Applied Mathematics & Information Sciences
An International Journal

http://dx.doi.org/10.12785/amis/090306

Enhanced Facet Ranking and Text Classifier for Opinion Mining

G. R. Brindha[1,*], *S. Prakash*[2], *B. Santhi*[1] *and P. Swaminathan*[1]

[1] School of Computing SASTRA University, India
[2] AVVM Sri Pushpam College Poondi, India

Received: 18 Jan. 2014, Revised: 3 Mar. 2014, Accepted: 4 Mar. 2014
Published online: 1 May 2015

Abstract: The enormous growth and usage of social networks offer positive ways to any business by sharing the emotions, feelings and experiences. Web users are benefited with valuable online reviews. To utilize the reviews effectively, researchers are working on necessary methods and ideas such as classification of positive and negative sense of reviews, ranking the facet in the reviews to make the effective classification etc. This study aims to propose a novel facet identification namely Facet Based Adjective identification method (FBAI) for efficient feature selection of reviews. The next algorithm FacetRank marks facet of each opinion from review set with positive, negative and neutral polarity. To classify the ranked facets, a novel Cluster based k Nearest Neighbor (C-kNN) algorithm is proposed. Constrained single pass clustering algorithm is combined with existing kNN classification algorithm to classify the review set as positive or negative. C-kNN reduces the resemblance checking calculation and can process high dimensional data which enable dynamic classification. This analysis takes household product reviews as input data set. The ranked review set (using FBAI+FacetRank) is given to kNN and C-kNN for classification. F1 score of C-kNN 2.43 % higher than kNN. Linear time complexity of C-kNN achieved is 68% of kNN.

Keywords: Ranking; FacetRank; FBAI algorithm; Classification; Clustered-kNN

1 Introduction

Blogging is increasing in exponential rate where people post their interest, emotions and experiences about a product, event or a person [1]. Reviews about products have much influence on the enterprise growth rate [2,3]. So the enterprises should follow and analyze their customers' opinions to have competitive intelligence [4]. Information retrieval and sentiment analysis plays a major role to achieve this goal and there are more studies to mine opinions [5,6,7]. For supervised learning, facet selection and classification is required to fine tune efficiency and over fitting [8]. Support Vector Machine (SVM) and Nave Bayes (NB) classify test data set based on the trained model, whereas kNN does not have training process. kNN works based on the distance between labeled data points and the test data. kNN does not need any assumption about data set [9]. This article provides a cluster based model to kNN to enhance the classification. The degree of preprocessing always influences the text classification. Hence this article takes the preprocessed review set and proposes a new module to rank the facet of online reviews. This module contains FBAI and FacetRank algorithm to select and rank the facets.

Next section talks about existing research work on ranking and classification of text data. The other sections are framed as follows. There are three phases namely Mining and Ranking (Phase-I), Classification and Clustered classification (Phase-II) and Performance analysis by comparison (Phase-III). Fig 1 depicts the flow of the study which includes proposed algorithms in dotted lines.

The Phase-I deals about a novel facet identification algorithm, FBAI (Facet Based Adjective Identification), which analyzes the general sentiments and determine the semantic direction of a particular facet present in a review. As a chain process of FBAI, another novel algorithm FacetRank is proposed to rank the features of each opinion. Phase-II explains text classification of kNN and a novel C-kNN algorithm to classify the online reviews, which are ranked by the algorithm proposed in Phase-I.

* Corresponding author e-mail: brindha.gr@ict.sastra.edu

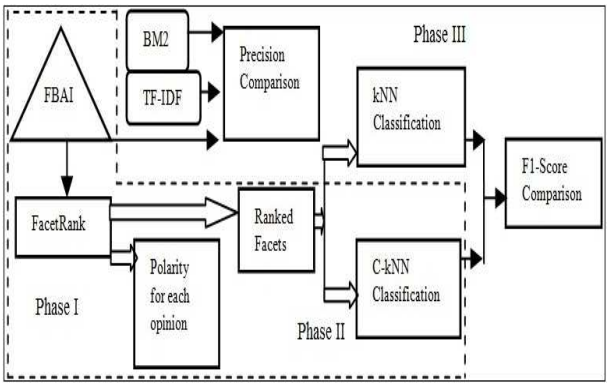

Fig. 1: Work flow of the study

In Phase-III, FBAI is compared with existing popular algorithm TF-IDF and BM25 to prove its facet selection efficiency. Further the comparative analysis of kNN and C-kNN is done in this phase.

