Journal of Theoretical and Applied Information Technology
15th June 2018. Vol.96. No 11

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

A FUZZY C-MEANS ALGORITHM AND SENTIMENT-LEXICONS-BASED MULTI-DIMENSIONAL VECTORS OF A SOKAL & SNEATH-IV COEFFICIENT USED FOR ENGLISH SENTIMENT CLASSIFICATION

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

Sentiment classification has long been the subject of research and there are many applications and many studies to service communities, commerce, politics, etc. In this research, we have proposed a new model for Big Data sentiment classification in the parallel network environment – a Cloudera system with Hadoop Map (M) and Hadoop Reduce (R). Our new model has used a Fuzzy C-Means Algorithm (FCM) with sentiment-lexicons-based multi-dimensional vectors and 3,000,000 documents of our training data set for document-level sentiment classification in English. First, we calculate the sentiment scores of English terms (verbs, nouns, adjectives, adverbs, etc.) by using a SOKAL & SNEATH-IV coefficient (SSIVC) through a Google search engine with AND operator and OR operator. Then, we transfer all the documents of both the testing data set and the training data set into many multi-dimensional vectors which are identified by using the sentiment lexicons. Finally, we implement the proposed model in both a sequential environment and a distributed system. Our new model can classify sentiment of millions of English documents based on many English documents in the parallel network environment. However, we tested our new model on our testing data set (including 5,500,000 English reviews, 2,750,000 positive and 2,750,000 negative) and achieved 87.82% accuracy. The results of this work can be widely used in applications and research of the English sentiment classification.

Keywords: *English sentiment dictionary; sentiment lexicons; English sentiment classification; Fuzzy C-Means; FCM; Cloudera; Hadoop Map; Hadoop Reduce; SOKAL & SNEATH-IV coefficient.*

1. INTRODUCTION

Clustering data is to process a set of objects into classes of similar objects. 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.

To implement our new model, we propose the following basic principles:

(1)Assuming that each English sentence has m English words (or English phrases).

(2)Assuming that the maximum number of one English sentence is m_max; it means that m is less than m_max or m is equal to m_max.

(3)Assuming that each English document has n English sentences.

(4)Assuming that the maximum number of one English document is n_max; it means that n is less than n_max or n is equal to n_max.

(5)Each English sentence is transferred into one vector (one-dimensional). Thus, the length of the vector is m. If m is less than m_max then each element of the vector from m to m_max-1 is 0 (zero).

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

(6)Each English document is transferred into one multi-dimensional vector. Therefore, the multi-dimensional vector has n rows and m columns. If n is less than n_max then each element of the multi-dimensional vector from n to n_max-1 is 0 (zero vector).

(7)All the documents of the English training data set are transferred into the multi-dimensional vectors which are based all the English sentiment lexicons. The positive English documents of the English training data set are transferred into the positive multi-dimensional vectors based on all the sentiment lexicons, the positive vector group. The negative English documents of the English training data set are transferred into the negative multi-dimensional vectors, the negative vector group.

(8)All English documents of the English testing data set are transferred into the multi-dimensional vectors based on all the sentiment lexicons. (9)One multi-dimensional vector (corresponding to one English document in the English testing data set) is the positive polarity if the vector is clustered into the positive vector group. One multi-dimensional vector (corresponding to one English document in the English testing data set) is the negative polarity if the vector is clustered into the negative vector group. One multi-dimensional vector (corresponding to one English document in the English testing data set) is the neutral polarity if the vector is not clustered into either the positive vector group or the negative vector group.

In this study, we propose a novel model by using the Fuzzy C-Means Algorithm (FCM) with sentiment-lexicons-based multi-dimensional vectors to classify emotions (positive, negative, neutral) of English documents in the parallel system.

First of all, we calculate the valences of the sentiment lexicons by using a SOKAL & SNEATH-IV coefficient (SSIVC) through a Google search engine with AND operator and OR operator. Based on Cosine, Ochiai, Sorensen, Tominato, Pointwise Mutual Information (PMI) and Jaccard measures, we built many equations related to SSIVC to calculate the valence of English terms. Methods based on sentiment lexicons relate to extraction and emotional score collection of terms, which are offered by lexicons to perform a prediction of emotions. This identifying sentiment scores of the sentiment lexicons was implemented in both a sequential system and a parallel environment. Next, we transfer all the document of both the testing data set and the training data set into the multi-dimensional vectors based on the sentiment lexicons above. This was implemented in both a sequential environment and a distributed system. Finally, we use the SSIVC to cluster one multi-dimentionsal vector (corresponding one document of the testing data set) into the positive vector group or the negative vector group. This was implemented in both a sequential system and a parallel environment.

The motivation of the work is as follows. Cosine, Ochiai, Sorensen, Tanimoto, PMI and Jaccard measures are used popularly to calculate the emotional values of the words. Thus, other similar measures can be used to identify the semantic scores of the words. Many algorithms in the data mining field can be applied to natural language processing, specifically sentiment classification for processing millions of English documents. This will result in many discoveries in scientific research, hence the motivation for this study.

The novelty of the proposed approach is as follows: a Fuzzy C-Means Algorithm (FCM) from the data mining field is applied to sentiment analysis. A Fuzzy C-Means Algorithm is applied to classify semantics of English documents based on many sentences. This algorithm can also be applied to identify the emotions of millions of documents. These above principles are proposed to classify the semantics of a document, and data mining is used in natural language processing. This survey can be applied to other parallel network systems. Hadoop Map (M) and Hadoop Reduce (R) are used in the proposed model. SOKAL & SNEATH-IV coefficient (SSIVC) is used in identifying the semantic values of the English words and phrases. Therefore, we will study this model in more detail.

The most significant contributions of our proposed model are displayed briefly as follows:

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

(2)SSIVC is used in identifying opinion scores of the English verb phrases and words through the Google search on the internet.

(3)The formulas are proposed in the paper.

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

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

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

(7)Many crucial contributions are listed in the Future Work section.

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

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

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

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

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

(12)The principles are proposed in the research.

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

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

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

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

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

(18)The FCM – related algorithms are proposed in this research.

This survey contains 6 sections. Section 1 introduces the study; Section 2 discusses the related works about the Fuzzy C-Means Algorithm (FCM), 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 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 generate several Norwegian sentiment lexicons by extracting sentiment information from two different types of Norwegian text corpus, namely, news corpus and discussion forums. The methodology is based on the Point wise Mutual Information (PMI), etc. The surveys related 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 [39-46].The survey in [40] catalogues the procedures and steps involved in agrodlimatic classification. These vary from conventional descriptive methods to modern computer-based numerical tecliniques, etc.

There are the surveys related to vector space modeling in [47-49]. We transfer all English sentences into many vectors, which are used in the VSM algorithm. In this research [47], the authors examine the vector space model, an information retrieval technique, and its variation. The rapid growth of the Internet and the abundance of documents and different forms of information available under-scores the need for good information retrieval technique. The vector space model is an algebraic model used for information retrieval, etc.

The research projects related to implementing algorithms, applications, studies in parallel network environment in [50, 51, 52]. In [50, 51], Hadoop is an Apache-based framework used to handle large data sets on clusters consisting of multiple computers, using the Map and Reduce programming model. The two main projects of the Hadoop are Hadoop Distributed File System (HDFS) and Hadoop M/R (Hadoop Map/Reduce). Hadoop M/R allows engineers to program for writing applications for parallel processing of large datasets on clusters consisting of multiple computers. A M/R task has two main components: (1) Map and (2) Reduce. This framework splits inputting data into chunks which multiple Map

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

tasks can handle with a separate data partition in parallel. The outputs of the map tasks are gathered and processed by the Reduce task ordered, etc. Cloudera [52], the global provider of the fastest, easiest, and most secure data management and analytics platform built on Apache™ Hadoop® and the latest open source technologies, announced today that it will submit proposals for Impala and Kudu to join the Apache Software Foundation (ASF). By donating its leading analytic database and columnar storage projects to the ASF, Cloudera aims to accelerate the growth and diversity of their respective developer communities, etc.

