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

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

A CO-TRAINING MODEL USING A FUZZY C-MEANS ALGORITHM, A K-MEANS ALGORITHM AND THE SENTIMENT LEXICONS - BASED MULTI-DIMENSIONAL VECTORS OF AN OTSUKA COEFFICIENT FOR ENGLISH SENTIMENT CLASSIFICATION

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

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

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

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

ABSTRACT

A semi-supervised learning of a machine learning used for a new model for big data sentiment classification has already been built in this survey. We have proposed a novel model using mainly a co-training (CT) approach to classify 10,500,000 documents of our testing data set comprising the 5,250,000 positive and the 5,250,000 negative into 4,000 documents of our training data set including the 2,000 positive and the 2,000 negative in English. In this co-training model (CTM), a Fuzzy C-Means algorithm has been used in training a first classifier and a K-Means algorithm has been used in training a second classifier based on many multi-dimensional vectors of sentiment lexicons of An OTSUKA coefficient (OM).After training the first classifier and the second classifier of the CTM of each loop, the 50 documents of the testing data set have certainly been chosen from the first classifier, then, they have been added to the second classifier, and the 50 documents of the testing data set have certainly been chosen from the second classifier, then, they have been added to the first classifier. The sentiment classification of all the documents of the testing data set has been identified after many loops of training the first classifier and the second classifier of the CTM certainly. In this survey, we do not use any vector space modeling (VSM). We do not use any one-dimensional vectors according to both the VSM and the sentiment classification. The OM is used in creating the sentiment classification of our basis English sentiment dictionary (bESD) through a Google search engine with AND operator and OR operator. The novel model has firstly been performed in a sequential system and then, we have secondly implemented the proposed model in a parallel network environment.The results of the sequential environment are less than that in the distributed system.We have achieved 89.25% accuracy of the testing data set. The results of the proposed model can widely be used in many commercial applications and surveys of the sentiment classification.

Keywords: *English sentiment classification; parallel system; Cloudera; Hadoop Map and Hadoop Reduce; Fuzzy C-Means; K-Means; OTSUKA coefficient; co-training.*

1. INTRODUCTION

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

The basic principles are proposed:

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.

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

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.

We perform our novel model as follows: we firstly calculate the valences of the sentiment lexicons of the bESD by using the OM through the Google search engine with AND operator and OR operator. One document is transferred into one multi-dimensional vector according to the sentiment lexicons. We transfer the positive documents of the training data set into the positive multi-dimensional vectors based on the sentiment lexicons, called the positive group. We identify the positive group 1 which is a haft of the number of the multi-dimensional vectors of the positive group of the training data set, and we also identify the positive group 2 which is a haft of the number of the multi-dimensional vectors of the positive group of the training data set. The co-training (CT) approach is mainly used for the novel model.We also transfer the negative documents of the training data set into the negative multi-dimensional vectors based on the sentiment leixcons, called the negative group. We identify the negative group 1 which is a haft of the number of the multi-dimensional vectors of the negative group of the training data set, and we also identify the negative group 2 which is a haft of the number of the multi-dimensional vectors of the negative group of the training data set. The co-training model uses the FCM as the first classifier and the KM as the second classifier. The documents of the testing data set are transferred into the multi-dimensional vectors according to the sentiment lexicons. The input of the FCM is the multi-dimensional vectors of the testing data set, the positive group 1 and the negative group 1 of the training data set. The input of the KM is the multi-dimensional vectors of the testing data set, the positive group 2 and the negative group 2 of the training data set. After training this classifier of the CTM of each loop, the 50 multi-dimensional vectors of the testing data set of the first classifier have certainly been chosen, and then, the 50 multi-dimensional vectors of the testing data set of the second classifier have certainly been selected. The 50 multi-dimensional vectors of the testing data set of the first classifier are added to either the positive group 2 or the negative group 2 of the second classifier. The 50 multi-dimensional vectors of the testing data set of the second classifier are added to either the positive group 1 or the negative group 1 of the first classifier.The sentiment classification of all the documents of the testing data set has been identified after many loops of training the classifier of the CTM certainly.

We perform all the above things in the sequential environment to get an accuracy of the result of the sentiment classification and an execution time of the result of the sentiment classification of the proposed model. Then, all the above things are secondly implemented in the parallel network environment to shorten the execution times of the proposed model to get the accuracy of the results of the sentiment classification and the execution times of the results of the sentiment classification of our novel model.The significant contributions of the novel model can be applied to many areas of research as well as commercial applications as follows:
1)Many surveys and commercial applications can use the results of this work in a significant way.
2)The algorithms are built in the proposed model.
3)This survey can certainly be applied to other languages easily.
4)The results of this study can significantly be applied to the types of other words in English.
5)The algorithm of data mining is applicable to semantic analysis of natural language processing.
6)This study also proves that different fields of scientific research can be related in many ways.
7)Millions of English documents are successfully processed for emotional analysis.
8)The sentiment classification is implemented in the parallel network environment.
9)The principles are proposed in the research.
10)The Cloudera distributed environment is used in this study.
11)The proposed work can be applied to other distributed systems.
12)This survey uses Hadoop Map (M) and Hadoop Reduce (R).
13)Our proposed model can be applied to many different parallel network environments such as a Cloudera system
14)This study can be applied to many different distributed functions such as Hadoop Map (M) and Hadoop Reduce (R).
15)TheOM– related equations are proposed in this survey.
16)TheFMC – related algorithms and the KM – related algorithms are built in this study.
17)The CT – related algorithms are proposed in this work.

This study contains 6 sections. Section 1 introduces the study; Section 2 discusses the related works about the OTSUKA coefficient (OM), Fuzzy C-Means algorithm (FCM), K-Means algorithm (KM), co-training (CT) algorithm, etc.; Section 3 is

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

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 OTSUKA coefficient (OM) 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 measues in [1-27]. In the research[1], the authors generated several Norwegian sentiment lexicons by extracting sentiment information from two different types of Norwegian text corpus, namely, news corpus and discussion forums. The methodology was based on the Point wise Mutual Information (PMI). The authors introduced a modification of the PMI that considered small "blocks" of the text instead of the text as a whole, etc.

The surveys related to the similarity coefficients to calculate the valences of words are in [28-32].

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

There are the works related to the OTSUKA coefficient (OM) in [39-44]. The authors in [39] collected 76 binary similarity and distance measures used over the last century and reveal their correlations through the hierarchical clustering technique, etc.

There are the researches related to theFuzzy C-Means algorithm (FCM) in [45-49]. The survey in [45] transmited a FORTRAN-IV coding of the fuzzy c-means (FCM) clustering program, etc.

The surveys related to the K-Means algorithm (KM) in [50-54]. The authors in [51] presented two algorithms which extended the k-means algorithm to categorical domains and domains with mixed numeric and categorical values. The k-modes algorithm used a simple matching dissimilarity measure to deal with categorical objects, replaces the means of clusters with modes, and used a frequency-based method to update modes in the clustering process to minimise the clustering cost function, etc.

The studies related to the Co-Training algorithm are in [55-59]: The authors in [55] proposed a novel Co-Training method for statistical parsing, etc.

There are the works related to vector space modeling (VSM) in [60-62]. In this study [60], the authors examined the Vector Space Model, an Information Retrieval technique and its variation, etc.

The latest researches of the sentiment classification are in [63-65]. In the research [63], the authors presented 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, we built the testing data set including 10,500,000 documents in the movie field, which contains 5,250,000 positive documents and 5,250,000 negative documents in English. All thedocuments in our English testing data set are automatically extracted from English Facebook, English websites and social networks. All the documents must be standardized. All the documents must be pre-processed carefully as follows: online text cleaning, white space removal, expanding abbreviation, stemming, and stop words removal. Then, we labeled positive and negative for them.

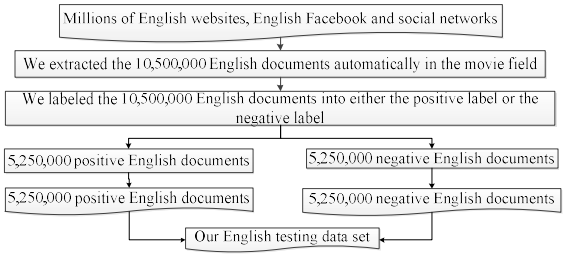

Fig. 1: Our English Testing Data Set.

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

In Fig 2 below, we built the training data set comprising 4,000 documents in the movie field, which contains 2,000 positive and 2,000 negative in English. All the documents in our training data set are automatically extracted from English Facebook, English websites and social networks. All the documents must be standardized. All the documents must be pre-processed carefully as follows: online text cleaning, white space removal, expanding abbreviation, stemming, and stop words removal. Then, we labeled positive and negative for them.

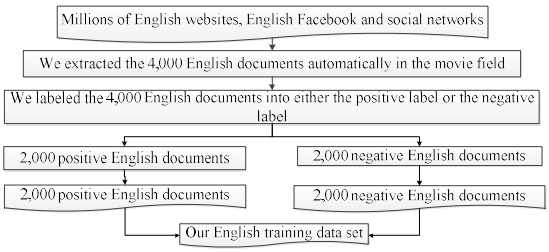

Fig. 2: Our English Training Data Set.

4. METHODOLOGY

We implement the proposed model in Figure 3. This methodology section comprises three parts: (4.1); (4.2); and (4.3).

