Novel Hybrid Feature Selection Models for Unsupervised Document Categorization

Amol P. Bhopale
Department of Information Technology
National Institute of Technology Karnataka
Surathkal, India
amolpbhopale@gmail.com

Sowmya Kamath S.
Department of Information Technology
National Institute of Technology Karnataka
Surathkal, India
sowmyakamath@nitk.edu.in

***Abstract*—Dealing with high dimensional data is a challenging and computationally complex task in the data pre-processing phase of text clustering. Conventionally, union and intersection approaches have been used to combine results of different feature selection methods to optimize relevant feature space for document collection. Union method selects all features from considered sub-models, whereas, intersection method selects only common features identified by sub-models. However, in reality, any type of feature selection can cause a loss of some potentially important features. In this paper, a hybrid feature selection model called *Modified Hybrid Union (MHU)* is proposed, which selects features by considering the individual strengths and weaknesses of each constituent component of the model. A comparative evaluation of its performance for K-means clustering and Bio-inspired Flock-based clustering is also presented on standard data sets such as OWL-S TC and Reuters-21578.**

Keywords*—*Text categorization; Feature selection; Dimensionality reduction; Unsupervised learning

I. Introduction

The era of big data has contributed significantly to the exponential increase in digital information available on the Internet. Much of this digital information is in the form of unstructured and semi-structured text, due to which, intelligent consumption of available information for targeted search is a challenging task. Text categorization is an effective way of identifying related documents based on their distinguishing domain-specific characteristics. Text categorization techniques use explicit information available in raw text documents to generate knowledge using which documents can be organized. Both supervised text classification and unsupervised text clustering approaches have been proposed over the years. Unsupervised approaches are better suited for document collections for which no prior class information is available. Because of this, clustering algorithms like k-means [1], Fuzzy C-Means [2], Expectation Maximization clustering [3], Quality Threshold clustering [4], Kernel k-means clustering [5], Density based clustering [6] and Minimum Spanning Tree based clustering [7] have been popularly applied for solving classification problems in multiple domains.

A significant challenge faced during the text clustering process is dealing with the high dimensionality of document-term space. For performing clustering, all documents need to be tokenized and a global dictionary containing all unique, distinct features has to be formed. As the number of distinct term increase, the performance of the clustering algorithm is adversely affected. This problem is compounded if the document corpus considered for the clustering is large and also has a wide variety of documents. All distinct terms obtained from the multiple documents may not be relevant for each document, moreover, the significance of a particular term may be different for different documents. It is therefore the primary goal of dimensionality reduction to select relevant, noise free and unique features by reducing the number of random variables under consideration from the feature space [8].

Existing dimension reduction techniques can be differentiated as *feature selection* and *feature extraction* based methods. In reality, a single dimension reduction method by itself is not capable of completely capturing all aspects of the original feature space for most optimal subset feature selections. Therefore, in recent years hybrid approaches such as FCD with LSI [9], MAMR-GA [10], IG-GA and IG-PCA [8] and many other approaches have received significant attention. The weaknesses of one model can be strengths of another model, which, when used together can achieve better accuracy in feature space optimization. Based on this observation, we propose a Modified Hybrid Union (MHU) approach, in which top-k relevance score features selected using union method are merged with the common features selected using the intersection approach, to achieve a more optimal feature space. The proposed MHU approach uses two different types of feature selection models, i.e., TV combined with DF and MAD combined with AC. To substantiate the efficacy of the proposed MHU method, two clustering algorithms, K-means document clustering and Flocking based document clustering [11] is applied to the feature space obtained from the modified hybrid union method.

The remainder of this paper is organized as follows: Section II briefly discusses existing work, while Section III provides details about the proposed technique. Section IV presents the two document clustering algorithms that were used to evaluate the proposed feature selection models. Section V presents a discussion on the experimental results and evaluation, followed by conclusions and references.

II. Related Work

To deal with shortcomings of individual dimension reduction techniques, several researchers have experimented with the possibility of applying various combinations of multiple

978-1-5090-6367-3/17/$31.00 ©2017 IEEE

dimension reduction methods to reduce redundant, irrelevant and noisy features and to enhance the resultant final feature space, resulting in hybrid models. As it is, *filter methods* [12] are more commonly used to perform statistical analysis of the feature set and select distinct features, whereas, *wrapper methods* [13] use learning techniques in order to assign relative importance of features in a set. The limitations of pure filter and wrapper methods are high computational cost and the feature set obtained is more biased towards learning methods.