There are the works related to the Fuzzy C-Means algorithm (FCM) in [53-67]. This survey in [53] transmits a FORTRAN-IV coding of the fuzzy c-means (FCM) clustering program. The FCM program is applicable to a wide variety of geo-statistical data analysis problems. This program generates fuzzy partitions and prototypes for any set of numerical data, etc.

The latest researches of the sentiment classification are [68-78]. In the research [68], the authors present their machine learning experiments with regard to sentiment analysis in blog, review and forum texts found on the World Wide Web and written in English, Dutch and French, etc.

3. DATA SET

In Fig 1 below, the training data set includes 3,000,000 documents in the movie field, which contains 1,500,000 positive documents and 1,000,000 negative documents in English. All the documents in our training data set are automatically extracted from English Facebook, English websites and social networks; then we labeled positive and negative for them.

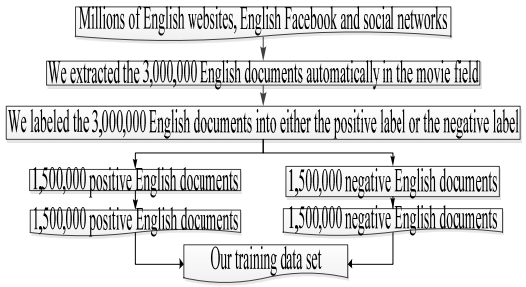

Fig. 1: Our training data set.

In Fig 2 below, the testing data set includes 5,500,000 documents in the movie field, which contains 2,750,000 positive documents and 2,750,000 negative documents in English. All the documents in our testing data set are automatically extracted from English Facebook, English websites and social networks; then we labeled positive and negative for them.

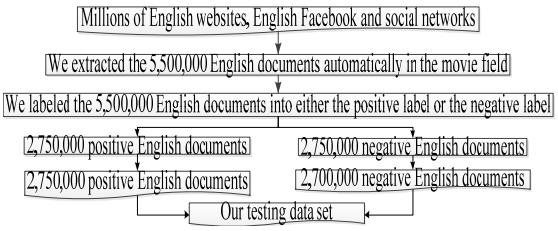

Fig. 2: Our English testing data set.

4. METHODOLOGY

This section comprises two parts. The first part is to transfer all the documents of both the testing data set and the training data set into the multi-dimensional vectors based on the sentiment lexicons in both the sequential environment and the distributed system in the sub-section (4.1). The second part is to cluster one multi-dimensional vectors (corresponding to one document of the testing data set) in both the senqential system and the parallel environment in the sub-section (4.2)

4.1 Tranferring the documents into the multi-dimensional vectors based on the sentiment lexicons

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

4.1.1 Calculating the valence of the sentiment lexicons

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, as the following diagram in Fig 3 below shows.

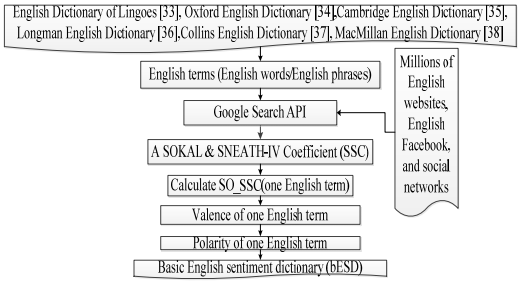

Fig. 3: Overview of English sentiment dictionary using SOKAL & SNEATH-IV coefficient (SSIVC)

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

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.

According to [14-22], Jaccard between two words wi and wj has the equations

$$Jaccard(wi, wj) = J(wi, wj) = \frac{|wi\ \cap wj|}{|wi\ \cup\ wj|} \quad (3)$$

and other type of the Jaccard equation between two words wi and wj has the equation

$$\mathrm{Jaccard(wi, wj)} = \mathrm{J(wi, wj)} = \mathrm{sim(wi, wj)} = \frac{\mathrm{F(wi, wj)}}{\mathrm{F(wi) + F(wj) - F(wi, wj)}} \quad (4)$$

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

$$\mathrm{SO(wi)} = \sum \mathrm{Sim(wi, positive)} - \sum \mathrm{Sim(wi, positive)} \quad (5)$$

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 this: 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 Vietnamese. 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 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 RTC 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 [39-46], we have the equation of the SSIVC:

$$\mathrm{SOKAL\ \&\ SNEATH - IV\ Coefficient\ (a, b) = SOKAL\ \&\ SNEATH - IV\ Measure(a, b) = SSIVC(a, b)} = \frac{\frac{(a \cap b)}{(a \cap b) + (\neg a \cap b)} + \frac{(a \cap b)}{(a \cap b) + (a \cap \neg b)} + \frac{(\neg a \cap \neg b)}{(\neg a \cap \neg b) + (\neg a \cap b)} + \frac{(\neg a \cap \neg b)}{(\neg a \cap \neg b) + (a \cap \neg b)}}{4} \quad (6)$$

with a and b are the vectors.

From the eq. (1), (2), (3), (4), (5), (6), 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. (6), when a has only one element, a is a word. When b has only one element, b is a word. In

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

eq. (6), a is replaced by w1 and b is replaced by w2.

$$\text{SOKAL \& SNEATH} - \text{IV Measure}(w1, w2) = \text{SOKAL \& SNEATH} - \text{IV Coefficient}(w1, w2) =$$

$$\text{SSIVC}(w1, w2) = \frac{\frac{P(w1,w2)}{P(w1,w2)+P(\neg w1,w2)} + \frac{P(w1,w2)}{P(w1,w2)+P(w1,\neg w2)} + \frac{P(\neg w1,\neg w2)}{P(\neg w1,\neg w2)+P(\neg w1,w2)} + \frac{P(\neg w1,\neg w2)}{P(\neg w1,\neg w2)+P(w1,\neg w2)}}{4} \quad (7)$$

Eq. (7) is similar to eq. (1). In eq. (2), eq. (1) is replaced by eq. (7). We have eq. (8) as follows:

$$\text{Valence}(w) = \text{SO\_SSIVC}(w) = \text{SSIVC}(w, \text{positive\_query}) - \text{SSIVC}(w, \text{negative\_query}) \quad (8)$$

In eq. (7), w1 is replaced by w and w2 is replaced by position_query. We have eq. (9).Eq. (9) is as follows:

$$\text{SSIVC}(w, \text{positive\_query}) = \frac{\frac{P(w,\text{positive\_query})}{a\_b\_9} + \frac{P(w,\text{positive\_query})}{a\_c\_9} + \frac{P(\neg w,\neg\text{positive\_query})}{b\_d\_9} + \frac{P(\neg w,\neg\text{positive\_query})}{c\_d\_9}}{4} \quad (9)$$

with

$$a\_b\_9 = P(w, \text{positive\_query}) + P(\neg w, \text{positive\_query})$$
$$a\_c\_9 = P(w, \text{positive\_query}) + P(w, \neg\text{positive\_query})$$
$$b\_d\_9 = P(\neg w, \neg\text{positive\_query}) + P(\neg w, \text{positive\_query})$$
$$c\_d\_9 = P(\neg w, \neg\text{positive\_query}) + P(w, \neg\text{positive\_query})$$

In eq. (7), w1 is replaced by w and w2 is replaced by negative_query. We have eq. (10).Eq. (10) is as follows:

$$\text{SSIVC}(w, \text{negative\_query}) = \frac{\frac{P(w,\text{negative\_query})}{a\_b\_10} + \frac{P(w,\text{negative\_query})}{a\_c\_10} + \frac{P(\neg w,\neg\text{negative\_query})}{b\_d\_10} + \frac{P(\neg w,\neg\text{negative\_query})}{c\_d\_10}}{4} \quad (10)$$

with $a\_b\_10 = P(w, \text{negative\_query}) + P(\neg w, \text{negative\_query})$

$$a\_c\_10 = P(w, \text{negative\_query}) + P(w, \neg\text{negative\_query})$$
$$b\_d\_10 = P(\neg w, \neg\text{negative\_query}) + P(\neg w, \text{negative\_query})$$
$$c\_d\_10 = P(\neg w, \neg\text{negative\_query}) + P(w, \neg\text{negative\_query})$$

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 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).