2 EXISTING SCHEMES

This section provides details on existing schemes in ranking and text classification process of opinion mining area. Popularly known ranking algorithms such as TF-IDF and BM25 are used in many studies. One of the researchers [10] used Language Model, TF-IDF, and Okapi BM25 to retrieve information and found Language model out performed. Term occurrence, binary occurrence and TF-IDF are used [11] for word set of trigram, bigram and unigram. The evaluation metrics namely precision, recall and F1-score are derived for three fold and tenfold cross validation for the above said method. From the comparison of these results they find that term occurrence does not perform well for all n-grams whereas TF-IDF & binary occurrence give better results. To find semantic orientation of reviews, high adjective algorithm is proposed and its effectiveness is proved with TF-IDF and TF [12]. Garcia Esparza et al. [13] compare CB10, CB100, BM25 and TF-IDF ranking algorithms for short messaging services (product recommendation) data set. Precision, recall, F1-score metrics shows TF-IDF performed better than others. Yang & Ko [14] use word count based TFIDF to focus on the specificity of bigram. To rank each sentences, Gong & Liu [15] apply weighted term frequency along singular value decomposition matrix. This is used to create semantic structure and to separate high valued sentences.
One of the traditional supervised machine learning method, support vector machines (SVM) [11] is experimented in different domains for variety of reviews with three corpus methods. NB and SVM are compared and the result is: character based bigrams do better, that too with NB than SVM [16]. Tan and Wang propose a new Model Adjustment (MA) algorithm to improve the performance of centroid classifier which gives a noticeable accuracy than SVM [17]. Including SVM four classifier are compared to find polarity shifters, negation and modifiers [18]. For the combination Bayesian classifier and kNN, relevance feedback gives better outcomes than a single classifier [19]. Following this wrapping method, six ensemble methods belonging to a same family, is applied for text categorization [20] and training [21]. Also different variety of feature sets, like higher-order n-grams [16,22] part-of-speech based features [23], dependency relations on words [22,24], are utilized to process sentiment classification output. Based on weighting schemes, combination of new and old data set of Amazon are classified [11] using Support Vector Machine. A novel FRank algorithm is applied on movie data set [25] and it is compared with existing count score algorithm. Further the performance of both algorithms is compared through SVM and NB classification. To process binary quantification issue, kNN is used and its behavior based on prevalence estimation is checked [26]. For this process two different weighting strategies are used and found only the statistical difference. Recently in a study [27], performance of ensemble methods are analyzed based on maximum entropy, NB, decision tree, kNN and SVM classifications. They find Random-Subspace method works better for sentiment classification. kNN is also used [28] for text categorization and comparatively poor performance against Rocchio classifier. To speed up kNN categorization, they employ improved Rocchio model. Jiang et al. [29] propose improved kNN which combines one pass clustering method with classification model for effective text classification. In comparison with SVM, NB and kNN this proposal has great scalability.
Arabic word net is used and kNN with various similarity processes are done [30]. Another paper [31] also uses different similarity measurement checking in kNN for hierarchical categories of documents. Multivariate Information Bottleneck method is used to view the related data from partitions. Among various real time classification applications, they apply Bayesian network to cluster data based on topics [32]. From getting motivation of above said analysis this article combines ranking and C-kNN classification.

3 MINING AND RANKING PHASE

The major focus of this method is to build up an opinion seek system to mine the emotions and extract the necessary information connected to the product facets. And also it can rate them as positive, negative or neutral. This facet based opinion mining will help to focus on the facets of the opinion or experience of persons. Fig 2 shows the structural design that implements the following processes. The web user, who needs decision making information about product from ton line reviews can use this proposed system in less time and can get polarity of

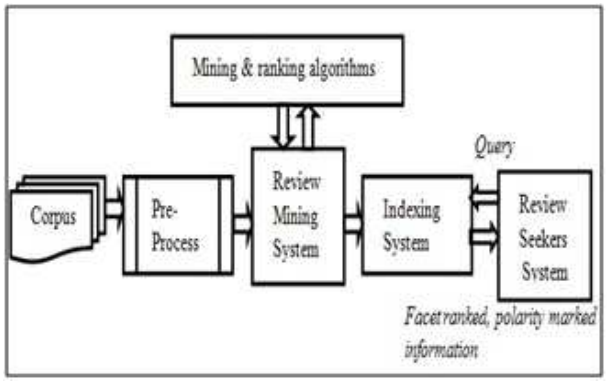

Fig. 2: Role of proposed algorithms in facet processing

each review individually.

Pre-Process: Reviews are processed in the acceptable structure of algorithms such as segregation of reviews in to text files, marking title for each review etc.

Review Mining System: This system comprises the following modules in addition to the significant processes of computing the distance of nearest adjective to a noun.

Tokenization: Splitting the words and providing identification numbers.

Part-Of-Speech Tagging: Grouping nouns and adjective collection

Stop Word Removal: removing unwanted words like 'a', 'the' etc.

Mining and Ranking Algorithms: FBAI extracts most relevant adjectives/emotions related to domain i.e., area from which reviews are taken. FacetRank weighs the facets based on intensity of adjective and the negation process. It also provides individual polarity of each opinion.

Indexing System: To handle recovery and to provide decision support information effectively, facets and reviews are indexed.

Review Seekers System: This system receives query about a product through user interface from users. It does pre process such as stop word removal (removing unwanted words like 'a', 'the' etc, stemming (finding root word) etc. Then it provides ranked facets and individual polarity of reviews to the user. The mining system contains the following modules:

1.Identification and extraction of facets which are recognized as significant for each review
2.Allocation of ranks to the selected facets and individual opinion polarity

3.1 Description of Facet Based Adjective Identification Method (FBAI)

The aim FBAI is to differentiate the most significant facets of opinions from the irrelevant content. This method recognizes the prospective adjectives and nouns from the input content. The weight of nouns is set to zero. Depending on the depth of the adjective the weight is increased. After processing all the reviews, these accumulated weights are utilized to rank the nouns. The allocations of weights are in such a way that each noun in the review has weight. These weights are used to extract the nouns which scores above the threshold value γ. Threshold is fixed depending on the reviews and evaluation. FBAI pseudo code is given below.

Algorithm 1 FBAI

Input: GageFacet(Array Of opinions)
Output: facet_set matrix with rows as weight and column as facets
for each opinion in the set do
parse the opinion and remove noise
for each line in the review do
for each adjective in a sentence do
obtain their nearest noun
$weight\_set[noun] = weight\_set[noun] + 1$
$prospective\_facet\_set$ *gets* $\{\}$
if weight_set[noun]$> \gamma$
prospective_facet_set[noun]= weight_set[noun]
return prospective_facet_set

The proposed FBAI algorithm performs better than the existing algorithms TF-IDF and BM25. The performance analysis in terms of precision is given in section 5.

3.2 Description of FacetRank Algorithm

The next step in proposed module is to rank the extracted facets through weights, assigned by FBAI method. FacetRank has two modules namely ranking and marking individual opinion polarity. This procedure needs four inputs:

1.Adjective list used in the review: mined words.

