A novel minimum redundancy-maximum relevance approach to feature selection for Korean sentiment analysis

Ju Yeon Heo[1] and Kun Chang Lee[2*]
[1]Department of Global Business Administration
Sungkyunkwan University
Seoul 03063, South Korea
heojuyeon12@gmail.com

[2]Professor, SKK Business School
Professor, Department of Health Sciences & Technology
Director, CSRI (Creativity Science Research Institute)
Samsung Advanced Institute for Health Sciences & Technology(SAIHST)
Sungkyunkwan University
Seoul 03063, South Korea
kunchanglee@gmail.com
*Corresponding author

Abstract

Feature selection plays an important role in sentiment analysis to perform well. In this paper, a supervised feature selection approach is presented, which is considering not only feature-feature and feature-target variable relationship but also trying to include only mutually exclusive and completely exhaustive features for efficiency. Applying a clustering, the two methods searches for a best set of features based on hybrid approach which is combination between filter and wrapper methods. Eventually, several experiments using 4 Korean datasets are presented to show the effectiveness of the features selected with two methods from the point of view of the number of features needed to reach certain classification accuracy.

Keywords:

1. Introduction

As social network services become more and more popular, we can obtain the information of social tendencies and preferences [1,2]. Consumer's evaluations of products or service are getting important because they are not only used for products or brand development, but also they impact other potential customers. The aim of sentiment analysis is to classify opinions and feelings from enormous reviews into two-class or multi-class polarity by creating accurate machine learning models.

In text datasets, both useful and useless high dimensional features which are contained. Because fruitless features are un-evenly distributed, irrelevant and noisy, enormous features and dimensions decrease the efficiency and performance [3]. Thus, the performance of sentiment analysis in text mining highly depends on feature selection. It is important in both supervised and unsupervised sentiment analysis by reducing dimensions and making models focus on most informative features [4]. Also, it can beat the problem of overfitting and can lower the cost of time and computational resources.

Because of grammatical uniqueness by language, it is often difficult to stipulate them with a consistent rule [23]. However, the studies on Korean sentiment analysis have the difficulties and there are still fewer studies on Korean sentiment analysis than English sentiment analysis, especially on the models that considering both official reviews and microblogging. Thus, this paper will be meaningful in studying feature selection methods in Korean data including both reviews and microblogging.

When it comes to feature selection or FS, two types of framework exist- (1) search-based framework and (2) correlation-based framework. For the search-based feature selection framework, 3 approaches are suggested; filtering, wrapper and embedded approaches. The hybrid method that are the mixture of filter and wrapper methods is proposed for complimenting the cons of both filter and wrapper methods. It can decrease cost compared to wrapper method and get rid of the limitation of the dependence of the feature subset selection in filter method [5]. In this sense, this paper takes hybrid approaches. The second framework, correlation-based framework, considers both feature–feature correlation and feature–class correlation. Generally, the feature–feature correlation is called as feature redundancy, while the correlation between features and class is viewed as feature relevance. The chi-square method that are used most for weighting features focus on how to select high feature relevance. If the target variable, for example class label, is independent of the feature variable, we can discard that feature variable. If they are dependent, the feature variable is important. However, they are not considering feature redundancy and they include all features that are relevant to target variables even if they are overlapped. If the features are sharing similar characteristics, we don't have to include all these features for feature vectors. For example, 'Love story' and 'Romance' in movie domain have really similar meanings and they have high possibilities to occur together in a same document. Even if both features have high feature relevance and highly dependent to the target variables, selecting both features can be inefficient. For considering both correlation, mRMR(minimum redundancy maximum relevance) is studied actively [13,14]. However, there's only a few works on considering both redundancy and relevance in feature selection with hybrid approaches in sentiment analysis. Also, for the best of our knowledge, few articles considered the meaning of features that have high redundancy. Especially these kind of researches about Korean are scares in sentiment analysis study fields. Studying in the field of not only official reviews but also microblogging in Korean is valuable when there are not many researches. Thus, this paper is going to compare the performance of three hybrid approaches. First, our two methods, ToCchi and ToCknn, second, chi-square methods hybrid approaches and last, other methods presented in 2017(OIFV) [5].

This paper takes a different look at mutually exclusive and completely exhaustive feature selection based on document prospects. We assume that features that appear similarly related to the documents share the similar characteristics. The things that features can share can be the similar topics, similar meaning, and similar emotions that can be used almost together <*Figure 1*>. For example, 'ㅜ' and 'ㅠㅠ' are the emoticons used in Korean that mean sad and they occurred almost in a same sentence or document. Thus, selecting the features equally balanced from the whole datasets not only can deal with the diverse and smaller features of the whole datasets harmoniously. But also it can be efficient and reduce the feature dimension without sacrifice representing the data.

Table 1. Korean language characteristics from a sentiment analysis

Similar topics		Similar meaning		Appearance almost together	
'캐릭터' *'character'* '배우' *'actor'* '연기' *'act'* '액션' *'action'* '감정' *'emotion'*	'우체국' *'post office'* '택배' *'delivery'* '배송' *'deliver'* '도착' *'arrive'* '상태' *'condition'*	'로맨스' *'Romance'* '러브스토리' *'Love story'* '사랑' *'Love'* '애정선' *'Affection'*	*(sad emoticon)* 'ㅠㅠ' 'ㅜㅜ' 'ㅠ' 'ㅜ' *(happy emoticon)* 'ㅎㅎ' 'ㅋㅋ'	'명예회손' *'defamation'* '침해' *'invasion'* '타인' *'others(formal)'* '욕설' *'swear word'*	*(exclamation for something)* '완전' '대박' '미치다' '진짜' '정말'

Also, our methods can play role as the goal of TF-IDF method. This is important because the number of reviews about the certain topics are the dependent to the training data, so choosing the features based on term frequency can lead to overfitting problem. The data is only collected for certain periods and it is only the small part of the whole reviews. Thus, when we depend on the frequency of the features, it can lead to wrong feature selection. However, by using our proposed methods, in the movie reviews, we can separate reviews of actors and the one of directions and then consider both topics equally balanced even if the number of reviews of actors are much larger than the one of directions. In other word, our methods can work as the role of TF-IDF unexpectedly.