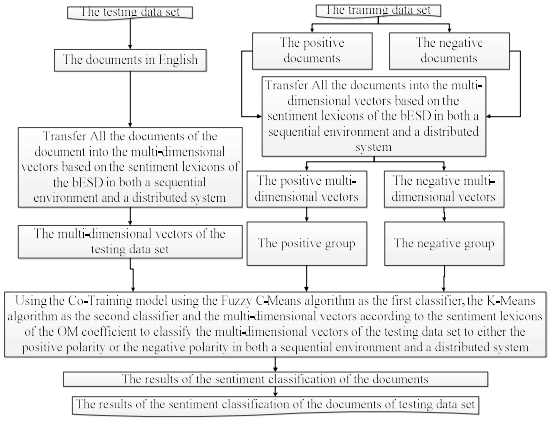

Fig. 3: Overview Of Our Novel Model.

4.1 Creating the sentiment lexicons in English

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

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

In this part, we calculate the valence and the polarity of one English word (or phrase) by using the OM through a Google search engine with AND operator and OR operator, as the following diagram in Figure 4 below shows.

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

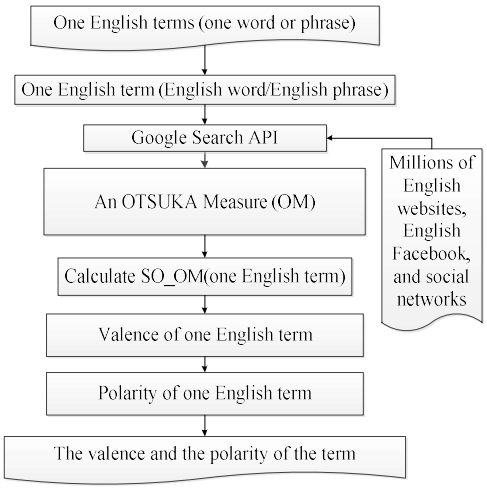

Fig. 4: Overview Of Identifying The Valence And The Polarity Of One Term In English Using The OM

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,

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

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.

The Jaccard equations with the Google search engine in English are used in [14, 15, 17]. [16] and [21] use the Jaccard equations in English. [20] and [22] use the Jaccard equations in Chinese. [18] uses the Jaccard equations in Arabic. The Jaccard equations with the Chinese search engine in Chinese are used in [19].

The authors in [28] used the Ochiai Measure through the Google search engine with AND operator and OR operator to calculate the sentiment values of the words in Vietnamese. The authors in [29] used the Cosine Measure through the Google search engine with AND operator and OR operator to identify the sentiment scores of the words in English. The authors in [30] used the Sorensen Coefficient through the Google search engine with AND operator and OR operator to calculate the sentiment values of the words in English. The authors in [31] used the Jaccard Measure through the Google search engine with AND operator and OR operator to calculate the sentiment values of the words in Vietnamese. The authors in [32] used the Tanimoto Coefficient through the Google search engine with AND operator and OR operator to identify the sentiment scores of the words in English

With the above proofs, we have the information as follows: PMI is used with AltaVista in English, Chinese, and Japanese with the Google in English; Jaccard is used with the Google in English, Chinese, and 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 OTSUKA coefficient (OM)are the similarity measures between two words, and they can perform the same functions and with the same characteristics; so OM is used in calculating the valence of the words. In addition, we prove that OM can be used in identifying the valence of the English word through the Google search with the AND operator and OR operator.

With the OM in [39-44], we have the equation of the OM:

$$\text{OTSUKA Coefficient}(a,b) = \text{OTSUKA Measure}(a,b) = \text{OM}(a,b) = \frac{(a \cap b)}{[[(a \cap b) + (\neg a \cap b)] * [(a \cap b) + (a \cap \neg b)]]^{0.5}} \quad (3)$$

with a and b are the vectors.

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

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

$$\text{OTSUKA Measure}(w1,w2) = \text{OTSUKA Coefficient}(w1,w2) = \text{OM}(w1,w2) = \frac{\text{P}(w1,w2)}{A4} \quad (4)$$

with A4 = $[[\text{P}(w1,w2) + \text{P}(\neg w1,w2)] * [\text{P}(w1,w2) + (\text{P}(w1,\neg w2)]]^{0.5}$

Eq. (3) is similar to eq. (1). In eq. (2), eq. (1) is replaced by eq. (4). We have eq. (5):

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

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

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

with A6 = $[[\text{P}(w, \text{positive\_query}) + \text{P}(\neg w, \text{positive\_query})] * [\text{P}(w, \text{positive\_query}) + (\text{P}(w, \neg \text{positive\_query})]]^{0.5}$

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

$$\text{OM}(w, \text{negative\_query}) = \frac{\text{P}(w, \text{negative\_query})}{[A7} \quad (7)$$

with A7 = $[[\text{P}(w, \text{negative\_query}) + \text{P}(\neg w, \text{negative\_query})] * [\text{P}(w, \text{negative\_query}) + (\text{P}(w, \neg \text{negative\_query})]]^{0.5}$

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

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

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

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_OM(w): valence of English word (or English phrase) w; is SO of word (or phrase) by using the OM
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, ¬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 ((not w) and [not (positive_query)]).
14)P(¬w,negative_query): number of returned results in Google search by keyword ((not w) and negative_query). We use the Google Search API to get the number of returned results in search online Google by keyword ((not w) and negative_query).
15)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))).
16)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 ((not 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 OM (w, positive_query) is as equal as 1.
The English word w is the farthest of positive_query if OM(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 OM(w, positive_query) > 0 and OM(w, positive_query) ≤ 1.
The English word w is the nearest of negative_query if OM(w, negative_query) is as equal as 1.
The English word w is the farthest of negative_query if OM(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 OM(w, negative_query) > 0 and OM(w, negative_query) ≤ 1.
So, the valence of the English word w is the value of OM(w, positive_query) substracting the value of OM(w, negative_query) and the eq. (7) is the equation of identifying the valence of the English word w.

We have the information about OM:
1)OM(w, positive_query) ≥ 0 and OM(w, positive_query) ≤ 1.
2)OM(w, negative_query) ≥ 0 and OM (w, negative_query) ≤ 1
3)IfOM (w, positive_query) = 0 and OM (w, negative_query) = 0 then SO_OM (w) = 0.

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

4)IfOM (w, positive_query) = 1 andOM (w, negative_query) = 0 then SO_OM (w) = 0.
5)IfOM (w, positive_query) = 0 and OM (w, negative_query) = 1 then SO_OM(w) = -1.
6)IfOM (w, positive_query) = 1 and OM (w, negative_query) = 1 then SO_OM(w) = 0.
So, SO_OM (w) ≥ -1 and SO_OM (w) ≤ 1.
The polarity of the English word w is positive polarity If SO_OM (w) > 0. The polarity of the English word w is negative polarity if SO_OM (w) < 0. The polarity of the English word w is neutral polarity if SO_OM (w) = 0. In addition, the semantic value of the English word w is SO_OM (w).

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

In summary, by using eq. (5), eq. (6), and eq. (7), we identify the valence and the polarity of one word (or one phrase) in English by using the OM through the Google search engine with AND operator and OR operator.

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

The comparisons of our model's benefits and drawbacks with the studies related to the OM in [39-44] are shown in Table 2.

4.2.1 Creating a bESD in a sequential environment

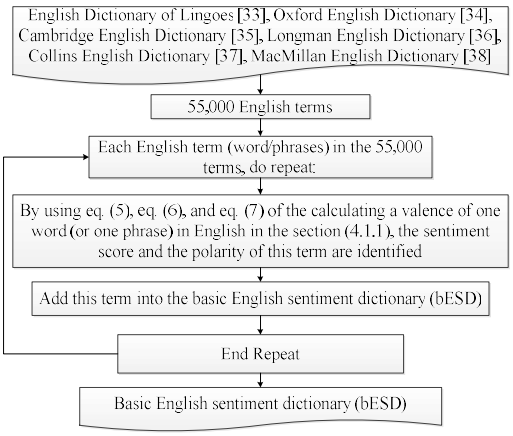

Fig. 5: Overview Of Creating A Besd In A Sequential Environment

In this part, we calculate the valence and the polarity of the English words or phrases for our bESD by using the OM in a sequential system, as the following diagram in Figure 5 below shows.

We proposed the algorithm 1 to perform this section:
Input: the 55,000 English terms; the Google search engine
Output: a bESD
Step 1: Each term in the 55,000 terms, do repeat:
Step 2: By using eq. (5), eq. (6), and eq. (7) of the calculating a valence of one word (or one phrase) in English in the section (4.1.1), the sentiment score and the polarity of this term are identified. The valence and the polarity are calculated by using the OM through the Google search engine with AND operator and OR operator.
Step 3: Add this term into the bESD;
Step 4: End Repeat – End Step 1;
Step 5: Return bESD;

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

4.3.1 Creating a bESD in a distributed system

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

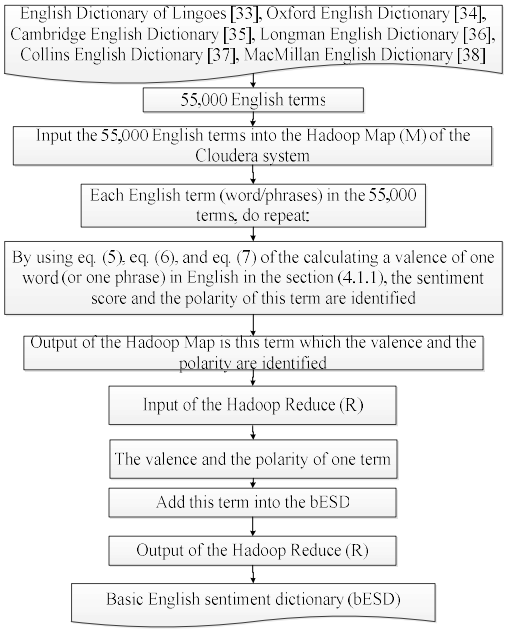

Fig. 6: Overview Of Creating A Besd In A Distributed Environment

In this part, we calculate the valence and the polarity of the English words or phrases for our bESD by using the OM in a parallel network environment, as the following diagram in Figure 6 below shows.

