Journal of Theoretical and Applied Information Technology
15th August 2018. Vol.96. No 15

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

THE BAG-OF-WORDS VECTORS AND A SOKAL & SNEATH-IV COEFFICIENT USED FOR A K-MEANS ALGORITHM OF ENGLISH SENTIMENT CLASSIFICATION IN A PARALLEL SYSTEM

[1]DR.VO NGOC PHU, [2]VO THI NGOC TRAN

[1]Nguyen Tat Thanh University, 300A Nguyen Tat Thanh Street, Ward 13, District 4, Ho Chi Minh City, 702000, Vietnam

[2]School of Industrial Management (SIM), Ho Chi Minh City University of Technology - HCMUT, Vietnam National University, Ho Chi Minh City, Vietnam

E-mail: [1]vongocphu03hca@gmail.com, vongocphu@ntt.edu.vn, [2]vtntran@HCMUT.edu.vn

ABSTRACT

Many different researches have already been used for sentiment classification for many years because the sentiment lexicons have already had many significant contributions to everyday life, such as in political activities, commodity production, and commercial activities. We have proposed a novel model using many bag-of-words vectors (BOWV) and a SOKAL & SNEATH-IV Coefficient (SSIVC) for a K-Means algorithm (KM) to classify all the documents of the testing data set comprising 13,500,000 documents, the 6,750,000 positive and the 6,750,000 negative , into either 2,500,000 positive sentences or 2,500,000 negative sentences of our training data set including 5,000,000 sentences in English. In this survey, the BOWVs are improved by using many sentiment lexicons of our basis English sentiment dictionary (bESD) by using the SSIVC through a Google search engine with AND operator and OR operator. One sentence in English is transferred into one BOWV. The KM is used in clustering one BOWV into either the positive BOWVs or the negative BOWVs of the training data set. The sentiment classification of one document is based on the results of the sentiment classification of its BOWVs. We have firstly tested the proposed model in a sequential environment, and then, the novel model has been tested in a distributed network system secondly. The results of the sequential system are not as good as that of the parallel environment. The accuracy of the testing data set has been achieved 88.76%. Many different fields can widely use the results of the novel model certainly.

Keywords: *English sentiment classification; parallel system; Cloudera; Hadoop Map and Hadoop Reduce; K-Means algorithm; SOKAL & SNEATH-IV Coefficient*

1. INTRODUCTION

A clustering data is a set of objects which is processed into classes of similar objects in a data mining field. One cluster is a set of data objects which are similar to each other and are not similar to objects in other clusters. A number of data clusters can be clustered, which can be identified following experience or can be automatically identified as part of clustering method.

There are the researches related to many sentiment lexicons in [1-38].

According to the survey related to the a vector space modeling (VSM) in [44-46], vector space model is a statistical model for representing text information for Information Retrieval, NLP, Text Mining as follows: (1) Representing documents in VSM is called "vectorizing text" (2) contains the following information: how many documents contain a term, and what are important terms each document. The vector space model procedure can be divided in to three stages. The first stage is the document indexing where content bearing terms are extracted from the document text. The second stage is the weighting of the indexed terms to enhance retrieval of document relevant to the user. The last stage ranks the document with respect to the query according to a similarity measure.

With the surveys related to the bag-of-words (BOW) [47-51], It is a model used in natural

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

language processing. One aim of BoW is to categorize documents. The idea is to analyse and classify different "bags of words" (corpus). And by matching the different categories, we identify which "bag" a certain block of text (test data) comes from.

Based on the studies related to the K-Means algorithm (KM) [52-56], It is one of the simplest unsupervised learning algorithms that solve the well known clustering problem. The procedure follows a simple and easy way to classify a given data set through a certain number of clusters (assume k clusters) fixed a priori. The main idea is to define k centroids, one for each cluster. These centroids shoud be placed in a cunning way because of different location causes different result. So, the better choice is to place them as much as possible far away from each other. The next step is to take each point belonging to a given data set and associate it to the nearest centroid. When no point is pending, the first step is completed and an early groupage is done. At this point we need to re-calculate k new centroids as barycenters of the clusters resulting from the previous step. After we have these k new centroids, a new binding has to be done between the same data set points and the nearest new centroid. A loop has been generated. As a result of this loop we may notice that the k centroids change their location step by step until no more changes are done. In other words centroids do not move any more.

The basic principles are proposed for our new model as follows: (1)It is assumed that each English sentence has m English words (or English phrases). (2)It is assumed that the maximum number of one English sentence is m_max terms (words or phrases); it means that m is less than m_max or m is equal to m_max. (3)It is assumed that each English document has n English sentences. (4)Assuming that the maximum number of one English document is n_max sentences; it means that n is less than n_max or n is equal to n_max.

The motivation of this new model is as follows: Many algorithms in the data mining field can be applied to natural language processing, specifically semantic classification for processing millions of English documents. The KM and a SOKAL & SNEATH-IV Coefficient (SSIVC) of the clustering technologies of the data mining filed can be applied to the sentiment classification in both a sequential environment and a parallel network system. The BOWVs can be improved according to many sentiment lexicons of our basis English sentiment dictionary (bESD). This will result in many discoveries in scientific research, hence the motivation for this study.

The novelty and originality of the proposed approach are as follows:
(1)The KM and the SSIVC were applied to the sentiment classification.
(2)This can also be applied to identify the sentiments (positive, negative, or neutral) of millions of many documents.
(3)This survey can be applied to other parallel network systems.
(4)The Cloudera system, Hadoop Map (M) and Hadoop Reduce (R) were used in the proposed model.
(5)The KM was performed in both the sequential system and the parallel environment.
(6)The SSIVC was implemented in both the sequential environment and the distributed system.
(7)We did not use a vector spae modeling (VSM).
(8)We did not use any one-dimensional vectors.
(9)We did not use any multi-dimensional vectors
(10)The sentiment values and the polarities of the sentiment lexicons of the bESD were calculated by using the PHI through a Google search engine with AND operator and OR operator in both the sequential environment and the distributed network system.
(11)The bag-of-words vectors (BOWVs) which were successfully improved according to the sentiment lexicons of the bESD were built in both the sequential system and the paralle environment.
(12)The input of this survey is the document of the testing data set and the sentences of the training data set. We studied to transfer them into the formats which the proposed model can process certainly.
(13)The KM using the BOWVs was used for the sentiment classification in English in both the sequential system and the parallel environment.

According to the purpose of the research, we always try to find a new approach to improve many accuracies of the results of the sentiment classification and to shorten many execution times of the proposed model with a low cost.

To get higher accuracy and shorten execution time of the sentiment classification, we did not use a vector spae modeling (VSM). We did not use any one-dimensional vectors. We did not use any multi-dimensional vectors. We improve the BOWVs by using the sentiment lexicons of the bESD. We calculate the valences and the polarities of the sentiment lexicons of the bESD through the Google search engine with AND operator and OR operator. One sentence in English is transferred into one BOWV.

In this survey, we implement the proposed model as follows: We calculate the valences and the

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

polarities of the sentiment lexicons of the bESD through the Google search engine with AND operator and OR operator. We improve the BOWVs by using the sentiment lexicons of the bESD. In this survey, one English sentence is transferred into one BOWV.

We perform the proposed model as follows: The sentiment scores and the polarities of the sentiment lexicons of the bESD are firstly calculated by using the SSIVC through the Google search engine with AND operator and OR operator. The BOWVs are improved by using the sentiment lexicons. In this survey, one English sentence is transferred into one BOWV. A positive group which we transfer the positive sentences of the training data set into the BOWVs and a negative group which we transfer the negative sentences of the training data set into the BOWVs. All the sentences of one document of the testing data set are transferred into the BOWVs of the document. We use the KM to cluster one BOWV of the document into either the positive group or the negative group of the training data set. The sentiment classification of the document is identified according to the results of its BOWVs. If the number of the BOWVs clustered into the positive is greater than that clustered into the negative in the document, this document is clustered into the positive. If the number of the BOWVs clustered into the positive is less than that clustered into the negative in the document, this document is clustered into the negative. If the number of the BOWVs clustered into the positive is as equal as that clustered into the negative in the document, this document is clustered into the neutral. Finally, the sentiment classification of all the documents of the testing data set is identified certainly.

All the above things are firstly implemented in the sequential system to get an accuracy of the result of the sentiment classification and an execution time of the result of the sentiment classification of the proposed model. All the above things are secondly performed in the parallel network environment to shorten the execution times of the proposed model to get the accuracy of the results of the sentiment classification and the execution times of the results of the sentiment classification of our new model

The significant contributions of our new model can be applied to many areas of research as well as commercial applications as follows:

(1)Many surveys and commercial applications can use the results of this work in a significant way.

(2)The algorithms are built in the proposed model.

(3)This survey can certainly be applied to other languages easily.

(4)The results of this study can significantly be applied to the types of other words in English.

(5)The algorithm of data mining is applicable to semantic analysis of natural language processing.

(6)This study also proves that different fields of scientific research can be related in many ways.

(7)Millions of English documents are successfully processed for emotional analysis.

(8)The semantic classification is implemented in the parallel network environment.

(9)The principles are proposed in the research.

(10)The Cloudera distributed environment is used in this study.

(11)The proposed work can be applied to other distributed systems.

(12)This survey uses Hadoop Map (M) and Hadoop Reduce (R).

(13)Our proposed model can be applied to many different parallel network environments such as a Cloudera system

(14)This study can be applied to many different distributed functions such as Hadoop Map (M) and Hadoop Reduce (R).

(15)The KM – related algorithms are proposed in this survey.

(16)The SSIVC – related algorithms are built in this work.

(17)The BOW – related algorithms are proposed in this study.

This study contains 6 sections. Section 1introduces the study; Section 2 discusses the related works about the bag-of-words (BOW), K-Means Algorithm (KM), SOKAL & SNEATH-IV Coefficient (SSIVC), etc.; Section 3 is about the English data set; Section 4 represents the methodology of our proposed model; Section 5 represents the experiment. Section 6 provides the conclusion. The References section comprises all the reference documents; all tables are shown in the Appendices section.

2. RELATED WORK

We summarize many researches which are related to our research. By far, we know that PMI (Pointwise Mutual Information) equation and SO (Sentiment Orientation) equation are used for determining polarity of one word (or one phrase), and strength of sentiment orientation of this word (or this phrase). Jaccard measure (JM) is also used for calculating polarity of one word and the equations from this Jaccard measure are also used for calculating strength of sentiment orientation this

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

word in other research. PMI, Jaccard, Cosine, Ochiai, Tanimoto, and Sorensen measure are the similarity measure between two words; from those, we prove that the SOKAL & SNEATH-IV Coefficient (SSIVC) is also used for identifying valence and polarity of one English word (or one English phrase). Finally, we identify the sentimental values of English verb phrases based on the basis English semantic lexicons of the basis English emotional dictionary (bESD).

There are the works related to the equations of the similarity measures in [1-27]. In the research [1], the authors generated several Norwegian sentiment lexicons by extracting sentiment information from two different types of Norwegian text corpus, namely, news corpus and discussion forums. The methodology was based on the Point wise Mutual Information (PMI). The authors introduced a modification of the PMI that considers small "blocks" of the text instead of the text as a whole, etc. The surveys related to the similarity coefficients to calculate the valences of words are in [28-32].

The English dictionaries are [33-38] and there are more than 55,000 English words (including English nouns, English adjectives, English verbs, etc.) from them.

There are the works related to the SOKAL & SNEATH-IV Coefficient (SSIVC) in [39, 43]. The authors in [39] collected 76 binary similarity and distance measures used over the last century and reveal their correlations through the hierarchical clustering technique, etc.

There are the works related to vector space modeling (VSM) in [44-46]. In this study [44], the authors examined the Vector Space Model, an Information Retrieval technique and its variation, etc.

The surveys related to the bag-of-words are in [47-51]. In the work [47], the authors explored a hierarchical generative probabilistic model that incorporates both n-gram statistics and latent topic variables by extending a unigram topic model to include properties of a hierarchical Dirichlet bigram language model, etc.

There are the researches related to the K-Means algorithm (KM) in [52-56]. In the study [52], the authors proposed a novel hybrid genetic algorithm (GA) that found a globally optimal partition of a given data into a specified number of clusters, etc.

The latest researches of the sentiment classification are [57-67]. The main approaches for sentiment analysis in [57] could be categorized into semantic orientation-based approaches, knowledge-based, and machine-learning algorithms, etc.

3. DATA SET

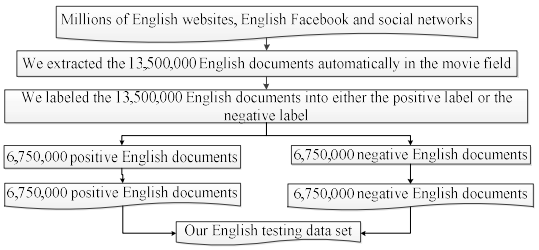

Fig. 1: Our testing data set in English.

Based on Fig 1, we built our the testing data set including the 13,500,000 documents in the movie field, which contains the 6,750,000 positive and 6,750,000 negative in English. All the documents in our English testing data set are automatically extracted from English Facebook, English websites and social networks; then we labeled positive and negative for them.