(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 ((notw) 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. (8) 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.

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

In Table 1, we show the comparisons of our model's results with the works related to [1-32].

The comparisons of our model's advantages and disadvantages with the works related to [1-32] are displayed in Table 2.

In Table 3, we present the comparisons of our model's results with the works related to the SOKAL & SNEATH-IV coefficient (SSIVC) in [39-46].

The comparisons of our model's benefits and drawbacks with the studies related to the SOKAL & SNEATH-IV coefficient (SSIVC) in [39-46] are presented in Table 4.

4.1.2 Transferring the documents into the multi-dimensional vectors in the sequential environment

In this section, we transfer the documents of both the testing data set and the training data set into the multi-dimensional vectors in the sequential system.

In Fig 4 below, we present how to transfer one English document into one multi-dimensional vector based on the sentiment lexicons in the sequential environment.

Then, we apply this part to transfer all the documents of both the testing data set and the training data set into the sequential system.

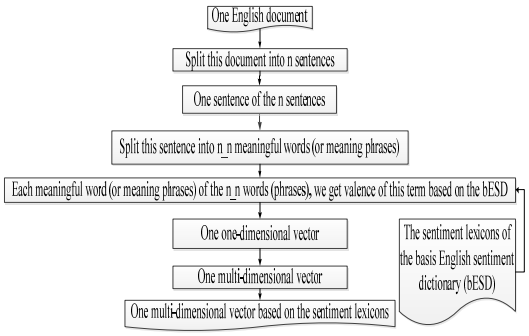

Fig. 4: Overview of transferring the documents into the multi-dimensional vectors in the sequential environment

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

We propose the algorithm 1 to transfer one English document into one multi-dimensional vectors based on the sentiment lexicons in the sequential system. The main ideas of the algorithm 1 are as follows:
Input: one English document
Output one multi-dimensional vector based the sentiment lexicons
Step 1: Split this document into the n sentences
Step 2: Each sentence in the n sentences, do repeat:
Step 3: Split this sentence into the n_n meaningful words (or meaningful phrases);
Step 4: Each term in the n_n terms, do repeat:
Step 5: Get the valence of this term based on the basis English sentiment dictionary (bESD);
Step 6: End Repeat- End Step 4;
Step 7: Add one one-dimensional vector (corresponding to this sentence) into the multi-dimensional vector;
Step 8: End Repeat – End Step 2;
Step 9: Return the multi-dimensional vector;

4.1.3 Transferring the documents into the multi-dimensional vectors in the distributed system

In this section, we transfer the documents of both the testing data set and the training data set into the mulit-dimensional vectors in the parallel network environment.

In Fig 5 below, we present how to transfer one English document into one multi-dimensional vector based on the sentiment lexicons in the parallel environment – the Cloudera system. The inputs of the Hadoop Map (M) in the Cloudera are one English document, the sentiment lexicons of the basis English dictionary (bESD). The output of the Hadoop Map is one one-dimensional vector (corresponding to one sentence of this document). The input of the Hadoop Reduce (R) in the Cloudera is the output of the Hadoop Map. The output of the Hadoop Reduce is one multi-dimensional vector (corresponding to this document).

Then, we apply this part to transfer all the documents of both the testing data set and the training data set into the Cloudera

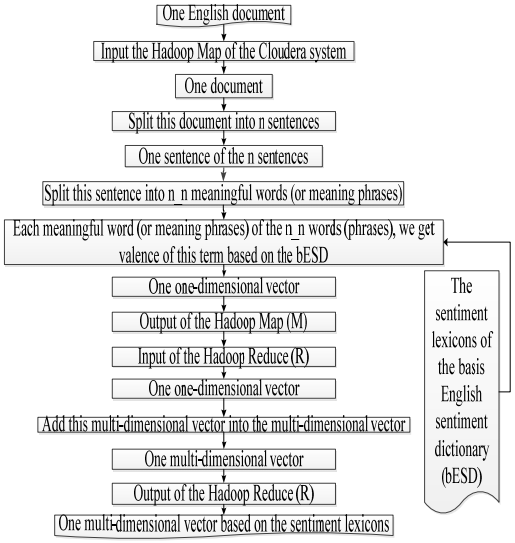

Fig 5: Overview of transferring the documents into the multi-dimensional vectors in the distributed system

In the Hadoop Map (M):
Input: One English document; The sentiment lexicons of the basis English sentiment dictionary (bESD).
Output: one one-dimensional vector;
Step 1:Input One English document; and The sentiment lexicons of the basis English sentiment dictionary (bESD) into the Hadoop Map in the Cloudera.
Step 2: Split this document into the n sentences;
Step 3: Each sentence in the n sentences, do repeat:
Step 4: Split this sentence into the n_n meaningful words (or meaningful phrases)
Step 5: Each term in the n_n terms, do repeat:
Step 6: Get valence of this term based on the basis English sentiment dictionary (bESD);
Step 7: Add this term into the one-dimensional vector;
Step 8: End Repeat – End Step 5;
Step 9: Return this one-dimensional vector;
Step 10: The output of the Hadoop Map is this one-dimensional vector;

In the Hadoop Reduce (R):
Input: one one-dimensional vector of the Hadoop Map (the input of the Hadoop Reduce is the output of the Hadoop Map)
Output: one multi-dimensional vector (corresponding to one English document)
Step 1: Receive one one-dimensional vector of the Hadoop Map
Step 2: Add this one-dimensional vector into the multi-dimensional vector;
Step 3: Return the multi-dimensional vector;

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

4.2 Implementing A K-Means Algorithms

This section has two parts: semantic classification for the documents of the testing in the sequential environment is presented in the first part; and in the second part, sentiment classification for the reviews of the testing in the parallel network environment is displayed.

With the English training data set, there are two groups. The first group includes the positive documents and the second group is the negative documents. The first group is called the positive cluster. The second group is called the negative cluster. All documents in both the first group and the second group go through the segmentation of words and stop-words removal; then, they are transferred into the multi-dimensional vectors (vector representation). The positive documents of the positive cluster are transferred into the positive multi-dimensional vectors which are called the positive vector group (or positive vector cluster). The negative documents of the negative cluster are transferred into the negative multi-dimensional vectors which are called the negative vector group (or negative vector cluster). Therefore, the training data set includes the positive vector group (or positive vector cluster) and the negative vector group (or negative vector cluster).

We have transferred all English sentences into one-dimensional vectors similar to the transferring the documents into the multi-dimensional vectors in the sequential environment (4.1.2) and the transferring the documents into the multi-dimensional vectors in the distributed system (4.1.3)

4.2.1 A Fuzzy C-Means Algorithm (FCM) in A Sequential Environment

In Fig. 6, in the sequential environment, the documents of the testing data set are transferred to the multi-dimensional vectors: each document of the testing data set is transferred to each multi-dimensional vector (each sentence of one document in the testing data set is transferred to the one-dimensional vector similar to the transferring the documents into the multi-dimensional vectors in the sequential environment (4.1.2). The positive documents in the training data set are transferred to the positive multi-dimensional vectors, called the positive vector group in the sequential environment: each document of the positive documents is transferred to each multi-dimensional vector (each sentence, of one document in the positive documents, is transferred to the one-dimensional vector similar to the transferring the documents into the multi-dimensional vectors in the sequential environment (4.1.2) in the sequential environment). The negative documents in the training data set are transferred to the negative multi-dimensional vectors, called the negative vector group in the sequential environment: each document of the negative documents is transferred to each multi-dimensional vector (each sentence, of one English document in the negative documents, is transferred to the one-dimensional vector similar to the transferring the documents into the multi-dimensional vectors in the sequential environment (4.1.2) in the sequential environment).