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

We developed the algorithm 2 to implement the Hadoop Map phase of this stage:

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

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

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

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

Step 3: Return this term;

We built the algorithm 3 to perform the Hadoop Reduce phase of this stage:

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

Output: a bESD

Step 1: Add this term into the bESD;

Step 2: Return bESD;

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

4.2 Transferring all the documents of the testing data set and the training data set into the multi-dimensional vectors in a sequential system and a parallel network environment.

In this section, we transfer all the documents of the testing data set and the training data set into the multi-dimensional vectors in a sequential system and a parallel network environment.

The section comprises three sub-sections as follows: (4.2.1) and (4.2.2).

4.2.1 Transferring all the documents of the testing data set and the training data set into the multi-dimensional vectors in a sequential system

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

We proposed the algorithm 4 to transfer one English document into one multi-dimensional vector according to the sentiment lexicons of the bESDin the sequential environment:

Input: one English document

Output: the multi-dimensional vector

Step 1: Split the English document into many separate sentences based on “.” Or “!” or “?”;

Step 2: Set Multi-dimensionalVector := { } { } with n_max rows and m_max columns;

Step 3: Set i := 0;

Step 4: Each sentence in the sentences of this document, do repeat:

Step 5: Multi-dimensionalVector[i][] := {};

Step 6: Set j := 0;

Step 7: Split this sentence into the meaningful terms (meaningful words or meaningful phrases);

Step 8: Get the valence of this term based on the sentiment lexicons of the bESD;

Step 9: Add this term into Multi-dimensionalVector[i];

Step 10: Set j := j+ 1;

Step 11: End Repeat – End Step 4;

Step 12: While j is less than m_max, repeat:

Step 13: Add {0} into Multi-dimensionalVector[i];

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

Step 14: Set j := j+1;
Step 15: End Repeat – End Step 12;
Step 16: Set i := i+1;
Step 17: End Repeat – End Step 4;
Step 18: While i is less than n_max, repeat:
Step 19: Add the vector {0} into Multi-dimensionalVector;
Step 20: Set i := i+1;
Step 21: End Repeat – End Step 18;
Step 22: Return Multi-dimensionalVector;

We proposed the algorithm 5 to transfer all the documents of the testing data set into the multi-dimensional vectors based on the sentiment lexicons of the bESD in the sequential environment:
Input: the documents of the testing data set
Output: the multi-dimensional vectors of the testing data set
Step 1: Set TheMulti-dimensionalVectors := {}
Step 2: Each document in the documents of the testing data set, do repeat:
Step 3: OneMulti-dimensionalVector := the algorithm 4 to transfer one English document into one multi-dimensional vector according to the sentiment lexicons of the bESD in the sequential environment with the input is this document;
Step 4: Add OneMulti-dimensionalVector into TheMulti-dimensionalVectors;
Step 5: End Repeat- End Step 2;
Step 6: Return TheMulti-dimensionalVectors;

We built the algorithm 6 to transfer all the positive documents of the training data set into all the multi-dimensional vectors based on the sentiment lexicons of the bESD, called the positive group of the training data set in the sequential system:
Input: all the positive documents of the training data set;
Output: the positive multi-dimensional vectors, called the positive group - ThePositiveMulti-dimensionalVectors
Step 1: Set ThePositiveMulti-dimensionalVectors := null;
Step 2: Each document in the positive documents, repeat:
Step 3: Multi-dimensionalVector := the algorithm 4 to transfer one English document into one multi-dimensional vector according to the sentiment lexicons of the bESD in the sequential environment with the input is this document;
Step 4: Add Multi-dimensionalVector into ThePositiveMulti-dimensionalVectors;
Step 5: End Repeat – End Step 2;
Step 6: Return ThePositiveMulti-dimensionalVectors;

We implemented the algorithm 7 to transfer all the negative documents of the training data set into all the one-dimensional vectors based on the sentiment lexicons of the bESD, called the negative group of the training data set in the sequential environment:
Input: all the negative sentences of the training data set;
Output the negative multi-dimensional vectors, called the negative vector group - TheNegativeMulti-dimensionalVectors;
Step 1: Set TheNegativeMulti-dimensionalVectors := null;
Step 2: Each document in the negative documents, repeat:
Step 3: Multi-dimensionalVector := the algorithm 4 to transfer one English document into one multi-dimensional vector according to the sentiment lexicons of the bESD in the sequential environment with the input is this document;
Step 4: Add Multi-dimensionalVector into TheNegativeMulti-dimensionalVectors;
Step 5: End Repeat – End Step 2;
Step 6: Return TheNegativeMulti-dimensionalVectors;

We developed the algorithm 8 to create the positive group 1 from the the positive group of the training data set in the sequential system with the positive group 1 is a haft of the number of multi-dimensional vectors of the positive group:
Input: the positive group of the training data set;
Output: the positive group 1
Step 1: Set the positive group 1 := a haft of the number of multi-dimensional vectors of the positive group of the training data set;
Step 2: Return the positive group 1;

We proposed the algorithm 9 to create the positive group 2 from the the positive group of the training data set in the sequential system with the positive group 2 is a haft of the number of multi-dimensional vectors of the positive group:
Input: the positive group of the training data set;
Output: the positive group 2
Step 1: Set the positive group 2 := a haft of the number of multi-dimensional vectors of the positive group of the training data set;
Step 2: Return the positive group 2;

We built the algorithm 10 to create the negative group 1 from the the positive group of the training

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

data set in the sequential system with the negative group 1 is a haft of the number of multi-dimensional vectors of the negative group:
Input: the negative group of the training data set;
Output: the negative group 1
Step 1: Set the negative group 1 := a haft of the number of multi-dimensional vectors of the negative group of the training data set;
Step 2: Return the negative group 1;

We proposed the algorithm 11 to create the negative group 2 from the the negative group of the training data set in the sequential system with the negative group 2 is a haft of the number of multi-dimensional vectors of the negative group:
Input: the negative group of the training data set;
Output: the negative group 2
Step 1: Set the negative group 2 := a haft of the number of multi-dimensional vectors of the negative group of the training data set;
Step 2: Return the negative group 2;

4.2.2 Transferring all the documents of the testing data set and the training data set into the multi-dimensional vectors in a parallel network environment

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

In Figure 7, we transfer one English sentence into one one-dimensional vector based on the sentiment lexicons of the bESD in Cloudera. This stage includes two phases: the Hadoop Map phase and the Hadoop Reduce phase. The input of the Hadoop Map phase is one sentence and the bESD. The output of the Hadoop Map phase is one term (one meaningful word/or one meaningful phrase) which the valence is identified. The input of the Hadoop Reduce phase is the output of the Hadoop Map, thus, the input of the Hadoop Reduce phase is one term (one meaningful word/or one meaningful phrase) which the valence is identified. The output of the Hadoop Reduce phase is one one-dimensional vector of this sentence.

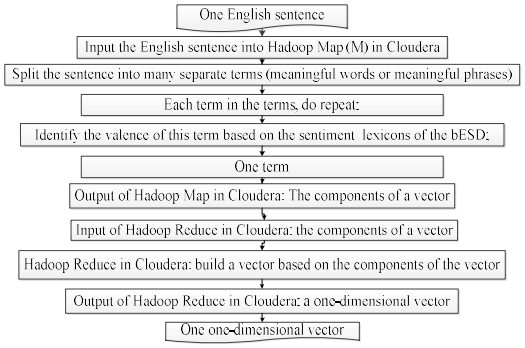

Fig. 7: Overview Of Transforming Each English Sentence Into One One-Dimensional Vector Based On The Sentiment Lexicons Of The Besd In Cloudera

We built the algorithm 12 to perform the Hadoop Map phase
Input: one sentence and the bESD;
Output: one term (one meaningful word/or one meaningful phrase) which the valence is identified
Step 1: Input this sentence and the bESD into the Hadoop Map in the Cloudera system;
Step 2: Split this sentence into the many meaningful terms (meaningful words/or meaningful phrases) based on the bESD;
Step 3: Each term in the terms, do repeat:
Step 4: Identify the valence of this term based on the bESD;
Step 5: Return this term;

We proposed the algorithm 13 to perform the Hadoop Reduce phase
Input:one term (one meaningful word/or one meaningful phrase) which the valence is identified – the output of the Hadoop Map phase
Output: one one-dimensional vector based on the sentiment lexicons of the bESD
Step 1: Receive one term;
Step 2: Add this term into the one-dimentional vector;
Step 3: Return the one-dimentional vector;

In Figure 8, we transfer one document into one multi-dimensional vector based on the sentiment lexicons of the bESD in the parallel system. This stage includes two phases: the Hadoop Map phase and the Hadoop Reduce phase. The input of the Hadoop Map phase is one document. The output of the Hadoop Map is one one-dimensional vector. The input of the Hadoop Reduce is the Hadoop Map, thus, the input of the Hadoop Reduce is one one-dimensional vector. The output of the Hadoop Reduce is the multi-dimensional vector of this document.