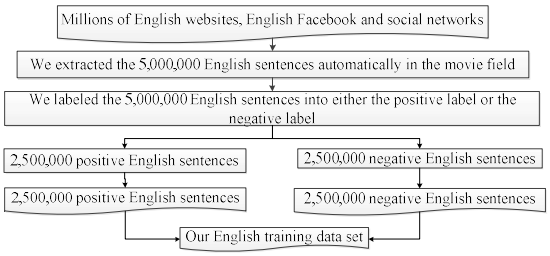

Fig 2: Our training data set in English.

In Fig 2, we built our the training data set including the 5,000,000 sentences in the movie field, which contains the 2,500,000 positive and 2,500,000 negative in English. All the sentences in our English training data set are automatically extracted from English Facebook, English websites and social networks; then we labeled positive and negative for them.

4. METHODOLOGY

This section comprises three sub-sections as follows: (4.1) and (4.2). In the sub-section (4.1), we create the sentiment lexicons of the bESD. In the sub-section (4.2), we improve the BOWVs based on the sentiment lexicons of the bESD. In the sub-section (4.3), we use the KM to cluster the documents of the testing data set into either the positive or the negative in both a sequential environment and a parallel distributed system.

4.1 Creating the sentiment lexicons in English

The section includes three parts as follows: (4.1.1); (4.1.2); and (4.1.3).

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

4.1.1 Calculating a valence of one word (or one phrase) in English

In this part, we calculate the valence and the polarity of one English word (or phrase) by using the SSIVC through a Google search engine with AND operator and OR operator, as the following diagram in Fig 3 below shows.

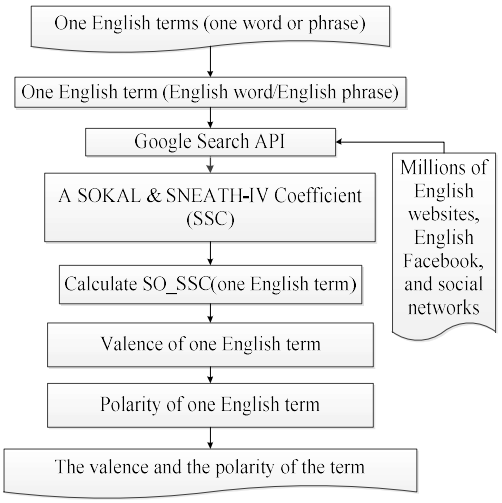

Fig. 3: Overview Of Identifying The Valence And The Polarity Of One Term In English Using A SOKAL & SNEATH-IV Coefficient (SSIVC)

According to [1-15], Pointwise Mutual Information (PMI) between two words wi and wj has the equation

$$PMI(wi, wj) = log_2(\frac{P(wi, wj)}{P(wi)xP(wj)}) \quad (1)$$

and SO (sentiment orientation) of word wi has the equation

$$SO\ (wi) = PMI(wi, positive) - PMI(wi, negative) \quad (2)$$

In [1-8] the positive and the negative of Eq. (2) in English are: positive = {good, nice, excellent, positive, fortunate, correct, superior} and negative = {bad, nasty, poor, negative, unfortunate, wrong, inferior}. The AltaVista search engine is used in the PMI equations of [2, 3, 5] and the Google search engine is used in the PMI equations of [4, 6, 8]. Besides, [4] also uses German, [5] also uses Macedonian, [6] also uses Arabic, [7] also uses Chinese, and [8] also uses Spanish. In addition, the Bing search engine is also used in [6]. With [9-12], the PMI equations are used in Chinese, not English, and Tibetan is also added in [9]. About the search engine, the AltaVista search engine is used in [11] and [12] and uses three search engines, such as the Google search engine, the Yahoo search engine and the Baidu search engine. The PMI equations are also used in Japanese with the Google search engine in [13]. [14] and [15] also use the PMI equations and Jaccard equations with the Google search engine in English. In [14-21] the positive and the negative of Eq. (5) in English are: positive = {good, nice, excellent, positive, fortunate, correct, superior} and negative = {bad, nasty, poor, negative, unfortunate, wrong, inferior}. The Jaccard equations with the Google search engine in English are used in [14, 15, 17]. [16] and [21] use the Jaccard equations in English. [20] and [22] use the Jaccard equations in Chinese. [18] uses the Jaccard equations in Arabic. The Jaccard equations with the Chinese search engine in Chinese are used in [19].The authors in [28] used the Ochiai Measure through the Google search engine with AND operator and OR operator to calculate the sentiment values of the words in Vietnamese. The authors in [29] used the Cosine Measure through the Google search engine with AND operator and OR operator to identify the sentiment scores of the words in English. The authors in [30] used the Sorensen Coefficient through the Google search engine with AND operator and OR operator to calculate the sentiment values of the words in English. The authors in [31] used the Jaccard Measure through the Google search engine with AND operator and OR operator to calculate the sentiment values of the words in Vietnamese. The authors in [32] used the Tanimoto Coefficient through the Google search engine with AND operator and OR operator to identify the sentiment scores of the words in English

With the above proofs, we have the information as follows: PMI is used with AltaVista in English, Chinese, and Japanese with the Google in English; Jaccard is used with the Google in English, Chinese, and Vietnamse. The Ochiai is used with the Google in Vietnamese. The Cosine and Sorensen are used with the Google in English.

According to [1-32], PMI, Jaccard, Cosine, Ochiai, Sorensen, Tanimoto and SOKAL & SNEATH-IV Coefficient (SSIVC) are the similarity measures between two words, and they can perform the same functions and with the same characteristics; so SSIVC is used in calculating the valence of the words. In addition, we prove that SSIVC can be used in identifying the valence of the English word through the Google search with the AND operator and OR operator.

With the SOKAL & SNEATH-IV Coefficient (SSIVC) in [39-43], we have the equation of the SSIVC as follows:

Journal of Theoretical and Applied Information Technology
15th August 2018. Vol.96. No 15

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

$$\text{SOKAL \& SNEATH} - \text{IV Coefficient}(a,b) = \text{SOKAL \& SNEATH} - \text{IV Measure}(a,b) = \text{SSIVC}(a,b) = \frac{A3}{4} \quad (3)$$

with a and b are the vectors.

$$A3 = \frac{(a\cap b)}{(a\cap b)+(\neg a\cap b)} + \frac{(a\cap b)}{(a\cap b)+(a\cap\neg b)} + \frac{(\neg a\cap\neg b)}{(\neg a\cap\neg b)+(\neg a\cap b)} + \frac{(\neg a\cap\neg b)}{(\neg a\cap\neg b)+(a\cap\neg b)}$$

From the eq. (1), (2), (3), we propose many new equations of the SSIVC to calculate the valence and the polarity of the English words (or the English phrases) through the Google search engine as the following equations below.

In eq. (3), when a has only one element, a is a word. When b has only one element, b is a word. In eq. (3), a is replaced by w1 and b is replaced by w2.

$$\text{SOKAL \& SNEATH} - \text{IV Measure}(w1,w2) = \text{SOKAL \& SNEATH} - \text{IV Coefficient}(w1,w2) = \text{SSIVC}(w1,w2) = \frac{A3\_1}{4} \quad (4)$$

$$\text{With} \quad A3\_1 = \frac{P(w1,w2)}{P(w1,w2)+P(\neg w1,w2)} + \frac{P(w1,w2)}{P(w1,w2)+P(w1,\neg w2)} + \frac{P(\neg w1,\neg w2)}{P(\neg w1,\neg w2)+P(\neg w1,w2)} + \frac{P(\neg w1,\neg w2)}{P(\neg w1,\neg w2)+P(w1,\neg w2)}$$

Eq. (4) is similar to eq. (1). In eq. (2), eq. (1) is replaced by eq. (4). We have eq. (5) as follows:

$$\text{Valence}(w) = \text{SO\_SSIVC}(w) = \text{SSIVC}(w, \text{positive\_query}) - \text{SSIVC}(w, \text{negative\_query}) \quad (5)$$

In eq. (4), w1 is replaced by w and w2 is replaced by position_query. We have eq. (6) as follows:

$$\text{SSIVC}(w, \text{positive\_query}) = \frac{A6}{4} \quad (6)$$

with $\text{a\_b\_6} = P(w, \text{positive\_query}) + P(\neg w, \text{positive\_query})$

$\text{a\_c\_6} = P(w, \text{positive\_query}) + P(w, \neg\text{positive\_query})$

$\text{b\_d\_6} = P(\neg w, \neg\text{positive\_query}) + P(\neg w, \text{positive\_query})$

$\text{c\_d\_6} = P(\neg w, \neg\text{positive\_query}) + P(w, \neg\text{positive\_query})$

$$A6 = \frac{P(w,\text{positive\_query})}{\text{a\_b\_6}} + \frac{P(w,\text{positive\_query})}{\text{a\_c\_6}} + \frac{P(\neg w,\neg\text{positive\_query})}{\text{b\_d\_6}} + \frac{P(\neg w,\neg\text{positive\_query})}{\text{c\_d\_6}}$$

In eq. (4), w1 is replaced by w and w2 is replaced by negative_query. We have eq. (7) is as follows:

$$\text{SSIVC}(w, \text{negative\_query}) = \frac{A7}{4} \quad (7)$$

with $\text{a\_b\_7} = P(w, \text{negative\_query}) + P(\neg w, \text{negative\_query})$

$\text{a\_c\_7} = P(w, \text{negative\_query}) + P(w, \neg\text{negative\_query})$

$\text{b\_d\_7} = P(\neg w, \neg\text{negative\_query}) + P(\neg w, \text{negative\_query})$

$\text{c\_d\_7} = P(\neg w, \neg\text{negative\_query}) + P(w, \neg\text{negative\_query})$

$$A7 = \frac{P(w,\text{negative\_query})}{\text{a\_b\_7}} + \frac{P(w,\text{negative\_query})}{\text{a\_c\_7}} + \frac{P(\neg w,\neg\text{negative\_query})}{\text{b\_d\_7}} + \frac{P(\neg w,\neg\text{negative\_query})}{\text{c\_d\_7}}$$

We have the information about w, w1, w2, and etc. as follows:

(1)w, w1, w2 : are the English words (or the English phrases).

(2)P(w1, w2): number of returned results in Google search by keyword (w1 and w2). We use the Google Search API to get the number of returned results in search online Google by keyword (w1 and w2).

(3)P(w1): number of returned results in Google search by keyword w1. We use the Google Search API to get the number of returned results in search online Google by keyword w1.

(4)P(w2): number of returned results in Google search by keyword w2. We use the Google Search API to get the number of returned results in search online Google by keyword w2.

(5)Valence(W) = SO_SSIVC(w): valence of English word (or English phrase) w; is SO of word (or phrase) by using the SOKAL & SNEATH-IV Coefficient (SSIVC)

(6) positive_query: { active or good or positive or beautiful or strong or nice or excellent or fortunate or correct or superior } with the positive query is the a group of the positive English words.

(7)negative_query: { passive or bad or negative or ugly or week or nasty or poor or unfortunate or wrong or inferior } with the negative_query is the a group of the negative English words.

(8)P(w, positive_query): number of returned results in Google search by keyword (positive_query and w). We use the Google Search API to get the number of returned results in search online Google by keyword (positive_query and w).

(9)P(w, negative_query): number of returned results in Google search by keyword (negative_query and w). We use the Google Search API to get the number of returned results in search online Google by keyword (negative_query and w).

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

(10)P(w): number of returned results in Google search by keyword w. We use the Google Search API to get the number of returned results in search online Google by keyword w.
(11)P(¬w,positive_query): number of returned results in Google search by keyword ((not w) and positive_query). We use the Google Search API to get the number of returned results in search online Google by keyword ((not w) and positive_query).
(12)P(w, ¬positive_query): number of returned results in the Google search by keyword (w and (not (positive_query))). We use the Google Search API to get the number of returned results in search online Google by keyword (w and [not (positive_query)]).
(13)P(¬w,negative_query): number of returned results in Google search by keyword ((not w) and negative_query). We use the Google Search API to get the number of returned results in search online Google by keyword ((not w) and negative_query).
(14)P(w,¬negative_query): number of returned results in the Google search by keyword (w and (not (negative_query))). We use the Google Search API to get the number of returned results in search online Google by keyword (w and (not (negative_query))).

As like Cosine, Ochiai, Sorensen, Tanimoto, PMI and Jaccard about calculating the valence (score) of the word, we identify the valence (score) of the English word w based on both the proximity of positive_query with w and the remote of positive_query with w; and the proximity of negative_query with w and the remote of negative_query with w.
The English word w is the nearest of positive_query if SSIVC (w, positive_query) is as equal as 1. The English word w is the farthest of positive_query if SSIVC(w, positive_query) is as equal as 0. The English word w belongs to positive_query being the positive group of the English words if SSIVC(w, positive_query) > 0 and SSIVC(w, positive_query) ≤ 1. The English word w is the nearest of negative_query if SSIVC(w, negative_query) is as equal as 1. The English word w is the farthest of negative_query if SSIVC(w, negative_query) is as equal as 0. The English word w belongs to negative_query being the negative group of the English words if SSIVC(w, negative_query) > 0 and SSIVC(w, negative_query) ≤ 1. So, the valence of the English word w is the value of SSIVC(w, positive_query) substracting the value of SSIVC(w, negative_query) and the eq. (5) is the equation of identifying the valence of the English word w.
We have the information about SSIVC as follows: (1)SSIVC(w, positive_query) ≥ 0 and SSIVC(w, positive_query) ≤ 1. (2)SSIVC(w, negative_query) ≥ 0 and SSIVC(w, negative_query) ≤ 1. (3)If SSIVC(w, positive_query) = 0 and SSIVC(w, negative_query) = 0 then SO_SSIVC(w) = 0. (4)If SSIVC(w, positive_query) = 1 and SSIVC(w, negative_query) = 0 then SO_SSIVC(w) = 0. (5)If SSIVC(w, positive_query) = 0 and SSIVC(w, negative_query) = 1 then SO_SSIVC(w) = -1. (6)If SSIVC(w, positive_query) = 1 and SSIVC(w, negative_query) = 1 then SO_SSIVC(w) = 0. So, SO_SSIVC(w) ≥ -1 and SO_SSIVC(w) ≤ 1.

