, of similarity between features f_i and f_j, $f_i \in ST'$. The feature f_i, that has similarity value $\left(sim(f_i, f_j)\right)$ more than $thres_{redundant}$, is identified as redundant feature of feature f_j. Next, we introduce a criterion in strategy for redundant feature elimination. The decision to remove a feature depends on a cumulative relevance (cr) measure [7].

$$cr = \sum_{i=1}^{|ST'|} s(f_i) \quad (2)$$

The cr value is used to calculate summation of relevance score of feature in subset ST'. Then, we propose removing the features $f_i$where $cr - s(f_i)$ is more than threshold, $Thres_{remove} = \left(1-\frac{\alpha}{2}\right) \cdot cr$. This means that f_i is redundant and removing f_i affects cr value a little. Thus, f_i can be removed from the feature set. On the other hand, some features f_i have $s(f_i)$ that make $cr - s(f_i) < thres_{remove}$, it means that the feature is redundant but it is influent to the cumulative relevance of feature set; thus, it should be kept in the selected feature set. Finally, we can select highly-relevant features and remove highly-redundant feature for data clustering.

4 Experiments

In the experiment, we use the RCV1v2 dataset [18] and choose the data samples with the highest four topic codes (CCAT, ECAT, GCAT, MCAT) in the "Topic Codes" hierarchy which contains 19,806 training documents and 16,942 testing documents. Furthermore, we also use Isolet dataset [19]. There are 617 real features with 7797 instances and 26 classes. We generate a vector model for training data without using class label based on our selected features. Then, we use K-Means clustering onto the vector model. The result, which is a set of clusters, will be used as a new model for clustering onto the testing data (also without using class label.) The labels assigned by clustering testing data are used to compare with class labels given from corpora. In order to assess clustering performance under different feature selection method, three qualitative measures are selected. For Average Accuracy (AA) [6, 21] and RS Rand Statistics (RS) [6, 22], we count number of documents, which have the same topics, in the same cluster and number of documents, which have different topics, in different clusters. In our clusters and in the corpus, both documents are placed in the same clusters: *ss*. In our clusters both documents are placed in the same clusters but in corpus they are in different clusters: *sd*. In our clusters documents are placed in different clusters but in the corpus they are in the same clusters: *ds*. In our clusters and in the corpus both documents are placed in different clusters: *dd*. Then, AA and RS are defined as follows:

$$AA = \frac{1}{2} \times \left(\frac{ss}{ss+ds} + \frac{dd}{sd+dd} \right) \tag{3}$$

$$RS = \frac{(ss+dd)}{ss+sd+ds+dd} \tag{4}$$

Another measure for evaluating clustering is the macro F1-measure (F1) [10, 21] that is evaluated as

$$F1 = \sum_i \frac{n_i}{N} \left[max_{j \in C} \left\{ \frac{2 \times Recall(i,j) \times Precision(i,j)}{Recall(i,j) + Precision(i,j)} \right\} \right] \tag{5}$$

which $Recall(i,j) = \frac{n_{ij}}{n_i}, Precision(i,j) = \frac{n_{ij}}{n_j}$, where n_{ij} is the number of instances belonging to class i in corpus that falls in cluster j, and n_i, n_j are the cardinalities of class i cluster j respectively.

Our proposed selection algorithm is evaluated by comparing the effectiveness of our optimal feature subset with other subsets selected by a baseline algorithm. We use ranking wrapper-based feature selection, which is the most preferable filter-based method to determine the best number of highly relevance feature as the baseline [6, 9, 13, 23]. We applied four unsupervised feature scoring schemes, DF, TC, TV and MM, to determine feature subset on training documents of RCV1v2 dataset with different cut-off number of features ranging from 500 to 4000. At each cut-off number, the performance of the K-Means clustering for the selected feature subset is estimated by 10-fold 10-time cross-validation. The results show that the optimal number of features for DF, TC, TV and MM are 1300, 500, 500 and 500, respectively.