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

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

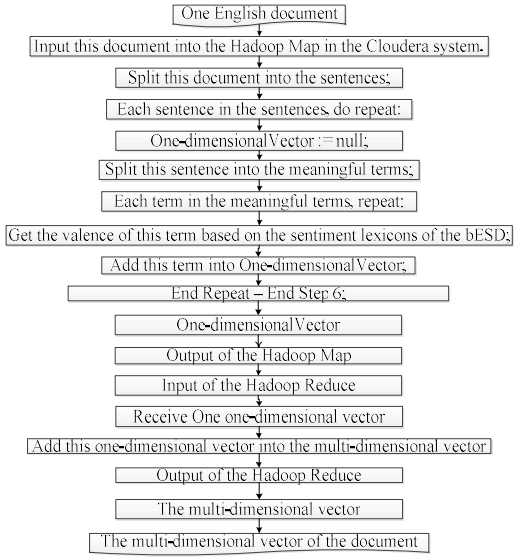

Fig. 8: Overview of transferring one document into one multi-dimensional vector based on the sentiment lexicons of the bESD in the parallel system

We proposed the algorithm 14 to implement the Hadoop Map phase
Input: one document
Output: one one-dimensional vector;
Step 1: Input this document into the Hadoop Map in the Cloudera system.
Step 2: Split this document into the sentences;
Step 3: Each sentence in the sentences, do repeat:
Step 4: One-dimensionalVector := null;
Step 5: Split this sentence into the meaningful terms;
Step 6: Each term in the meaningful terms, repeat:
Step 7: Get the valence of this term based on the sentiment lexicons of the bESD;
Step 8: Add this term into One-dimensionalVector;
Step 9 End Repeat – End Step 6;
Step 10: Return this One-dimensionalVector;
Step 11: The output of the Hadoop Map is this One-dimensionalVector;

We built the algorithm 15 to implement the Hadoop Reduce phase
Input: One-dimensionalVector - one one-dimensional vector of the Hadoop Map (the input of the Hadoop Reduce is the output of the Hadoop Map)
Output: the multi-dimensional vector of the English document – Multi-dimensionalVector;
Step 1: Receive One-dimensionalVector;
Step 2: Add this One-dimensionalVector into One-dimensionalVector;
Step 3: Return Multi-dimensionalVector;

In Figure 9, we transfer the documents of the testing data set into the multi-dimensional vectors of the document based on the sentiment lexicons of the bESD in the parallel system:

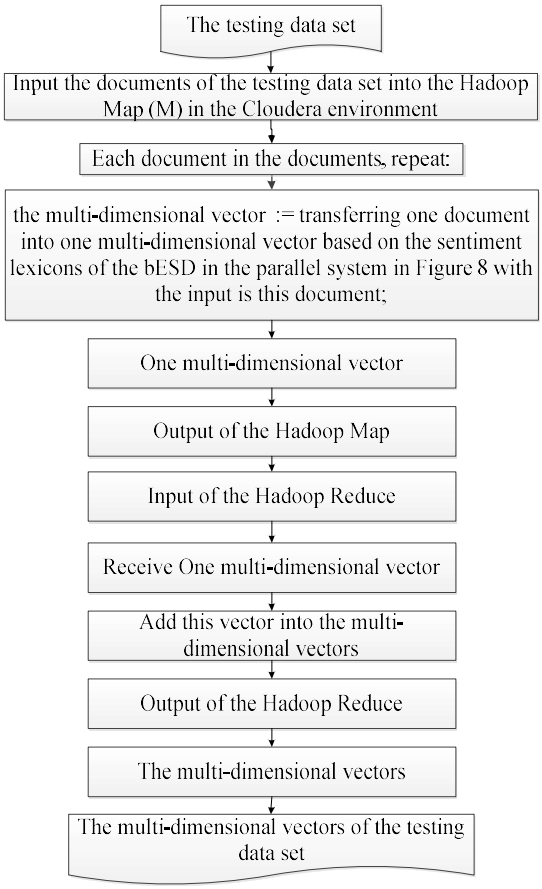

Fig. 9: Overview of transferring the documents of the testing data set into the multi-dimensional vectors of the document based on the sentiment lexicons of the bESD in the parallel system

This stage includes two phases: the Hadoop Map phase and the Hadoop Reduce phase. The input of the Hadoop Map phase is the documents of the testing data set. The output of the Hadoop Mp is one multi-dimensional vector. The input of the Hadoop Reduce is the Hadoop Map, thus, the input of the Hadoop Reduce is one multi-dimensional vector. The output of the Hadoop Reduce is the multi-dimensional vectors of the testing data set

We built the algorithm 16 to implement the Hadoop Map phase
Input: the documents of the testing data set
Output: one multi-dimensional vector (corresponding to one document)
Step 1: Input the documents of the testing data set into the Hadoop Map in the Cloudera system.
Step 3: Each document in the documents of the testing data set, do repeat:

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

Step 4: the multi-dimensional vector := transferring one document into one multi-dimensional vector based on the sentiment lexicons of the bESD in the parallel system in Figure 8 with the input is this document;
Step 5: Return this multi-dimensional vector;
Step 6: The output of the Hadoop Map is this multi-dimensional vector;

We proposed the algorithm 17 to implement the Hadoop Reduce phase
Input: one multi-dimensional vector of the Hadoop Map (the input of the Hadoop Reduce is the output of the Hadoop Map)
Output: the multi-dimensional vectors of the English documents of the testing data set
Step 1: Receive one multi-dimensional vector of the Hadoop Map
Step 2: Add this multi-dimensional vector into the multi-dimensional vectors of the testing data set;
Step 3: Return the multi-dimensional vectors of the testing data set;

In Figure 10, we transfer the positive documents of the training data set into the positive multi-dimensional vectors (called the positive group of the training data set) based on the sentiment lexicons of the bESDin the distributed system. The stage includes two phases: the Hadoop Map (M) phase and the Hadoop Reduce (R) phase. The input of the Hadoop Map phase is the positive documents of the training data set. The output of the Hadoop Map phase is one multi-dimensional vector (corresponding to one document of the positive documents of the training data set). The input of the Hadoop Redude phase is the output of the Hadoop Map phase, thus, the input of the Hadoop Reduce phase isone multi-dimensional vector (corresponding to one document of the positive documents of the training data set). The output of the Hadoop Reduce phase is the positive multi-dimensional vectors, called the positive group (corresponding to the positive documents of the training data set)

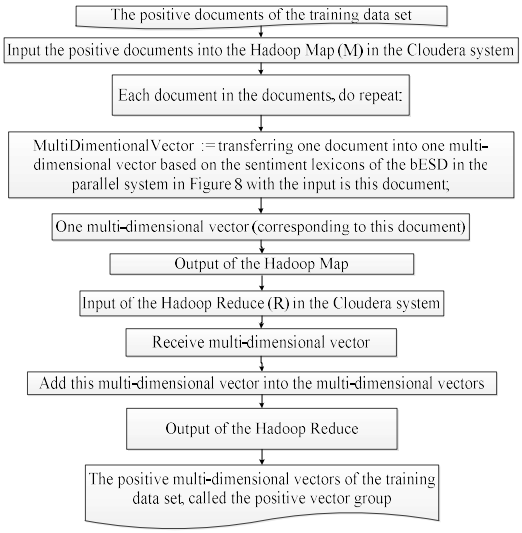

Fig. 10: Overview of transferring the positive documents of the training data set into the positive multi-dimensional vectors (called the positive group) based on the sentiment lexicons of the bESD in the distributed system.

We built the algorithm 18 to perform the Hadoop Map phase
Input: the positive documents of the training data set
Output: one multi-dimensional vector (corresponding to one document of the positive documents of the training data set)
Step 1: Input the positive documents into the Hadoop Map in the Cloudera system.
Step 2: Each document in the documents, do repeat:
Step 3: MultiDimentionalVector := transferring one document into one multi-dimensional vector based on the sentiment lexicons of the bESD in the parallel system in Figure 8 with the input is this document;
Step 4: Return MultiDimentionalVector ;

We proposed the algorithm 19 to implement the Hadoop Reduce phase
Input: one multi-dimensional vector (corresponding to one document of the positive documents of the training data set)
Output: the positive multi-dimensional vectors, called the positive group (corresponding to the positive documents of the training data set)
Step 1: Receive one multi-dimensional vector;
Step 2: Add this multi-dimensional vector into PositiveVectorGroup;
Step 3: Return PositiveVectorGroup - the positive multi-dimensional vectors, called the positive group (corresponding to the positive documents of the training data set);

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

In Figure 11, we transfer the negative documents of the training data set into the negative multi-dimensional vectors (called the negative group of the training data set) based on the sentiment lexicons of the bESD in the distributed system.