The polarity of the English word w is positive polarity If SO_SSIVC(w) > 0. The polarity of the English word w is negative polarity if SO_SSIVC(w) < 0. The polarity of the English word w is neutral polarity if SO_SSIVC(w) = 0. In addition, the semantic value of the English word w is SO_SSIVC(w).

We calculate the valence and the polarity of the English word or phrase w using a training corpus of approximately one hundred billion English words — the subset of the English Web that is indexed by the Google search engine on the internet. AltaVista was chosen because it has a NEAR operator. The AltaVista NEAR operator limits the search to documents that contain the words within ten words of one another, in either order. We use the Google search engine which does not have a NEAR operator; but the Google search engine can use the AND operator and the OR operator. The result of calculating the valence w (English word) is similar to the result of calculating valence w by using AltaVista. However, AltaVista is no longer.
In summary, by using eq. (4), eq. (5), eq. (6), and eq. (7), we identify the valence and the polarity of one word (or one phrase) in English by using the SSIVC through the Google search engine with AND operator and OR operator.

To see the advantages of the proposed model, we show the comparisons of this model's results with the surveys in the tables as follows: Table 3, and Table 4.

In Table 3, we present the comparisons of our model's advantages and disadvantages with the works related to [1-32].

The comparisons of our model's benefits and drawbacks with the studies related to the SOKAL & SNEATH-IV coefficient (SSIVC) in [39-43] are shown in Table 4.

4.1.2 Creating a basis English sentiment dictionary (bESD) in a sequential environment

According to [33-38], we have at least 55,000 English terms, including nouns, verbs, adjectives,

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

etc. In this part, we calculate the valence and the polarity of the English words or phrases for our basis English sentiment dictionary (bESD) by using the SSIVC in a sequential system, as the following diagram in Fig 4 below shows.

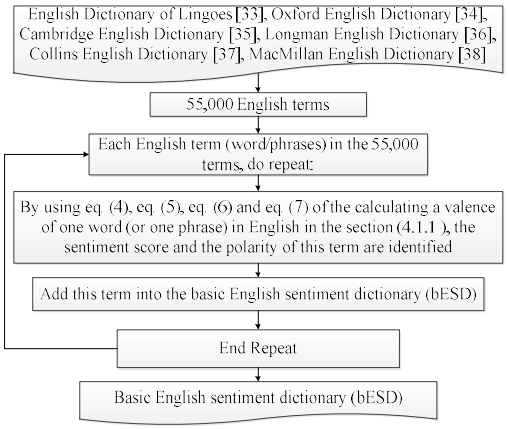

Fig. 4: Overview Of Creating A Basis English Sentiment Dictionary (Besd) In A Sequential Environment

We proposed the algorithm 1 to perform this section.

Input: the 55,000 English terms; the Google search engine

Output: a bESD

Step 1: Each term in the 55,000 terms, do repeat:

Step 2: By using eq. (4), eq. (5), eq. (6) and eq. (7) of the calculating a valence of one word (or one phrase) in English in the section (4.1.1), the sentiment score and the polarity of this term are identified. The valence and the polarity are calculated by using the SSIVC through the Google search engine with AND operator and OR operator.

Step 3: Add this term into the basis English sentiment dictionary (bESD);

Step 4: End Repeat – End Step 1;

Step 5: Return bESD;

Our bESD has more 55,000 English words (or English phrases) and bESD is stored in Microsoft SQL Server 2008 R2.

4.1.3 Creating a basis English sentiment dictionary (bESD) in a distributed system

According to [33-38], we have at least 55,000 English terms, including nouns, verbs, adjectives, etc. In this part, we calculate the valence and the polarity of the English words or phrases for our basis English sentiment dictionary (bESD) by using the SSIVC in a parallel network environment, as the following diagram in Fig 5 below shows.

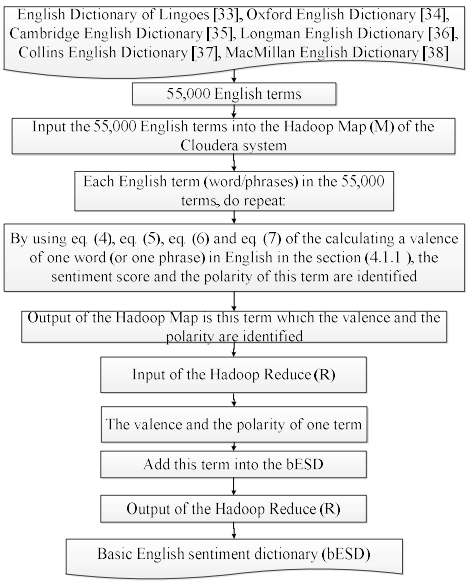

Fig. 5: Overview Of Creating A Basis English Sentiment Dictionary (Besd) In A Distributed Environment

This section includes two phases: the Hadoop Map (M) phase and the Hadoop Reduce (R) phase. The input of the Hadoop Map phase is the 55,000 terms in English in [33-38]. The output of the Hadoop Map phase is one term which the sentiment score and the polarity are identified. The output of the Hadoop Map phase is the input of the Hadoop Reduce phase. Thus, the input of the Hadoop Reduce phase is one term which the sentiment score and the polarity are identified. The output of the Hadoop Reduce phase is the basis English sentiment dictionary (bESD).

We built the algorithm 2 to implement the Hadoop Map phase.

Input: the 55,000 English terms; the Google search engine

Output: one term which the sentiment score and the polarity are identified.

Step 1: Each term in the 55,000 terms, do repeat:

Step 2: By using eq. (4), eq. (5), eq. (6) and eq. (7) of the calculating a valence of one word (or one phrase) in English in the section (4.1.1), the sentiment score and the polarity of this term are identified. The valence and the polarity are calculated by using the SSIVC through the Google search engine with AND operator and OR operator.

Step 3: Return this term;

We proposed the algorithm 3 to perform the Hadoop Reduce phase

Input: one term which the sentiment score and the polarity are identified – The output of the Hadoop

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Map phase.
Output: a basis English sentiment dictionary (bESD)
Step 1: Add this term into the basis English sentiment dictionary (bESD);
Step 2: Return bESD;

Our bESD has more 55,000 English words (or English phrases) and bESD is stored in Microsoft SQL Server 2008 R2.

4.2 Transferring the sentences into the BOWVs according to the sentiment lexicons in both a sequential environment and a parallel network system

This section has three parts as follows: (4.2.1), (4.2.2), and (4.2.3).

4.2.1 Improving the BOWVs based on the sentiment lexicons of the bESD

In this section, we reform the bag-of-words vectors (BOWVs) according to the sentiment lexicons of the bESD.

Based on the surveys related to the BOWVs [47-51], the Bag of Words (BOW) model is widely used in text classification and natural language processing. It models the text of the document as an unordered collection of words. This collection does not take into account the grammatical significance or order of these words, with the mere presence of the word in the document qualifying the word to be in the BOW model. Consider the following two documents (each with one sentence):

d1= "Machine learning is important in classification. Document classification is an important problem"

d2= "Machine learning is a Computer Science concept"

From these two documents, a dictionary is constructed as follows:
B = 1:"Machine", 2:"learning", 3:"is", 4:"important", 5:"in", 6:"document", 7:"classification", 8:"an", 9:"problem", 10: "computer", 11:"science", 12:"concept", 13:"a"
Based on the BOW model the vectors for these documents become:

Vd1= [1,1,2,2,1,1,2,1,1,0,0,0,0]
Vd2= [1,1,1,0,0,0,0,0,0,1,1,1,1]

Here, each column represents the count of the occurrences of a word in the document.

We use two formats of BOW as inputs.
1)BOW presence of words - Here, the values in the vector is either 1 indicating presence of the word, or 0, indicating the absence of the word.
2) BOW normalized frequency occurrence of words - Here, we normalize the vectors to unit length as follows: norm(Vi) =

Vi/||Vi||

As known, in one sentence, there are sometimes many terms (meaningful words or meaningful phrases) bearing the neutral sentiment, the positive sentiment, or the negative sentiment. For example, we assume that in the bESD, "important" is the positive sentiment and its valence is +2.1. "problem" is the positive sentiment and its sentiment score is +0.3.

We see that the neutral terms are not the important role in a sentence. Thus, if we still use them in calculating the sentiment of the sentence, there are many noises for this calculating.
We also see that the negative terms make many noises for calculating the sentiment of the positive polarity and the positive terms make many noises for calculating the sentiment of the negative polarity

With Vd1= [1,1,2,2,1,1,2,1,1,0,0,0,0] and Vd2= [1,1,1,0,0,0,0,0,0,1,1,1,1], we use the valences of the terms combined with their frequencies to remove many noises of identifiying the sentiment classification of one sentence. We apply the valences of the sentiment lexicons of the bESD into the Vd1 and Vd2 as follows: According to the bESD, it is assumed that "Machine" is 0 of its valence; "learning" is 0 of its sentiment value; "is" is 0 of its sentiment scorel "important" of +2.1 of its valence; "in" is 0 of its sentiment score; "document" is 0 of its valence; "classification" is 0 of its sentiment value; "an" is 0 of its valence; "problem" is +0.3 of its sentiment score; the sentiment value of "computer" is 0; the valence of "science" is 0; the sentiment score of "concept" is 0; and the sentiment value of "a" is 0. Therefore, we have Vd1 and Vd2 as follows: Vd1 = [0,0,0,2*(+2.1),0,0,0,0,+0.3,0,0,0,0] and Vd1 = [0,0,0,0,0,0,0,0,0,0,0,0,0] => emphasizing on the positive terms and the negative terms in one sentence.

In one BOWV, the value of each element is (its valences) x (its frequency).

4.2.2 Transferring the sentences into the BOWVs according to the sentiment lexicons in a sequential environment

In this section, we transfer all the sentences of the training data set into the BOWVs based on the sentiment lexicons and all the sentences of one document of the testing data set into the BOWVs according to the sentiment lexicons in the sequential system.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

We built the algorithm 4 to create an order list of the BOWV which comprises all the meaningful terms of both the testing data set and the training data set in the sequential system.

Input: the documents of the testing data set and the sentences of the training data set.

Output: an order list of the BOWV – AnOrderListOfTheBOWV

Step 1: Set AnOrderListOfTheBOWV := {}

Step 2: Each sentence in the sentences of the training data set, repeat:

Step 3: Split this sentence into the meaningful terms based on the sentiment lexicons of the bESD;

Step 4: Each term in the meaningful terms, repeat:

Step 5: If checking this term in AnOrderListOfTheBOWV is false Then

Step 6: Add this term into AnOrderListOfTheBOWV;

Step 7: End If –Step 5;

Step 8: End Repeat – End Step 2;

Step 9: Each document in the documents of the testing data set, repeat:

Step 10: Split this document into the sentences;

Step 11: Each sentence into the sentences, repeat:

Step 12: Split this sentence into the meaningful terms according to the sentiment lexicons of the bESD;

Step 13: Each term in the meaningful terms, repeat:

Step 14: If checking this term in AnOrderListOfTheBOWV is false Then

Step 15: Add this term into AnOrderListOfTheBOWV;

Step 16: End If –Step 13;

Step 17: End Repeat – Step 11;

Step 18: End Repeat – Step 9;

Step 19: Return AnOrderListOfTheBOWV;

We proposed the algorithm 5 to transfer one sentence in English into one BOWV in the sequential system.

Input: one sentence in English and an order list of the BOWV – AnOrderListOfTheBOWV

Output: one BOWV;

Step 1: Set BOWV := {} with its length is the length of AnOrderListOfTheBOWV;

Step 2: Set i := 0;

Step 3: Each term in AnOrderListOfTheBOWV, repeat:

Step 4: Number := Count this term in this sentence;

Step 5: Valence := Get the valence of this term based on the sentiment lexicons of the bESD;

Step 6: Set BOWV[i] := Number * Valence;

Step 7: Set i := i + 1;

Step 8: End Repeat – End Step 3;

Step 9: Return BOWV;

We built the algorithm 6 to transfer all the sentences of one document into the BOWVs in the sequential system.

Input: one document in English

Output: the BOWVs of this document - BOWVs;

Step 1: Split this document into the sentences;

Step 2: Each sentence into the sentences, repeat:

Step 3: BOWV := the algorithm 5 to transfer this sentence in English into one BOWV in the sequential system;

Step 4: BOWV into BOWVs;

Step 5: End Repeat – End Step 2;

Step 6: Return BOWVs;

We proposed the algorithm 7 to create a positive group of the training data set from the positive sentences of the training data set in the sequential system.