Akadi et al [10] presented a study on the processes adopted for selecting gene subsets using two-stage dimension reduction methods such as genetic algorithms (GA) and maximum relevance-minimum redundancy (MRMR). They used support vector machines (SVM) and Naive Bayes (NB) classifiers, by which they were able to select the smallest gene subset with good accuracy. However, their method fails to deal with datasets with noisy features. A Hybrid reduction method approach using a combination of feature selection methods such as Information Gain and Genetic algorithm (IG-GA) and combination of feature selection and feature extraction methods such as information gain and principle component analysis (IG-PCA) was presented by Uguz et al [8] and Ghareb et al [14], which achieved good accuracy, but were computationally intensive.

Hoque et al [15] proposed a greedy method for feature selection based on Mutual Information. They studied mutual information between feature-class and feature-feature to select optimal feature subset of data sets. However, this method only focuses on selecting relevant and non-redundant features and any noisy features adversely affects the performance. Li et al [16] introduced new supervised feature selection method CHIR (based on the Chi-square method) which selects relevant terms known to categories by utilizing known class label information. Forsati et al [17] proposed ant colony optimization (ACO) based techniques for feature selection and a hybrid approach to avoid local optima called enRiched ACO (RACO). They considered previously determined traversals as a good source to guide future explorations. They proposed three RACO-based feature selection (RACOFS) algorithms, with an assumption that newer features have a higher priority and to keep track of the globally optimal solution. The algorithm showed better performance for small and medium data sets, but required additional optimization for large data sets, thus making it computationally intensive.

Kumar et al [18] presented a video summarization technique where, visual features are extracted from key frames of adjoining events of different segments. K-means clustering is applied to group similar frames together and the frame which is nearer to the centroid is considered as a key-frame. Our proposed approach is applied to document clustering, but can be easily extended for multimedia content, as we consider feature selection methods for optimal feature vector generation, while video summarization methods use clustering based methods for determining the best-suited features in frames.

In most of these works, different combinations of feature selection models have been proposed. Although hybrid approaches reduced the number of final features to be considered by a marginal number, solely using union or intersection methods have some drawbacks. This could either lead to the selection of larger amount of irrelevant features or could cause the loss of some important features. This motivates us to design better models that can build on the merits of both union and intersection methods. In this paper, an enhanced hybrid feature selection model, i.e., modified hybrid union (MHU) method is proposed which aims to overcome these shortcomings.

III. Proposed System

Figure 1 depicts major processes adopted by methodology for MHU model generation and unsupervised learning. We describe each of these processes in detail next.

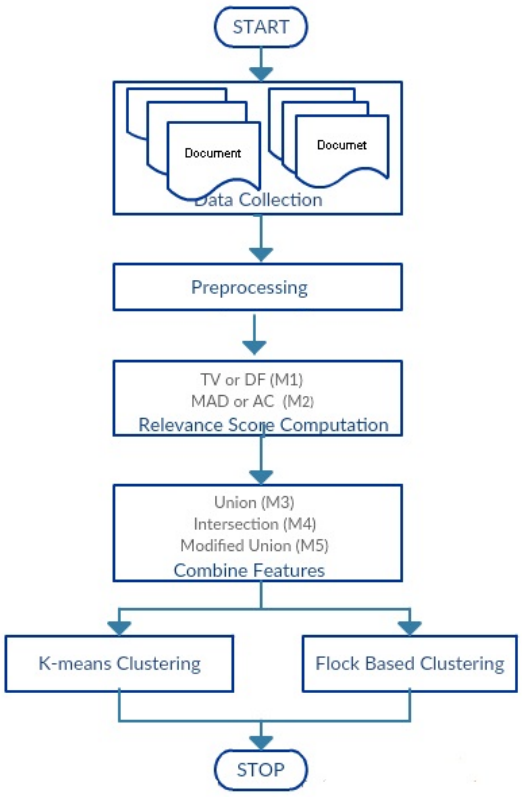

Fig. 1: Proposed Work flow

A. Data Collection

For experimental validation, we used two different datsets - Reuters News dataset (unstructured) and OWL-S TC Web service dataset (semi-structured). The Reuters-21,578 dataset [19] contains 8 categories and 2449 documents as - acq (700), crude (253), earn (700), grain (41), interest (190), money-fx(206), ship(108), trade(251). The OWL-S TC Service collection which is available online on http://projects.semwebcentral.org/projects/owls-tc/ consists of a collection of 1090 web service descriptions from different domains like - Communication (58), Economy (359), Education (285), Food (34), Geography (60), Medical (73), Simulation (16), Travel (165) and Weapon (40). Our objective is to validate the effectiveness of the proposed feature selection approach in capturing the most relevant features for each domain, so that highest clustering purity can be obtained.

B. Preprocessing

For each document in the dataset, several preprocessing steps are applied as detailed below.