We used both chi-square and KNN for ordering the features within the cluster. Because the chi-square method is focusing on feature relation, chi-square scoring after clustering the features can consider feature relevance and feature redundancy at once. On the other hand, the clustering - KNN method scores the features that are in core of the clusters high. Even if it can reflect high feature relevance and MECE, they can't represent the whole cluster because they put center features before other features. Also, it doesn't take account of feature redundancy. Thus, we predict that chi-square after clustering method will show better performance than KNN after clustering.

As a result, we expect considering selecting features balanced related to the features can increase the performance by reducing the number of unnecessary features. In other words, we can reach high accuracy with only small number of features by mutually exclusive and completely exhaustive feature selection. This paper proposed two method with hybrid feature selection approaches; Topic clustering with CHI(ToCchi) and Topic clustering with kNN(ToCknn) model. Comparative experiments were done on 4 datasets with different themes written in Korean.

In this sense, we propose two research questions as follows:

RQ1 : Is it empirically supported that the number of features necessary for the ToCchi and ToCknn to reach a certain level of accuracy is lesser than the base hybrid method and OIFV method ?

RQ2 : When using ToCchi and ToCknn, which classifier among LR and RF shows robust performance when compared with the base hybrid method and OIFV method ?

The rest of paper is arranged accordingly. Section 2 addresses previous studies related to this paper. Section 3 explains proposed methods and experiments. In Section 4, proposed methodologies for feature selection are

described with algorithms. The comparative experiments and evaluation is showed in Section 5. Lastly, conclusion and future works are given in Section 6.

2. Previous Studies

Sentiment analysis is studied well from early 2000 in NLP [15]. Numerous approaches have been reported to classify sentiment from languages, both in supervised and unsupervised method [16]. The main process of sentiment analysis can be divided into three steps. The first step includes representation of the data and second one involves extraction and selection of the features from large sets. The last is classifying each sample into binary or multi classes with single-classifiers such as NB, LR and ensemble-classifiers like RF, AB [6].

In this process, feature selection is the key steps because high-dimensional feature space is a significant challenge in text mining. It can reduce lots of cost occurred in computational time and power and increase efficiency of text analysis [7]. Also, it can improve the performance by eliminating unnecessary text features that increases noisy [9]. Unlike general structured data, texts in different languages have lexical and grammatical uniqueness by language. And as the forms of expression are varied and complex, it is often difficult to stipulate them with a consistent rule [23]. The studies on Korean sentiment analysis have the difficulties. The use of artificially modified Korean makes lexical analysis difficult. The combination of consonants and vowels made up the Korean language. However, the texts in SNS often distorted, such as using consonants or vowels only and not the combination for emoticons. In that case, the features will be exploding and it will be hard to tell the emoticons and frequently used typos. There are still fewer studies on Korean sentiment analysis than English sentiment analysis, especially on the models that considering both official reviews and microblogging. The research of emotion analysis in Korean microblog texts [22] applied the machine learning model based on Korean documents and classified human sentiments into seven emotions.

Feature selection techniques are categorized into filter, wrapper, and embedded methods [8]. Filter methods are for statistical scoring of text feature excluding the consideration of performance of learning mechanisms. Wrapper methods are trying to find the best subset of features by considering the interaction with learning algorithms. In the embedded approach, the feature selection is done in the process of training algorithms and the best subset of features are found by a classifier used. The hybrid method that are the mixture of filter and wrapper methods are also included in the selection categories [10]. It is the complement of the filter and wrapper that reduces computational cost compared to wrapper method and overcomes the limitation of the dependence of the feature subset selection in filter method. Yousefpour, Ibrahim, 2017 investigated the hybrid methods of feature selection for text classification. They scored features with 5 filter methods and tested the performance of each subsets from the vector generated by ordinal-based and frequency-based integration (OIFV) [5]. Wang, Suge, et al also researched the hybrid feature selecting method based on category distinguishing ability of words and information gain [11]. They concluded that hybrid methods is superior to the one with directly using information gain. Le Nguyen Hoai Nam et al proposed a hybrid filter feature selection, called FCFS and related filter feature selection methods as CMFS, OCFS, CIIC, IG, CHI with two datasets about news and medicine [21].

These papers focus on the hybrid method by integrating the process used in filter and wrapper methods. In the first step, features within the topics are ordered to create feature vectors based on the score of relation between the feature and the each topic. The next is making subset of the vector and evaluating the performance of each subset and finding the best one which is the method of wrapper approach.

The correlation-based framework considers both feature–feature correlation and feature–class correlation. First in the feature-feature correlation study, Vinh, Nguyen X., and James Bailey, a method for supervised feature selection based on clustering the features into groups is proposed, using a conditional mutual information based distance measure. They find that there is a reasonable condition, namely when all features are independent given the class variable (as assumed by the popular naive Bayes classifier) [12]. Second, in the feature-class correlation, the method by Sotoca, José Martínez, and Filiberto Pla builds a dissimilarity space using information theoretic measures, in particular conditional mutual information between features with respect to a relevant variable that represents the class labels [20].

For considering both correlation, mRMR which means minimum redundancy maximum relevance is studying actively. In the work of Assi, E. Bou, et al, based on a Support Vector Machine and an Adaptive Neuro Fuzzy inference system, data reduction was performed by mRMR features selection approach for electrodes selection and a genetic algorithm. The selected subset of features performed equally and sometimes even better than the whole features set [13]. In the article of Niphat Claypo and Saichon Jaiyen, the mRMR feature selection is used to select the features of data in order to reduce the number of features in the data set. Consequently, the computational times of learning algorithms are reduced for neural networks with mRMR approaches based on Thai restaurant reviews [14].