Input: the positive sentences of the training data set

Output: the positive BOWVs of the training dat set – a positive group – APositiveGroup;

Step 1: Set APositiveGroup := {}

Step 2: Each sentence in the positive sentences of the training data set, repeat:

Step 3: BOWV : = the algorithm 5 to transfer this sentence in English into one BOWV in the sequential system;

Step 4: Add BOWV into APositiveGroup;

Step 5: End Repeat – End Step 2;

Step 6: Return APositiveGroup;

We proposed the algorithm 8 to create a negative group of the training data set from the negative sentences of the training data set in the sequential system.

Input: the negative sentences of the training data set

Output: the negative BOWVs of the training dat set – a negative group – ANegativeGroup;

Step 1: Set ANegativeGroup:= {}

Step 2: Each sentence in the negative sentences of the training data set, repeat:

Step 3: BOWV : = the algorithm 5 to transfer this sentence in English into one BOWV in the sequential system;

Step 4: Add BOWV into ANegativeGroup;

Step 5: End Repeat – End Step 2;

Step 6: Return ANegativeGroup;

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

4.2.3 Transferring the sentences into the BOWVs according to the sentiment lexicons in a parallel network system

In this section, we transfer all the sentences of the training data set into the BOWVs based on the sentiment lexicons and all the sentences of one document of the testing data set into the BOWVs according to the sentiment lexicons in the distributed system.

In Fig 6, we create an order list of the BOWV which comprises all the meaningful terms of both the testing data set and the training data set in the distributed network system. This stage comprises two phases as follows: Hadoop Map phase and Hadoop Reduce phase. The input of the Hadoop Map is the documents of the testing data set and the sentences of the training data set. The output of the Hadoop Map phase is one term. The input of the Hadoop Reduce is the output of the Hadoop Map, thus, the input of the Hadoop Reduce is one term. The output of the Hadoop Reduce is an order list of the BOWV – AnOrderListOfTheBOWV.

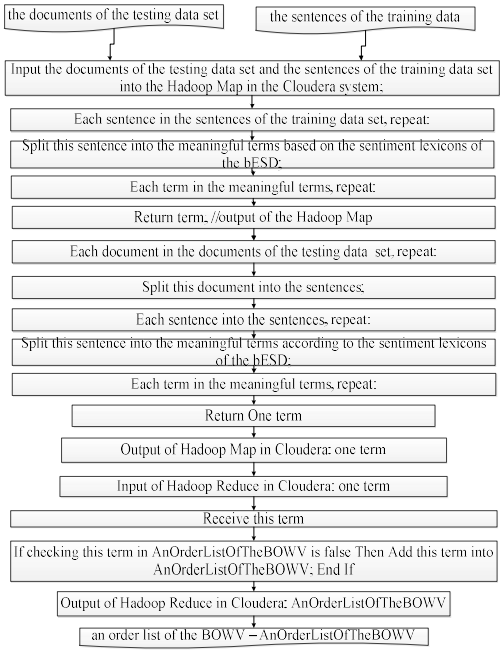

Fig. 6: Overview Of Creating An Order List Of The BOWV Which Comprises All The Meaningful Terms Of Both The Testing Data Set And The Training Data Set In The Distributed Network **system**

We built the algorithm 9 to perform the Hadoop Map

Input: the documents of the testing data set and the sentences of the training data set.

Output: one term;

Step 1: Input the documents of the testing data set and the sentences of the training data set into the Hadoop Map in the Cloudera system;

Step 2: Each sentence in the sentences of the training data set, repeat:

Step 3: Split this sentence into the meaningful terms based on the sentiment lexicons of the bESD;

Step 4: Each term in the meaningful terms, repeat:

Step 5: Return term; //output of the Hadoop Map

Step 6: Each document in the documents of the testing data set, repeat:

Step 7: Split this document into the sentences;

Step 8: Each sentence into the sentences, repeat:

Step 9: Split this sentence into the meaningful terms according to the sentiment lexicons of the bESD;

Step 10: Each term in the meaningful terms, repeat:

Step 11: Return term; //output of the Hadoop Map

We proposed the algorithm 10 to implement the Hadoop Reduce

Input: one term; //output of the Hadoop Map

Output: an order list of the BOWV – AnOrderListOfTheBOWV

Step 1: If checking this term in AnOrderListOfTheBOWV is false Then

Step 2: Add this term into AnOrderListOfTheBOWV;

Step 3: End If –Step 1;

Step 4: Return AnOrderListOfTheBOWV;

In Fig 7, we transfer one English sentence into one BOWV in Cloudera. This stage includes two phases: the Hadoop Map phase and the Hadoop Reduce phase. The input of the Hadoop Map phase is one sentence and AnOrderListOfTheBOWV. The output of the Hadoop Map phase is the number of one term and the valence of one term. The input of the Hadoop Reduce phase is the output of the Hadoop Map, thus, the input of the Hadoop Reduce phase is the number of one term and the valence of one term. The output of the Hadoop Reduce phase is one BOWV of this sentence.

We built the algorithm 11 to perform the Hadoop Map phase

Input: one sentence and AnOrderListOfTheBOWV;

Output: Number and Valence; //the output of the Hadoop Map phase.

Step 1: Input this sentence and AnOrderListOfTheBOWV into the Hadoop Map in the Cloudera system;

Step 2: Each term in the AnOrderListOfTheBOWV, do repeat:

Step 3: Number := Count this term in this sentence;

Step 4: Valene := Get the valence of this term based on the sentiment lexicons of the bESD;

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Step 5: Return (Number, Valence); //the output of the Hadoop Map phase.

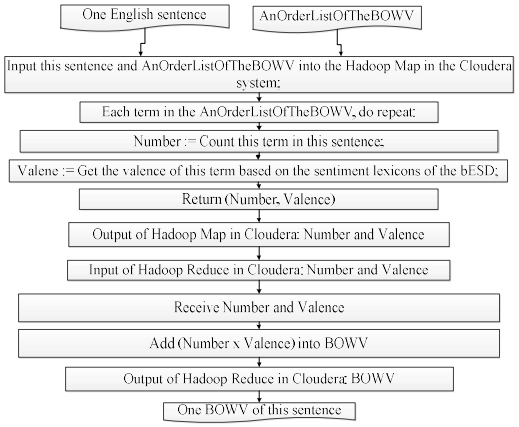

Fig. 7: Overview Of Transferring One English Sentence Into One BOWV In Cloudera

We proposed the algorithm 12 to perform the Hadoop Reduce phase
Input: Number and Valence; //the output of the Hadoop Map phase.
Output: one BOWV of this sentence
Step 1: Receive Number and Valence;
Step 2: Add (Number x Valence) into BOWV;
Step 3: Return BOWV;

In Fig 8, we transfer all the sentences of one document of the testing data set into the BOWVs of the document of testing data set in the parallel network environment. This stage comprise two phases: the Hadoop Map phase and the Hadoop Reduce phase. The input of the Hadoop Map is one document of the testing data set and AnOrderListOfTheBOWV. The output of the Hadoop Reduce is one BOWV (corresponding to one sentence) of this document. The input of the Hadoop Reduce is the output of the Hadoop Map, thus, the input of the Hadoop Reduce is one BOWV (corresponding to one sentence) of this document. The output of the Hadoop Reduce is the BOWVs of this document.

We built the algorithm 13 to perform the Hadoop Map phase
Input: one document of the testing data set and AnOrderListOfTheBOWV;
Output: one BOWV of this document
Step 1: Input this document and AnOrderListOfTheBOWVinto the Hadoop Map in the Cloudera system;
Step 2: Split this document into the sentences;
Step 3: Each sentence in the sentences, do repeat:
Step 4: BOWV := the transforming one English sentence into one BOWV in Cloudera in Fig 7 with the input is this sentence
Step 5: Return BOWV; //the output of the Hadoop Map phase.

We proposed the algorithm 14 to perform the Hadoop Reduce phase
Input: one BOWV of this document
Output: the BOWVs of this document
Step 1: Receive one BOWV;
Step 2: Add this BOWV into BOWVs;
Step 3: Return BOWVs;

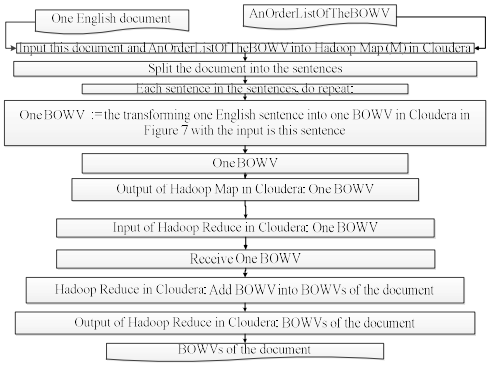

Fig. 8: Overview Of Transferring All The Sentences Of One Document Of The Testing Data Set Into The Bowvs Of The Document Of Testing Data Set In The Parallel Network Environment.

In Fig 9, we transfer the positive sentences of the training data set into the positive BOWVs (called the positive group of the training data set) in the distributed system.

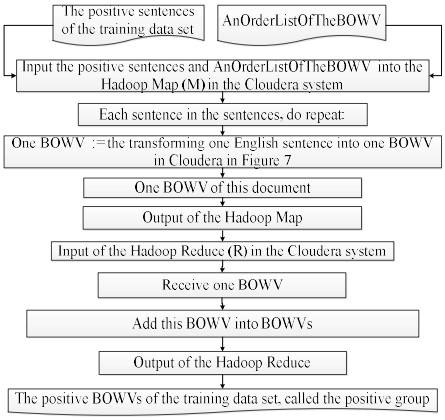

Fig. 9: Overview Of Transferring The Positive Sentences Of The Training Data Set Into The Positive Bowvs (Called The Positive Group Of The Training Data Set) In The Distributed System.

The stage includes two phases: the Hadoop Map (M) phase and the Hadoop Reduce (R) phase. The input of the Hadoop Map phase is the positive sentences of the training data set and AnOrderListOfTheBOWV. The output of the Hadoop Map phase is one BOWV of the positive sentences of the training data set. The input of the

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Hadoop Redude phase is the output of the Hadoop Map phase, thus, the input of the Hadoop Reduce phase is one BOWV of one sentence of the positive sentences of the training data set. The output of the Hadoop Reduce phase is the positive BOWVs, called the positive group (corresponding to the positive sentences of the training data set)

We proposed the algorithm 15 to perform the Hadoop Map phase:

Input: the positive sentences of the training data set and AnOrderListOfTheBOWV

Output: one BOWV of the positive sentences of the training data set

Step 1: Input the positive sentences and AnOrderListOfTheBOWV into the Hadoop Map in the Cloudera system.

Step 2: Each sentences in the positive sentences, do repeat:

Step 3: One BOWV := the transforming one English sentence into one BOWV in Cloudera in Fig 7

Step 4: Return One BOWV ;

We built the algorithm 16 to implement the Hadoop Reduce phase

Input: one BOWV of the positive sentenceS of the training data set

Output: the positive BOWVs, called the positive group (corresponding to the positive sentences of the training data set)

Step 1: Receive one BOWV;

Step 2: Add this BOWV into PositiveGroup;

Step 3: Return PositiveGroup - the positive BOWVs, called the positive group (corresponding to the positive sentences of the training data set);

In Fig 10, we transfer the negative sentences of the training data set into the negative BOWVs (called the negative group of the training data set) in the distributed system. The stage includes two phases: the Hadoop Map (M) phase and the Hadoop Reduce (R) phase. The input of the Hadoop Map phase is the negative sentences of the training data set and AnOrderListOfTheBOWV. The output of the Hadoop Map phase is one BOWV of the negative sentences of the training data set. The input of the Hadoop Redude phase is the output of the Hadoop Map phase, thus, the input of the Hadoop Reduce phase is one BOWV of one sentence of the negative sentences of the training data set. The output of the Hadoop Reduce phase is the negative BOWVs, called the negative group (corresponding to the negative sentences of the training data set)

We proposed the algorithm 17 to perform the Hadoop Map phase

Input: the negative sentences of the training data set and AnOrderListOfTheBOWV

Output: one BOWV of the negative sentences of the training data set

Step 1: Input the negative sentences and AnOrderListOfTheBOWV into the Hadoop Map in the Cloudera system.

Step 2: Each sentences in the positive sentences, do repeat:

Step 3: One BOWV := the transforming one English sentence into one BOWV in Cloudera in Fig 7

Step 4: Return One BOWV ;

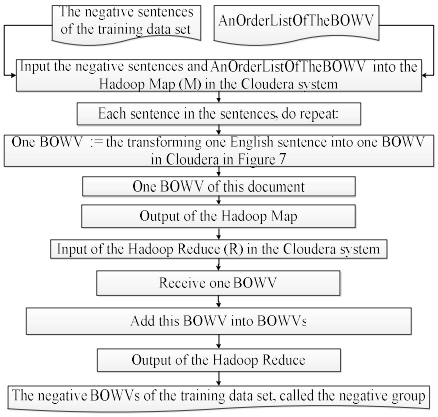

Fig. 10: Overview Of Transferring The Negative Sentences Of The Training Data Set Into The Negative Bowvs (Called The Negative Group Of The Training Data Set) In The Distributed System.