1) Element name extraction: Since OWL-S documents, which are basically XML files. The service-specific information is contained with element names which represent the service's functionality. Hence, we use a OWL-S parser to specifically extract only element name-phrases from XML DOM tree of the OWL-S documents. This step is not required for Reuters documents.

2) Tokenization: Tokenization is a process of splitting any document into words or symbols, and removal of special characters. The element name-phrases obtained from OWL-S and Reuters-21578 documents are processed to obtain term tokens which form the initial feature space for each document.

3) Stop-word Removal: In the initial feature space, low-value words often occur (for example, *is, am, we, thus, where, a, the, who, be, also, on* etc), which contribute very little towards the domain of each service. Hence, these stop-words are removed by using a standard English language stop-word list, thus reducing the computational complexity.

4) Stemming: A process of stemming is used to identify the root word from the derivationally related formatted term. For example, consider words like *nationalist, nationalism, national* etc, which are derived from the original root word 'nation'. Removing these multiple terms which have the stem can further reduce the original term space. We used the Porter Stemmer [20] for performing stemming of the terms obtained from the element name-phrases.

5) Term Weighting: Documents can be mapped to vector space as per the Vector Space Model [21], and the relative importance of each term in the document-term space can be obtained by using term weighting. Various term weighting schemes are available in literature [22], of which, we used the Tf-idf (Term frequency-inverse term frequency) weighting scheme (given by Eq. 1).

$$Tf - idf(i,|d) = (\sqrt{wf_{id}})Ln(\frac{N}{df_i}) \quad if \quad wf_{id} \geq 1 \qquad (1)$$

where, Tf is the frequency of a word present in document and idf is the number of other documents in the corpus which contain that specific word.

6) Document feature vector generation: After generating a global dictionary of features obtained from all documents in the corpus, a $Document \times Feature$ matrix is created, which contains the feature frequency w.r.t to each document. This matrix representation of the documents in terms of features is also known as a bag-of-word (BOW) representation.

7) Similarity calculation: For combining and separating documents based on characteristics of data, we need to measure the similarity between two documents represented in vector format. The similarity score helps in identifying similar services so that clustering can be performed. For this purpose, the cosine correlation measure was used (Eq. 2).

$$cos(x_p, x_j) = \frac{x_p x_j}{|x_p||x_j|} \qquad (2)$$

where, x_p and x_j are two document feature vectors.

C. Relevance Score Computation

After representing each service document in a 2D vector format by its feature vector, with the frequency of occurrence as a weighted value, different feature selection methods are applied for relevance score calculation. Each of these techniques are described next.

1) Term Variance (TV): Term variance is a process of calculating each term's score on the basis of deviation of term with respect to the mean, w.r.t all other documents. Hence, a term which is not uniformly spread over the corpus is considered as more important than other features. Mathematically, term variance can be calculated as per equation 3.

$$TV_i = \frac{1}{n}\sum_{j=1}^{n}(X_{ij} - \bar{X}_i)^2 \qquad (3)$$

where, X_{ij} is the value of i^{th} feature with respect to j^{th} document.

2) Document Frequency (DF): This is an effective feature selection method used to calculate the relevance of a feature in a particular document as it generates a score by counting the number of documents that contain a particular term. It is assumed that more the number of documents covered by term more it is important.

3) Mean Absolute Difference (MAD): This method is used to assign a relevance score based on the difference of sample weight and mean value of a term w.r.t all other documents. For experimental purposes, a threshold relevance score value of 0.9 was used.

$$MAD_i = \frac{1}{n}\sum_{j=1}^{n}(X_{ij} - \bar{X}_i) \qquad (4)$$

Here, X_{ij} is the value of i_{th} feature with j_{th} document and mean value can be calculated as per equation 5.

$$\bar{X}_i = \frac{1}{n}\sum_{j=1}^{n}(X_{ij}) \qquad (5)$$

4) Absolute cosine (AC): This method is used to remove any redundant features based on their similarity score. The number of similar characters present in a word will decide the similarity score and if it is greater than a threshold value then, that term can be discarded. A threshold value of 0.40 was considered for similarity in this case. Absolute Cosine values for a term can be calculated using equation 6.

$$cos(\theta_{w_i}, w_t) = |\frac{W_i.W_t}{||W_i|||W_t||}| \qquad (6)$$

D. Combining Features

Conventionally, the union and intersection approaches are used for merging features sublists. Initially, we consider Term Variance as M_1 (Model 1) and Document Frequency as M_2 (Model 2), explained next.