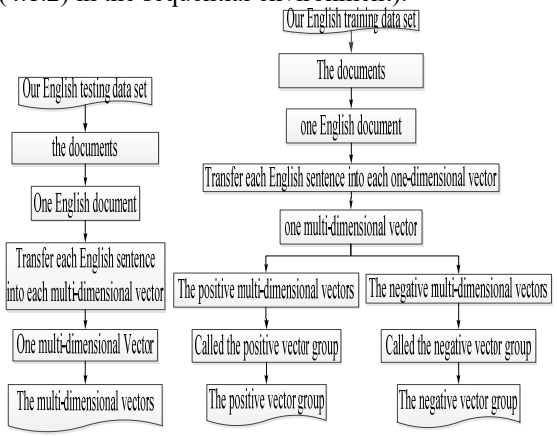

Fig. 6: Overview of transferring all English documents into the multi-dimensional vectors

We perform this part as follows in Fig. 7 below: In the sequential environment, the FCM is implemented to cluster one multi-dimensional vector (called A) of the English testing data set to the positive vector group or the negative vector group. The document (corresponding to A) is the positive polarity if A is clustered to the positive vector group. The document (corresponding to A) is the negative polarity if A is clustered to the negative vector group. The document (corresponding to A) is the neutral polarity if A is not clustered to both the positive vector group and the negative vector group.

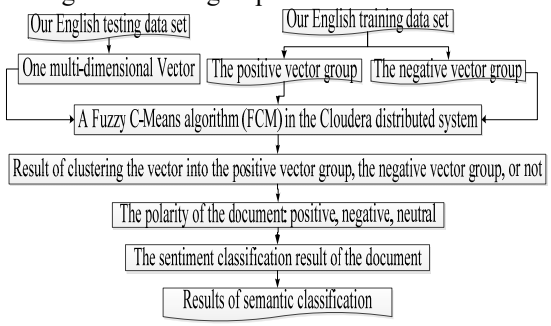

Fig. 7. A Fuzzy C-Means Algorithm in the Sequential Environment

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

We built many algorithms to perform the FCM in the sequential environment. We build the algorithm 1 to transfer one English document into one multi-dimensional vector. Each document is split into many sentences. Each sentence in each document is transferred to one one-dimensional vector based on the transferring the documents into the multi-dimensional vectors in the sequential environment (4.1.2) in the sequential environment. We insert all the one-dimensional vectors of the sentences into one multi-dimensional vector of one document. The main ideas of the algorithm 1 are as follows:

Input: one English document
Output: one multi-dimensional vector
Step 1: Split the English document into many separate sentences based on "." Or "!" or "?";
Step 2: Each sentence in the n sentences of this document, do repeat:
Step 3: Transfer this sentence into one vector (one dimension) based on the transferring the documents into the multi-dimensional vectors in the sequential environment (4.1.2)
Step 4: Add the transferred vector into one multi-dimensional vector
Step 5: End Repeat – End Step 2
Step 6: Return one multi-dimensional vector;

We build the algorithm 2 to create the positive vector group. Each document in the positive documents in the English training data set is split into many sentences. Each sentence of the document is transferred to one one-dimensional vector based on the transferring the documents into the multi-dimensional vectors in the sequential environment (4.1.2) in the sequential environment. We insert all the one-dimensional vectors of the sentences of the document into one one-dimensional vector of the document. Then, the positive documents in the English training data are transferred to the positive multi-dimensional vectors. The main ideas of the algorithm 2 are as follows:

Input: the positive English documents of the English training data set.
Output: the positive vector group PositiveVectorGroup
Step 1:Each document in the positive document of the training data set, do repeat:
Step 2: OneMultiDimensionalVector := Call Algorithm 1 with the positive English document in the English training data set;
Step 3: Add OneMultiDimensionalVector into PositiveVectorGroup;
Step 4: End Repeat – End Step 1
Step 5: Return PositiveVectorGroup;

We build the algorithm 3 to create the negative vector group. Each document in the negative documents in the English training data set is split into many sentences. Each sentence of the document is transferred to one one-dimensional vector based on the transferring the documents into the multi-dimensional vectors in the sequential environment (4.1.2) in the sequential environment. We insert all the one-dimensional vectors of the sentences of the document into one one-dimensional vector of the document. Then, the negative documents in the English training data set are transferred to the negative multi-dimensional vectors. The main ideas of the algorithm 3 are as follows:

Input: the negative English documents of the English training data set.
Output: the negative vector group PositiveVectorGroup
Step 1:Each document in the negativedocument of the training data set, do repeat:
Step 2: OneMultiDimensionalVector := Call Algorithm 1 with the negativeEnglish document in the English training data set;
Step 3: Add OneMultiDimensionalVector into NegativeVectorGroup;
Step 4: End Repeat – End Step 1
Step 5: Return NegativeVectorGroup;

We build the algorithm 4 to cluster one multi-dimensional vector (corresponding to one document of the English testing data set) into the positive vector group PositiveVectorGroup, the negative vector group NegativeVectorGroup, or not.The main ideas of the algorithm 4 are as follows:

Input: one multi-dimensional vector A (corresponding to one English document of the English testing data set), the positive vector group PositiveVectorGroup, the negative vector group NegativeVectorGroup;
Output: positive, negative, neutral;
Step 1: Implement the Fuzzy C-Means Algortihm based on the Fuzzy C-Means algorithm (FCM) in [53-67] with input is one multi-dimensional vector (corresponding to one English document of the English testing data set), the positive vector group PositiveVectorGroup, the negative vector group NegativeVectorGroup;
Step 2: With the results of Step 1, If the vector is clustered into the positive vector group Then Return positive;
Step 3: Else If the vector is clustered into the negative vector group Then Return negative; End If – End Step 2
Step 4: Return neutral;

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

The FCM uses Euclidean distance to calculate the distance between two vectors.

4..2.2 A Fuzzy C-Means Algorithm (FCM)in A Parallel Network Environment

In Fig. 8, all documents of both the English testing data set and the English training data set are transferred into all the multi-dimensional vectors in the Cloudera parallel network environment. With the documents of the English training data set, we transferred them into the multi-dimensional vectors by using Hadoop Map (M)/Reduce (R) in the Cloudera parallel network environment with the purpose of shortening the execution time of this task. The positive documents of the English training data set are transferred into the positive vectors in the Cloudera parallel system and are called the positive vector group. The negative documents of the English training data set are transferred into the negative vectors in the Cloudera parallel system and are called the negative vector group. Besides, the documents of the English testing data set are transferred to the multi-dimensional vectors by using Hadoop Map (M)/Reduce (R) in the Cloudera parallel network environment with the purpose of shortening the execution time of this task.

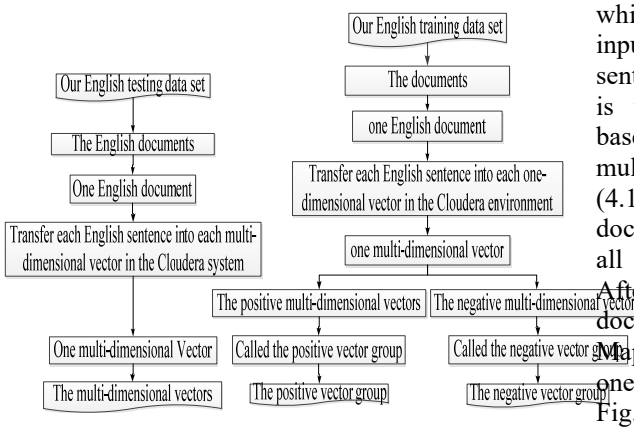

Fig. 8: Overview of transferring all English documents into the multi-dimensional vectors in the Cloudera distributed system.