We proposed the algorithm 18 to implement the Hadoop Reduce phase

Input: one BOWV of the negative sentences of the training data set

Output: the negative BOWVs, called the negative group (corresponding to the negative sentences of the training data set)

Step 1: Receive one BOWV;

Step 2: Add this BOWV into NegativeGroup;

Step 3: Return NegativeGroup - the negative BOWVs, called the negative group (corresponding to the negative sentences of the training data set);

4.3 Using the KM to cluster the documents of the testing data set into either the positive or the negative in both a sequential environment and a parallel system

In Fig 11, we use the KM and the one-dimensional vectors to classify the documents of the testing data set into either polarity or the negative polarity in both a sequential environment and a distributed network environment as follows:

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

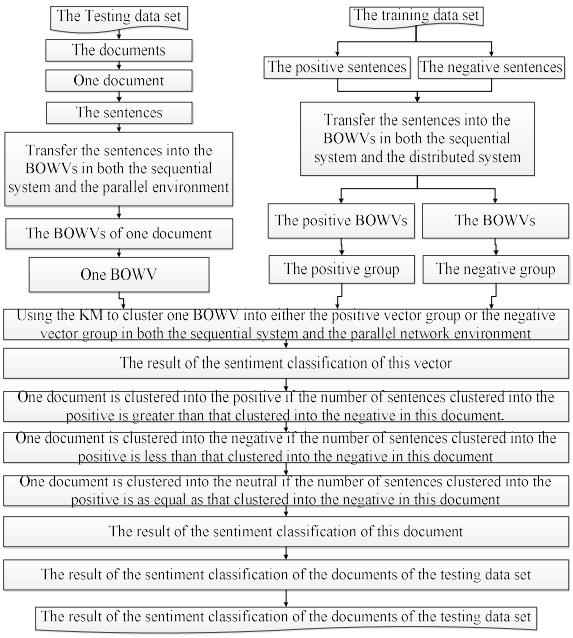

Fig. 11: Overview Of Using The KM And The Bowvs To Classify The Documents Of The Testing Data Set Into Either The Positive Polarity Or The Negative Polarity In Both The Sequential System And The Distributed Network Environment

This section includes two sub-sections as follows: (4.3.1) and (4.3.2).

4.3.1 Using the K-Means Algorithm to cluster the documents of the testing data set into either the positive or the negative in a sequential environment

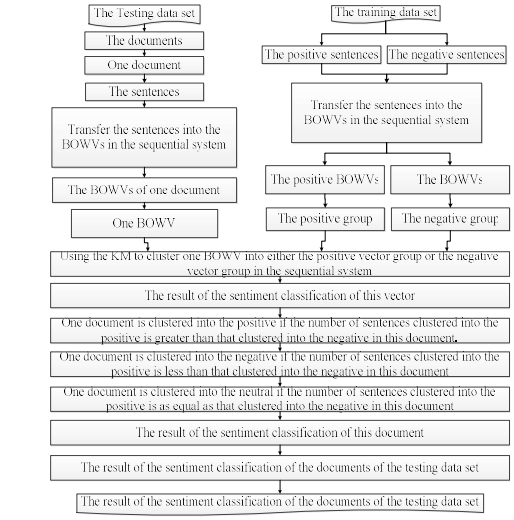

Fig 12: Overview Of Using The KM And The Bowvs To Classify The Documents Of The Testing Data Set Into Either The Positive Polarity Or The Negative Polarity In The Sequential System.

In Fig 12, we use the KM to classify the documents of the testing data set into either polarity or the negative polarity in a sequential environment

According to the surveys related the K-Means algorithm (KM) in [52-56], we proposed the algorithm 19 to use the KM to cluster one BOWV (corresponding one sentence of one document of the testing data set) into either the positive vector group or the negative vector group of the training data set in the sequential environment as follows:
Input: one BOWV of a document in the testing data set; the positive vector group and the negative vector group of the training data set.
Output: the result of clustering the vector into either the positive vector group or the negative vector group.
Step 1: Select randomly k centres (centroid) of k clusters. Each cluster is represented by the centre of this cluster.
Step 2: Calculate the distance between objects to k centres using Euclidean distance.
Step 3: Group objects into the closest group.
Step 4: Identify the new centre of the clusters.
Step 5: Repeat step 2 until no object groups change.
Step 6: Return the result of clustering the vector into either the positive vector group or the negative vector group.

We built the algorithm 20 to cluster one document of the testing data set into either the positive or the negative in the sequential system.
Input: one document of the testing data set; the positive group and the negative group of the training data set.
Output: The result of the sentiment classification of this document
Step 1: TheBOWVs := the algorithm 6 to transfer all the sentences of one document into the BOWVs in the sequential system with the input is this document;
Step 2: Set count_positive := 0; and count_negative := 0;
Step 3: Each BOWVs in TheBOWVs, do repeat:
Step 4: OneResult := the algorithm 19 to use the KM to cluster one BOWV (corresponding one sentence of one document of the testing data set) into either the positive vector group or the negative vector group of the training data set in the sequential environment
Step 5: If OneResult is the positive Then count_positive := count_positive + 1;
Step 6: Else If OneResult is the negative Then count_negative := count_negative + 1;
Step 7: End Repeat – End Step 3;

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Step 8: If count_positive is greater than count_negative Then Return positive;
Step 9: Else If count_positive is less than count_negative Then Return negative;
Step 10: Return neutral;

We built the algorithm 21 to cluster the documents of the testing data set into either the positive or the negative in the sequential environment.
Input: the documents of the testing data set and the training data set
Output: the results of the sentiment classification of the documents of the testing data set;
Step 1: the algorithm 4 to create an order list of the BOWV which comprises all the meaningful terms of both the testing data set and the training data set in the sequential system.
Step 2: the algorithm 7 to creat a positive group of the training data set from the positive sentences of the training data set in the sequential system
Step 3: the algorithm 8 to creat a negative group of the training data set from the negative sentences of the training data set in the sequential system
Step 4: Each document in the documents of the testing data set, do repeat:
Step 5: OneResult := the algorithm 20 to cluster one document of the testing data set into either the positive or the negative in the sequential system.
Step 6: Add OneResult into the results of the sentiment classification of the documents of the testing data set;
Step 7: Return the results of the sentiment classification of the documents of the testing data set;

4.3.2 Using the K-Means Algorithm to cluster the documents of the testing data set into either the positive or the negative in a distributed system

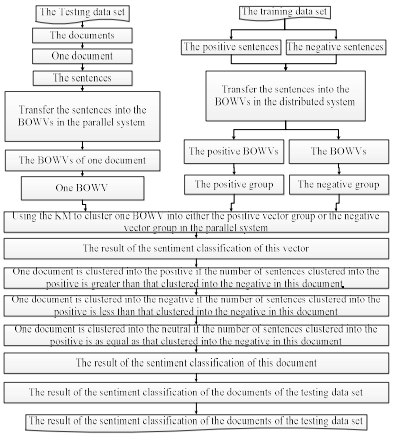

Fig. 13: Overview Of Using The KM And The Bowvs To Classify The Documents Of The Testing Data Set Into Either The Positive Polarity Or The Negative Polarity In The Distributed Network Environment

In Fig 13, we use the KM to classify the documents of the testing data set into either polarity or the negative polarity in a distributed network environment

In Fig 14, we use the KM to cluster one BOWV (corresponding one sentence of one document of the testing data set) into either the positive group or the negative group of the training data set in the parallel environment.

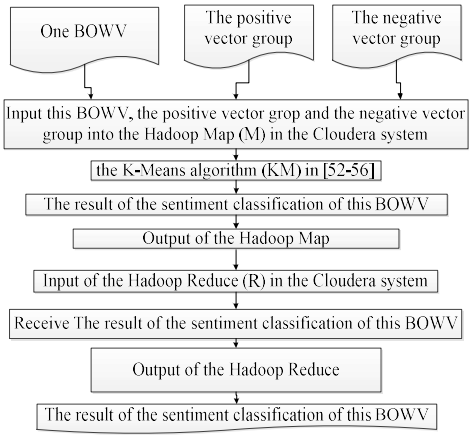

Fig. 14: Overview Of Using The KM To Cluster One BOWV (Corresponding One Sentence Of One Document Of The Testing Data Set) Into Either The Positive Vector Group Or The Negative Vector Group Of The Training Data Set In The Parallel Environment

This stage has two phases: the Hadoop Map phase and the Hadoop Reduce phase. The input of the Hadoop Map is one BOWV (corresponding one sentence of one document of the testing data set), the positive group and the negative group the training data set. The output of the Hadoop Map is the result of the sentiment classification of this BOWV. The input of the Hadoop Reduce is the output of the Hadoop Map, thus the input of the Hadoop Reduce is the result of the sentiment classification of this BOWV. The output of the Hadoop Reduce is the the result of the sentiment classification of this BOWV.

We proposed the algorithm 22 to perform the Hadoop Map phase
Input: one BOWV of a document in the testing data set; the positive vector group and the negative vector group of the training data set.
Output: the result of clustering the BOWV into either the positive vector group or the negative vector group.
Step 1: Input this BOWV, the positive vector group and the negative vector group into the Hadoop Map in the Cloudera system.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Step 2: Select randomly k centres (centroid) of k clusters. Each cluster is represented by the centre of this cluster.
Step 3: Calculate the distance between objects to k centres using Euclidean distance.
Step 4: Group objects into the closest group.
Step 5: Identify the new centre of the clusters.
Step 6: Repeat step 2 until no object groups change.
Step 7: Return the result of clustering the BOWV into either the positive vector group or the negative vector group;// the output of the Hadoop Map

We built the algorithm 23 to implement the Hadoop Reduce phase
Input: the result of clustering the BOWV into either the positive vector group or the negative vector group – the output of the Hadoop Map
Output: the result of clustering the BOWV into either the positive vector group or the negative vector group.
Step 1: Receive the result of clustering the BOWV into either the positive vector group or the negative vector group;
Step 2: Return the result of clustering the BOWV into either the positive vector group or the negative vector group;

In Fig 15, we use the KM and the one-dimensional vectors to cluster one document of the testing data set into either the positive or the negative in the distributed environment. The input of the Hadoop Map is one document of the testing data set, the positive group and the negative group of the training data set. The output of the Hadoop Map is the result of the sentiment classification of one BOWV (corresponding to one sentence of this document) into either the positive vector group or the negative vector group. The input of the Hadoop Reduce is the output of the Hadoop Map, thus, the input of the Hadoop Reduce is the result of the sentiment classification of one BOWV (corresponding to one sentence of this document) into either the positive vector group or the negative vector group. The output of the Hadoop Reduce is the result of the sentiment classification of this document.

We propose the algorithm 24 to perform the Hadoop Map phase of use the KM and the one-dimensional vectors to cluster one document of the testing data set into either the positive or the negative in the distributed environment. The main ideas of the algorithm 24 are as follows:
Input: one document of the testing data set; the positive vector group and the negative vector group of the training data set.
Output: the result of the sentiment classification of one one-dimensional vector (corresponding to one sentence of this document) into either the positive vector group or the negative vector group
Step 1: Input this document, the positive vector group and the negative vector group into the Hadoop Map in the Cloudera system.
Step 2: TheOne-dimensionalVectors := the of transferring all the sentences of one document of the testing data set into the one-dimensional vectors of the document of testing data set based on the sentiment lexicons of the bESD in the parallel network environment in Fig 8;
Step 3: Each one-dimensional vector in TheOne-dimensionalVectors, do repeat:
Step 4: OneResult := the using the KM to cluster one one-dimensional vector (corresponding one sentence of one document of the testing data set) into either the positive vector group or the negative vector group of the training data set int the parallel environment in Fig 15;
Step 5: Return OneResult; // the output of the Hadoop Map

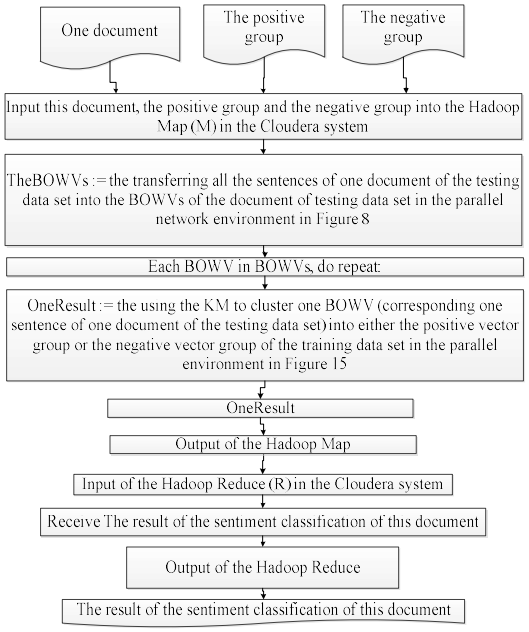

Fig. 15: Overview Of Using The KM To Cluster One Document Of The Testing Data Set Into Either The Positive Or The Negative In The Distributed Environment

We built the algorithm 25 to perform the Hadoop Reduce phase
Input: OneResult - the result of the sentiment classification of one one-dimensional vector (corresponding to one sentence of this document) into either the positive vector group or the negative vector group

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Output: the result of the sentiment classification of this document.
Step 1: Receive OneResult - the result of the sentiment classification of one one-dimensional vector (corresponding to one sentence of this document) into either the positive vector group or the negative vector group;
Step 2: Add OneResult into the result of the sentiment classification of this document;
Step 3: Return the result of the sentiment classification of this document;

In Fig 16, we use the KM to cluster the documents of the testing data set into either the positive or the negative in the parallel network environment. This stage comprises two phases: the Hadoop Map phase and the Hadoop Reduce phase. The input of the Hadoop Map is the documents of the testing data set and the training data set. The output of the Hadoop Map is the result of the sentiment classification of one document of the testing data set. The input of the Hadoop Reduce is the output of the Hadoop Map, thus, the input of the Hadoop Reduce is the result of the sentiment classification of one document of the testing data set. The output of the Hadoop Reduce is the results of the sentiment classification of the documents of the testing data set.