Let D be the set of documents present in dataset, after pre-processing. Let F be the original feature set and FS_1 be the number of features selected after applying model M_1. Let

FS_2 be the number of features selected after applying model M_2.

$$F = \{f_1, f_2, f_3, f_4, f_5, ..., f_t\}$$

$$FS_1 = \{f_{11}, f_{12}, f_{13}, f_{14}, f_{15}, ..., f_q\}$$

$$FS_2 = \{f_{21}, f_{22}, f_{23}, f_{24}, f_{25}, ..., f_l\}$$

Where f_q highlights the total terms selected with FS_1 i.e. Model M_1, and $q \ll$f. f_l highlights the number of terms selected using Model M_2 and $l \ll$f.

Definition 1. Union: FS_1 highlights the total terms selected with model M_l and FS_2 highlights the total terms selected with model M_2. Then, the union approach simply merges the two results to form a new feature sublist, FS_3, where, $FS_3 = FS_l \cup FS_2$ and number of features created i.e. f$^{'}$ are always greater than or equal to q+l.

Definition 2. Intersection: Intersection approach finds the common features obtained from model M_1 and model M_2 which is FS_4 where, $FS_4 = FS_l \cap FS_2$ and number of features created i.e. f$^{"}$ is always less than or equal to q or l.

In this work, we considered the top 50% of features selected by different models, which were hybridized using the union and intersection approaches. These two approaches are used together to counter-balance their individual ill effects, as the union approach may increase the count of features and the intersection approach tends to discard some important features.

Definition 3. Modified Union: For selection of both highest ranked terms and more common terms, we propose a Modified Hybrid Union (MHU) approach, where C_1% of FS_3 (i.e. features obtained after union approach) and C_2% of FS_4 (i.e. features obtained after intersection approach) are merged together. Heuristically, a ratio of 20:80 was found to be best, i.e., C_1 = 20% of union and C_2= 80% of intersection. As the proportion of union model was increased, computing time also increased, while decreasing the proportion of union model affected cluster purity badly. After performing several experiments with different rations of Union and Intersection models, the ratio 20:80 generated the best clustering results. Here, the number of features created i.e. f$^{"}$ obtained are always less than or equal to q+l.

$$FS_5 = C_1\%ofFS_3 \cup \{C_2\%ofFS_4\} \quad (7)$$

Finally, features selected with different methods are combined i.e. added together to form new models. The MHU model is formed using the combination of top ranked features selected using union and intersection model. Table I gives the brief about formation of each model.

TABLE I: Formation of different models

Models	Composition	Description
M1	TV	Terms selected using TV method
M2	DF	Terms selected using DF method
M3	M1 ∪ M2	Terms got after union of M1 & M2 models
M4	M1 ∩ M2	Terms got after intersection of M1 & M2 models
M5	(20% of M3)+(80% of M4)	MHU Model using 20% of M3 and 80% of M4
M6	MAD	Terms selected using MAD method
M7	AC	Terms selected using AC method
M8	M6 ∪ M7	Terms got after union of M6 & M7 models
M9	M6 ∩ M7	Terms got after intersection of M6 & M7 models
M10	(20% of M8) +(80% of M9)	MHU Model using 20% of M8 and 80% of M9

IV. Document Clustering

To analyze the data for knowledge gain, two different clustering algorithms, k-means and Flock based clustering were applied to the processed data. Clustering algorithms help in determining the various latent patterns and domain-specific features in an unknown dataset. We used k-means and Flock based algorithms for this purpose.

A. K-means Clustering

The k-means algorithm takes a parameter K as input and then partitions n-samples into K clusters iteratively. The first step is to initialize the value of K (in this case, K=9, as the number of classes in the OWL-S TC dataset is 9). Then, it randomly select K documents as cluster centroids. Next, the similarity score between each service and each of the K centroids is computed using cosine similarity measure. If the minimum score to assign each document sample to different clusters is met, then the documents are assigned to those clusters. In the next iteration, the new cluster centroids are recomputed from newly formed clusters. Now, with these newly selected centroids, we recalculate the similarity score and assign documents to those clusters with which the maximum similarity was obtained. This process is repeated until no new reassignment of documents happens from one iteration to another, thus reaching a stable clustering point. Though simple, the main drawback of k-means clustering is that it doesn't consider global optimization to generate optimal number of clusters. Due to this limitation, a heuristic clustering algorithm based on bird flocking in nature, that incorporates a self-organization strategy by considering each document as a social entity was applied to the processed dataset.