3. Proposed Method and Experiments

When generating the feature vectors in the first step in hybrid feature selection approaches, several methods are used and studied for its performance. According to Yousefpour et al, feature selection methods such as the IG and CHI methods were found to achieve better accuracy than other methods in filter approaches. Compare to the CHI method, kNN is less studied in feature selection process [5]. kNN score the features according to how they share the similar patterns or characteristics. If we set one feature which is strongly related to the topic as a center and use the kNN method within the specific topic group of features, we can pick features that have strong patterns within themselves and they can represent the topic group well. In other word, kNN method also can be an indicator that can show the relationship between features and the topic as CHI can be. In this paper, we compare the CHI and kNN method in terms of indicator that can score the features with strong association to the topic before using wrapper method in hybrid approach. This paper proposes two methods in scoring the relevance which is the ToCchi and ToCknn.

3.1 Datasets and feature preprocessing

The detection of features from the datasets should be done before selecting feature for classifier models. Feature representation are done with several methods such as bag-of-words [16], lexicon etc. The unigram BOW is adopted for feature extraction.

Table 2. Datasets

Datasets	subject	# of positive	# of negative	total	# of features (After count vectorizer)	Language	Gathering period
Google App Store	Various app review	1688	1845	3533	3374	Korean	2017.7~8
11Street Shopping	Various products	9551	10449	20000	6378	Korean	2017.5~6
Twitter	Idol (Wanna One)	6792	7751	14543	6382	Korean	2017.7~8
Watcha-movie	Movie	4215	3481	7696	6352	Korean	2011

Google App Store

We collected the data from Google App store by crawling the reviews of various apps. The Google App is the biggest market of the apps on Android OS. For each download page, there are reviews and the ratings of the apps. The problems and the benefits of the apps are pointed by numerous active users.

Twitter

We collected the data from Twitter about 'Wanna One' which is popular idol singer group in Korea. It is picked out as a search word because it is a hot topic in July, 2017 so that there are various sentiment reactions on Twitter. We gather the results of the topic except for sentences repeated meaninglessly, advertisement and sentences which are too short. For the rating of the text data, we employed the students and ask them for making polarity consisted of positive and negative.

11 street shopping

In '11 street shopping[1]', consumers can compare the price of products from various categories and read the reviews and ratings from 1 to 5. We save the reviews of several categories with active reviews from May to June, 2017 and divided into positive and negative based on ratings. Due to the characteristic of the culture in this shopping site, customers almost rate the products from 3 to 5. Thus, we give positive polarity to under 4 and negative polarity to the upper 4 after the relationship between reviews and ratings are checked by humans.

Watcha movie

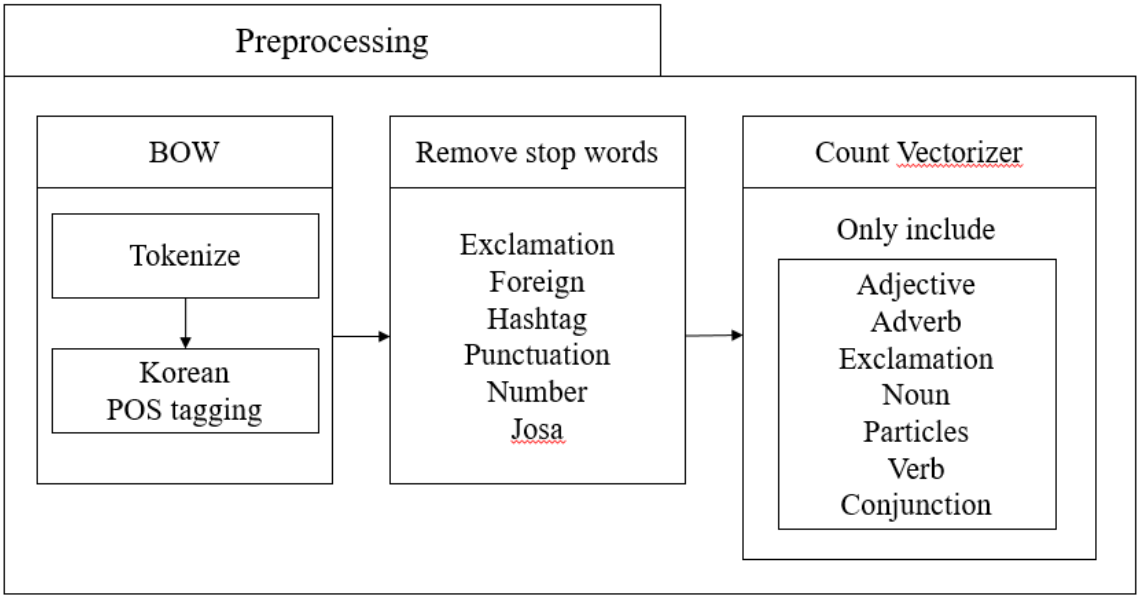
'Whatcha[2]' is the movie recommend site in Korea. After accumulating the data of preference movie of users, they advise some movies to watch. It has lots of movie fan users and active review system. Diverse users write down their feelings and evaluations and they write down opinion, impression and deep thoughts of movies and its topics. We choose movies among movies passed over million viewers mark and place sentences with score 5 to positive and 0.5, 1 to negative.

[1] www.11st.co.kr/

[2] https://watcha.net/

Fig 2 summarizes the procedures taken in this study for the sake of feature preprocessing,.