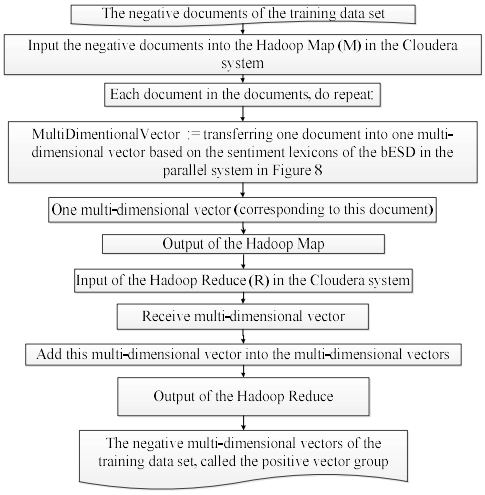

Fig. 11: Overview Of Transferring The Negativedocuments Of The Training Data Set Into The Negativemulti-Dimensional Vectors (Called The Negativegroup) Based On The Sentiment Lexicons Of The Besd In The Distributed System.

The stage includes two phases: the Hadoop Map (M) phase and the Hadoop Reduce (R) phase. The input of the Hadoop Map phase is the negative documents of the training data set. The output of the Hadoop Map phase is one multi-dimensional vector (corresponding to one document of the negative documents of the training data set). The input of the Hadoop Redude phase is the output of the Hadoop Map phase, thus, the input of the Hadoop Reduce phase isone multi-dimensional vector (corresponding to one document of the negative documents of the training data set). The output of the Hadoop Reduce phase is the negative multi-dimensional vectors, called the negative group (corresponding to the negativedocuments of the training data set)

We built the algorithm 20 to perform the Hadoop Map phase

Input: the negative documents of the training data set

Output: one multi-dimensional vector (corresponding to one document of the negative documents of the training data set)

Step 1: Input the negative documents into the Hadoop Map in the Cloudera system.

Step 2: Each document in the documents, do repeat:

Step 3: MultiDimentionalVector := the transferring one document into one multi-dimensional vector based on the sentiment lexicons of the bESDin the parallel system in Figure 7

Step 4: Return MultiDimentionalVector ;

We proposed the algorithm 21 to implement the Hadoop Reduce phase

Input: one multi-dimensional vector (corresponding to one document of the negativedocuments of the training data set)

Output: the negative multi-dimensional vectors, called the negativegroup (corresponding to the negativedocuments of the training data set)

Step 1: Receive one multi-dimensional vector;

Step 2: Add this multi-dimensional vector into NegativeVectorGroup;

Step 3: Return NegativeVectorGroup - the megative multi-dimensional vectors, called the negativegroup (corresponding to the negativedocuments of the training data set);

4.3 Using the Co-Training model with the Fuzzy C-Means algorithm as the first classifier, the K-Means algorithm as the second classifier and the multi-dimensional vectors according to the sentiment lexicons of the OTSUKA coefficient to classify the documents of the testing data set into either the positive vector group or the negative vector group in both a sequential environment and a distributed system.

In section, we use the Co-Training model with the Fuzzy C-Means algorithm as the first classifier, the K-Means algorithm as the second classifier and the multi-dimensional vectors according to the sentiment lexicons of the OTSUKA coefficient to classify the documents of the testing data set into either the positive vector group or the negative vector group in both a sequential environment and a distributed system.

The section compises two parts as follows: (4.3.1) and (4.3.2).

4.3.1 Using the Co-Training model with the Fuzzy C-Means algorithm as the first classifier, the K-Means algorithm as the second classifier and the multi-dimensional vectors according to the sentiment lexicons of the OTSUKA coefficient to classify the documents of the testing data set into either the positive vector group or the negative vector group in a sequential environment.

In this section, we use the Co-Training model with the Fuzzy C-Means algorithm as the first classifier, the K-Means algorithm as the second classifier and the multi-dimensional vectors according to the sentiment lexicons of the OTSUKA coefficient to classify the documents of

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

the testing data set into either the positive vector group or the negative vector group in a sequential environment in Figure 12:

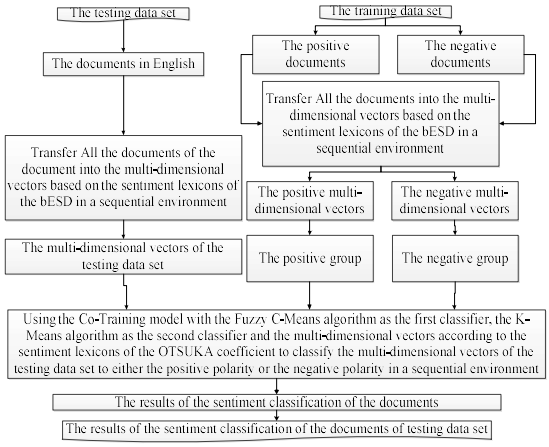

Fig. 12: Overview Of Our Novel Model In A Sequential System.

Based on the studies related to the Fuzzy C-Means algorithm (FCM) in [45-49], 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

According to the surveys related to the K-Means algorithm (KM) in [50-54], the main ideas of the the KM are as follows:

1)Place K points into the space represented by the objects that are being clustered. These points represent initial group centroids.

2)Assign each object to the group that has the closest centroid.

3)When all objects have been assigned, recalculate the positions of the K centroids.

4)Repeat Steps 2 and 3 until the centroids no longer move. This produces a separation of the objects into groups from which the metric to be minimized can be calculated.

We built the algorithm 22 to classify all the documents of the testing data set into either the positive or the negative in the sequential system by using the Co-Training model with the Fuzzy C-Means algorithm as the first classifier, the K-Means algorithm as the second classifier and the multi-dimensional vectors according to the sentiment lexicons of the OTSUKA coefficient:

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

Output: the results of the sentiment classification of the documents of the testing data set(positive, negative, or neutral);

Step 1: the creating a bESD in a sequential environment (4.1.2);

Step 2: the algorithm 6 to transfer all the positive documents of the training data set into all the multi-dimensional vectors based on the sentiment lexicons of the bESD (called the positive group) in the sequential system

Step 3: the algorithm 7 to transfer all the negative documents of the training data set into all the multi-dimensional vectors based on the sentiment lexicons of the bESD (called the negative group) in the sequential environment.

Step 4: the algorithm 5 to transfer all the documents of the testing data set into the multi-dimensional vectors based on the sentiment lexicons of the bESD in the sequential environment.

Step 5: An initial collection U of unlabeled examples, set U := the multi-dimensional vectors of the testing data set;

Step 6: Training set L1 for classifer h1, L1 = the positive group 1 and the negative group 1 of the training data set;

Step 7:Training set L2 for classifer h2, L2 = the positive group 2 and the negative group 2 of the training data set;

Step 8: While (U is not empty), repeat:

Step 9: Using the KM with the input is U, and L1 according to the surveys related to the K-Means algorithm (KM) in [50-54];

Step 10: Using the FCM with the input is U, and L2 based on the studies related to the Fuzzy C-Means algorithm (FCM) in [45-49];

Step 11: Selecting the 50 multi-dimensional vectors from the best results of the sentiment classification of the KM;

Step 12: U := U - the 50 multi-dimensional vectors;

Step 13: Add the 50 multi-dimensional vectors into either the positive group 2 or the negative group 2 of L2;

Step 14: Choosing the 50 multi-dimensional vectors from the best results of the sentiment classification of the FCM;

Step 15: U := U - the 50 multi-dimensional vectors;

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

Step 16: Add the 50 multi-dimensional vectors into either the positive group 1 or the negative group 1 of L1;
Step 17: End Repeat – End Step 8;
Step 18: Return the results of the sentiment classification of the documents of the testing data set (positive, negative, or neutral);

4.3.2 Using the Co-Training model with the Fuzzy C-Means algorithm as the first classifier, the K-Means algorithm as the second classifier and the multi-dimensional vectors according to the sentiment lexicons of the OTSUKA coefficient to classify the documents of the testing data set into either the positive vector group or the negative vector group in a distributed system.

In this section, we use the self-training model withthe Co-Training model using the Fuzzy C-Means algorithm as the first classifier, the K-Means algorithm as the second classifier and the multi-dimensional vectors according to the sentiment lexicons of the OTSUKA coefficient to classify the documents of the testing data set into either the positive vector group or the negative vector group in a distributed network environmentin Figure 13:

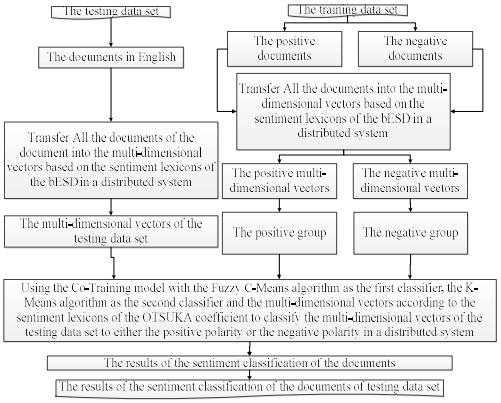

Fig. 13: Overview Of Our Novel Model In A Parallel Network System.

In Figure 14, we create the positive group 1 from the the positive group of the training data set in the sequential system with the positive group 1 is a haft of the number of multi-dimensional vectors of the positive groupin the distributed system. This stage comprises two phases: the Hadoop Map phase and the Hadoop Reduce phase. The input of the Hadoop Map is the positive group of the training data set. The output of the Hadoop Map is the positive group 1. The input of the Hadoop Reduce is the output of the Hadoop Map, thus, the input of the Hadoop Reduce is the positive group 1. The output of the Hadoop Reduce is the positive group 1

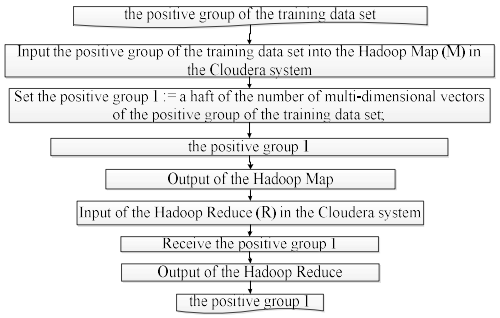

Fig. 14: Overview Of Creating The Positive Group 1 From The The Positive Group Of The Training Data Set In The Sequential System With The Positive Group 1 Is A Haft Of The Number Of Multi-Dimensional Vectors Of The Positive Group In The Distributed System.