B. Flock based Clustering

Flock based clustering algorithms do not require prior knowledge about number of partitions for given data. Reynolds [11] defined the flocking model and used it for implementing computer graphics based animations of flocks of birds or school of fish. We adapted this flocking model to promote clustering behavior for a given set of documents, characterized by their optimized feature vectors. We use the three fundamental steering rules that govern the movement of each interacting entity, *alignment*, *cohesion* and *separation* as defined by Reynolds in the Flocking based clustering algorithm. Figure II depicts the flocking model pictorially. Each of these rules are applied to all interacting entities during each clustering iteration. We describe each steering rule and their importance during clustering below.

- *Rule 1: Alignment.* ensures that each object in the defined clustering space tries to match its velocity with the average velocity of its neighbor in the cluster.

- *Rule 2: Separation.* ensures adequate distance between each document object so as to avoid collisions in the same feature space.
- *Rule 3: Cohesion.* promotes a change to the average position of each entity to move towards that of its neighboring entities i.e., all document objects try to align their movement in the direction of the centroid (average spatial position) of the local flock.
- *Rule 4: Similarity/Dissimilarity.* For promoting document clustering, we enhance the basic flocking model by defining a fourth rule, that of feature similarity and dissimilarity. This is required to influence the change of position of each entity based on its computed similarity with its neighbor. Hence, this computed similarity value can be used to ensure that similar entities are '*attracted*' to each other and dissimilar entities '*repulse*' one another, thus helping in achieving the dual objectives of compact clusters (low intra-cluster distance) and well-separated clusters (high inter-cluster distance).

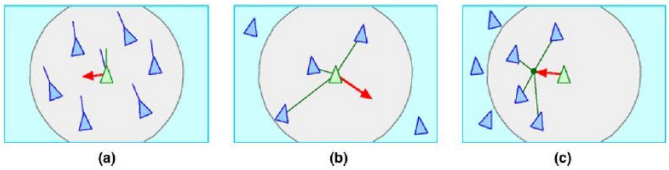

Fig. 2: Basic rules in the flock for boid (a) Alignment, (b) separation, (c) cohesion

For clustering on the document collection, the 2-d matrix of $Document \times Features$ is considered, where, each document is modeled as a randomly moving object in virtual space. Tf-idf score is used as a feature vector of each boid. The behavior (velocity) of each object B with position P_b is influenced by all neighboring objects X within position P_x in its neighborhood. The process flocking based clustering is described below:

- *Step 1:* Initialize the variable MAX-ITERATION count and take input from INPUT-MATRIX.
- *Step 2:* Randomly Initialize initial centroid vectors.
- *Step 3:* Clusters are formed based on the similarity between centroid vectors and other document vectors.
- *Step 4:* For each cluster member, we computed fitness value using Schaffer's F6 function [23]. This is a testing function which is used to evaluate the performance of optimization algorithms. It calculates a fitness value by using oscillations or peaks along 2D plot. We incorporated this fitness function, by plotting documents on the X-axis and features along the Y-axis. The Schaffer's F6 function can be formulated as in equation 8.

$$f(x,y) = 0.5 + \frac{sin^2(x^2 - y^2) - 0.5}{(1 + 0.001(x^2 + y^2))^2} \quad (8)$$

where x and y are dimensions of each document vector. We considered the highest fitness value boid i.e., local maxima generating boid as the BEST-FIT boid and used it as a new centroid of cluster in next iteration.

- *Step 5:* For other members of same cluster. we increase similarity score by the value of difference between BEST-FIT and fitness value of member document vector.
- *Step 6:* Repeat the same from step 3 until we reach either MAX-ITERATION or the condition new-formed-cluster = old-cluster becomes true.

V. Experimental Evaluation

In this section, the experimental setup and the observed effect of the various feature selection models when used for clustering are discussed. The performance of clustering algorithms are evaluated depending on the purity of clusters formed for the given data. The objective is to minimize intra-cluster distance so that compact clusters can be obtained, and to maximize inter-cluster distance, so that well-defined clusters with minimal overlap can be achieved. We used widely known cluster purity measure to know the quality of cluster. Purity can be calculated as the summation of correctly placed objects in each cluster divided by the total number of objects considered for study. The results obtained from different models for k-means and Flock based clustering algorithms are shown in Tables II, III, IV, V and VI.

$$Purity(\%) = \frac{\sum_0^k \max_0^j (\#\ of\ records\ in\ each\ class)}{Total\ number\ of\ Documents} \quad (9)$$

TABLE II: Experimental statistics for K-means and Flock based algorithms using conventional feature selection models on OWL-S TC dataset with original 1370 features