2.Weight for adjectives: includes positive, negative or neutral and also depth of the word. For example 'good' is positive, but 'excellent' is stronger than 'good'. The weight allotment is between -5 to 5 (negative opinion and positive opinion) to each reviewed word. Top weight specifies stronger positive review.

Weights and weighing: Consider the sentences below.

–This Bread is good.
–This Bread is better.
–This Bread is best.
–This Bread is very nice.

Obviously all are positive sentences. Each and every sentence has its own emphasis level. There is a need of a method to find positivity or negativity of opinion, to make conclusion about the opinion target. Weight level of the opinion really matters when the things are compared. The

following paragraph explains opinion weighing process. Opinion can be weighed based on the adjectives used. For example in automobile domain, words often used are long lasting, good, work, amazing, awesome, worst, out of control, sucks, troublesome etc., Table 1 defines some sample weights. But some words get direction in the way it is used. Those words act sometimes positive and some times negative.

Table 1: Word and Weight

Sample Words	Weight	Direction
long lasting	1	+1
Good	1	+1
Work	1	+1
Amaze	3	+1
Awesome	3	+1
Bad	1	-1
Worst	2	-1
Outofcontrol	1	-1
Suck	1	-1
Troublesome	2	-1

3.Direction determiner: The direction determiner or inverters, determine the direction of other words. For example, consider the sentence: 'Food is not good', here the word 'not' determines the direction of opinion, and leads to a separate logic to find the weight of an opinion when a determiner is present.

Table 2: Direction determiner

Words in negative sense	Direction
Example :NOT	-1

In the above example, 'good' gets +1 from weight level chart. NOT is direction determiner and it is negative interpreter so it takes -1 (Table 2). The weighing formula is,

$$OpinionWeight = WL * DD \quad (1)$$

Where WL is weight level and DD is direction of determiner. By applying this to the above example: Opinion Weight of 'Food is not good'.

$$OpinionWeight = 1 * -1 = -1 \quad (2)$$

Neutral words: There are few words which do not convey any direction. For example, consider the sentence: 'Food is ok.' It is neither positive nor negative. But the same word in other place gives different meaning when we use it with NOT: 'Food is not ok.' Now this statement is definitely a negative comment.

4.Set of prospective facets: Facet Based Adjective Identification (FBAI) procedure processes the review line by line, and for each line words are retrieved, which are nearest to the required facet. The weights of a facet are the summation of weight of opinionated words related to that facet. The weight of the identified facets are added together to evaluate the review. For each facet, average weight is calculated per reviewed word. This weight is utilized to rank the facets of opinion, with the base of perception such a way that positive means like and negative means dislike. Our ranking algorithm is given below:

Algorithm 2 FacetRank

Input: adj_weights, inverted_words, prospective_facets, opinions

Module I: Output: Ranked_ facets

universal_noun_weight={}
universal_noun_adjective_numerate={}
for all opinions in the set do
opinion_noun_weight={}
opinion_noun_adjective_numerate={}

for all sentences of the opinion do
left_content={} // two words can be retained
sentence_weight=0
for every words in a sentence do
if word in adj_weights
weight=adj_weight[word]
if inverted_word in left content
inverted = true;
else
inverted =false;
weight = Level_Val(adj_weight[word],inverted)
nearest_noun=find_nearest_noun(word)
opinion_noun_weight[nearest_noun]+=weight
opinion_noun_adjective_numerate[nearest_noun]++
universal_noun_weight[nearest_noun]+=weight
universal_noun_adj_numerate[nearest_noun]++

Module II: Output: polarity of individual opinions

sentence_weight+=weight
revise left_content
cumulative_weight=Σ opinion_noun_weight
cumulative_adj=Σ opinion_noun_adj_numerate
average_weight= cumulative_weight/cumulative_adj
if average_weight>0
mark polarity $\leftarrow$ positive
else if average_weight<0
mark polarity $\leftarrow$ negative
else mark polarity $\leftarrow$ neutral
average_facet_weight<={}

for all nouns in universal_noun_weight do
average_facet_weight[noun]=universal_noun_weight[noun]/
universal_noun_adj_numerate[noun]
Rank the facets by average_facet_weight

Algorithm 3 Level_Val

Input: adjective, inverted
Output: Level_val
get the weight level and direction from weight table for the adjective
val = weight level
If (inverted)
val = weight level * -1
end

The weighing for the facets extracted is as follows:

$$OW = (x.Headingweight + Contentweight)/(x+1) \quad (3)$$

In the above equation 3 (Opinion Weight), x is weight coefficient,

$$Headingweight = \Sigma aw_h/|a_h| \quad (4)$$

and

$$Contentweight = \Sigma aw_c/|a_c| \quad (5)$$

where aw_h and aw_c indicates adjective weights of heading and content. Adjectives in heading and content are a_h and a_c. The interpretation of heading is generally a good outline which retrieves the mood of the reviewer. So it should be awarded top weight *(x)*. While applying this method, parameter x is used suitably based on data.
FacetRank provides facet wise rank and positive, negative or neutral polarity for individual opinions of review set.

4 Classification and Clustered classification Phase

This section includes kNN classification and the proposed C-kNN classification of ranked review set. Both classifiers process the opinion set to provide the overall polarity. To check the performance of C-kNN, kNN is applied on the review set.