Fig. 2. Procedures for feature preprocessing



To annotate sentence with POS, this article used the 'KoNLPy'. It splits the word into morphemes. After being excluded stop words which is above, features in datasets are selected including POS mentioned above by 'Count Vectorizer' in scikit-learn open source[3]. The particles are contained because they are used as emoticons widely in Korean. For example, 'ㅋㅋ', 'ㅎㅎ' show laughing and 'ㅠㅠ' shows crying.

3.2 Proposed mRMR method

Before discussing the proposed mRMR method for feature selection, let us consider basic definitions of Chi-square, kNN (k-nearest neighbor), and K-means that are necessary for our proposed method.

The Chi-square calculates the degree of the relationship between the feature and the category.

$$CHI(f, c_i) = \frac{N \times (AB - CD)^2}{(A+C) \times (A+D) \times (B+C) \times (B+D)}$$

$$i = 1, 2, 3, ..., M$$

Where f: feature, c: category, N: number of all documents, M: the number of the categories (data in this paper use M=2 for binary polarity), A: the number of times f and c occur together, B: the number of time neither c nor f occurs, C: the number of times f occurs without c and D is the number of times c occurs without f. k-NN or k-

[3] http://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.CountVectorizer.html

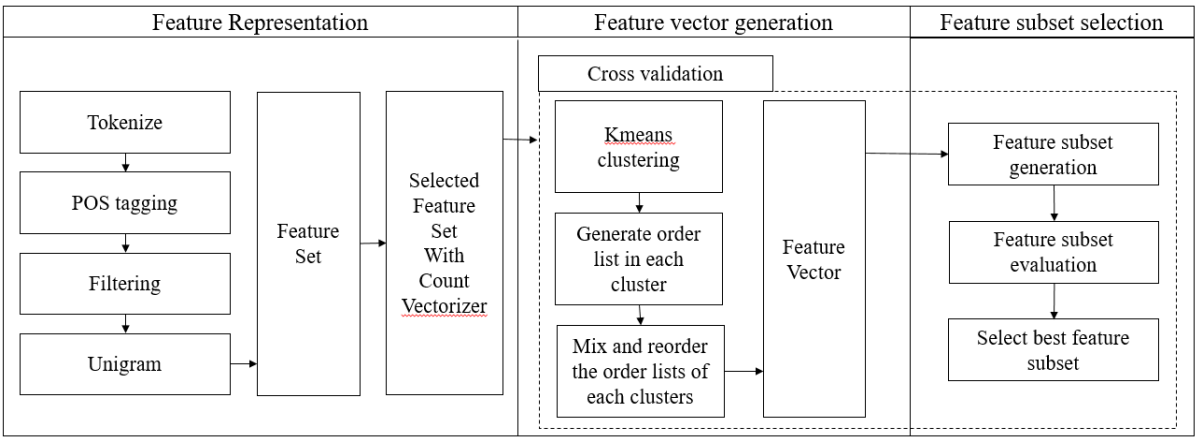
Nearest Neighbor is a widely applied text classifier due to its simplicity and efficiency. Its training-stage consists of storing all training examples as classifier, thus it has often been called as lazy learner since 'it defers the decision on how to generalize beyond the training data until each new query instance is encountered' [18]. K-means clustering [19] is a method commonly used to automatically partition a data set into k groups. After selecting k initial cluster centers, this model iteratively refines them. Each instance d_i is assigned to its closest cluster center. Each cluster center C_j is updated to be the mean of its constituent instances. The algorithm converges when there is no further change in assignment of instances to clusters.

This paper creates feature vectors based on scoring and ordinal feature subsets as a hybrid approach. This is better to overcome the limitation occurred when choosing the size of feature subsets in filter methods and reduce the computational cost compared to wrapper methods because it decreases the number of possible feature subsets. We propose two sub-mechanisms to generate feature vectors which are the core stage in the whole feature selection process. The first sub-mechanism is a topic clustering with CHI (abbreviated as ToCchi) and the second sub-mechanism is a topic clustering with kNN (abbreviated as ToCknn) for feature selection.

These methods consider different clusters of features to select the features equally balanced. They assume that selecting the features equally balanced can increase the performance by reducing the number of unnecessary features with feature redundancy, feature relevance, and MECE. Selecting features without considering the characteristics of the features is not efficient way because it can only choose features about one or two specific topics or contain similar features sharing same topic, same meaning or same emotions that can be used almost together. The data is only collected for certain periods and it is only the small part of the whole reviews. If the model depends on the number of appearance of the features in the training datasets, it can be overfitting because the frequency of the features is the characteristic of the data.

For the ToCknn, the center of the feature is selected based on the nearest centroid scoring. Then k-nearest neighbor method is used to make an ordinal list of features. For the ToCchi, the feature vector is made based on the chi-square score of the features. The unigram BOW with counter vectorized process for feature selection is used as the way of representing the raw text sets in both sub-mechanisms. Figure 3 depicts procedures included in the ToCknn and ToCchi.

Fig. 3. Process of ToCchi and ToCknn mechanisms



First step is necessary to make clusters from the documents. Each data point represents original features which are computed out of unigram BOW. By using Kmeans, features are clustered into several groups which are used as parameter of the number of clustering changes. Because it is hard to decide hyper parameter for the number of clusters, cross validation is used to find best set.