We built the algorithm 26 to implement the Hadoop Map phase
Input: the documents of the testing data set and the training data set
Output: the result of the sentiment classification of one document of the testing data set;
Step 1: the creating an order list of the BOWV which comprises all the meaningful terms of both the testing data set and the training data set in the distributed network system in Fig 8
Step 2: the transferring the positive sentences of the training data set into the positive BOWVs (called the positive group of the training data set) in the distributed system in Fig 9
Step 3: the transferring the negative sentences of the training data set into the negative BOWVs (called the negative group of the training data set) in the distributed system in Fig 10
Step 4: Input the documents of the testing data set, the positive group and the negative group into the Hadoop Map in the Cloudera system
Step 5: Each document in the documents of the testing data set, do repeat:
Step 6: OneResult := the using the KM to cluster one document of the testing data set into either the positive or the negative in the distributed environment in Fig 15 with the input is this document, the positive group and the negative group.
Step 7: Return OneResult - the result of the sentiment classification of one document of the testing data set;//the output of the Hadoop Map

We proposed the algorithm 27 to perform the Hadoop Reduce phase
Input: OneResult - the result of the sentiment classification of one document of the testing data set;//the output of the Hadoop Map
Output: the results of the sentiment classification of the documents of the testing data set;
Step 1: Receive OneResult ;
Step 2: Add OneResult into the results of the sentiment classification of the documents of the testing data set;
Step 3: Return the results of the sentiment classification of the documents of the testing data set;

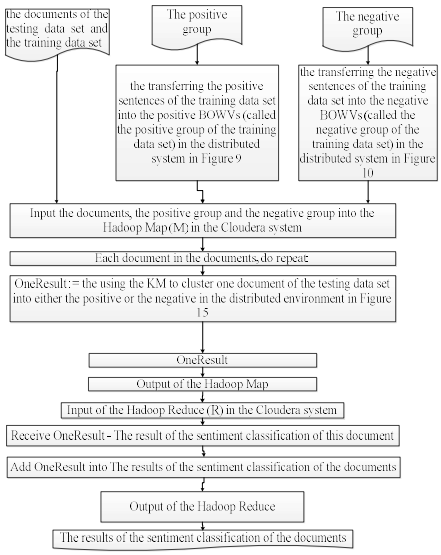

Fig. 16: Overview Of Using The KM To Cluster The Documents Of The Testing Data Set Into Either The Positive Or The Negative In The Parallel Network Environment.

5. EXPERIMENT

We have measured an Accuracy (A) to calculate the accuracy of the results of emotion classification. We used a Java programming language for programming to save data sets, implementing our proposed model to classify the 13,500,000 documents of the testing data set and the 5,000,000 sentences of the training data set. To implement the proposed model, we have already used the Java programming language to save the English testing data set and to save the results of emotion classification. The proposed model was

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

implemented in both the sequential system and the distributed network environment.

A novel model using many bag-of-words vectors (BOWV) and a SOKAL & SNEATH-IV Coefficient (SSIVC) for a K-Means algorithm (KM) is implemented in the sequential environment with the configuration as follows: The sequential environment in this research includes 1 node (1 server). The configuration of the server in the sequential environment is: Intel® Server Board S1200V3RPS, Intel® Pentium® Processor G3220 (3M Cache, 3.00 GHz), 2GB CC3-10600 ECC 1333 MHz LP Unbuffered DIMMs. The operating system of the server is: Cloudera. The Java language is used in programming the novel model using many bag-of-words vectors (BOWV) and a SOKAL & SNEATH-IV Coefficient (SSIVC) for a K-Means algorithm (KM)

The novel model using many bag-of-words vectors (BOWV) and a SOKAL & SNEATH-IV Coefficient (SSIVC) for a K-Means algorithm (KM) is performed in the Cloudera parallel network environment with the configuration as follows: This Cloudera system includes 9 nodes (9 servers). The configuration of each server in the Cloudera system is: Intel® Server Board S1200V3RPS, Intel® Pentium® Processor G3220 (3M Cache, 3.00 GHz), 2GB CC3-10600 ECC 1333 MHz LP Unbuffered DIMMs. The operating system of each server in the 9 servers is: Cloudera. All 9 nodes have the same configuration information. The Java language is used in programming the application of the novel model using many bag-of-words vectors (BOWV) and a SOKAL & SNEATH-IV Coefficient (SSIVC) for a K-Means algorithm (KM) in the Cloudera

In Table 1, we present the results of the documents in the testing data set and the accuracy of our novel model for the documents in the testing data set.

The average execution times of the classification of our new model for the documents in testing data set are shown in Table 2.

6. CONCLUSION

In this survey, a new model has been proposed to classify sentiment of many documents in English using many bag-of-words vectors (BOWV) and a SOKAL & SNEATH-IV Coefficient (SSIVC) for a K-Means algorithm (KM) with Hadoop Map (M) /Reduce (R) in the Cloudera parallel network environment. Based on our proposed new model, we have achieved 88.76% accuracy of the testing data set in Table 1. Until now, not many studies have shown that the clustering methods can be used to classify data. Our research shows that clustering methods are used to classify data and, in particular, can be used to classify the sentiments (positive, negative, or neutral) in text. The proposed model can be applied to other languages although our new model has been tested on our English data set. Our model can be applied to larger data sets with millions of English documents in the shortest time although our model has been tested on the documents of the testing data set in which the data sets are small in this survey.

According to Table 2, the average time of the sentiment classification of using the bag-of-words vectors (BOWV) and a SOKAL & SNEATH-IV Coefficient (SSIVC) for a K-Means algorithm (KM) in the sequential environment is 59,168,044 seconds/13,500,000 documents and it is greater than the average time of the sentiment classification of using the bag-of-words vectors (BOWV) and a SOKAL & SNEATH-IV Coefficient (SSIVC) for a K-Means algorithm (KM) in the Cloudera parallel network environment with 3 nodes which is 18,732,681 seconds/13,500,000 documents. The average time of the sentiment classification of using the bag-of-words vectors (BOWV) and a SOKAL & SNEATH-IV Coefficient (SSIVC) for a K-Means algorithm (KM) in the Cloudera parallel network environment with 9 nodes is 6,593,249 seconds/13,500,000 documents, and It is the shortest time in the table. Besides, the average time of the sentiment classification of using the bag-of-words vectors (BOWV) and a SOKAL & SNEATH-IV Coefficient (SSIVC) for a K-Means algorithm (KM) in the Cloudera parallel network environment with 6 nodes is 9,361,540 seconds/13,500,000 documents

The accuracy of the proposed model is dependent on many factors as follows:
(1)The KM – related algorithms
(2)The testing data set
(3)The documents of the testing data set and the sentences of the training data set must be standardized carefully.
(4)Transferring one sentence into one BOWV.
(5)The SSIVC – related algorithms

The execution time of the proposed model is dependent on many factors as follows:
(1)The parallel network environment such as the Cloudera system.
(2)The distributed functions such as Hadoop Map (M) and Hadoop Reduce (R).
(3)The KM – related algorithms
(4)The performance of the distributed network system.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

(5)The number of nodes of the parallel network environment.
(6)The performance of each node (each server) of the distributed environment.
(7)The sizes of the training data set and the testing data set.
(8)Transferring one sentence into one BOWV.
(9)The SSIVC – related algoritms.

The proposed model has many advantages and disadvantages. Its positives are as follows: It uses the bag-of-words vectors (BOWV) and a SOKAL & SNEATH-IV Coefficient (SSIVC) for a K-Means algorithm (KM) to classify semantics of English documents based on sentences. The proposed model can process millions of documents in the shortest time. This study can be performed in distributed systems to shorten the execution time of the proposed model. It can be applied to other languages. Its negatives are as follows: It has a low rate of accuracy. It costs too much and takes too much time to implement this proposed model.

To understand the scientific values of this research, we have compared our model's results with many studies in the tables below.

In Table 5, we present the comparisons of our model's benefits and drawbacks with the studies related to the K-Means algorithm (KM) in [52-56] .

The comparisons of our model's advantages and disadvantages with the works in [44-46] are shown in Table 6.

In Table 7, we display the comparisons of our model's benefits and drawbacks with the studies related to the bag-of-words (BOW) in [47-51]

The comparisons of our model's positives and negatives the latest sentiment classification models (or the latest sentiment classification methods) in [57-67] are presented in Table 8.

REFRENCES:

[1] Aleksander Bai, Hugo Hammer, "*Constructing sentiment lexicons in Norwegian from a large text corpus*", 2014 IEEE 17th International Conference on Computational Science and Engineering, 2014

[2] P.D.Turney, M.L.Littman, "*Unsupervised Learning of Semantic Orientation from a Hundred-Billion-Word Corpus*", arXiv:cs/0212012, Learning (cs.LG); Information Retrieval (cs.IR), 2002

[3] Robert Malouf, Tony Mullen, "*Graph-based user classification for informal online political discourse*", In proceedings of the 1st Workshop on Information Credibility on the Web, 2017

[4] Christian Scheible, "*Sentiment Translation through Lexicon Induction*", Proceedings of the ACL 2010 Student Research Workshop, Sweden, pp 25–30, 2010

[5] Dame Jovanoski, Veno Pachovski, Preslav Nakov, "*Sentiment Analysis in Twitter for Macedonian*", Proceedings of Recent Advances in Natural Language Processing, Bulgaria, pp 249–257, 2015

[6] Amal Htait, Sebastien Fournier, Patrice Bellot, "*LSIS at SemEval-2016 Task 7: Using Web Search Engines for English and Arabic Unsupervised Sentiment Intensity Prediction*", Proceedings of SemEval-2016, California, pp 481–485, 2016

[7] Xiaojun Wan, "*Co-Training for Cross-Lingual Sentiment Classification*", Proceedings of the 47th Annual Meeting of the ACL and the 4th IJCNLP of the AFNLP, Singapore, pp 235–243, 2009

[8] Julian Brooke, Milan Tofiloski, Maite Taboada, "*Cross-Linguistic Sentiment Analysis: From English to Spanish*", International Conference RANLP 2009 - Borovets, Bulgaria, pp 50–54, 2009

[9] Tao Jiang, Jing Jiang, Yugang Dai, Ailing Li, "*Micro–blog Emotion Orientation Analysis Algorithm Based on Tibetan and Chinese Mixed Text*", International Symposium on Social Science (ISSS 2015), 2015

[10] Tan, S.; Zhang, J., "*An empirical study of sentiment analysis for Chinese documents*", Expert Systems with Applications (2007), doi:10.1016/j.eswa.2007.05.028, 2007

[11] Weifu Du, Songbo Tan, Xueqi Cheng, Xiaochun Yun, "*Adapting Information Bottleneck Method for Automatic Construction of Domain-oriented Sentiment Lexicon*", WSDM'10, New York, USA, 2010

[12] Ziqing Zhang, Qiang Ye, Wenying Zheng, Yijun Li, "*Sentiment Classification for Consumer Word-of-Mouth in Chinese: Comparison between Supervised and Unsupervised Approaches*", The 2010 International Conference on E-Business Intelligence, 2010

[13] Guangwei Wang, Kenji Araki, "*Modifying SO-PMI for Japanese Weblog Opinion Mining by Using a Balancing Factor and Detecting Neutral Expressions*", Proceedings of NAACL HLT 2007, Companion Volume, NY, pp 189–192, 2007

[14] Shi Feng, Le Zhang, Binyang Li Daling Wang, Ge Yu, Kam-Fai Wong, "*Is Twitter A Better Corpus for Measuring Sentiment Similarity?* ",

Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, USA, pp 897–902, 2013

[15] Nguyen Thi Thu An, Masafumi Hagiwara, "*Adjective-Based Estimation of Short Sentence's Impression"*, (KEER2014) Proceedings of the 5th Kanesi Engineering and Emotion Research; International Conference; Sweden, 2014

[16] Nihalahmad R. Shikalgar, Arati M. Dixit, "*JIBCA: Jaccard Index based Clustering Algorithm for Mining Online Review*", International Journal of Computer Applications (0975 – 8887), Volume 105 – No. 15, 2014

[17] Xiang Ji, Soon Ae Chun, Zhi Wei, James Geller, "*Twitter sentiment classification for measuring public health concerns*", Soc. Netw. Anal. Min. (2015) 5:13, DOI 10.1007/s13278-015-0253-5, 2015

[18] Nazlia Omar, Mohammed Albared, Adel Qasem Al-Shabi, Tareg Al-Moslmi, "*Ensemble of Classification algorithms for Subjectivity and Sentiment Analysis of Arabic Customers' Reviews*", International Journal of Advancements in Computing Technology(IJACT), Volume5, 2013

[19] Huina Mao, Pengjie Gao, Yongxiang Wang, Johan Bollen, "*Automatic Construction of Financial Semantic Orientation Lexicon from Large-Scale Chinese News Corpus*", 7th Financial Risks International Forum, Institut Louis Bachelier, 2014

[20] Yong REN, Nobuhiro KAJI, Naoki YOSHINAGA, Masaru KITSUREGAW, "*Sentiment Classification in Under-Resourced Languages Using Graph-based Semi-supervised Learning Methods*", IEICE TRANS. INF. & SYST., VOL.E97–D, NO.4, DOI: 10.1587/transinf.E97.D.1, 2014