4.1 KNN classifier

The k-Nearest Neighbor classifier operates to classify unknown features by relating the unknown to the known, based on distance between two words. Distance between two words conveys similarity. Similarity between two features conveys the distance. When any two points are similar then the distance between the points are less and vice-versa. The distance is computed with a distance function. The feature distance is to get the nearest ranked features. During learning process kNN does not abstract information. Hence kNN is called lazy learners unlike feed-forward neural networks, which is called eager network, since proper abstraction is happened while learning [33]. Unlike Nave Bayes which does parameter estimation, kNN memorizes the labeled data and compares the test data set. KNN works with simple logic by locating the nearest neighbor in problem space and labeling the unknown points with a known label based on neighbor points.
Consider this test review t (ranked by FacetRank) for kNN classifier to find the k nearest neighbor from the labeled reviews and gain the entrant categories based on the group of k neighbors. The resemblance of t with each neighbor review is the gain for the group of the neighbor review. When more number of k nearest reviews belongs to same group, the sum of gain of that group is the resemblance gain group for the test review t. Then the highest gain value is given to test review t. Prediction rule of kNN (F(t)) is as follows:

$$maxgain(t, G_j) = \Sigma_{r_i \in kNN} resem(t, r_i) \quad (6)$$

Where F(t) is test review label for t.
$maxgain(t, G_j)$: gain for the group point G_j based on t.
$resem(t, r_i)$ is the resemblance between t and the labeled review r_i.
Though this non parametric method is easy and effective, it takes more time to classification and accuracy is affected by noisy reviews.

4.1.1 Distance functions

The kNN classification in this article takes Euclidean distance to calculate the distance between two review points.
Euclidean Distance(ED): The straight line between two points can be calculated using Euclidean distance. Suppose if A and B are two points and the distance between these points is,

$$ED(A,B) = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \quad (7)$$

To calculate the distance for the data points in N dimensions for continuous attributes,

$$ED(A,B)\,forN = \sqrt{(\Sigma_{k=1}^{n}(A_k - B_k)^2)} \quad (8)$$

A_k and B_k are the coordinates of A and B in the dimension k.

Normalize Scale: Normalization improves the performance of kNN for the continuous attributes.
Thus kNN classifies the review set and provides positive negative or neutral opinion of the whole review set. Next section proposes C-kNN to classify the above ranked review set. Comparative analysis of kNN and C-kNN algorithms are given in section 5.

4.2 Novel C-kNN Review Mining classifier

As explained in the previous section, kNN utilizes all labeled reviews (data set) to classify the test review which includes vast similarity calculations. To handle this problem, cluster based kNN for text categorization is proposed and named as Cluster-kNN (C-kNN). This proposed model gets classification model by using conditioned cluster algorithm. Then kNN is applied for the classification of test review documents. Word ranks for the reviews are calculated using FBAI+FacetRank module.

4.2.1 Cluster Based Model Construction

Clustering method is used to segregate review documents (on line opinions) into clusters of related objects. This is an unsupervised method to learn unknown data. Text clustering takes resemblance degree of reviews in same cluster is more and different cluster is least. So preprocess the review documents using clustering is helpful to find the arrangement and distribution of corpus.

Algorithm 4 Single Pass Clustering

Step 1:Initialize set of clusters S_0, as null set and get input of new text n.
Step 2: Create a new cluster with n and its label is considered as label of new cluster.
Step 3: when no reviews left in review collections, process step 9, else go to step 4.
Step 4: Read a new review n, calculate the resemblance between n and all clusters G in S_0 with cosine function. Then find cluster G_j^0.
Step 5: G_j^0 in S_0, nearest to review n with $resem(n,G_j^0) >= resem(n,G)$ for all G in S_0.
Step 6: If $resem(n,G_j^0) < \gamma$ or label of review n is not same as the label of nearest cluster then go to step 2.
Step 7: Combine review n into cluster G_j^0.
Step 8: Update rank of words in the cluster G_j^0, then go to step 3.
Step 9: End the clustering process to obtain clustering output $S_0 = G_1^0, G_2^0, G_3^0, ...G_m^0$ Each one of the cluster in S_0 is the classification model.

Method to update rank of words in the cluster (step 7 and 8) is explained below:

$$R_{G_j^0}^{k+1}(w) = (R_{G_j^0}^k(w)|G_j^0| + R(w)_n)/(|G_j^0|+1) \quad (9)$$

Where $R_{G_j^0}^{k+1}(w)$ denotes new rank for the word w in clusterG_j^0;
$R_{G_j^0}^k(w)$is rank of word w in clusterG_j^0;
$R(w)_n$ is rank of word w in review n;
$|G_j^0|$ is number of reviews in cluster G_j^0.

Single pass clustering algorithm takes linear time to segregate reviews, which is to be used to construct classification model in this section.

4.2.2 Review Documents Classification

The classifier kNN has the capability to process text data set effectively. Hence, this proposal combines kNN to classify test review documents with the help of acquired classification model. Classification specifications are as follows.

The test document t as input.
Mark each cluster by gain in S_0 based on t with the help of following formula.
Then set label for cluster with the maximum gain obtained by the test review t.

$$F(t) = ClusterGain(t,G_j) = \Sigma_{(G_j^0 \in kNN)} resem(t,G_j^0) x(G_j^0, G_k) \quad (10)$$

Where, F(t) is the label set to test review t; ClusterGain(t,G_k) is the gain of entrant category G_k based on t;
$resem(t,G_j^0)$ is resemblance of t with the cluster G_j^0 in the model S_0;
$x(G_j^0, G_k \in 0,1$ is the cluster G_j^0 based on G_k;
$x = 1$ denotes cluster G_j^0 is an element of category G_k, or
$x = 0$ denotes cluster G_j^0 is not an element of category G_k.