We then used the proposed outlier-based feature selection in Algorithm 1 to select a feature subset with these four feature scores. We set merely a parameter α, to identify the optimal threshold. From preliminary parameter setting, numbers of features selected by the proposed algorithm for DF, TC, TV, and MM are 486, 483, 537, and 470, respectively. The result shows that the proposed algorithm almost selects a smaller feature subset when compared with the feature subset selected using the baseline algorithm. The selected feature subsets are used to train the clustering; then the clustering performance is evaluated on testing documents. Table 1 shows the performance of features selected by each algorithm for DF, IC, IV, and MM. The performance values in each table are 10-run average values. We compute Student's independent two-tailed t-test in order to evaluate the statistical significance of the difference between the two averaged values: the one from the proposed method and the one from baseline method. The *p*-Val is the probability associated with an independent two-tailed t-Test. The "compare" means that the proposed method is statistically significant (at the 0.05 level) win or loss over the baseline methods and equal means no statistically significant difference. The experimental result shows that the proposed method with four feature scores get comparably equal clustering performance to the baseline method.

Table 1. Comparing three performance measures between feature subsets selected by the UFCBF method and the baseline method for DF, TC, TV and MM on RCV1v2 dataset.

Feature Score	Method	#feature	F1	AA	RS
DF	baseline	1300	0.688	0.553	0.626
	UFCBF	486	0.693	0.554	0.632
	p-Value		0.857	0.969	0.722
	compare		equal	equal	equal
TC	baseline	500	0.667	0.525	0.600
	UFCBF	483	0.684	0.557	0.633
	p-Value		0.588	0.278	0.234
	compare		equal	equal	equal
TV	baseline	500	0.683	0.552	0.626
	UFCBF	537	0.645	0.555	0.629
	p-Value		0.204	0.837	0.841
	compare		equal	equal	equal
MM	baseline	500	0.665	0.539	0.613
	UFCBF	470	0.702	0.563	0.638
	p-Value		0.149	0.143	0.127
	compare		equal	equal	equal

Moreover, we also evaluate performance of the proposed method on the Isolet with of baseline method presented in [10]. The Table 2 shows comparison of validation. The number of selected features from the proposed method is lower than number from the baseline method. Table 2 shows that proposed algorithm achieve higher F1 than the baseline and RS value of both methods are equal.

Table 2. A comparison of the performance on Isolet dataset

Method	#feature	F1	Rand	AA
baseline [10]	617	0.365	-	-
	274	0.336	0.94	-
	275	0.344	0.94	-
Purposed	263	0.530	0.94	0.62
Compare		Win	equal	-

5 Conclusions

In this paper, we proposed an effective and computationally efficient algorithm that dramatically reduces size of feature set in high dimensional datasets. The proposed algorithm eliminates a large number of irrelevant and redundant features and selects a subset of informative features that provide more discriminating power for unsupervised learning model. Our algorithm is developed and tested it on the RCV1v2 and Isolet data corpus. Our experimental results confirm that the proposed algorithm can greatly reduce the size of feature sets while maintaining the clustering performance of learning algorithms. The algorithm uses a simple statistic based threshold determination to develop a novel filter-based feature selection technique. Our approach does not require iterative empirical processing or prior knowledge. Compared to traditional hybrid feature selection the optimal subset from our proposed algorithm is significantly comparable or even better than the baseline algorithm. Experiments showed that proposed method works well and can be used with the unsupervised learning algorithm which class information is unavailable and the dimension of data is extremely high. Our proposed method can not only reduce the computation cost of text document analysis but can also be applied to other textual analysis applications.

References

1. Almeida, L.P., Vasconcelos, A.R., Maia, M.G.: A Simple and Fast Term Selection Procedure for Text Clustering. In: Nedjah, N., Macedo Mourelle, L., Kacprzyk, J., França, F.G., De Souza, A. (eds.) Intelligent Text Categorization and Clustering, vol. 164, pp. 47-64. Springer Berlin Heidelberg (2009)
2. Alelyani, S., Tang, J., Liu, H.: Feature Selection for Clustering: A Review. In: Aggarwal, C., Reddy, C. (eds.) Data Clustering: Algorithms and Applications. CRC Press (2013)