Second step is to create feature vectors within each cluster. Feature vector is represented as follows:

$$Features = [f_1, f_2, f_3, ..., f_n]$$

where f_i is a feature and N is a number of features. For the ToCchi method, features are ordered based on its chi-square scores. On the other hand, for the ToCknn method, first center feature is selected based on the Chi-square among the features. Next, for each cluster, the list of features is generated as the order of relevance with the center by using K-nearest neighbors. For ToCchi, features are ordered by the chi-square in cluster. For example, for cluster A, feature vector F_{CHI-A} is represented as follows.

$$F_{CHI-A} = [f_1, f_2, f_3, ..., f_n]$$

$$= [f_{CHI-a1}, f_{CHI-a2}, f_{CHI-a3}, ..., f_{CHI-an}]$$

For ToCknn, feature vector for cluster A $F_{KNN-A} = [f_1, f_2, f_3, ..., f_n]$ is denoted as follows:

$$F_{KNN-A} = [f_1, f_2, f_3.., f_n]$$

$$= [f_{KNN-a1}, f_{KNN-a2}, f_{KNN-a3}, ..., f_{KNN-an}]$$

Third step is to generate feature subsets. Feature vectors are integrated into FV_{final} vector as follows.

$$FV_{final} = [f_{Cluster\_a1}, f_{Cluster\_b1}, f_{Cluster\_c1}, f_{Cluster\_a2}, f_{Cluster\_b2}, f_{Cluster\_c2} \cdots f_{Cluster\_aN}, f_{Cluster\_bN}, f_{Cluster\_cN}]$$

Finally, feature subsets are made as follows.

$$ToChi, ToCknn = [x_1, x_2, ..., x_N], \quad \forall_{i,j} i < j \rightarrow rank(x_i) \geq rank(x_j)$$
$$Feature\ subsets = \{\{x_1\}, \{x_1, x_2\}, \{x_1, x_2, x_3\}, \ ... \ , \{x_1, x_2, ..., x_N\}\}$$

Where x_i is the feature and N is a total number of features. x_1 is the highest rank and x_2 is the second highest rank. After creating the feature vectors like this, feature subsets are generated and the subsets with highest accuracy are chosen as a final result. In summary, the algorithmic procedures are shown as follows, where cross validation is used to find the hyper parameter about the number of clusters made in Kmeans model.

Table 3. Algorithmic Procedures of the Proposed Method

Input : unigram or bigram of text datasets
Output : feature subset which gain the highest accuracy
Shuffle the datasets
Weight data basis on TF-IDF
For fold in numRepitition:
    For CV in CrossValidation:
        Clustering based on documents with Kmeans method when number of clusters=CV
        If ToCchi model:
            For cluster in each clusters:
                score features basis on Chi-square
                make feature vectors by ordering them according to the score
            End cluster
        If ToCknn model:
            For cluster in each clusters:
                select the feature with Nearest Centroid as a center
                (e.g. If 3 clusters, 3 centers are elected.)
                Get the feature vectors in order of degree of nearness to the center by using kNN
            End cluster
        Generate the set of feature subsets incrementally as Equation 7
        For sub in the set of feature subsets:
            Create data with only features in sub from original datasets
            Separate data into test and train set
            Classification: LR, RF
            End sub
        Save feature subset with highest accuracy among the set of feature subsets
        End CV
    Save feature subset with highest accuracy among the CV in CrossValidation
    End fold
Average the accuracy, precision, recall and f-value on all saved feature subsets.

3.3 Experiment Results

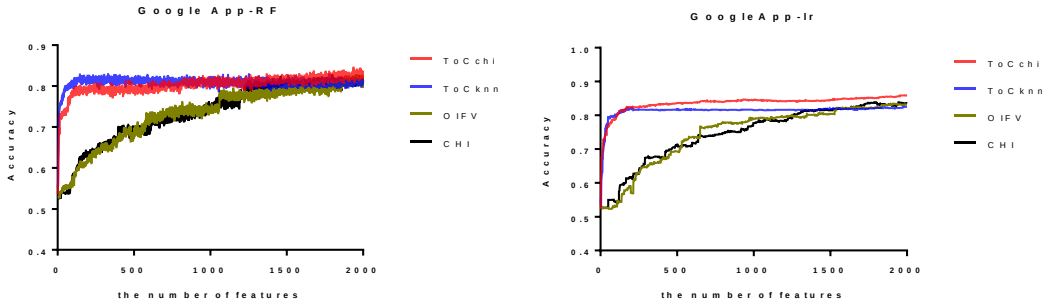
We compare the performance of the proposed ToCknn and ToCchi methods with the results of base hybrid model and Chi model. For the base model, feature vectors are generated by the order of scoring based on Chi-square. After creating the feature vectors, the same process of producing feature subsets vector is employed. All the subsets vectors are evaluated its performance. OIFV method [5] is used to compare the result of our two methods with other hybrid feature selection methods. For the sake of performance comparison, the four indices are adopted-Accuracy, Precision, Recall and F1-score. Accuracy is the ratio of all true predictions to all predicted samples. Precision is the ratio of true predicted samples against all predicted samples. The ratio of true predicted samples against all actual samples is called Recall. F1 is the average of recall and precision. In order to produce reliable results, all the computation processes were repeated twice and then results were averaged to obtain final result.

The classification classifiers adopted for our performance comparison include logistic regression (LR) and random forest (RF). LR is typically a basic classifier that has been used extensively in the fields of machine learning-based classification works. RF is famous as a typical ensemble classifier which is used as a bench marking purpose.

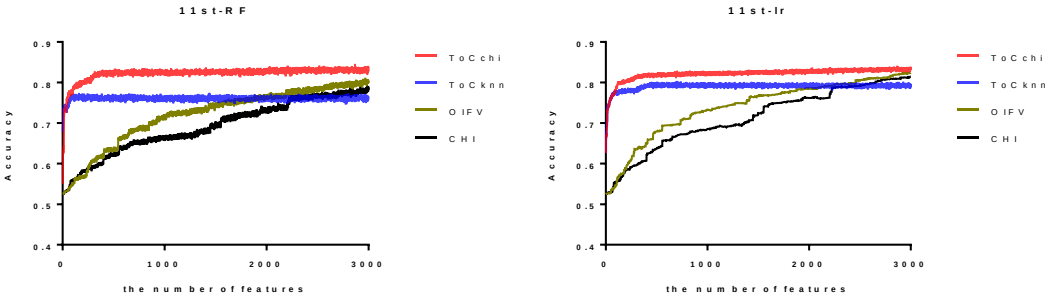
By using the four datasets as shown in Table 2 and the algorithmic procedures in Table 3, we computed experiment results. Figure 4 depicts performance trajectory computed from using each dataset. Table 4 also shows summary results from LR and RF.