4.2.3 Revision of Model

Linearly revise the model $S_0 = G_1^0, G_2^0, G_3^0, ...G_m^0$ based on the new training review document set using the technique of fixing the model and get new classification model. C-kNN constructs the classification model by utilizing single pass clustering algorithm; it alters the learning process of kNN. The conditioned clustering gains less number of cluster compared to training samples. Consequently with the help of kNN in review classification, the resemblance calculation is significantly reduced. The huge growth of reviews in Internet needs to be processed successfully to utilize the conveyed values. There is no end to process the reviews. Hence, for new documents, it is not possible to rebuild the model again, where more time is spent. Thus the proposed C-kNN processes that review document using clustered model in which the model is built linearly. The time taken by C-kNN is notably less compared to kNN, which takes all labeled data set to test the given reviews. Comparative analyses of kNN and C-kNN are given in the next section.

5 INVESTIGATIVE ASSESMENT

Proposed method is investigated on various data group sizes, to enhance the measurement of many factors in the

system. Particularly different group of assessments are done. In the first investigation, we assess the functioning of the proposed FBAI algorithm with ordinary simple algorithm BM25 and TF-IDF. Then the performance proposed C-kNN is analyzed. The outcome is given in the following sections.

5.1 Assessment of Facet Extraction Procedure

This section compares three implemented algorithms, to extract facets from opinion. The proposed procedure FBAI, and the popular TF-IDF and BM25 are compared. The analysis is done by comparing the precision of first N outcomes provided by the algorithms. Here the precision (P) is
P=no_of_relevant_facets/Total_number_of_reviews(N)
The comparison of significant five facets for a mobile phone review between TF-IDF and FBAI algorithms are given below:
General Words: 'battery', 'screen', 'keypad', 'quality'

Words in FBAI: 'internet', 'signal', 'durability', 'compact', 'apps'

Words in TF-IDF: 'user friendly', 'airtel', 'nokia', 'reminder', 'options'

Now, from the above example it is obvious that the words form FBAI are better matched as potential facets of a mobile phone. But TF-IDF algorithm has the word 'nokia', which is a brand name and 'airtel', a service provider, which cannot be taken for review. Thus the manual rating for TF-IDF is 0.8 and for FBAI is 0.93.
Fig 3-6 depict the precision comparison for FBAI, TF-IDF and BM25. The data set are for the products dishwasher, Mobile phone, LCD small screen and LCD large screen. The comparative analyses are discussed in following cases.

Case 1: Fig 3 shows the precision values for range of facet counts. In this chart proposed FBAI is compared with TF-IDF and BM25. The data set is only 30 reviews of dishwasher in which TF-Precision of FBAI and BM25 are similar to TF-IDF except during the selection of less facets. From 30 reviews, during the selection 15 facets to 50 facets, precision range is 0.4 to 0.5.

Case 2: Fig 4 shows the precision values for selection of facets from mobile data set. Since the data set has 90 mobile phone reviews, FBAI performs well (0.99) which varies from previous case. But as the number of facet extraction is increasing, precision level is decreasing for FBAI. Though precision value is less for BM25 (0.8) and TF-IDF (0.7), their precision level is stable Compared to FBAI even when facet count is increasing.

Case 3: Fig 5 shows the precision values for selection of facets from LCD small screen television data set. Number of reviews is increased with just 10 reviews (from 90 to 100) compared to the previous case. But FBAI precision is decreased from 0.99 to 0.8. From this variation, it is obvious that the performance of algorithm is also depends on data set. Since the prospective facets are different for

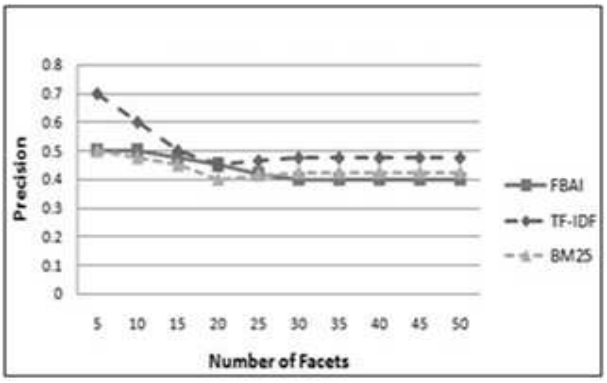

Fig. 3: Precision comparison for 30 Reviews(Dishwasher)

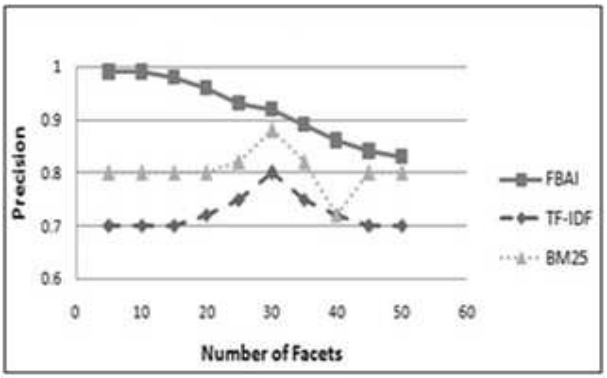

Fig. 4: Precision comparison for 90 Reviews(Mobile Phones)

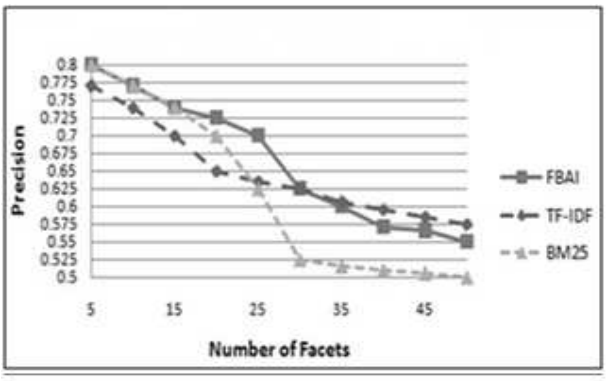

Fig. 5: Precision comparison for 100 Reviews(LCD small screen television)