We implement this part in Fig. 9, below. In the Cloudera distributed network environment, by using the FCM, one multi-dimensional vector (called A) of one document in the English testing data set is clustered into the positive vector group or the negative vector group. The document (corresponding to A) is the positive polarity if A is clustered into the positive vector group. The document (corresponding to A) is the negative polarity if A is clustered into the negative vector group. The document (corresponding to A) is the neutral polarity if A is not clustered into both the positive vector group and the negative vector group.

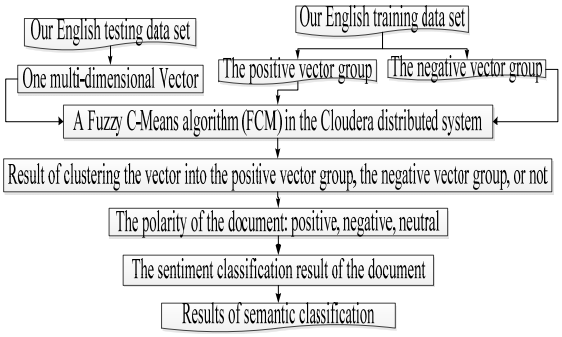

Fig. 9: A Fuzzy C-Means Algorithm (FCM) in the Parallel Network Environment.

An overview of transferring each English sentence into one vector in the Cloudera network environment is follows in Fig. 10. In Fig. 10, transferring each English document into one vector in the Cloudera network environment includes two phases: Map (M) phases and Reduce (R) phases. Input of the Map phase is one document and Output of the Map phase is many components of a vector which corresponds to the document. One document, input into Hadoop Map (M), is split into many sentences. Each sentence in the English document is transferred into one one-dimensional vector based on the transferring the documents into the multi-dimensional vectors in the distributed system (4.1.3). This is repeated for all the sentences of the document until all the sentences are transferred into all the one-dimensional vectors of the document. After finishing to transfer each sentence of the document into one one-dimensional vector, the Map phase of Cloudera automatically transfers the one-dimensional vector into the Reduce phase. In Fig. 10, the input of the Reduce phase is the output of the Map phase, and this input comprises many components (many one-dimensional vectors) of a multi-dimensional vector. The output of the Reduce phase is a multi-dimensional vector which corresponds to the document. In the Reduce phase of Cloudera, those components of the vector are built into one multi-dimensional vector.

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

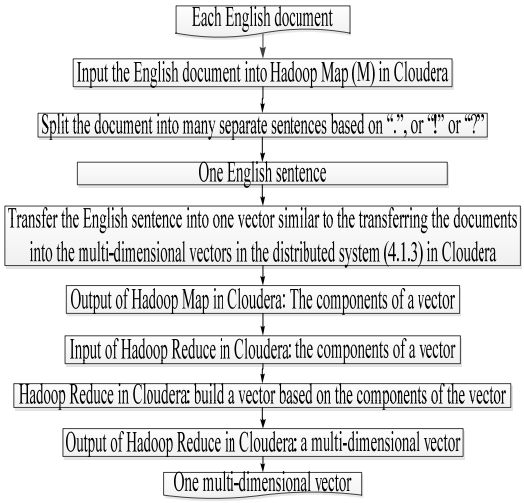

Fig. 10: Overview of transforming each English sentence into one vector in Cloudera

The documents of the English testing data set are transferred into the multi-dimensional vectors based on Fig. 10. The FCM in the Cloudera parallel network environment has two main phases: the first main phase is Hadoop Map (M) phase in Cloudera and the second main phase is Hadoop Reduce (R) phase in Cloudera. In the Map phase of Cloudera, the input of the phase is the multi-dimensional vector of one English document (which is classified), the positive vector group, the negative vector group; and the output of the phase is the clustering results of the multi-dimensional vector of the document to the positive vector group or the negative vector group, or not. With the Reduce phase of the Cloudera, the input of the phase is the output of the Map phase of the Cloudera and this input is the clustering results of the multi-dimensional vector of the document to the positive vector group or the negative vector group or not; and the output of the phase is the sentiment classification result of the document into the positive polarity, the negative polarity, or the neutral polarity. In the Reduce phase, the document is classified as the positive emotion if the multi-dimensional vector is clustered into the positive vector group; the document is classified as the negative semantic if the multi-dimensional vector into the negative vector group; and the document is classified as the neutral sentiment if the multi-dimensional vector is not clustered into the positive vector group, or the negative vector group, or not.

4.2.2.1 Hadoop Map (M)

This phase is done as illustrated in Fig. 11, below. The Fuzzy C-Means algorithm (FCM) in Cloudera is based on the Fuzzy C-Means algorithm (FCM) in [53-67]. The input is one multi-dimensional vector in the English testing data set, the positive vector group and the negative vector group of the English training data set. The output of the FCM is the clustering results of the multi-dimensional vector into the positive vector group or the negative vector group, or not.

The main ideas of the FCM are as follows:

1)Enter values for the two parameters: c (1 <c< N),m and initializing the sample matrix

2)Repeat

2.1 j = j + 1;

2.2 Calculating fuzzy partition matrix Uj following formula (1)

2.3 Updating centers V(j) [v1(j), v2(j), ..., vc(j)]basing on (2) and Ujmatrix;

Step 3: Untill ($|| U_{(j+1)}–U_{(j)}||_F \leq \varepsilon$);

Step 4: Performing results of the clusters.

with $||U||^2_F = \sum i \sum k U^2_{ik}$

The FCM uses Euclidean distance to calculate the distance between two vectors

After finishing to cluster the multi-dimensional vector into the positive vector group, or the negative vector group, or not, Hadoop Map transfers this results into Hadoop Reduce in the Cloudera system.

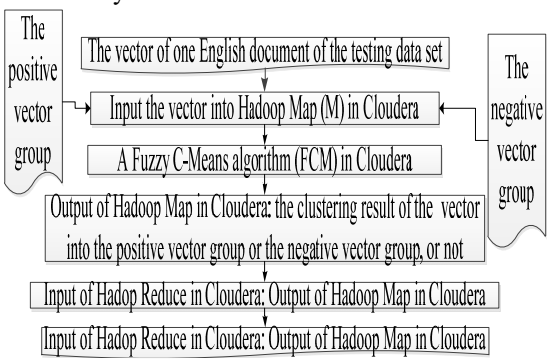

Fig. 11.Overview of the FCM in Hadoop Map (M) in Cloudera

4.2.2.2 Hadoop Reduce (R)

This phase is done as illustrated below in Fig. 12. After receiving the clustering result of Hadoop Map, Hadoop Reduce labels the semantics polarity for the multi-dimensional vector which is classified. Then, the output of Hadoop Reduce will return the semantics polarity of one document (corresponding to the multi-dimensional vector) in the English testing data set. One document is the positive polarity if the multi-dimensional vector is clustered into the positive vector group. One document is the negative polarity if the multi-dimensional vector is clustered into the negative vector group. One

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

document is the neutral polarity if the multi-dimensional vector is not clustered into both the positive vector group and the negative vector group.