Fig. 4. Performance trajectory for each dataset

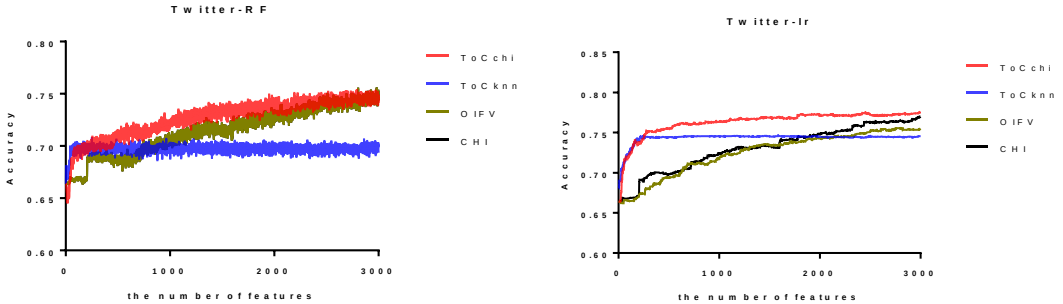
(a) Google App Store - RF(left), LR(right)



(b) 11 Street Shopping - RF(left), LR(right)



(c)Twitter - RF(left), LR(right)



(d) Watcha Movie - RF(left), LR(right)

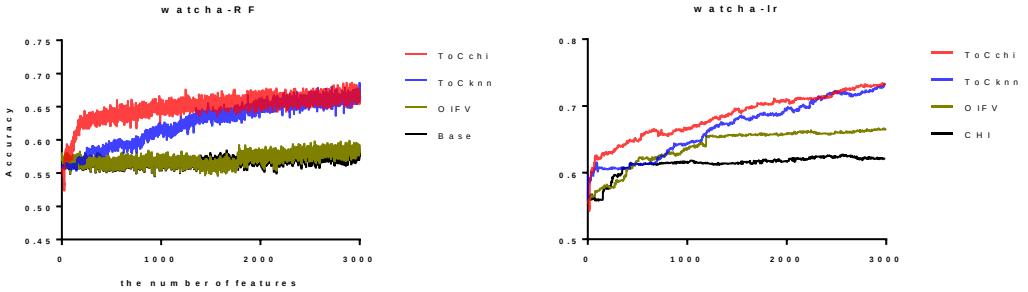


Table 4. Experiment results with LR and RF classifiers

(a) LR

Dataset	Model	# of features needed to reach a certain level of accuracy				Performance of subset with best accuracy				
		start	0.6	0.7	0.8	Accuracy	Features	Precision	Recall	F1
Google App Store	Base	0.525	170.0	478.0	1277.5	0.850	3369.0	0.850	0.857	0.851
	OIFV	0.527	220.0	527.0	1354.0	0.861	3369.0	0.861	0.861	0.861
	ToCchi	0.523	7.5	20.5	117.0	0.873	3370.5	0.877	0.871	0.873
	ToCknn	0.552	10.0	25.0	110.0	0.861	3370.0	0.867	0.858	0.860
11street Shopping	Base	0.525	329.5	1373.0	2660.0	0.849	6374.5	0.839	0.845	0.840
	OIFV	0.525	226.0	741.5	2469.5	0.824	6072.5	0.872	0.872	0.870
	ToCchi	0.626	1.0	17.5	164.5	0.848	6374.5	0.839	0.845	0.840
	ToCknn	0.627	1.0	15.0	317.0	0.848	6372.0	0.836	0.838	0.836
Twitter	Base	0.664	1.0	556.5	None	0.794	6378.0	0.794	0.794	0.794
	OIFV	0.664	1.0	224.0	None	0.794	6377.5	0.794	0.794	0.794
	ToCchi	0.664	1.0	44.5	None	0.795	6378.5	0.793	0.730	0.746
	ToCknn	0.680	1.0	32.0	None	0.766	6378.5	0.756	0.694	0.707
Watcha movie	Base	0.558	356.0	5353.5	None	0.730	6128.0	0.730	0.730	0.730
	OIFV	0.563	395.5	5210.5	None	0.748	6114.5	0.748	0.748	0.749
	ToCchi	0.556	38.0	1715.0	None	0.775	6034.5	0.775	0.775	0.775
	ToCknn	0.564	65.5	2150.5	None	0.789	6122.0	0.785	0.784	0.788

(b) RF

Dataset	Model	# of features needed to reach a certain level of accuracy				Performance of subset with best Accuracy				
		start	0.6	0.7	0.8	Accuracy	Features	Precision	Recall	F1
Google App Store	Base	0.525	140.5	620	1418.5	0.765	6380.00	0.757	0.690	0.704
	OIFV	0.528	145.5	580	1853	0.848	3370.00	0.832	0.832	0.832
	ToCchi	0.531	7.0	17	573	0.853	3332.05	0.844	0.822	0.824
	ToCknn	0.540	6.5	8.5	337	0.843	3373.05	0.832	0.822	0.824
11street Shopping	Base	0.525	139.5	621	1425	0.846	7534.00	0.857	0.855	0.856
	OIFV	0.525	283.0	901.5	3017.5	0.856	6373.00	0.843	0.843	0.843
	ToCchi	0.702	0.0	4.5	119.5	0.852	7539.05	0.860	0.858	0.858
	ToCknn	0.627	5.0	15	272	0.844	6378.00	0.836	0.831	0.832
Twitter	Base	0.664	1.0	3916.5	None	0.774	6378.05	0.766	0.686	0.700
	OIFV	0.664	1.0	3471.5	None	0.772	6377.05	0.763	0.763	0.763
	ToCchi	0.664	1.0	208	None	0.773	6379.00	0.753	0.680	0.693
	ToCknn	0.664	1.0	1147	None	0.770	6379.00	0.756	0.685	0.699
Watcha	Base	0.580	4825.5	6267.5	None	0.727	6348.05	0.690	0.690	0.690

movie	OIFV	0.558	4764.0	6238.5	None	0.716	6352.05	0.690	0.690	0.690
	ToCchi	0.558	165.5	5887	None	0.722	6348.00	0.698	0.695	0.696
	ToCknn	0.558	750.4	5108.5	None	0.722	6350.00	0.706	0.701	0.703