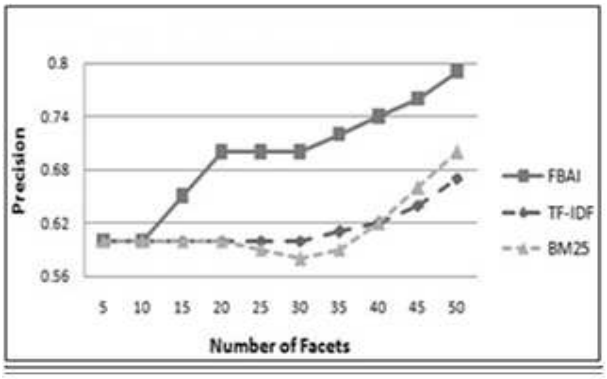

Fig. 6: Precision comparison for 250 Reviews(LCD large screen television)

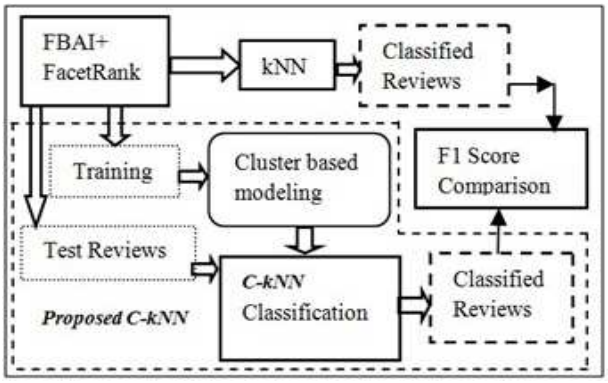

Fig. 7: Cluster based C-kNN and kNN: classification process

television and mobile phone, this variation occurs. TF-IDF performance is increased from 0.7 to 0.775, but BM25 remains same (0.8). But for this data set precision of TF-IDF and BM25 is not stable, as they are in the previous case. The precision of all three algorithms is decreasing as the number of facet selection is increasing.

Case 4: Fig 6 shows the precision values for selection of facets from LCD small screen television data set. This testing takes more than double times dataset compared to previous case. But the review area is same i.e. LCD television (change in attribute i.e. size only). Initially for 5 to 10 facet selection all algorithms perform with less precision of 0.6. TF-IDF and BM are stable up to 40 facet selection. Then up to 50 facets, the precision is increasing and final precision is 0.7. But the proposed FBAI precision increased drastically for 10 to 20 facets (0.6 to 0.7), stable during 20 to 30 facet selection (0.7) and again drastically increased for 30 to 50 facets (0.7 to 0.78)

Thus this analysis concludes FBAI performs well for large data set and it is not stable which depends on the review area. FBAI algorithm ranks each noun based on the number of adjectives, inverter and neutral word, better than BM25 and TF-IDF algorithms.
Though FBAI algorithm performs better than the other algorithms on all reviews sets, TF-IDF performs better than FBAI in few cases. For example the Dishwasher review comparison gets more accuracy in TF-IDF than in FBAI. This is because the smaller reviews are not benefitted by our algorithm. The significant words which are taken for review influence the accuracy, which is clearly depicted in fig 4 and 5. Note the review count is more or less same.

5.2 Performance Analysis of kNN and C-kNN

This topic analyses the classification performance of kNN and C-kNN by feeding the dataset which is processed through FBAI+FacetRank. Fig 7 explains process flow of kNN and C-kNN.

The proposed C-kNN uses single pass, linear cluster based model, whereas kNN takes whole labeled set each and every time to test the review. So every time the resemblance checking of labeled set with test review is computed, which means kNN takes more time.
Data set: One of the popularly known researchers, Hu and Liu provide data set of electronic products (http://www.cs.uic.edu/ liub) for opinion mining. This corpus has 13 categories with 7842 training documents and for testing 7861 new data set from Amazon is taken with the ratio of 1:1. Table 3 shows the details of review documents and F1-score of kNN and C-kNN.

Table 3: Specification of electronics goods data set along classification results

Category	Training /labeled set	Testing set	C-kNN	k=45	kNN; k=10
			Cluster vectors	F1-Score	F1-Score
C1	384	386	181	0.6775	0.6436
C2	554	555	263	0.7645	0.6528
C3	1011	1013	196	0.9555	0.9376
C4	577	578	178	0.6534	0.6759
C5	531	533	213	0.8403	0.8137
C6	312	313	121	0.9167	0.8933
C7	300	302	134	0.7832	0.7611
C8	229	231	125	0.9364	0.9248
C9	546	548	136	0.8741	0.8506
C10	346	347	203	0.9798	0.9701
C11	1716	1717	304	0.6238	0.6046
C12	597	598	209	0.7406	0.7349
C13	739	740	193	0.6875	0.6692
Total	7842	7861	2456	0.8026	0.7794

For different k values, F1-score of kNN and C-kNN are tested (Fig 8) on the mobile phone data set. C-kNN performs better all most in all points. When k takes value 1, kNN performs better, since C-kNN is affected more by