Model	Parameter	K-means	Flock Based
M1	Features Taken	685	685
	Cluster Purity	59.74%	66.66 %
	Total Exec Time	2.37 min	2.34 min
M2	Features Taken	685	685
	Cluster Purity	64.26%	61.49 %
	Total Exec Time	3.14 min	2.39 min
M3	Features Taken	924	924
	Cluster Purity	64.64%	65.93 %
	Total Exec Time	3.18 min	6.15 min
M4	Features Taken	446	446
	Cluster Purity	56.04%	75.71 %
	Total Exec Time	2.09 min	2.11 min
M6	Features Taken	460	460
	Cluster Purity	68.88 %	65.18 %
	Total Exec Time	4.59 min	4.07 min
M7	Features Taken	120	120
	Cluster Purity	47.02 %	43.76 %
	Total Exec Time	1.09 min	2 min
M8	Features Taken	540	540
	Cluster Purity	58.36 %	59.01 %
	Total Exec Time	2.16 min	4.44 min
M9	Features Taken	40	40
	Cluster Purity	47.46 %	44.59 %
	Total Exec Time	1.03 min	1.10 min

Table II represents the results of k-means and flock based algorithms with emphasis on the effect of feature selection models M1 to M9 on OWL-S TC dataset. Each model is composed of various base models and models M3 & M8 and M4 & M9 are built using the union and intersection methods respectively. From table II it is clearly observed that, model

M6 which applies MAD technique to select features, produced best results using k-means algorithm whereas model M4 which is an intersection of features selected using TV & DF achieved best clustering purity for flock based algorithm.

TABLE III: Experimental statistics for K-means and Flock based algorithm using conventional feature selection models on Reuters-21578 dataset with original 9915 features

Model	Parameter	K-means	Flock Based
M1	Features Taken	4957	4957
	Cluster Purity	48.01%	50.55 %
	Total Exec Time	55.25 min	50.18 min
M2	Features Taken	4957	4957
	Cluster Purity	45.44%	46.20 %
	Total Exec Time	48.52 min	48.26 min
M3	Features Taken	6018	6018
	Cluster Purity	47.28%	48.22 %
	Total Exec Time	57.57 min	57.18 min
M4	Features Taken	3896	3896
	Cluster Purity	47.03%	48.42 %
	Total Exec Time	39.56 min	41.38 min
M6	Features Taken	2564	2564
	Cluster Purity	48.34%	45.69%
	Total Exec Time	29.09 min	29.14 min
M7	Features Taken	539	539
	Cluster Purity	31.68 %	32.99%
	Total Exec Time	10.49 min	08.23 min
M8	Features Taken	3034	3034
	Cluster Purity	52.75 %	52.47 %
	Total Exec Time	24.50 min	24.41 min
M9	Features Taken	69	69
	Cluster Purity	33.23%	31.60%
	Total Exec Time	05.07 min	06.32 min

From Table III, it can be seen that the hybrid union model based flock clustering approach produced better results than the k-means algorithm. For k-means algorithm, it was observed that the hybrid union model with a combination of MAD and AC produced good results (shown in Table V). Although the clustering purity in few conventional models is greater, but when it compare to time for execution MHU performs better. Table VI represents the purity and time for execution for MHU model applied to both datasets with k-means and Flock based algorithm. It can be seen that, for flock based clustering approach results produced by a combination of MAD and AC hybrid union model are better when compared to all other conventional models. Figure 6, 7 and 8 show the comparison between k-means and flock based algorithm results when MHU models are used.

TABLE IV: Comparative performance statistics of K-means and Flock based clustering algorithms

Dataset	MHU Models	Parameter	k-means	Flock based
OWL-S TC	M5	Features Taken	462	462
		Cluster Purity	61.77 %	61.21 %
		Total Exec Time	1.58 min	2.01 min
OWL-S TC	M10	Features Taken	143	143
		Cluster Purity	60.20 %	75.161 %
		Total Exec Time	1.28 min	1.15 min
Reuters-21578	M5	Features Taken	3580	3580
		Cluster Purity	47.81 %	50.63 %
		Total Exec Time	38.43 min	33.47 min
Reuters-21578	M10	Features Taken	651	651
		Cluster Purity	62.76 %	63.38 %
		Total Exec Time	12.13 min	10.39 min