We built the algorithm 23 to perform the Hadoop Map phase
Input: the positive group of the training data set;
Output: the positive group 1;
Step 0: Input the positive group of the training data set into the Hadoop Map in the Cloudera system;
Step 1: Set the positive group 1 := a haft of the number of multi-dimensional vectors of the positive group of the training data set;
Step 2: Return the positive group 1;

We proposed the algorithm 24 to implement the Hadoop Reduce phase
Input: the positive group 1;
Output: the positive group 1
Step 1: Receive the positive group 1;
Step 2: Return the positive group 1;

In Figure 15, we create the positive group 2 from the the positive group of the training data set in the sequential system with the positive group 2 is a haft of the number of multi-dimensional vectors of the positive group in the distributed system. This stage comprises two phases: the Hadoop Map phase and the Hadoop Reduce phase. The input of the Hadoop Map is the positive group of the training data set. The output of the Hadoop Map is the positive group 2. The input of the Hadoop Reduce is the output of the Hadoop Map, thus, the input of the Hadoop Reduce is the positive group 2. The output of the Hadoop Reduce is the positive group 2

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

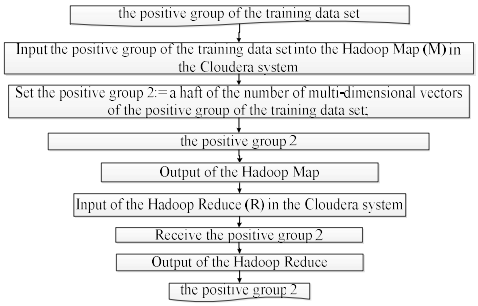

Fig. 15: Overview Of Creating The Positive Group 2 From The The Positive Group Of The Training Data Set In The Sequential System With The Positive Group 2 Is A Haft Of The Number Of Multi-Dimensional Vectors Of The Positive Group In The Distributed System.

We proposed the algorithm 25 to perform the Hadoop Map phase
Input: the positive group of the training data set;
Output: the positive group 2;
Step 0: Input the positive group of the training data set into the Hadoop Map in the Cloudera system;
Step 1: Set the positive group 2 := a haft of the number of multi-dimensional vectors of the positive group of the training data set;
Step 2: Return the positive group 2;

We built the algorithm 26 to implement the Hadoop Reduce phase
Input: the positive group 2;
Output: the positive group 2
Step 1: Receive the positive group 2;
Step 2: Return the positive group 2;

In Figure 16, we create the negative group 1 from the the negative group of the training data set in the sequential system with the negative group 1 is a haft of the number of multi-dimensional vectors of the negative group in the distributed system. This stage comprises two phases: the Hadoop Map phase and the Hadoop Reduce phase. The input of the Hadoop Map is the negativegroup of the training data set. The output of the Hadoop Map is the negative group 1. The input of the Hadoop Reduce is the output of the Hadoop Map, thus, the input of the Hadoop Reduce is the negativegroup 1. The output of the Hadoop Reduce is the negativegroup 1

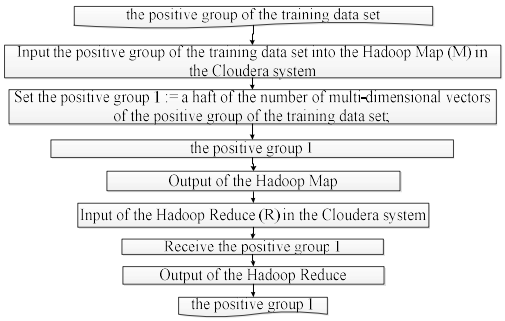

Fig. 16: Overview Of Creating The Negative Group 1 From The The Positive Group Of The Training Data Set In The Sequential System With The Negative Group 1 Is A Haft Of The Number Of Multi-Dimensional Vectors Of The Negative Group In The Distributed System.

We built the algorithm 27 to perform the Hadoop Map phase
Input: the negative group of the training data set;
Output: the negative group 1;
Step 0: Input the negative group of the training data set into the Hadoop Map in the Cloudera system;
Step 1: Set the negativegroup 1 := a haft of the number of multi-dimensional vectors of the negativegroup of the training data set;
Step 2: Return the negativegroup 1;

We proposed the algorithm 28 to implement the Hadoop Reduce phase
Input: the negativegroup 1;
Output: the negativegroup 1
Step 1: Receive the negativegroup 1;
Step 2: Return the negativegroup 1;

In Figure 17, we create the negativegroup 2 from the the negativegroup of the training data set in the sequential system with the negative group 2 is a haft of the number of multi-dimensional vectors of the negative group in the distributed system. This stage comprises two phases: the Hadoop Map phase and the Hadoop Reduce phase. The input of the Hadoop Map is the negativegroup of the training data set. The output of the Hadoop Map is the negative group 2. The input of the Hadoop Reduce is the output of the Hadoop Map, thus, the input of the Hadoop Reduce is the negative group 2. The output of the Hadoop Reduce is the negative group 2

We proposed the algorithm 29 to perform the Hadoop Map phase
Input: the negative group of the training data set;
Output: the negative group 2;
Step 0: Input the negative group of the training data set into the Hadoop Map in the Cloudera

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

system;
Step 1: Set the negative group 2 := a haft of the number of multi-dimensional vectors of the negativegroup of the training data set;
Step 2: Return the negative group 2;

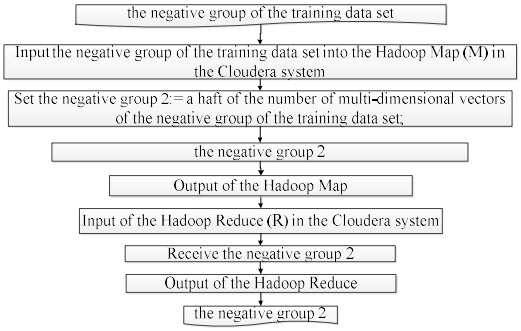

Fig. 17: Overview of creating the negative group 2 from the the negative group of the training data set in the sequential system with the negative group 2 is a haft of the number of multi-dimensional vectors of the negative group in the distributed system.

We built the algorithm 30 to implement the Hadoop Reduce phase
Input: the negativegroup 2;
Output: the positive group 2
Step 1: Receive the negativegroup 2;
Step 2: Return the negativegroup 2;

Based on the studies related to the Fuzzy C-Means algorithm (FCM) in [45-49], 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

According to the surveys related to the K-Means algorithm (KM) in [50-54], the main ideas of the the KM are as follows:
1)Place K points into the space represented by the objects that are being clustered. These points represent initial group centroids.
2)Assign each object to the group that has the closest centroid.
3)When all objects have been assigned, recalculate the positions of the K centroids.
4)Repeat Steps 2 and 3 until the centroids no longer move. This produces a separation of the objects into groups from which the metric to be minimized can be calculated.

In Figure 18, we use the Co-Training model with the Fuzzy C-Means algorithm as the first classifier, the K-Means algorithm as the second classifier and the multi-dimensional vectors according to the sentiment lexicons of the OTSUKA coefficient to classify the documents of the testing data set into either the positive or the negative in the distributed system. This stage comprises two phases: the Hadoop Map phase and the Hadoop Reduce phase. The input of the Hadoop Map is thedocuments of the testing data set and the training data set. The output of the Hadoop Map is the 50 multi-dimensional vectors from the best results of the sentiment classification of the KM of the first classifier and the 50 multi-dimensional vectors from the best results of the sentiment classification of the FCM of the second classifier. The input of the Hadoop Reduce is the output of the Hadoop Map, thus, the input of the Hadoop Reduce is the 50 multi-dimensional vectors from the best results of the sentiment classification of the KM of the first classifier and the 50 multi-dimensional vectors from the best results of the sentiment classification of the FCM of the second classifier. The output of the Hadoop Reduce is the results of the sentiment classification of the documents of the testing data set.