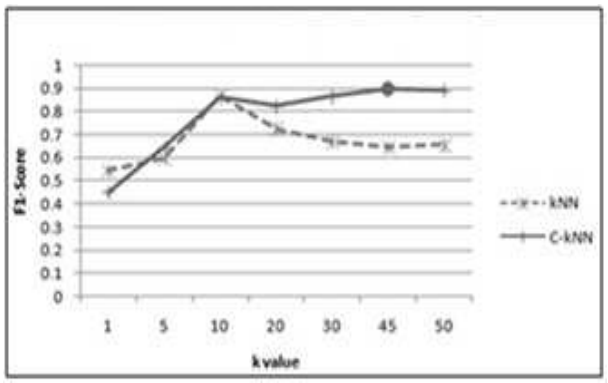

Fig. 8: F1-Score vs k-Value on Mobile phones

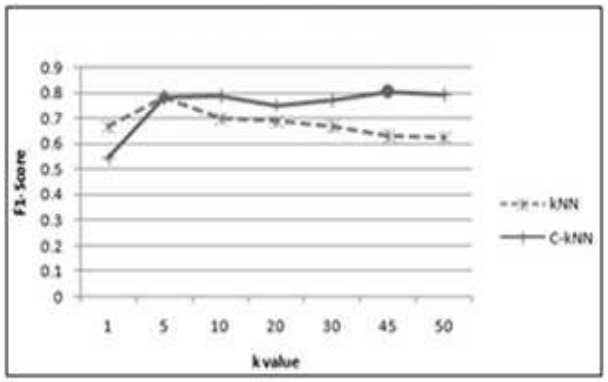

Fig. 9: F1-Score vs k-Value on Electronic Goods

the noisy reviews than kNN. When kNN and C-kNN take k value as 10, both gain same score and kNN gains maximum F1-score. Hence optimum k value is 10 for kNN (86.99%) and 45 for C-kNN (89.71%), with the difference of 2.72%.

For electronic goods reviews, kNN and C-kNN performance is charted in fig 9. Excluding the k value 1, C-kNN performs better compared to kNN. KNN gets maximum F1-score for k=1 and C-kNN gets 80.25% of F1-score for k=45, in average 4.61% higher than kNN.

From table 4 it is clear that the aggregated time cost of C-kNN is 67% of kNN for electronic goods document classification. For mobile data set, the process time exceeds kNN, since test document size is very small compared to training reviews. C-kNN is given training and test documents in the ratio of 5:1. So C-kNN takes more time to construct model compared to classification time. But for real time requirements the model can be constructed off line to enhance classification accuracy. The average time of classification by C-kNN is one and half time lesser than kNN.

Table 4: The time efficiency of C-kNN and kNN

Corpus	Mobile reviews			Electronic Goods		
Time (min)	Model	Classification	Total	Model	Classification	Total
C-kNN	2.13	1.02	3.15	51.33	124.68	176.01
kNN	0	2.01	2.01	0	262.88	272.3

Thus the evaluation results shows the precision score analysis of proposed ranking methods. From C-kNN results, it is clear that the linear approach of modeling is valuable and reasonable and helpful for real time applications.

6 Conclusion

On considering the importance of information in quality reviews, a new facet selection algorithm (FBAI) and FacetRank algorithm are developed based on significant facet rather than the whole product. Comparative analysis on different size of reviews on different product shows clearly that FBAI algorithm works with high level of precision compared to TF-IDF and BM25. Though kNN is effective for text classification, it is not an efficient one in terms of distance computation. This article proposes C-kNN, an effective and efficient method. Clustering is a good method for multifaceted document to find the spread of training reviews. This single pass method catches the relevancy of category and its subsets through confined clause. Hence clean subcategories are formed and reflect the spread of multifaceted documents than actual test review samples. C-kNN classifies test reviews by means of kNN with respect to clusters, in alternate to actual review samples. This enhancing factor of kNN in C-kNN (in linear time) is reflected in the performance measurements. For real time issues such as document tracking, spam filtration, and other text classifications can use this linearly revised mode, since C-kNN is ascendable and applicable. Thus the proposed work benefits to the user of opinions by offering facet wise ranking, individual opinion polarity (FacetRank) and fast, efficient overall polarity classification of online reviews.

References

[1] L.S.Chena, C.H.Liub, H.J.Chiua,A neural network based approach for sentiment classification in the Blogosphere, Journal of Informetrics, **5**, 313-322 (2011).

[2] R.Bapna, P.Goes, R.Gopal, J.R.Marsden, Moving from data-constrained to dataenabled research: experiences and challenges in collecting, validating, and analyzing large-scale e-commerce data, Statistical Science, **21** (2),116-130. (2006).

[3] J.R.Marsden, The internet and DSS ? massive, real-time data availability is changing the DSS landscape, Information Systems and e-Business Management, **6** (2),193-203 (2008).

[4] Kaiquan Xu , Stephen Shaoyi Liao, Jiexun Li, Yuxia Song. Mining comparative opinions from customer reviews for