- Results of the ToCchi and ToCknn are highlighted only when they outperform results of both Base and OIFV.

To show how the experiment results shown in Figure 4 and Table 4 support RQ1 and RQ2, let us discuss the results here in line with each RQ.

3.4 Answers for RQ1 and RQ2

Firstly, we need to answer RQ1. As described in introduction, RQ1 is whether it is empirically supported that the proposed ToCchi and ToCknn methods require lesser number of features than the benchmarking methods and OIFV method to reach a certain level of accuracy. Experiment results from Table 4 reveal that the answer for RQ1 is definitely yes. It is quite remarkable that the number of features required for the proposed methods to reach a certain level of accuracy is extremely small compared with the benchmarking hybrid method and OIFV method. With the Google App store datasets, ToCchi reaches the accuracy level 0.7 with only 17 features, and ToCknn with only 9 features. Meanwhile, the base hybrid model requires 618 features and OIFV needs 578 features when they are experimented with RF. In the case of Google App Store dataset, it is quite clear that the proposed methods need smaller number of features to reach satisfiable accuracy level, compared with bench-marking methods. The benefit like this is very promising when considering the fact that reduced number of features have huge implications especially for practitioners who need to fight against response time to satisfy customers on a timely basis. Same results were obtained with other datasets such as 11 street shopping, Twitter, and Watcha movie. Advantage of the ToCchi and ToCknn like this can also be verified again in Fig. 4 in which performance trajectory given number of features is depicted.

Secondly, RQ2 is about which classifier among LR and RF shows robust performance when using the ToCchi and ToCknn, compared with the base hybrid method and OIFV method. ToCchi seems to be better than ToCknn due to its stable performance in several kind of datasets. This result is driven from the fact that ToCchi outperform both base and previous hybrid method in all four datasets when ToCknn just do in only three datasets. When datasets seem to have lots of noisy such as type errors and neologisms used only in small scale, ToCchi is better in performance. Because chi-square scoring after clustering the features can consider feature relevance, MECE, and feature redundancy at once, it can outperform even if the datasets have lots of noise like Twitter. On the other hand, the clustering - KNN method can reflect high feature relevance and MECE, but they can’t represent the whole cluster because they put center features before other features. Also, it doesn’t take account of feature redundancy. When the features are not in a good shape such as containing typing error and unofficial words, it seems like strong relevance between features are rather shows worse performance. As we expected, chi-square after clustering method gives better results than KNN after clustering.

3.5 Implications

Academic implications are as follows.

Firstly, the loan words, they are used mixed with the words in the other language that have the same meaning. For example, '러브' which is the loan word meaning 'love' is used mixed with '사랑' that is the Korean with the same meaning. For the researchers, it will be useful to filter these words before applying to the classifiers. In Figure1, it is shown that the two methods in this paper, ToCchi and ToCknn have functions to distinct and not include all the words with similar meanings. Even if, two methods are not selected as the main feature selection methods, they can adopt them to roughly solve these kinds of loan word problems.

Secondly, for the unofficial resources such as Twitter, feature selection methods that emphasize feature relevance seem to be better rather than feature redundancy. When we compare the performance between two feature selection methods, ToCchi show the better performance with the datasets having lots of noisy such as type errors and neologisms used only in small scale. Thus, with the unformal datasets or sources, it would be better to use methods considering the relationship between features and target variables like ToCchi.

Thirdly, for the clustering, this paper used K-means and it require 2 main hyper-parameters which are the number of clusters and the number of initialization features within each clusters. Because there are more than 1 parameters, grid search is useful for finding proper hyper-parameters. Even if it can pass the optimal point because of grid, this method is good for the researchers with limited time and computing powers.

Practical Implications are as follows.

Firstly, even if the domain of the datasets is the same, the characteristic of the data and the words used can be completely different. For example, the reviews of the movie in Twitter include lots of interjection and swear words used for expressing strong emotions. However, the movie reviews in Watcha or Naver that are the Korean movie review webpages, contain more formal and professional words. Thus, when hands on workers want to balance considering the characteristics of sources, it will be useful to apply the ToCchi and ToCknn in this paper. Because they select small number of features with high performance, users can gather features with them from each sources and apply for the mixed datasets with various sources.

Secondly, because hyper feature selection method which is the complement between filter and wrapper still need lots of time and computing power, this paper used counter-vectorizer to pick out the features roughly before applying the ToCchi and ToCknn. However, if the practical users have enough time and computing powers, it would be better to use those two methods without any processing of picking out.

Thirdly, if the agents have enough computing power and time, it would be better to try both grid search and random search for finding the hyper-parameter. Grid search will help the workers to narrow range of the hyper-parameters and random search can complement a weakness of the grid search that it can overlook the area with optimal solutions.