[21] Oded Netzer, Ronen Feldman, Jacob Goldenberg, Moshe Fresko, "*Mine Your Own Business: Market-Structure Surveillance Through Text Mining*", Marketing Science, Vol. 31, No. 3, pp 521-543, 2012

[22] Yong Ren, Nobuhiro Kaji, Naoki Yoshinaga, Masashi Toyoda, Masaru Kitsuregawa, "*Sentiment Classification in Resource-Scarce Languages by using Label Propagation*", Proceedings of the 25th Pacific Asia Conference on Language, Information and Computation, Institute of Digital Enhancement of Cognitive Processing, Waseda University, pp 420 – 429, 2011

[23] José Alfredo Hernández-Ugalde, Jorge Mora-Urpí, Oscar J. Rocha, "*Genetic relationships among wild and cultivated populations of peach palm (Bactris gasipaes Kunth, Palmae): evidence for multiple independent domestication events*", Genetic Resources and Crop Evolution, Volume 58, Issue 4, pp 571-583, 2011

[24] Julia V. Ponomarenko, Philip E. Bourne, Ilya N. Shindyalov, "*Building an automated classification of DNA-binding protein domains*", BIOINFORMATICS, Vol. 18, pp S192-S201, 2002

[25] Andréia da Silva Meyer, Antonio Augusto Franco Garcia, Anete Pereira de Souza, Cláudio Lopes de Souza Jr, "*Comparison of similarity coefficients used for cluster analysis with dominant markers in maize (Zea maysL)* ", Genetics and Molecular Biology, 27, 1, 83-91, 2004

[26] Snežana MLADENOVIĆ DRINIĆ, Ana NIKOLIĆ, Vesna PERIĆ, "*Cluster Analysis of Soybean Genotypes Based on RAPD Markers*", Proceedings. 43rd Croatian and 3rd International Symposium on Agriculture. Opatija. Croatia, 367- 370, 2008

[27] Tamás, Júlia; Podani, János; Csontos, Péter, "*An extension of presence/absence coefficients to abundance data:a new look at absence*", Journal of Vegetation Science 12: 401-410, 2001

[28] Vo Ngoc Phu, Vo Thi Ngoc Chau, Vo Thi Ngoc Tran, Nguyen Duy Dat, "*A Vietnamese adjective emotion dictionary based on exploitation of Vietnamese language characteristics*", International Journal of Artificial Intelligence Review (AIR), doi:10.1007/s10462-017-9538-6, 67 pages, 2017

[29] Vo Ngoc Phu, Vo Thi Ngoc Tran, "*Latent Semantic Analysis using A Dennis Coefficient for English Sentiment Classification in A Parallel System*", International Journal of Computers, Communications and Control ISSN 1841-9836, Volume 13, No 3, 390-410, 23 pages, 2018

[30] Vo Ngoc Phu, Vo Thi Ngoc Chau, Vo Thi Ngoc Tran, "*Shifting Semantic Values of English Phrases for Classification*", International Journal of Speech Technology (IJST), 10.1007/s10772-017-9420-6, 28 pages, 2017

[31] Vo Ngoc Phu, Vo Thi Ngoc Chau, Vo Thi Ngoc Tran, Nguy Duy Dat, Khanh Ly Doan Duy, "*A Valence-Totaling Model for Vietnamese Sentiment Classification",* International Journal of Evolving Systems (EVOS), DOI: 10.1007/s12530-017-9187-7, 47 pages, 2017

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

[32] Vo Ngoc Phu, Vo Thi Ngoc Chau, Vo Thi Ngoc Tran, Nguyen Duy Dat, Khanh Ly Doan Duy, "*Semantic Lexicons of English Nouns for Classification*", International Journal of Evolving Systems, DOI: 10.1007/s12530-017-9188-6, 69 pages, 2017

[33] English Dictionary of Lingoes, *http://www.lingoes.net/,* 2017

[34] Oxford English Dictionary, *http://www.oxforddictionaries.com/*, 2017

[35] Cambridge English Dictionary, *http://dictionary.cambridge.org/*, 2017

[36] Longman English Dictionary, *http://www.ldoceonline.com/*, 2017

[37] Collins English Dictionary, *http://www.collinsdictionary.com/dictionary/english*, 2017

[38] MacMillan English Dictionary, *http://www.macmillandictionary.com/*, 2017

[39] Seung-Seok Choi, Sung-Hyuk Cha, Charles C. Tappert, "*A Survey Of Binary Similarity And Distance Measures*", Systemics, Cybernetics And Informatics, Issn: 1690-4524, Volume 8 - Number 1, 2010

[40] Jair Moura Duarte, João Bosco dos Santos, Leonardo Cunha Melo, "*Comparison Of Similarity Coefficients Based On Rapd Markers In The Common Bean*", Genetics and Molecular Biology, 22, 3, 427-432, 1999

[41] Warrens, Matthijs Joost, "*Similarity Coefficients for Binary Data. Properties of Coefficients, Coefficient Matrices, Multi-way Metrics and Multivariate Coefficients*", Dissertation Leiden University – With References – With Summary in Dutch, ISBN 978-90-8891-0524, 2008

[42] Rodham E. Tulloss, "*Assessment of Similarity Indices for Undesirable Properties and a new Tripartite Similarity Index Based on Cost Functions*", Offprint from Palm, M. E. and I. H. Chapela, eds. 1997. MSSCology in Sustainable Development: Expanding Concepts, Vanishing Borders. (Parkway Publishers, Boone, North Carolina): 122-143, 1997

[43] T. Affolder et al., "*Double Diffraction Dissociation at the Fermilab Tevatron Collider*", Phys. Rev. Lett. 87, 141802, 2001

[44] Víctor Carrera-Trejo, Grigori Sidorov, Sabino Miranda-Jiménez, Marco Moreno Ibarra and Rodrigo Cadena Martínez, "*Latent Dirichlet Allocation complement in the vector space model for Multi-Label Text Classification*", International Journal of Combinatorial Optimization Problems and Informatics, Vol. 6, No. 1, pp. 7-19, 2015

[45] Pascal Soucy, Guy W. Mineau, "*Beyond TFIDF Weighting for Text Categorization in the Vector Space Model*", Proceedings of the 19th International Joint Conference on Artificial Intelligence, pp. 1130-1135, USA, 2015

[46] Pascal Soucy, Guy W. Mineau, "*Beyond TFIDF Weighting for Text Categorization in the Vector Space Model*", Proceedings of the 19th International Joint Conference on Artificial Intelligence, pp. 1130-1135, USA, 2015

[47] Hanna M. Wallach, "Topic modeling: beyond bag-of-words", ICML '06 Proceedings of the 23rd international conference on Machine learning , Pages 977-984 , Pittsburgh, Pennsylvania, USA, 2006

[48] Jun Yang, Yu-Gang Jiang, Alexander G. Hauptmann, Chong-Wah Ngo, "*Evaluating bag-of-visual-words representations in scene classification*", MIR '07 Proceedings of the international workshop on Workshop on multimedia information retrieval , Pages 197-206 , Augsburg, Bavaria, Germany, 2007

[49] Bharath Sriram, Dave Fuhry, Engin Demir, Hakan Ferhatosmanoglu, Murat Demirbas, "*Short text classification in twitter to improve information filtering*", SIGIR '10 Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval, Pages 841-842 , Geneva, Switzerland, 2010

[50] Lei Wu; Steven C. H. Hoi; Nenghai Yu, "*Semantics-Preserving Bag-of-Words Models and Applications*", IEEE Transactions on Image Processing, Volume: 19, Issue: 7, DOI: 10.1109/TIP.2010.2045169, 2010

[51] J. R. R. Uijlings, A. W. M. Smeulders, R. J. H. Scha, "*Real-time bag of words, approximately*", CIVR '09 Proceedings of the ACM International Conference on Image and Video Retrieval, Article No. 6 , Santorini, Fira, Greece, 2009

[52] K. Krishna; M. Narasimha Murty, "*Genetic K-means algorithm*", IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics) , Volume: 29, Issue: 3, DOI: 10.1109/3477.764879, 1999

[53] Zhexue Huang, "*Extensions to the k-Means Algorithm for Clustering Large Data Sets with Categorical Values*", Data Mining and Knowledge Discovery, Volume 2, Issue 3, pp 283–304, 1998

[54] Liping Jing; Michael K. Ng; Joshua Zhexue Huang, "*An Entropy Weighting k-Means Algorithm for Subspace Clustering of High-Dimensional Sparse Data*", IEEE Transactions on Knowledge and Data Engineering, Volume:

19, Issue: 8, DOI: 10.1109/TKDE.2007.1048, 2007

[55] J.M Peña 1, J.A Lozano, P Larrañaga, "*An empirical comparison of four initialization methods for the K-Means algorithm*", Pattern Recognition Letters, Volume 20, Issue 10, Pages 1027-1040, https://doi.org/10.1016/S0167-8655(99)00069-0, 1999

[56] B.-H. Juang; L.R. Rabiner, "*The segmental K-means algorithm for estimating parameters of hidden Markov models"*, IEEE Transactions on Acoustics, Speech, and Signal Processing, Volume: 38, Issue: 9, DOI: 10.1109/29.60082,1999

[57] Basant Agarwal, Namita Mittal, "*Machine Learning Approach for Sentiment Analysis*", Prominent Feature Extraction for Sentiment Analysis, Print ISBN 978-3-319-25341-1, 10.1007/978-3-319-25343-5_3, 21-45,2014

[58] Basant Agarwal, Namita Mittal, "*Semantic Orientation-Based Approach for Sentiment Analysis*", Prominent Feature Extraction for Sentiment Analysis, Print ISBN 978-3-319-25341-1, 10.1007/978-3-319-25343-5_6, 77-88, 2016

[59] Sérgio Canuto, Marcos André, Gonçalves, Fabrício Benevenuto, "*Exploiting New Sentiment-Based Meta-level Features for Effective Sentiment Analysis*", Proceedings of the Ninth ACM International Conference on Web Search and Data Mining (WSDM '16), 53-62, New York USA, 2016

[60] Shoiab Ahmed, Ajit Danti, "*Effective Sentimental Analysis and Opinion Mining of Web Reviews Using Rule Based Classifiers*", Computational Intelligence in Data Mining, Volume 1, Print ISBN 978-81-322-2732-8, DOI 10.1007/978-81-322-2734-2_18, 171-179, India, 2016

[61] Vo Ngoc Phu, Phan Thi Tuoi, "*Sentiment classification using Enhanced Contextual Valence Shifters*", International Conference on Asian Language Processing (IALP), 224-229, 2014

[62] Vo Thi Ngoc Tran, Vo Ngoc Phu and Phan Thi Tuoi, "*Learning More Chi Square Feature Selection to Improve the Fastest and Most Accurate Sentiment Classification*", The Third Asian Conference on Information Systems (ACIS 2014), 2014

[63] Nguyen Duy Dat, Vo Ngoc Phu, Vo Thi Ngoc Tran, Vo Thi Ngoc Chau, "*STING Algorithm used English Sentiment Classification in A Parallel Environment*", International Journal of Pattern Recognition and Artificial Intelligence, January 2017.

[64] Vo Ngoc Phu, Nguyen Duy Dat, Vo Thi Ngoc Tran, Vo Thi Ngoc Tran, "*Fuzzy C-Means for English Sentiment Classification in a Distributed System*", International Journal of Applied Intelligence (APIN), DOI: 10.1007/s10489-016-0858-z, 1-22, November 2016.

[65] Vo Ngoc Phu, Chau Vo Thi Ngoc, Tran Vo Thi Ngoc, Dat Nguyen Duy, "*A C4.5 algorithm for english emotional classification*", Evolving Systems, pp 1-27, doi:10.1007/s12530-017-9180-1, April 2017.

[66] Vo Ngoc Phu, Vo Thi Ngoc Chau, Vo Thi Ngoc Tran, "*SVM for English Semantic Classification in Parallel Environment*", International Journal of Speech Technology (IJST), 10.1007/s10772-017-9421-5, 31 pages, May 2017.

[67] Vo Ngoc Phu, Vo Thi Ngoc Tran, Vo Thi Ngoc Chau, Nguyen Duy Dat, Khanh Ly Doan Duy, "*A Decision Tree using ID3 Algorithm for English Semantic Analysis*", International Journal of Speech Technology (IJST), DOI: 10.1007/s10772-017-9429-x, 23 pages, 2017

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

APPENDICES:

Table 1: The results of the documents in the testing data set and the accuracy of our novel model for the documents in the testing data set

	Testing Dataset	Correct Classification	Incorrect Classification	Accuracy
Negative	6,750,000	5,991,676	758,324	88.76%
Positive	6,750,000	5,990,924	759,076	
Summary	13,500,000	11,982,600	1,517,400	

Table 2: The average execution times of the classification of our new model for the documents in testing data set.

	Average time of the classification / 13,500,000 documents.
The novel model in the sequential environment	59,168,044 seconds
The novel model in the Cloudera distributed system with 3 nodes	18,732,681 seconds
The novel model in the Cloudera distributed system with 6 nodes	9,361,540 seconds
The novel model in the Cloudera distributed system with 9 nodes	6,593,249 seconds

Table 3: Comparisons of our model's advantages and disadvantages with the works related to [1-32].