Competitive Intelligence Decision Support Systems, **50**,743-754 (2011).
[5] M.Chau, J.Xu, Mining communities and their relationships in blogs: a study of online hate groups, International Journal of Human Computer Studies, **65** (1), 57-70 (2007).
[6] B.Liu, Web Data Mining?Exploring Hyperlinks, Contents and Usage Data, Springer, Verlag 2nd edition (2006).
[7] T.S.Raghu, H.Chen, Cyber infrastructure for homeland security: advances in information sharing, data mining, and collaboration systems, Decision Support Systems, **43** (4),1321-1323 (2007).
[8] J.Barranquero, P.Gonza lez, J.D?ez, J.Jose del Coz, On the study of nearest neighbor algorithms for prevalence estimation in binary problems, Pattern Recognition, **46**,472-482 (2013).
[9] [9] Peter Harrington, Machine Learning in Action, Manning Publications, Shelter Island, ISBN 9781617290183 (2012).
[10] TK.Fan, CH.Chang, Blogger-Centric Contextual Advertising, Expert Systems with Applications, **38**,1777-1788 (2011).
[11] M.RushdiSaleh, M.T.Martn-Valdivia, A.Montejo-Rez, L.A.Urea-Lpez, Experiments with SVM to classify opinions in different domains, Expert Systems with Applications, **38**,14799-14804 (2011).
[12] M.Eirinaki, S.Pisal, J.Singh, Feature-based opinion mining and ranking. Journal of Computer and System Sciences, **78**,1175-1184 (2012).
[13] S.Garcia Esparza,MP.O'Mahony, B.Smyth, Mining the real-time web: A novel approach to product recommendation, Knowledge-Based Systems.**29**, 3-11 (2012).
[14] S.Yang, Y.Ko, Finding relevant features for Korean comparative sentence extraction, Pattern Recognition Letters. **32**, 293-296 (2011).
[15] Y.Gong and X.Liu, Generic text summarization using relevance measure and latent semantic analysis, In: Proceedings of SIGIR-2001, 19-25 (2001).
[16] Z.Zhang, Q.Ye, Z.Zhang, Y.Li, Sentiment classification of Internet restaurant reviews written in Cantonese, Expert Systems with Applications, **38**, 7674-7682 (2011).
[17] S.Tan, Y.Wang, G.Wu, Adapting centroid classifier for document categorization, Expert Systems with Applications, **38**, 10264-10273 (2011).
[18] T.Wilson, J.Wiebe, P.Hoffmann, Recognizing Contextual Polarity: An Exploration of Features for Phrase-Level Sentiment Analysis, Computational Linguistics, **35**,(3), 399-433 (2008).
[19] L.Larkey, W.Croft, Combining classifiers in text categorization, in: Proceeding of ACM SIGIR Conference, 289-297 (1996).
[20] Y.S.Dong, K.S.Han, A comparison of several ensemble methods for text categorization. in: The 2004 IEEE International Conference on Services Computing (SCC), 419-422 (2004).
[21] S.Li, C.Zong, X.Wang, Sentiment classification through combining classifiers with multiple feature sets. In: Proceedings of the IEEE International Conference on Natural Language Processing and Knowledge Engineering (NLP-KE 07), 135-140 (2007).
[22] K.Dave, S.Lawrence, D.M.Pennock, Mining the peanut gallery: opinion extraction and semantic classification of product reviews. In: Proceedings of the International World Wide Web Conference (WWW), 519-528, (2003).
[23] V.Hatzivassiloglou, J.Wiebe, Effects of adjective orientation and gradability on sentence subjectivity, in: Proceedings of the International Conference on Computational Linguistics (COLING), pages 299-305 (2000).
[24] Subrahmanian.V, Reforgiato.D. AVA: adjective-verb-adverb combinations for sentiment analysis. IEEE Intelligent Systems, **23**, 43-50, (2008).
[25] N.Prabha, S.Vignesh, B.Santhi, G.R.Brindha, Polarity Categorization with Fine Tuned Pipeline Process of Online Reviews, International Journal of Engineering and Technology (IJET), 5:2221-2226, (2012).
[26] G.Wang, J.Sun, J.Ma, K.Xu, J.Gu, Sentiment classification: The contribution of ensemble learning, Decision Support Systems, **57**,77-93 (2014).
[27] G.Pang, S.Jiang, A generalized cluster centroid based classifier for text categorization. Information Processing and Management, **49**, 576-586 (2013).
[28] S.Jiang, G.Pang, M.Wu, L.Kuang, An improved K-nearest-neighbor algorithm for text categorization, Expert Systems with Applications, **39**,1503-1509 (2012).
[29] Z.Elberrichi, K.Abidi, Arabic text categorization: a comparative study of different representation modes. Int. Arab J. Inf. Technol., **2**, (4),210-215 (2012).
[30] R.Duwairi, R.Al-Zubaidi, A Hierarchial kNN classifier for textual data, Int. Arab J. Inf. Technol, **8**, (3), 251-159 (2009).
[31] N.Slonim, N.Friedman, N.Tishby, Multivariate Information Bottleneck, Neural Computation **18**, 1739-1789 (2006).
[32] L.Chen,L.Qi,F.Wang, Comparison of feature-level learning methods for mining online consumer reviews, Expert Systems with Applications, textbf39, 9588-9601 (2012).
[33] F.L.Cruz, J.A.Troyano, F.Enrquez, F.J.Ortega, C.G.Vallejo, 'Long autonomy or long delay' The importance of domain in opinion mining, Expert Systems with Applications, **40**,3174-3184 (2013).

G. R. Brindha obtained the Bachelor Degree in Computer Science, Masters degree in Computer applications and MPhil degree in Computer Science from Bharathidasan University,India. Now, she is currently working as Assistant Professor in School of Computing at SASTRA University and moreover she is pursuing the Ph.D. degree as Part-Time at the same university. Her area of interest includes Data Mining, Machine learning and Artificial intelligence. She has published more than 10 research papers in the field of her interest.

S. Prakash obtained the Bachelor Degree in Computer Science and Masters degree in Computer applications from Bharathidasan University, India.Now, he is a research scholar in AVVM SriPushpam College, Poondi. His area of interest includes Machine learning, Artificial intelligence and Database management. He has published more than 4 research papers in the field of his interest.

B. Santhi obtained the Bachelor Degree in Mathematics, Masters degree in mathematics and Masters degree in Computer applications from Bharathidasan University,India. Further she received Masters of technology in Computer Science and Doctorate degree in Computer Science from SASTRA University,India. Now, She is working as Professor in School of Computing, SASTRA University. Her area of interest includes Image Processing, Machine learning, Data mining and Sensor Networks. She has published around 70 research papers in the field of her interest.

P. Swaminathan holds Honours degree in Electronics and Communication Engineering and Doctorate degree in Electronics Engineering. He is Fellow of Institution of Engineers, India. Currently, he is working as Dean in School of Computing at SASTRA University. His research interest includes Embedded Systems, Software Engineering and Expert Systems.