4. Concluding Remarks

In this paper, two methods is presented which are considering not only feature-feature and feature-target variable relationship but also trying to include only mutually exclusive and completely exhaustive features for efficiency. From the view of the previous researches, it is meaningful because there are only a few researches about feature selection considering MECE. Eventually, the results with 4 Korean datasets show the effectiveness of the features selected with two methods from the point of view of the number of features needed to reach certain classification accuracy. From RQ1, Both models get the certain level of accuracy with only few number of features compare to the base and OIFV hybrid method in three datasets with both classifiers. From RQ2, ToCchi seems to be better than ToCknn due to its stable performance even in unofficial and noisy datasets.

However, this article also has the limitations. Because of the cost of time and computing power, counter vectorizer is used to pick features out roughly first before using our two methods. Thus the full features within each datasets are about 6000 and because of that, the effectiveness of the two methods are not shown effectively. Also the classifiers used in this article are only LR and RF, which are the basic of the linear classifier and ensemble classifier. That's also because of the time and memory cost, and it would be better to show the results using other classifiers such as SVM and AB(AdaBoost). For active research in this field, progressing the experiment with known English data will be useful because it is easy to compare with other methods. Therefore, for the further research, experiments with well-known English data without any counter-vectorizer process using various classifiers should be proceeded. Also, it would be better to use various BOW such as POS and n-gram because this paper used only unigram for reducing cost of time and memory.

References

[1] Hu, Nan, Noi Sian Koh, and Srinivas K. Reddy. "Ratings lead you to the product, reviews help you clinch it? The mediating role of online review sentiments on product sales." Decision support systems 57 (2014): 42-53.

[2] Cheung, Christy MK, Bo Sophia Xiao, and Ivy LB Liu. "Do actions speak louder than voices? The signaling role of social information cues in influencing consumer purchase decisions." Decision Support Systems 65 (2014): 50-58.

[3] Zheng, Ling, Ren Diao, and Qiang Shen. "Self-adjusting harmony search-based feature selection." Soft Computing 19.6 (2015): 1567-1579.

[4] Liu, Luying, et al. "A comparative study on unsupervised feature selection methods for text clustering." Natural Language Processing and Knowledge Engineering, 2005. IEEE NLP-KE'05. Proceedings of 2005 IEEE International Conference on. IEEE, 2005.

[5] Yousefpour, Alireza, Roliana Ibrahim, and Haza Nuzly Abdel Hamed. "Ordinal-based and frequency-based integration of feature selection methods for sentiment analysis." Expert Systems with Applications 75 (2017): 80-93.

[6] Aggarwal, U., and G. Aggarwal. "Sentiment Analysis: A Survey." International Journal of Computer Sciences and Engineering 5.5 (2017): 222-225.

[7] Nassirtoussi, Arman Khadjeh, et al. "Text mining of news-headlines for FOREX market prediction: A Multi-layer Dimension Reduction Algorithm with semantics and sentiment." Expert Systems with Applications 42.1 (2015): 306-324.

[8] Saeys, Yvan, Iñaki Inza, and Pedro Larrañaga. "A review of feature selection techniques in bioinformatics." bioinformatics 23.19 (2007): 2507-2517.

[9] Dash, Manoranjan, and Huan Liu. "Feature selection for classification." Intelligent data analysis 1.1-4 (1997): 131-156.

[10] Lin, Kuan-Cheng, et al. "Feature selection based on an improved cat swarm optimization algorithm for big data classification." The Journal of Supercomputing 72.8 (2016): 3210-3221.

[11] Wang, Suge, et al. "A hybrid method of feature selection for Chinese text sentiment classification." Fuzzy Systems and Knowledge Discovery, 2007. FSKD 2007. Fourth International Conference on. Vol. 3. IEEE, 2007.

[12] Vinh, Nguyen X., and James Bailey. "Comments on supervised feature selection by clustering using conditional mutual information-based distances." Pattern Recognition 46.4 (2013): 1220-1225.

[13] Assi, E. Bou, et al. "A hybrid mRMR-genetic based selection method for the prediction of epileptic seizures." Biomedical Circuits and Systems Conference (BioCAS), 2015 IEEE. IEEE, 2015.

[14] Claypo, Niphat, and Saichon Jaiyen. "Opinion mining for Thai restaurant reviews using neural networks and mRMR feature selection." *Computer Science and Engineering Conference (ICSEC), 2014 International*. IEEE, 2014.

[15] Pang, Bo, and Lillian Lee. "Opinion mining and sentiment analysis." Foundations and Trends® in Information Retrieval 2.1–2 (2008): 1-135.

[16] Pang, Bo, Lillian Lee, and Shivakumar Vaithyanathan. "Thumbs up?: sentiment classification using machine learning techniques." Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10. Association for Computational Linguistics, 2002.

[17] Breiman, Leo. "Random forests." Machine learning 45.1 (2001): 5-32.

[18] Sebastiani, Fabrizio. "Machine learning in automated text categorization." ACM computing surveys (CSUR) 34.1 (2002): 1-47.

[19] MacQueen, James. "Some methods for classification and analysis of multivariate observations." Proceedings of the fifth Berkeley symposium on mathematical statistics and probability. Vol. 1. No. 14. 1967.

[20] Sotoca, José Martínez, and Filiberto Pla. "Supervised feature selection by clustering using conditional mutual information-based distances." Pattern Recognition 43.6 (2010): 2068-2081.

[21] Nam, Le Nguyen Hoai, and Ho Bao Quoc. "A Combined Approach for Filter Feature Selection in Document Classification." Proceedings of the 2015 IEEE 27th International Conference on Tools with Artificial Intelligence (ICTAI). IEEE Computer Society, 2015.

[22] Lee, C., Choi, D., Kim, S., Kang, J.: Classification and analysis of emotion in korean microblog texts. KIISE 40(3), 159–167 (2013)

[23] Jung, Younghee, et al. "A corpus-based approach to classifying emotions using Korean linguistic features." *Cluster Computing*20.1 (2017): 583-595.