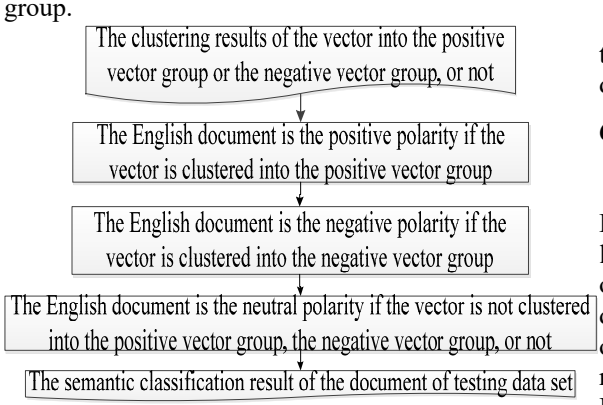

Fig. 12: Overview of Hadoop Reduce (R) in Cloudera

5. EXPERIMENT

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

The sequential environment in this research includes 1 node (1 server). The Java language is used in programming the FCM. The configuration of the server in the sequential environment is: Intel® Server Board S1200V3RPS, Intel® Pentium® Processor G3220 (3M Cache, 3.00 GHz), 2GB PC3-10600 ESSIVC 1333 MHz LP Unbuffered DIMMs. The operating system of the server is: Cloudera. We perform the FCM in the Cloudera parallel network environment; this Cloudera system includes 9 nodes (9 servers). The Java language is used in programming the application of the FCMin the Cloudera. The configuration of each server in the Cloudera system is: Intel® Server Board S1200V3RPS, Intel® Pentium® Processor G3220 (3M Cache, 3.00 GHz), 2GB PC3-10600 ESSIVC 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.

In Table 5, we show the results of the documents in the testing data set.

The Accuracy of our new model for the documents in the testing data set is presented in Table 6.

In Table 7, we display the average execution times of the classification of our new model for the documents in testing data set.

6. CONCLUSION

Although our new model has been tested on our English data set, it can be applied to many other languages. In this paper, our model has been tested on the 5,500,000 English documents of the testing data set in which the data sets are small. However, our model can be applied to larger data sets with millions of English documents in the shortest time. In this work, we have proposed a new model to classify sentiment of English documents using the Fuzzy C-Means Algorithm (FCM) with Hadoop Map (M)/Reduce (R) in the Cloudera parallel network environment. With our proposed new model, we have achieved 87.82% accuracy of the testing data set in Table 6. 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 emotion in text.

In Table 7, the average time of the semantic classification of the FCM algorithm in the sequential environment is 23,457,821 seconds /5,500,000 English documents and it is greater than the average time of the emotion classification of the FCM in the Cloudera parallel network environment with 3 nodes which is 8,142,573 seconds/5,500,000 English documents. The average time of the emotion classification of the FCM in the Cloudera parallel network environment with 9 nodes, which is 2,528,643 seconds /5,500,000 English documents, is the shortest time. Besides, the average time of the emotion classification of the FCM in the Cloudera parallel network environment with 6 nodes is 4,021,386 seconds /5,500,000 English documents

The execution time of the FCM in the Cloudera is dependent on the performance of the Cloudera parallel system and also dependent on the performance of each server on the Cloudera system.

The proposed model has many advantages and disadvantages. Its positives are as follows: It uses the Fuzzy C-Means algorithm 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. 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 8, we display the comparisons of our model's results with the works in [47, 48, 49].

The comparisons of our model's advantages and disadvantages with the works in [47, 48, 49] are shown in Table 9.

In Table 10, we present the comparisons of our model's merits and demerits with the works related to the K-Nearest Neighbors algorithm (K-NN) in [53-67].

The comparisons of our model's positives and negatives the latest sentiment classification models (or the latest sentiment classification methods) in [68-73] are displayed in Table 11.

7. FUTURE WORK

Based on the results of this proposed model, many future projects can be proposed, such as creating full emotional lexicons in a parallel network environment to shorten execution times, creating many search engines, creating many translation engines, creating many applications that can check grammar correctly. This model can be applied to many different languages, creating applications that can analyze the emotions of texts and speeches, and machines that can analyze sentiments.

REFERENCES:

[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, 2010, pp 25–30.

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

[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, 2016, pp 481–485.

[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, 2009, pp 235–243.

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

[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, 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, 2007, pp 189–192.

[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, 2013, pp 897–902.

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

[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.

[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), Volume 5, 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, 2012, pp 521-543.

[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, 2011, pp 420 - 429.

[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, 2011, pp 571-583.

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

[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, 2004, 83-91.

[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, 2008, 367- 370.

[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, 2017, 67 pages.

[29] Vo Ngoc Phu, Vo Thi Ngoc Chau, Nguyen Duy Dat, Vo Thi Ngoc Tran, Tuan A. Nguyen, "*A Valences-Totaling Model for English Sentiment Classification*", International Journal of Knowledge and Information Systems, DOI: 10.1007/s10115-017-1054-0, 2017, 30 pages.

[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, 2017, 28 pages.

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

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/s12530-017-2017, 9188-6, 69 pages.

[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] Tarik S.K.M. Rabie, "*Implementation Of Some Similarity Coefficients In Conjunction With Multiple Upgma And Neighbor-Joining Algorithms For Enhancing Phylogenetic Trees*", Egypt. Poult. Sci. Vol (30) (II): (607-621), 2010

[40] Bruce M. Campbell, "*Similarity coefficients for classifying relevés*", Vegetatio, Volume 37, Issue 2, 1978, pp 101–109

[41] 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. MSSIVCology in Sustainable Development: Expanding Concepts, Vanishing Borders. (Parkway Publishers, Boone, North Carolina): 122-143, 1997

[42] Sung-Hyuk Cha, "*Comprehensive Survey On Distance/Similarity Measures Between Probability Density Functions*", International Journal Of Mathematical Models And Methods In Applied Sciences, Issue 4, Volume 1, 2007.

[43] G. N. Lance, W. T. Williams, "*Computer Programs for Hierarchical Polythetic Classification ("Similarity Analyses")*", The Computer Journal, Volume 9, Issue 1, https://doi.org/10.1093/comjnl/9.1.60, 1996, Pages 60–64

[44] G. N. Lance, W. T. Williams, "*Computer programs for monothetic classification ("Association analysis")*", The Computer Journal, Volume 8, Issue 3, 1965, Pages 246–249, https://doi.org/10.1093/comjnl/8.3.246

[45] W. T. Williams, H. T. Clifford, "*Group-size dependence: a rationale for choice between numerical classifications*", The Computer Journal, Volume 14, Issue 2, 1971, Pages 157–162, https://doi.org/10.1093/comjnl/14.2.157

[46] L. WATSON, W. T. WILLIAMS, G. N. LANCE, "*Angiosperm taxonomy: a comparative study of some novel numerical techniques*", Botanical Journal of the Linnean Society, Volume 59, Issue 380, https://doi.org/10.1111/j.1095-8339.1966.tb00075.x, 2008, Pages 491–501

[47] Vaibhav Kant Singh, Vinay Kumar Singh, "*Vector Space Model: An Information Retrieval System*", Int. J. Adv. Engg. Res. Studies/IV/II/Jan.-March,2015/141-143, 2015

[48] 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, 2015, pp. 7-19.

[49] 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, USA, 2015, pp. 1130-1135.