We built the algorithm 31 to perform the Hadoop Map phase
Input: the documents of the testing data set and the training data set
Output: the 50 multi-dimensional vectors from the best results of the sentiment classification of the KM of the first classifier and the 50 multi-dimensional vectors from the best results of the sentiment classification of the FCM of the second classifier;
Step 1: Creating a basis English sentiment dictionary (bESD) in a parallel environment (4.1.3);
Step 2: Transferring the documents of the testing data set into the multi-dimensional vectors of the document based on the sentiment lexicons of the bESD in the parallel system Figure 9
Step 3: Transferring the positive documents of the training data set into the positive multi-dimensional vectors (called the positive group of the training data set) based on the sentiment lexicons of the bESD in the distributed system in Figure 10
Step 4: Transferring the negative documents of the training data set into the negative multi-

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

dimensional vectors (called the negative group of the training data set) based on the sentiment lexicons of the bESD in the distributed system in Figure 11
Step 5: Creating the positive group 1 from the the positive group of the training data set in the sequential system with the positive group 1 is a haft of the number of multi-dimensional vectors of the positive group in the distributed system in Figure 14
Step 6: Creating the positive group 2 from the the positive group of the training data set in the sequential system with the positive group 2 is a haft of the number of multi-dimensional vectors of the positive group in the distributed system in Figure 15
Step 7: Creating the negative group 1 from the the negative group of the training data set in the sequential system with the negative group 1 is a haft of the number of multi-dimensional vectors of the negative group in the distributed system in Figure 16
Step 8: Creating the negative group 2 from the the negative group of the training data set in the sequential system with the negative group 2 is a haft of the number of multi-dimensional vectors of the negative group in the distributed system in Figure 17
Step 9: Input the multi-dimensional vectors of the testing data set, the positive group and the negative group of the training data set into the Hadoop Map in the Cloudera;
Step 10: An initial collection U of unlabeled examples, set U := the multi-dimensional vectors of the testing data set;
Step 11: Training set L1 for classifer h1, L1 = the positive group 1 and the negative group 1 of the training data set;
Step 12: Training set L2 for classifer h2, L2 = the positive group 2 and the negative group 2 of the training data set;
Step 13: While (U is not empty), repeat:
Step 14: Using the KM with the input is U, and L1 according to the surveys related to the K-Means algorithm (KM) in [50-54];
Step 15: Using the FCM with the input is U, and L2 based on the studies related to the Fuzzy C-Means algorithm (FCM) in [45-49];
Step 16: Selecting the 50 multi-dimensional vectors from the best results of the sentiment classification of the KM;
Step 17: U := U - the 50 multi-dimensional vectors;
Step 18: Add the 50 multi-dimensional vectors into either the positive group 2 or the negative group 2 of L2;
Step 19: Choosing the 50 multi-dimensional vectors from the best results of the sentiment classification of the FCM;
Step 20: U := U - the 50 multi-dimensional vectors;
Step 21: Add the 50 multi-dimensional vectors into either the positive group 1 or the negative group 1 of L1;
Step 22: End Repeat – End Step 8;
Step 23: Return the 50 multi-dimensional vectors from the best results of the sentiment classification of the KM of the first classifier and the 50 multi-dimensional vectors from the best results of the sentiment classification of the FCM of the second classifier;

We proposed the algorithm 32 to implement the Hadoop Reduce phase
Input: the 50 multi-dimensional vectors from the best results of the sentiment classification of the KM of the first classifier and the 50 multi-dimensional vectors from the best results of the sentiment classification of the FCM of the second classifier;
Output: the results of the sentiment classification of the documents of the testing data set (positive, negative, or neutral);
Step 1:Receive the 50 multi-dimensional vectors from the best results of the sentiment classification of the KM of the first classifier and the 50 multi-dimensional vectors from the best results of the sentiment classification of the FCM of the second classifier;
Step 2: Add them into the results of the sentiment classification of the documents of the testing data set (positive, negative, or neutral);
Step 3: Return the results of the sentiment classification of the documents of the testing data set (positive, negative, or neutral);

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

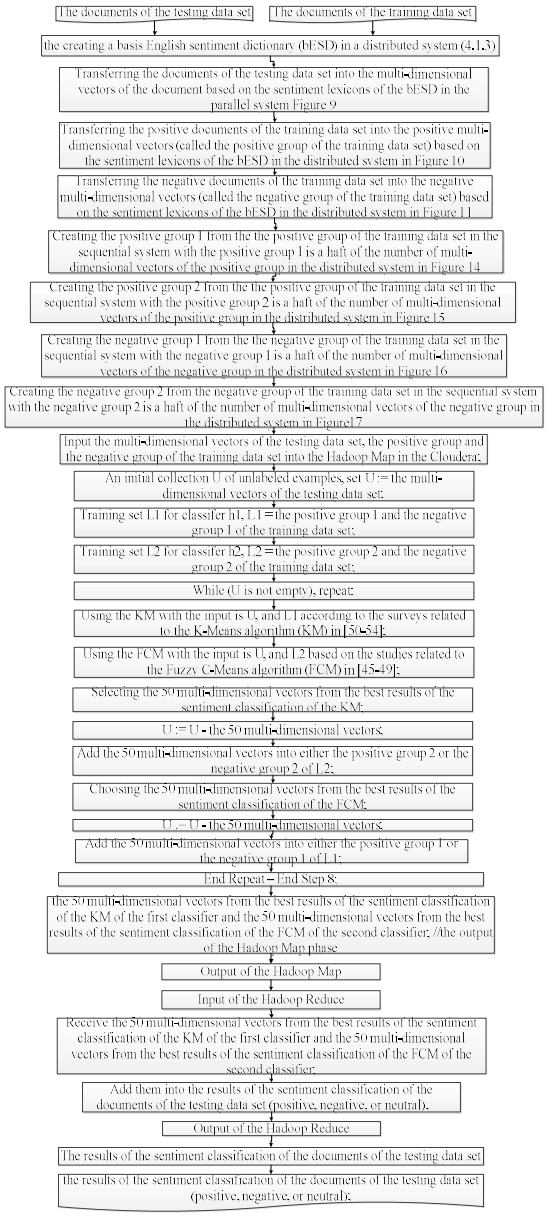

Fig. 18: Overview of using the Co-Training model with the Fuzzy C-Means algorithm as the first classifier, the K-Means algorithm as the second classifier and the multi-dimensional vectors according to the sentiment lexicons of the OTSUKA coefficient to classify the documents of the testing data set into either the positive or the negative in the distributed system.

5. EXPERIMENT

We have measured an Accuracy (A) to calculate the accuracy of the results of emotion classification.

We used a Java programming language for programming to save data sets, implementing our proposed model to classify the 10,500,000 documents of the testing data set and the 4,000 documents of the training data set. To implement the proposed model, we have already used the Java programming language to save the English testing data set and to save the results of emotion classification.

The proposed model was implemented in both the sequential system and the distributed network environment.

Our model related to the Co-Training model using the Fuzzy C-Means algorithm as the first classifier, the K-Means algorithm as the second classifier and the multi-dimensional vectors according to the sentiment lexicons of the OTSUKA coefficient is implemented in the sequential environment with the configuration as follows: The sequential environment in this research includes 1 node (1 server). The configuration of the server in the sequential environment is: Intel® Server Board S1200V3RPS, Intel® Pentium® Processor G3220 (3M Cache, 3.00 GHz), 2GB CC3-10600 ECC 1333 MHz LP Unbuffered DIMMs. The operating system of the server is: Cloudera. The Java language is used in programming our model related to the Co-Training model using the Fuzzy C-Means algorithm as the first classifier, the K-Means algorithm as the second classifier and the multi-dimensional vectors according to the sentiment lexicons of the OTSUKA coefficient.

The novel model related to the self-training model using the Co-Training model using the Fuzzy C-Means algorithm as the first classifier, the K-Means algorithm as the second classifier and the multi-dimensional vectors according to the sentiment lexicons of the OTSUKA coefficient is performed in the Cloudera parallel network environment with the configuration as follows: This Cloudera system includes 9 nodes (9 servers). The configuration of each server in the Cloudera system is: Intel® Server Board S1200V3RPS, Intel® Pentium® Processor G3220 (3M Cache, 3.00 GHz), 2GB CC3-10600 ECC 1333 MHz LP Unbuffered DIMMs. The operating system of each server in the 9 servers is: Cloudera. All 9 nodes have the same configuration information. The Java language is used in programming the application of the proposed model related to the Co-Training model using the Fuzzy C-Means algorithm as the first classifier, the K-Means algorithm as the second classifier and the multi-dimensional vectors according to the sentiment lexicons of the OTSUKA coefficient in the Cloudera

In Table 3, we present the results of the documents in the testing data set and the accuracy

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

of our novel model for the documents in the testing data set.

The average times of the classification of our new model for the documents in testing data set are displayed in Table 4.

6. CONCLUSION

In this survey, a new model has been proposed to classify sentiment of many documents in English using the Co-Training model withthe Fuzzy C-Means algorithm as the first classifier, the K-Means algorithm as the second classifier and the multi-dimensional vectors according to the sentiment lexicons of the OTSUKAcoefficientwith Hadoop Map (M) /Reduce (R) in the Cloudera parallel network environment. Based on our proposed new model, we have achieved 89.29% accuracy of the testing data set in Table 3. Until now, not many studies have shown that the clustering methods can be used to classify data. Our research shows that clustering methods are used to classify data and, in particular, can be used to classify the sentiments(positive, negative, or neutral) in text.

The proposed model can be applied to other languages although our new model has been tested on our English data set. Our model can be applied to larger data sets with millions of English documents in the shortest time although our model has been tested on the documents of the testing data set in which the data sets are small in this survey.