Surveys	Approach	Advantages	Disadvantages
[1]	Constructing sentiment lexicons in Norwegian from a large text corpus	Through the authors' PMI computations in this survey they used a distance of 100 words from the seed word, but it might be that other lengths that generate better sentiment lexicons. Some of the authors' preliminary research showed that 100 gave a better result.	The authors need to investigate this more closely to find the optimal distance. Another factor that has not been investigated much in the literature is the selection of seed words. Since they are the basis for PMI calculation, it might be a lot to gain by finding better seed words. The authors would like to explore the impact that different approaches to seed word selection have on the performance of the developed sentiment lexicons.
[2]	Unsupervised Learning of Semantic Orientation from a Hundred-Billion-Word Corpus.	This survey has presented a general strategy for learning semantic orientation from semantic association, SO-A. Two instances of this strategy have been empirically evaluated, SO-PMI-IR and SO-LSA. The accuracy of SO-PMI-IR is comparable to the accuracy of HM, the algorithm of Hatzivassiloglou and SSIVCKeown (1997). SO-PMI-IR requires a large corpus, but it is simple, easy to implement, unsupervised, and it is not restricted to adjectives.	No Mention
[3]	Graph-based	The authors describe several experiments	There is still much

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

	user classification for informal online political discourse	in identifying the political orientation of posters in an informal environment. The authors' results indicate that the most promising approach is to augment text classification methods by exploiting information about how posters interact with each other	left to investigate in terms of optimizing the linguistic analysis, beginning with spelling correction and working up to shallow parsing and co-reference identification. Likewise, it will also be worthwhile to further investigate exploiting sentiment values of phrases and clauses, taking cues from methods
[4]	A novel, graph-based approach using SimRank.	The authors presented a novel approach to the translation of sentiment information that outperforms SOPMI, an established method. In particular, the authors could show that SimRank outperforms SO-PMI for values of the threshold x in an interval that most likely leads to the correct separation of positive, neutral, and negative adjectives.	The authors' future work will include a further examination of the merits of its application for knowledge-sparse languages.
Our work	We have proposed a novel model using many bag-of-words vectors (BOWV) and a SOKAL & SNEATH-IV Coefficient (SSIVC) for a K-Means algorithm (KM) to classify all the documents of the testing data set into either the positive sentences or the negative sentences of our training data set in English. The advantages and disadvantages of this survey are shown in the Conclusion section.		

Table 4: Comparisons of our model's benefits and drawbacks with the studies related to the SOKAL & SNEATH-IV coefficient (SSIVC) in [39-43]

Surveys	Approach	Benefits	Drawbacks
[39]	A Survey of Binary Similarity and Distance Measures	Applying appropriate measures results in more accurate data analysis. Notwithstanding, few comprehensive surveys on binary measures have been conducted. Hence the authors collected 76 binary similarityand distance measures used over the last century and reveal their correlations through the hierarchical clustering technique	No mention
[40]	Comparison of similarity oefficients based on rapdmarkers in the common bean	The employment of different similarity coefficients caused few alterationsin cultivar classification, since correlations among genetic distances were larger than 0.86. Nevertheless, the different similarity	No mention

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

		coefficients altered the projection efficiency in a two-dimensional space and formed different numbers of groups by Tocher'soptimization procedure. Among these coefficients, Russel and Rao's was the most discordant and the Sorensen-Dice wasconsidered the most adequate due to a higher projection efficiency in a two-dimensional space. Even though few structuralchanges were suggested in the most different groups, these coefficients altered some relationships between cultivars withhigh genetic similarity.	
[41]	Similarity Coefficients for Binary Data	The authors show how to use Similarity Coefficients for Binary Data	No mention
[42]	Assessment of Similarity Indices for Undesirable Properties and a new Tripartite Similarity Index Based on Cost Functions	The purpose of this study is to motivate, describe, and offer animplementation for, a working similarity index that avoids the difficultiesnoted for the others.	No mention
[43]	Double Diffraction Dissociation at the Fermilab Tevatron Collider	The authors' results are compared with previous measurements and with predictions based on Regge theory and factorization	No mention
Our work	We have proposed a novel model using many bag-of-words vectors (BOWV) and a SOKAL & SNEATH-IV Coefficient (SSIVC) for a K-Means algorithm (KM) to classify all the documents of the testing data set into either the positive sentences or the negative sentences of our training data set in English. The advantages and disadvantages of this survey are shown in the Conclusion section.		

Table 5: Comparisons of our model's benefits and drawbacks with the studies related to the K-Means algorithm (KM) in [52-56]

Surveys	Approach	Benefits	Drawbacks
[52]	Genetic K-means algorithm	The authors define K-means operator, one-step of K-means algorithm, and use it in GKA as a search operator instead of crossover. The authors also define a biased mutation operator specific to clustering called distance-based-mutation. Using finite Markov chain theory, the authors prove that the GKA converges to the global optimum. It is observed in the simulations that GKA converges to the best known optimum corresponding to the given data in concurrence with the convergence result. It is also observed that GKA searches faster than some of the other evolutionary	No mention

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

		algorithms used for clustering.	
[53]	Extensions to the k-Means Algorithm for Clustering Large Data Sets with Categorical Values	The authors use the well known soybean disease and credit approval data sets to demonstrate the clustering performance of the two algorithms. The authors' experiments on two real world data sets with half a million objects each show that the two algorithms are efficient when clustering large data sets, which is critical to data mining applications.	No mention
[54]	An Entropy Weighting k-Means Algorithm for Subspace Clustering of High-Dimensional Sparse Data	In the new algorithm, the authors extend the k-means clustering process to calculate a weight for each dimension in each cluster and use the weight values to identify the subsets of important dimensions that categorize different clusters. This is achieved by including the weight entropy in the objective function that is minimized in the k-means clustering process. An additional step is added to the k-means clustering process to automatically compute the weights of all dimensions in each cluster. The experiments on both synthetic and real data have shown that the new algorithm can generate better clustering results than other subspace clustering algorithms. The new algorithm is also scalable to large data sets.	No mention
[55]	An empirical comparison of four initialization methods for the K-Means algorithm	The results of the authors' experiments illustrate that the random and the Kaufman initialization methods outperform the rest of the compared methods as they make the K-Means more effective and more independent on initial clustering and on instance order. In addition, the authors compare the convergence speed of the K-Means algorithm when using each of the four initialization methods. The authors' results suggest that the Kaufman initialization method induces to the K-Means algorithm a more desirable behaviour with respect to the convergence speed than the random initialization method.	No mention
[56]	The segmental K-means algorithm for estimating parameters of hidden Markov models	The authors prove the convergence of the algorithm and compare it with the traditional Baum-Welch reestimation method. They also print out the increased flexibility this algorithm offers in the general speech modeling framework	No mention
Our work	We have proposed a novel model using many bag-of-words vectors (BOWV) and a SOKAL & SNEATH-IV Coefficient (SSIVC) for a K-Means algorithm (KM) to classify all the documents of the testing data set into either the positive sentences or the negative sentences of our training data set in English. The advantages and disadvantages of this survey are shown in the Conclusion section.		

Table 6: Comparisons of our model's advantages and disadvantages with the works in [44-46]

Researches	Approach	Advantages	Disadvantages
[44]	Examining the vector space model, an information retrieval technique and its variation	In this work, the authors have given an insider to the working of vector space model techniques used for efficient retrieval techniques. It is the bare fact that each system has its own strengths and weaknesses. What we have sorted out in the authors' work for	The draw backs are that the system yields no theoretical findings.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

		vector space modeling is that the model is easy to understand and cheaper to implement, considering the fact that the system should be cost effective (i.e., should follow the space/time constraint. It is also very popular. Although the system has all these properties, it is facing some major drawbacks.	Weights associated with the vectors are very arbitrary, and this system is an independent system, thus requiring separate attention. Though it is a promising technique, the current level of success of the vector space model techniques used for information retrieval are not able to satisfy user needs and need extensive attention.
[45]	+Latent Dirichlet allocation (LDA). +Multi-label text classification tasks and apply various feature sets. +Several combinations of features, like bi-grams and uni-grams.	In this work, the authors consider multi-label text classification tasks and apply various feature sets. The authors consider a subset of multi-labeled files of the Reuters-21578 corpus. The authors use traditional TF-IDF values of the features and tried both considering and ignoring stop words. The authors also tried several combinations of features, like bi-grams and uni-grams. The authors also experimented with adding LDA results into vector space models as new features. These last experiments obtained the best results.	No mention
[46]	The K-Nearest Neighbors algorithm for English sentiment classification in the Cloudera distributed system.	In this study, the authors introduce a new weighting method based on statistical estimation of the importance of a word for a specific categorization problem. One benefit of this method is that it can make feature selection implicit, since useless features of the categorization problem considered get a very small weight. Extensive experiments reported in the work show that this new weighting method improves significantly the classification	Despite positive results in some settings, Gain Ratio failed to show that supervised weighting methods are gener

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

		accuracy as measured on many categorization tasks.	ally higher than unsupervised ones. The authors believe that ConfWeight is a promising supervised weighting technique that behaves gracefully both with and without feature selection. Therefore, the authors advocate its use in further experiments.
Our work	We have proposed a novel model using many bag-of-words vectors (BOWV) and a SOKAL & SNEATH-IV Coefficient (SSIVC) for a K-Means algorithm (KM) to classify all the documents of the testing data set into either the positive sentences or the negative sentences of our training data set in English. The advantages and disadvantages of the proposed model are shown in the Conclusion section.		

Table 7: Comparisons of our model's benefits and drawbacks with the studies related to the bag-of-words (BOW) in [47-51]

Surveys	Approach	Benefits	Drawbacks
[47]	Topic modeling: beyond bag-of-words	The model hyperparameters are inferred using a Gibbs EM algorithm. On two data sets, each of 150 documents, the new model exhibits better predictive accuracy than either a hierarchical Dirichlet bigram language model or a unigram topic model. Additionally, the inferred topics are less dominated by function words than are topics discovered using unigram statistics, potentially making them more meaningful.	No mention
[48]	Evaluating bag-of-visual-words representations in scene classification	Given the analogy between this representation and the bag-of-words representation of text documents, we apply techniques used in text categorization, including term weighting, stop word removal, feature selection, to generate image representations that differ in the dimension, selection, and weighting of visual words. The impact of these representation choices to scene classification is studied through extensive experiments on the TRECVID and PASCAL collection. This study provides an empirical basis for designing visual-word	No mention

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

		representations that are likely to produce superior classification performance	
[49]	Short text classification in twitter to improve information filtering	As short texts do not provide sufficient word occurrences, traditional classification methods such as "Bag-Of-Words" have limitations. To address this problem, we propose to use a small set of domain-specific features extracted from the author's profile and text. The proposed approach effectively classifies the text to a predefined set of generic classes such as News, Events, Opinions, Deals, and Private Messages.	No mention
Our work	We have proposed a novel model using many bag-of-words vectors (BOWV) and a SOKAL & SNEATH-IV Coefficient (SSIVC) for a K-Means algorithm (KM) to classify all the documents of the testing data set into either the positive sentences or the negative sentences of our training data set in English. The advantages and disadvantages of this survey are shown in the Conclusion section.		

Table 8: Comparisons of our model's positives and negatives the latest sentiment classification models (or the latest sentiment classification methods) in [57-67]

Studies	Approach	Positives	Negatives
[57]	The Machine Learning Approaches Applied to Sentiment Analysis -Based Applications	The main emphasis of this survey is to discuss the research involved in applying machine learning methods, mostly for sentiment classification at document level. Machine learning-based approaches work in the following phases, which are discussed in detail in this work for sentiment classification: (1) feature extraction, (2) feature weighting schemes, (3) feature selection, and (4) machine-learning methods. This study also discusses the standard free benchmark datasets and evaluation methods for sentiment analysis. The authors conclude the research with a comparative study of some state-of-the-art methods for sentiment analysis and some possible future research directions in opinion mining and sentiment analysis.	No mention
[58]	Semantic Orientation-Based Approach for Sentiment Analysis	This approach initially mines sentiment-bearing terms from the unstructured text and further computes the polarity of the terms. Most of the sentiment-bearing terms are multi-word features unlike bag-of-words, e.g., "good movie," "nice cinematography," "nice actors," etc. Performance of semantic orientation-based approach has been limited in the literature due to inadequate coverage of multi-word features.	No mention
[59]	Exploiting New Sentiment-Based Meta-Level Features for Effective Sentiment Analysis	Experiments performed with a substantial number of datasets (nineteen) demonstrate that the effectiveness of the proposed sentiment-based meta-level features is not only superior to the traditional bag-of-words representation (by up to 16%) but also is also superior in most cases to state-of-art meta-level features previously proposed in the literature for text classification tasks that	A line of future research would be to explore the authors' meta features with other

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

		do not take into account any idiosyncrasies of sentiment analysis. The authors' proposal is also largely superior to the best lexicon-based methods as well as to supervised combinations of them. In fact, the proposed approach is the only one to produce the best results in all tested datasets in all scenarios.	classification algorithms and feature selection techniques in different sentiment analysis tasks such as scoring movies or products according to their related reviews.
Our work	We have proposed a novel model using many bag-of-words vectors (BOWV) and a SOKAL & SNEATH-IV Coefficient (SSIVC) for a K-Means algorithm (KM) to classify all the documents of the testing data set into either the positive sentences or the negative sentences of our training data set in English. The positives and negatives of the proposed model are given in the Conclusion section.		