[50] Hadoop, http://hadoop.apache.org., 2017

[51] Apache, http://apache.org ., 2017

[52] Cloudera, http://www.cloudera.com., 2017

[53] James C. Bezdek, Robert Ehrlich, William Full, "*FCM: The fuzzy c-means clustering algorithm*", Computers & Geosciences, https://doi.org/10.1016/0098-3004(84)90020-7, Volume 10, Issues 2–3, 1984Pages 191-203

[54] Dao-Qiang Zhang, Song-Can Chen, "*A novel kernelized fuzzy C-means algorithm with application in medical image segmentation*", Artificial Intelligence in Medicine, https://doi.org/10.1016/j.artmed.2004.01.012, Volume 32, Issue 1, 2004, Pages 37-50

[55] N.R. Pal, J.C. Bezdek, "*On cluster validity for the fuzzy c-means model*", IEEE Transactions on Fuzzy Systems, Volume: 3, Issue: 3,DOI: 10.1109/91.413225, 1995

[56] N.R. Pal, K. Pal, J.M. Keller, J.C. Bezdek, "*A possibilistic fuzzy c-means clustering algorithm*", IEEE Transactions on Fuzzy Systems, Volume: 13, Issue: 4,DOI: 10.1109/TFUZZ.2004.840099, 2005

[57] Dzung L. Phamab, Jerry L. Princea (1999) , "*An adaptive fuzzy C-means algorithm for image segmentation in the presence of intensity*

inhomogeneities", Pattern Recognition Letters, Volume 20, Issue 1, https://doi.org/10.1016/S0167-8655(98)00121-4, 1999, Pages 57-6

[58] Dao-Qiang Zhang, Song-Can Chen, "*Clustering Incomplete Data Using Kernel-Based Fuzzy C-means Algorithm*", Neural Processing Letters, Volume 18, Issue 3, 2003, pp 155–162

[59] L. Szilagyi, Z. Benyo, S.M. Szilagyi, H.S. Adam, "*MR brain image segmentation using an enhanced fuzzy C-means algorithm*", Proceedings of the 25th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, 2003, DOI: 10.1109/IEMBS.2003.1279866

[60] Keh-Shih Chuang, Hong-Long Tzeng, Sharon Chen, Jay Wu, Tzong-Jer Chen, "*Fuzzy c-means clustering with spatial information for image segmentation*", Computerized Medical Imaging and Graphics,https://doi.org/10.1016/j.compmedimag.2005.10.001 , Volume 30, Issue 1, 2006, Pages 9-15

[61] R.J. Hathaway; J.C. Bezdek; Yingkang Hu, "*Generalized fuzzy c-means clustering strategies using L/sub p/ norm distances*", IEEE Transactions on Fuzzy Systems, Volume: 8, Issue: 5, DOI: 10.1109/91.873580, 2000

[62] Weiling Cai, Songcan Chen, Daoqiang Zhang, "*Fast and robust fuzzy c-means clustering algorithms incorporating local information for image segmentation*", Pattern Recognition, https://doi.org/10.1016/j.patcog.2006.07.011, Volume 40, Issue 3, 2007, Pages 825-838

[63] J.F. Kolen; T. Hutcheson, "*Reducing the time complexity of the fuzzy c-means algorithm*", IEEE Transactions on Fuzzy Systems, Volume: 10, Issue: 2, DOI: 10.1109/91.995126, 2002

[64] Young Won Lim, Sang Uk Lee, "*On the color image segmentation algorithm based on the thresholding and the fuzzy c-means techniques*", Pattern Recognition, https://doi.org/10.1016/0031-3203(90)90103-R, Volume 23, Issue 9, 1990, Pages 935-952

[65] BirendraBiswal; P. K. Dash; B. K. Panigrahi, "*Power Quality Disturbance Classification Using Fuzzy C-Means Algorithm and Adaptive Particle Swarm Optimization*", IEEE Transactions on Industrial Electronics, Volume: 56, Issue: 1, DOI: 10.1109/TIE.2008.928111, 2009

[66] Zheng, Yuhui; Jeon, Byeungwoo; Xu, Danhua; Wu, Q.M. Jonathan; Zhang, Hui; "*Image segmentation by generalized hierarchical fuzzy C-means algorithm*", Journal of Intelligent & Fuzzy Systems, vol. 28, no. 2, DOI: 10.3233/IFS-141378, 2015, pp. 961-973

[67] Long Chen; C. L. P. Chen; Mingzhu Lu, "*A Multiple-Kernel Fuzzy C-Means Algorithm for Image Segmentation*", IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), Volume: 41, Issue: 5, DOI: 10.1109/TSMCB.2011.2124455, 2011

[68] 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, 2016, 21-45.

[69] 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, 2016, 77-88

[70] 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), New York USA, 2016, 53-62

[71] 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, India, 2016, 171-179

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

[73] 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

[74] 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.

[75] Vo Ngoc Phu, Nguyen Duy Dat, Vo Thi Ngoc Tran, Vo Thi Ngoc Tran, "*Fuzzy C-Means for English Sentiment Classification in a*

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

Distributed System", International Journal of Applied Intelligence (APIN), DOI: 10.1007/s10489-016-0858-z, November 2016, 1-22

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

[77] 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, 2017, 31 pages

[78] 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, 2017, 2017, 23 pages

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

APPENDICES:

Table 1: Comparisons of our model's results with the works related to [1-32].

SOKAL & SNEATH-IV coefficient (SSIVC)

Semantic classification, sentiment classification: SC

Studies	PMI	JM	Language	SD	DT	SSIVC	SC	Other measures	Search engines
[1]	Yes	No	English	Yes	Yes	No	Yes	No	No Mention
[2]	Yes	No	English	Yes	No	No	Yes	Latent Semantic Analysis (LSA)	AltaVista
[3]	Yes	No	English	Yes	Yes	No	Yes	Baseline; Turney-inspired; NB; Cluster+NB; Human	AltaVista
[4]	Yes	No	English German	Yes	Yes	No	Yes	SimRank	Google search engine
[5]	Yes	No	English Macedonian	Yes	Yes	No	Yes	No Mention	AltaVista search engine
[6]	Yes	No	English Arabic	Yes	No	No	Yes	No Mention	Google search engine Bing search engine
[7]	Yes	No	English Chinese	Yes	Yes	No	Yes	SVM(CN); SVM(EN); SVM(ENCN1); SVM(ENCN2); TSVM(CN); TSVM(EN); TSVM(ENCN1); TSVM(ENCN2); CoTrain	No Mention
[8]	Yes	No	English Spanish	Yes	Yes	No	Yes	SO Calculation SVM	Google
[9]	Yes	No	Chinese Tibetan	Yes	Yes	No	Yes	- Feature selection -Expectation Cross Entropy -Information Gain	No Mention
[10]	Yes	No	Chinese	Yes	Yes	No	Yes	DF, CHI, MI andIG	No Mention
Our work	No	No	English Language	No	No	Yes	Yes	No	Google search engine

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

Table 2: 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 andSO-LSA. The Accuracy of SO-PMI-IR is comparable to the Accuracy of HM, the algorithm of Hatzivassiloglou and McKeown (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 user classification for informal online political discourse	The authors describe several experiments 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	There is still much 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]	Anovel, 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.
[5]	Analysis in Twitter for Macedonian	The authors' experimental results show an F1-score of 92.16, which is very strong and is on par with the best results for English, which were achieved in recent SemEval competitions.	In future work, the authors are interested in studying the impact of the raw corpus size, e.g., the authors could only collect half a million tweets for creating lexicons and analyzing/evaluating the system, while Kiritchenko et al. (2014) built their lexicon on million tweets and evaluated their system on 135 million English tweets. Moreover, the authors are interested not only in quantity but also in quality, i.e., in studying the quality of the individual words and phrases used as seeds.
Our work	-SOKAL & SNEATH-IV coefficient (SSIVC) through the Google search engine with AND operator and OR operator. -The Fuzzy C-Means algorithm (FCM) with the multi-dimensional vectors based on the sentiment lexicons for English sentiment classification in the Cloudera distributed system. -The advantages and disadvantages of this survey are shown in the Conclusion section.		

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

Table 3: Comparisons of our model's results with the worksrelated to the SOKAL & SNEATH-IV coefficient (SSIVC) in [39-46].

Studies	PMI	JM	SOKAL & SNEATH-IV coefficient (SSIVC)	Language	SD	DT	Sentiment Classification
[39]	Yes	Yes	Yes	English	NM	NM	No mention
[40]	No	No	Yes	NM	NM	NM	No mention
[41]	No	No	Yes	NM	NM	NM	No mention
[42]	No	No	Yes	NM	NM	NM	No mention
[43]	No	No	Yes	NM	NM	NM	No mention
[44]]	No	No	Yes	NM	NM	NM	No mention
[45]	No	No	Yes	NM	NM	NM	No mention
[46]	No	No	Yes	NM	NM	NM	No mention
Our work	No	No	Yes	English Language	Yes	Yes	Yes

Table 4: Comparisons of our model's benefits and drawbacks with the studies related to the SOKAL & SNEATH-IV coefficient (SSIVC) in [39-46].