3. Sebastiani, F.: Machine learning in automated text categorization. ACM Comput. Surv. 34, 1-47 (2002)
4. Ferreira, A.J., Figueiredo, M.A.T.: An unsupervised approach to feature discretization and selection. Pattern Recognition 45, 3048-3060 (2012)
5. Shamsinejadbabki, P., Saraee, M.: A new unsupervised feature selection method for text clustering based on genetic algorithms. J Intell Inf Syst 38, 669-684 (2012)
6. Luying, L., Jianchu, K., Jing, Y., Zhongliang, W.: A comparative study on unsupervised feature selection methods for text clustering. In: Natural Language Processing and Knowledge Engineering, 2005. IEEE NLP-KE '05. Proceedings of 2005 IEEE International Conference on, pp. 597-601. (Year)
7. Ferreira, A.J., Figueiredo, M.A.T.: Efficient feature selection filters for high-dimensional data. Pattern Recognition Letters 33, 1794-1804 (2012)
8. Ferreira, A., Figueiredo, M.: Efficient unsupervised feature selection for sparse data. In: EUROCON - International Conference on Computer as a Tool (EUROCON), 2011 IEEE, pp. 1-4. (Year)
9. Yanjun, L., Congnan, L., Chung, S.M.: Text Clustering with Feature Selection by Using Statistical Data. Knowledge and Data Engineering, IEEE Transactions on 20, 641-652 (2008)
10. Mitra, S., Kundu, P.P., Pedrycz, W.: Feature selection using structural similarity. Information Sciences 198, 48-61 (2012)
11. Guyon, I., Andr, #233, Elisseeff: An introduction to variable and feature selection. J. Mach. Learn. Res. 3, 1157-1182 (2003)
12. Liu, H., Yu, L.: Toward integrating feature selection algorithms for classification and clustering. IEEE Transactions on Knowledge and Data Engineering 17, 491-502 (2005)
13. Somol, P., Novovicova, J., Pudil, P.: Efficient Feature Subset Selection and Subset Size Optimization. Pattern Recognition Recent Advances 75-97 (2010)
14. Yu, L., Liu, H.: Efficient Feature Selection via Analysis of Relevance and Redundancy. J. Mach. Learn. Res. 5, 1205-1224 (2004)
15. Liu, T., Liu, S., Chen, Z.: An Evaluation on Feature Selection for Text Clustering. In: In ICML, pp. 488-495. (Year)
16. Yang, Y., Pedersen, J.O.: A Comparative Study on Feature Selection in Text Categorization. In: 14th International Conference on Machine Learning, pp. 412-420. Morgan Kaufmann Publishers Inc., 657137 (Year)
17. Zonghu, W., Zhijing, L., Donghui, C., Kai, T.: A new partitioning based algorithm for document clustering. In: Fuzzy Systems and Knowledge Discovery (FSKD), 2011 Eighth International Conference on, pp. 1741-1745. (Year)
18. Lewis, D.D., Yang, Y., Rose, T., Li, F.: RCV1: A New Benchmark Collection for Text Categorization Research. Journal of Machine Learning Research 5, 361-397 (2004)
19. Bache, K., Lichman, M.: UCI Machine Learning Repository. University of California, Irvine, School of Information and Computer Sciences, Irvine, CA (2013)
20. Mitra, P., Murthy, C.A., Pal, S.K.: Unsupervised feature selection using feature similarity. Pattern Analysis and Machine Intelligence, IEEE Transactions on 24, 301-312 (2002)
21. Shamsinejadbabki, P., Saraee, M.: A new unsupervised feature selection method for text clustering based on genetic algorithms. J Intell Inf Syst 1-16 (2011)
22. Achtert, E., Goldhofer, S., Kriegel, H.-P., Schubert, E., Zimek, A.: Evaluation of Clusterings - Metrics and Visual Support. In: ICDE'12, pp. 1285-1288. (2012)
23. Ruiz, R., Riquelme, J., Aguilar-Ruiz, J.: Heuristic Search over a Ranking for Feature Selection. In: Cabestany, J., Prieto, A., Sandoval, F. (eds.) Computational Intelligence and Bioinspired Systems, vol. 3512, pp. 498-503. Springer Berlin / Heidelberg (2005)