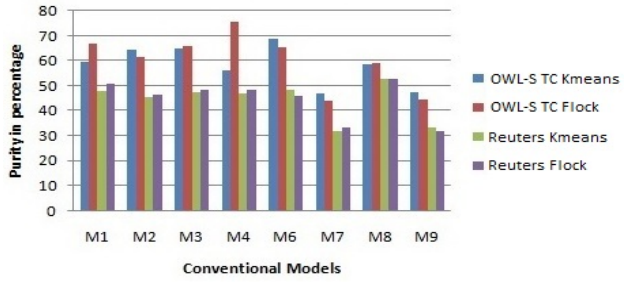

Fig. 3: Observed clustering purity with conventional feature selection models

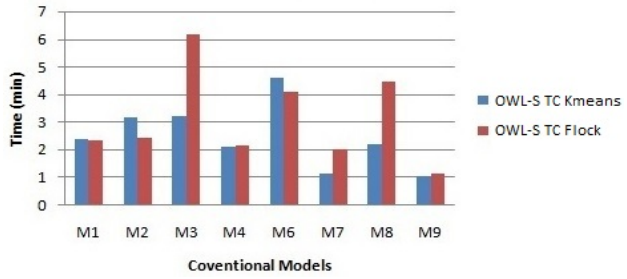

Fig. 4: Observed execution time with conventional feature selection models for OWL-S TC dataset

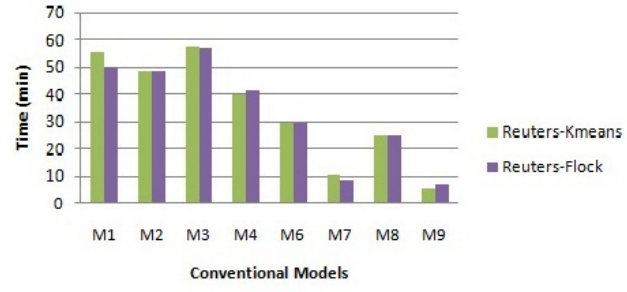

Fig. 5: Observed execution time with conventional feature selection models for Reuters-21578 dataset

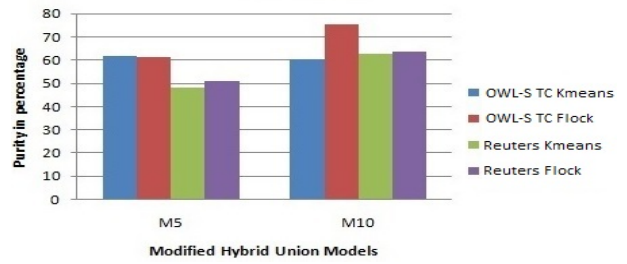

Fig. 6: Observed clustering purity for K-means and Flock based clustering using Modified Hybrid Union Models (M5 and M10)

VI. Conclusion

In this paper, multiple feature selection models are evaluated and their effect on document clustering accuracy has

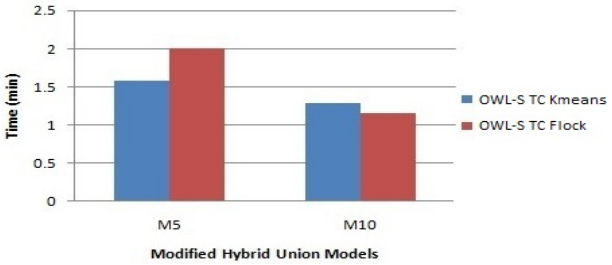

Fig. 7: Observed execution time for K-means and Flock based clustering using Modified Hybrid Union Models (M5 and M10) for OWL-S TC dataset

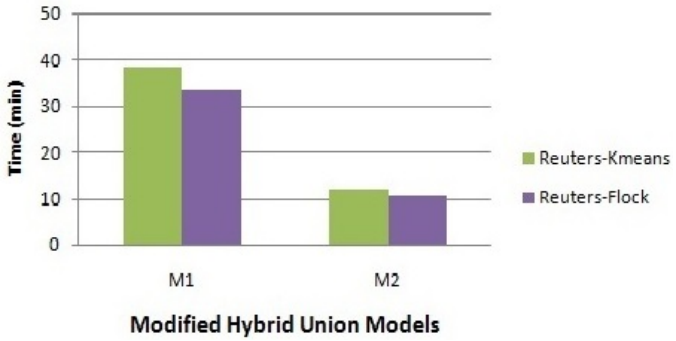

Fig. 8: Observed execution time for K-means and Flock based clustering using Modified Hybrid Union Models (M5 and M10) for Reuters-21578 dataset

been measured. We proposed hybrid feature selection models called Modified Hybrid Union (MHU) which are designed based on different combinations of feature selection methods and by combining both union & intersection strategies. MHU helps in identifying highly relevant features by considering the individual strengths and weaknesses of each component in the underlying models. We used K-means clustering and Flock-based clustering on the OWL-S TC and Reuters-21578 data set to evaluate for effectiveness of the proposed hybrid feature selection models. Experimental results clearly indicate that the proposed hybrid feature selection methodology significantly improved the performance of clustering algorithms, in terms of optimizing the clustering time and reducing the dimensionality of the feature space. Moreover, in future the performance can be improved using a data parallelism approach applied in TFIDF computations and also further research is possible by avoiding the use of sparse matrix generated for bag-of-words.