Surveys	Approach	Benefits	Drawbacks
[39]	Implementation Of Some Similarity Coefficients In Conjunction With Multiple Upgma And Neighbor-Joining Algorithms For Enhancing Phylogenetic Trees	The population genetic distances were estimated by using two cluster algorithms (UPGMA & NJ neighbor-joining) accompanied with ten similarity coefficients comprising Jaccard, Sørensen-Dice, Russel& Rao, Rogers & Tanimoto, Simple Matching, Pearson Phi, Lance &Williams, Mountford, Michael, and Kulchenzky-1. The results demonstrated that for almost all methodologies, the Jaccard and Sørensen-Dice followed by Simple Matching coefficients revealed extremely close results, because both of them exclude negative co-occurrences. Due to the fact that there is no guarantee that the DNA regions with negative co- occurrences between two strains are indeed identical, the use of coefficients such as Jaccard and Sørensen-Dice that do not include negative occurrences was imperative for closely related organisms along with the NJ neighbor-joining cluster algorithm.	No mention
[40]	Similarity coefficients for classifying relevés	In this study, the clustering procedure of group average sorting was used to construct the dendrogram. It gives an average similarity value within the dendrogram groups. These values can be used to give quantitative definitions to syntaxonomic rank.	No mention

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

[41]	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
[42]	Comprehensive Survey on Distance/Similarity Measures between Probability Density Functions	Various distance/similarity measures that are applicable to compare two probabilitydensity functions, pdf in short, are reviewed and categorized inboth syntactic and semantic relationships. A correlation coefficient and a hierarchical clustering technique are adopted to reveal similarities among numerous distance/similarity measures	No mention
Our work	-SOKAL & SNEATH-IV coefficient (SSIVC) through the Google search engine with AND operator and OR operator. -The Fuzzy C-Means algorithm (FCM) with the multi-dimensional vectors based on the sentiment lexicons for English sentiment classification in the Cloudera distributed system. -The advantages and disadvantages of this survey are shown in the Conclusion section.		

Table 5: The results of the documents in the testing data set.

	Testing Dataset	Correct Classification	Incorrect Classification
Negative	2,750,000	2,415,341	334,659
Positive	2,750,000	2,414,759	335,241
Summary	5,500,000	4,830,100	669,900

Table 6: The accuracy of our novel model for the documents in the testing data set.

Proposed Model	Class	Accuracy
Our new model	Negative	87.82%
	Positive	

Table 7: The average execution times of the classification of our novel model for the documents in testing data set.

	The average execution times of the classification /5,500,000 English documents.
The Fuzzy C-Means Algorithm (FCM) in the sequential environment	23,457,821 seconds
The Fuzzy C-Means Algorithm (FCM) in the Cloudera distributed system with 3 nodes	8,142,573 seconds
The Fuzzy C-Means Algorithm (FCM) in the Cloudera distributed system with 6 nodes	4,021,386 seconds
The Fuzzy C-Means Algorithm (FCM) in the Cloudera distributed system with 9 nodes	2,528,643 seconds

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

Table 8: Comparisons of our model's results with the works in [47, 48, 49].

Clustering technique: CT.
Parallel network system: PNS (distributed system).
Special Domain: SD.
Depending on the training data set: DT.
Vector Space Model: VSM
No Mention: NM
English Language: EL.

Studies	FCM	CT	Sentiment Classification	PNS	SD	DT	Language	VSM
[47]	No	No	No	No	Yes	No	EL	Yes
[48]	No	No	Yes	No	Yes	No	EL	Yes
[49]	No	No	Yes	No	Yes	Yes	EL	Yes
Our work	Yes	Yes	Yes	Yes	Yes	Yes	EL	Yes

Table 9: Comparisons of our model's advantages and disadvantages with the works in [47, 48, 49].

Researches	Approach	Advantages	Disadvantages
[47]	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 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.	The drawbacks are that the system yields no theoretical findings. 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.
[48]	+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
Our work	The Fuzzy C-Means algorithm for English sentiment classification in the Cloudera distributed system. The advantages and disadvantages of the proposed model are shown in the Conclusion section.		

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

Table 10: Comparisons of our model's merits and demerits with the works related to the K-Nearest Neighbors algorithm (K-NN) in [53-67].

Works	Approach	Merits	Demerits
[53]	FCM: The fuzzy c-means clustering algorithm	These partitions are useful for corroborating known substructures or suggesting substructure in unexplored data. The clustering criterion used to aggregate subsets is a generalized least-squares objective function. Features of this program include a choice of three norms (Euclidean, Diagonal, or Mahalonobis), an adjustable weighting factor that essentially controls sensitivity to noise, acceptance of variable numbers of clusters, and outputs that include several measures of cluster validity.	No mention
[54]	A novel kernelized fuzzy C-means algorithm with application in medical image segmentation	Experimental results on both synthetic and real MR images show that the proposed algorithms have better performance when noise and other artifacts are present than the standard algorithms.	No mention
[55]	On cluster validity for the fuzzy c-means model	Limit analysis indicates, and numerical experiments confirm, that the Fukuyama-Sugeno index is sensitive to both high and low values of m and may be unreliable because of this. Of the indexes tested, the Xie-Beni index provided the best response over a wide range of choices for the number of clusters, (2-10), and for m from 1.01-7. Finally, our calculations suggest that the best choice for m is probably in the interval [1.5, 2.5], whose mean and midpoint, m=2, have often been the preferred choice for many users of FCM.	No mention
[56]	A possibilistic fuzzy c-means clustering algorithm	The authors derive the first-order necessary conditions for extrema of the PFCM objective function, and use them as the basis for a standard alternating optimization approach to finding local minima of the PFCM objective functional. Several numerical examples are given that compare FCM and PCM to PFCM. The authors' examples show that PFCM compares favorably to both of the previous models. Since PFCM prototypes are less sensitive to outliers and can avoid coincident clusters, PFCM is a strong candidate for fuzzy rule-based system identification.	No mention
Our work	-SOKAL & SNEATH-IV coefficient (SSIVC) through the Google search engine with AND operator and OR operator. -The Fuzzy C-Means algorithm (FCM) with the multi-dimensional vectors based on the sentiment lexicons for English sentiment classification in the Cloudera distributed system. -Our research's merits and demerits are shown in the Conclusion section.		

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

Table 11: Comparisons of our model's positives and negatives the latest sentiment classification models (or the latest sentiment classification methods) in [68-73].

Studies	Approach	Positives	Negatives
[68]	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
[69]	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
[70]	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 or text classification tasks that 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.	A line of future research would be to explore the authors' meta features with other classification algorithms and feature selection techniques in different sentiment analysis tasks such as scoring movies or products According to their related reviews.
[71]	Rule-Based Machine Learning Algorithms	The proposed approach is tested by experimenting with online books and political reviews and demonstrates the efficacy through Kappa measures, which have a higher Accuracy of 97.4% and a lower error rate. The weighted average of different Accuracy measures like Precision, Recall, and TP-Rate depicts higher efficiency rate and lower FP-Rate. Comparative experiments on various rule-based machine learning algorithms have been performed through a ten-fold cross validation training model for sentiment classification.	No mention
Our work	-SOKAL & SNEATH-IV coefficient (SSIVC) through the Google search engine with AND operator and OR operator. -The Fuzzy C-Means algorithm (FCM) with the multi-dimensional vectors based on the sentiment lexicons for English sentiment classification in the Cloudera distributed system. -The positives and negatives of the proposed model are given in the Conclusion section.		