REFERENCES

[1] J. MacQueen, "Some methods for classification and analysis of multivariate observations," *Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability*, pp. 281—297, 1967.

[2] J. C. Bezdek, *Pattern Recognition with Fuzzy Objective Function Algorithms*. MA, USA: Kluwer Academic Publishers Norwell, 1981.

[3] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the em algorithm," *Journal Of The Royal Statistical Society, Series B*, vol. 39, no. 1, pp. 1–38, 1977.

[4] L. J. Heyer, S. Kruglyak, and S. Yooseph, "Exploring expression data: Identification and analysis of coexpressed genes," *Genome Res.*, vol. 9, no. 11, pp. 1106–1115, 1999.

[5] M. Girolami, "Mercer kernel-based clustering in feature space," *IEEE transactions on neural networks*, vol. 13, no. 3, pp. 780–784, 2002.

[6] K. P. S. J. Kriegel, H.-P. and A. Zimek, "Density-based clustering," *WIREs Data Mining Knowl Discov*, vol. 1, no. 3, pp. 231–240, May/June 2011.

[7] C. T. Zahn, "Graph-theoretical methods for detecting and describing gestalt clusters," *IEEE Transactions on Computers*, vol. 20, no. 1, pp. 68–86, 1971.

[8] H. Uguz, "A two-stage feature selection method for text categorization by using information gain, principal component analysis and genetic algorithm," *Knowledge-Based Systems*, pp. 1024–1032, 2012.

[9] J. Meng, H. Lin, and Y. Yu, "A two-stage feature selection method for text categorization," *Computers & Mathematics with Applications*, vol. 62, no. 7, pp. 2793–2800, 2011.

[10] A. Akadi, A. Amine, A. Ouardighi, and D. Aboutajdine, "A two-stage gene selection scheme utilizing mrmr filter and ga wrapper," *Knowledge and Information System*, pp. 487–500, 2011.

[11] C. Reynolds, "Flocks, herds, and schools: a distributed behavioral model," *Computer Graphics 21*, pp. 25–34, 1987.

[12] H.-H. Hsu, C.-W. Hsieh, and M.-D. Lu, "Hybrid feature selection by combining filters and wrappers," *Expert Systems with Applications*, vol. 38, no. 7, pp. 8144–8150, July 2011.

[13] Y. Peng, Z. Wu, and J. Jiang, "A novel feature selection approach for biomedical data classification," *Journal of Biomedical Informatics*, vol. 43, no. 1, pp. 15–23, August 2009.

[14] A. S. Ghareb, A. A. Bakar, and A. R. Hamdan, "Hybrid feature selection based on enhanced genetic algorithm for text categorization," *Science direct Expert Systems with Applications*, vol. 49, pp. 31—-47, 2016.

[15] N. Hoque, D. Bhattacharyya, and J. Kalita, "Mifs-nd: A mutual information-based feature selection method," *Expert Systems with Applications*, vol. 41, no. 14, pp. 6371–6385, October 2014.

[16] Y. Li, C. Luo, and S. M. Chung, "Text clustering with feature selection by using statistical data," *IEEE Transactions On Knowledge And Data Engineering*, vol. 20, no. 5, pp. 641–652, May 2008.

[17] R. Forsati, A. Moayedikia, R. Jensen, M. Shamsfard, and M. R. Meybodi, "Enriched ant colony optimization and its application in feature selection," *SI Computational Intelligence Techniques for New Product Development*, vol. 142, pp. 354–371, October 2014.

[18] K. Kumar, D. D. Shrimankar, and N. Singh, "Eratosthenes sieve based key-frame extraction technique for event summarization in videos," *Multimedia Tools and Applications*, pp. 1–22, 2017.

[19] D. D. Lewis, "Reuters-21578 text categorization test collection, distribution 1.0," *http://www. research. att. com/˜ lewis/reuters21578. html*, 1997.

[20] M. F. Porter, "An algorithm for suffix stripping," *Program*, vol. 14, no. 3, pp. 130–137, 1980.

[21] G. Salton, A. Wong, and C.-S. Yang, "A vector space model for automatic indexing," *Communications of the ACM*, vol. 18, no. 11, pp. 613–620, 1975.

[22] G. Salton and C. Buckley, "Term-weighting approaches in automatic text retrival," *Information Processing and Management*, vol. 24, no. 5, pp. 513–523, 1988.

[23] J. M. Dieterich and B. Hartke, "Empirical review of standard benchmark functions using evolutionary global optimization," *Applied Mathematics*, vol. 3, no. 10, pp. 641–652, July 2012.