According to Table 4, the average time of the sentiment classification of usingthe Co-Training model with the Fuzzy C-Means algorithm as the first classifier, the K-Means algorithm as the second classifier and the multi-dimensional vectors according to the sentiment lexicons of the OTSUKA coefficient in the sequential environment is 44,206,310 seconds/10,500,000 English documents and it is greater than the average time of the sentiment classification of using the Co-Training model with the Fuzzy C-Means algorithm as the first classifier, the K-Means algorithm as the second classifier and the multi-dimensional vectors according to the sentiment lexicons of the OTSUKA coefficient in the Cloudera parallel network environment with 3 nodes which is 13,402,103 seconds/10,500,000 English documents. The average time of the sentiment classification of using the Co-Training model with the Fuzzy C-Means algorithm as the first classifier, the K-Means algorithm as the second classifier and the multi-dimensional vectors according to the sentiment lexicons of the OTSUKA coefficientin the Cloudera parallel network environment with 9 nodes is 4,934,034 seconds/10,500,000 English documents, and It is the shortest time in the table. Besides, the average time of the sentiment classification of using the Co-Training model with the Fuzzy C-Means algorithm as the first classifier, the K-Means algorithm as the second classifier and the multi-dimensional vectors according to the sentiment lexicons of the OTSUKA coefficientin the Cloudera parallel network environment with 6 nodes is7,401,051seconds /10,500,000 English documents

The accuracy of the proposed model is dependent on many factors as follows:

1)The co-training – related algorithms
2)The testing data set
3)The documents of the testing data set must be standardized carefully.
4)Transferring one document into one multi-dimensional vector based on the sentiment lexicons.
5)TheOM – related equations.
6)The FMC – related algorithms and the KM – related algorithms.

The execution time of the proposed model is dependent on many factors as follows:

1)The parallel network environment such as the Cloudera system.
2)The distributed functions such as Hadoop Map (M) and Hadoop Reduce (R).
3)The co-training – related algorithms
4)The performance of the distributed network system.
5)The number of nodes of the parallel network environment.
6)The performance of each node (each server) of the distributed environment.
7)The sizes of the training data set and the testing data set.
8)Transferring one document into one multi-dimensional vector according to the sentiment lexicons.
9)The Google search engine.
10)TheOM – related equations.
11)The FCM – related algorithms and the KM – related algorithm.

The proposed model has many advantages and disadvantages. Its positives are as follows: It uses the Co-Training model with the Fuzzy C-Means algorithm as the first classifier, the K-Means algorithm as the second classifier and the multi-dimensional vectors according to the sentiment lexicons of the OTSUKA coefficient with the multi-dimensional vectors based on the sentiment lexicons to classify the sentiments 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

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

systems to shorten the execution time of the proposed model. It can be applied to other languages. Its negatives are as follows: It has a low rate of accuracy. It costs too much and takes too much time to implement this proposed model.

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

In Table 5, we show the comparisons of our model's benefits and drawbacks with the studies related to the Fuzzy C-Means algorithm (FCM) in [45-49]

The comparisons of our model's benefits and drawbacks with the studies related to the K-Means algorithm (KM) in [50-54] are presented in Table 6.

In Table 7, we display the comparisons of our model's benefits and drawbacks with the studies related to the Co-Training algorithm in [55-59]

The comparisons of our model's advantages and disadvantages with the works in [60-62] are shown in Table 8.

In Table 9, we present the comparisons of our model's positives and negatives the latest sentiment classification models (or the latest sentiment classification methods) in [63-65]

REFRENCES:

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

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

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

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

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

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

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

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

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

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

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

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

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

[14] Shi Feng, Le Zhang, Binyang Li Daling Wang, Ge Yu, Kam-Fai Wong, "*Is Twitter A Better Corpus for Measuring Sentiment Similarity?* ", Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, USA, pp 897–902, 2013

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

[16] Nihalahmad R. Shikalgar, Arati M. Dixit, "*JIBCA: Jaccard Index based Clustering*

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

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, pp 521-543, 2012

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

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

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

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

[26] Snežana Mladenović Drinić, Ana Nikolić, Vesna Perić, "*Cluster Analysis Of Soybean Genotypes Based On RAPD Markers*", Proceedings. 43rd Croatian And 3rd International Symposium On Agriculture. Opatija. Croatia, 367- 370, 2008

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

[28] Vo Ngoc Phu, Vo Thi Ngoc Chau, Vo Thi Ngoc Tran, Nguyen Duy Dat, "*A Vietnamese adjective emotion dictionary based on exploitation of Vietnamese language characteristics*", International Journal of Artificial Intelligence Review (AIR), doi:10.1007/OTSUKA462-017-9538-6, 67 pages, 2017

[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/OTSUKA115-017-1054-0, 30 pages, 2017

[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/OTSUKA772-017-9420-6, 28 pages, 2017

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

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

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

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

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

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

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

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

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

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

[40] J. John Sepkoski Jr. , "*Quantified coefficients of association and measurement of similarity*", Journal of the International Association for Mathematical Geology, Volume 6, Issue 2, pp 135–152, 1973

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

[42] Shiraishi, Minoru M.D.; Mizuta, Hiroshi M.D.; KubOTa, Kenji M.D.; OTsuka, Yutaka M.D.; NagamOTo, Noriyoshi M.D.; Takagi, Katsumasa M.D. , "*Stabilometric Assessment in the Anterior Cruciate Ligament-Reconstructed Knee*", Clinical Journal of Sport Medicine, http://journals.lww.com/cjsportsmed/abstract/1996/01000/stabilometric_assessment_in_the_anterior_cruciate.8.aspx, 1996

[43] Sony Hartono Wijaya, Farit Mochamad Afendi, Irmanida Batubara, Latifah K. Darusman, Md Altaf-Ul-Amin, Shigehiko Kanaya, "*Finding an appropriate equation to measure similarity between binary vectors: Case studies on Indonesian and Japanese herbal medicines*", BMC Bioinformatics BMC series – open, inclusive and trusted 2016 17:520, https://doi.org/10.1186/s12859-016-1392-z, 2016

[44] K. A. Harris, T. H. Myers, R. W. Yanka, L. M. Mohnkern, N. OTsuka, "*A high quantum efficiency in situ doped mid-wavelength infrared p-on-n homojunctionsuperlattice detector grown by phOToassisted molecular-beam epitaxy. Journal of Vacuum Science & Technology B, NanOTechnology and Microelectronics: Materials*", Processing, Measurement, and Phenomena 9, 1752 (1991); doi: http://dx.doi.org/10.1116/1.585411, 1998

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

[46] M.N. Ahmed; S.M. Yamany; N. Mohamed; A.A. Farag; T. Moriarty, "*A modified fuzzy c-means algorithm for bias field estimation and segmentation of MRI data*", IEEE Transactions on Medical Imaging, Volume: 21, Issue: 3, DOI: 10.1109/42.996338, 2002

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

[48] 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

[49] 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

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

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

[52] Liping Jing; Michael K. Ng; Joshua Zhexue Huang, "*An Entropy Weighting k-Means Algorithm for Subspace Clustering of High-Dimensional Sparse Data*", IEEE Transactions on Knowledge and Data Engineering, Volume: 19, Issue: 8, DOI: 10.1109/TKDE.2007.1048, 2007

[53] Mu-Chun Su; Chien-Hsing Chou, "*A modified version of the K-means algorithm with a*

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

distance based on cluster symmetry", IEEE Transactions on Pattern Analysis and Machine Intelligence, Volume: 23, Issue: 6, DOI: 10.1109/34.927466, 2001

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

[55] Anoop Sarkar, "*Applying co-training methods to statistical parsing*", NAACL '01 Proceedings of the second meeting of the North American Chapter of the Association for Computational Linguistics on Language technologies, Pages 1-8, Pittsburgh, Pennsylvania, 2001

[56] Jia Liu; Chun Chen; Jiajun Bu; Mingyu You; Jianhua Tao, "*Speech Emotion Recognition using an Enhanced Co-Training Algorithm*", IEEE International Conference on Multimedia and Expo,DOI: 10.1109/ICME.2007.4284821 , Beijing, China, 2007

[57] Levin; Viola; Freund, "*Unsupervised improvement of visual detectors using cotraining*", Proceedings. Ninth IEEE International Conference on Computer Vision, DOI: 10.1109/ICCV.2003.1238406 , Nice, France, 2003

[58] Xiaojun Wan Peking, "*Co-training for cross-lingual sentiment classification*", ACL '09 Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 1 - Volume 1 , Pages 235-243 , Suntec, Singapore, 2009

[59] Mi Zhang, Jie Tang, Xuchen Zhang, Xiangyang Xue, "*Addressing cold start in recommender systems: a semi-supervised co-training algorithm*", SIGIR '14 Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval, Pages 73-82 , Gold Coast, Queensland, Australia, 2014

[60] 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

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

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

[63] Basant Agarwal, Namita Mittal, "*Machine Learning Approach for Sentiment Analysis*", Prominent Feature Extraction for Sentiment Analysis, Print ISBN 978-3-319-25341-1, 10.1007/978-3-319-25343-5_3, 21-45, 2016

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

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

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

APPENDICES:

Table 1: 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 surveythey used a distance of 100 words from the seed word, but it mightbe that other lengths that generate better sentiment lexicons. Some of the authors' preliminary research showed that 100 gave a better result.	No mention
[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 ofHatzivassiloglou 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
Our work	-Our novel model, the advantages and disadvantages of this survey are shown in the Conclusion section.		

Table 2: Comparisons of our model's benefits and drawbacks with the studies related to the OTSUKA coefficient (OM) in [39-44].

Surveys	Approach	Benefits	Drawbacks
[39]	A Survey of Binary Similarity and Distance Measures	Applying appropriate measures results in more accurate data analysis. Notwithstanding, few comprehensive surveys on binary measures have been conducted. Hence the authors collected 76 binary similarity and distance measures used over the last century and reveal their correlations through the hierarchical clustering technique	No mention
[40]	Quantified coefficients of association and measurement of similarity	Quantified coefficients of association avoid several problems of shape and size that are associated with correlation coefficients and measures of Euclidean distance. However, when measuring similarity, quantified coefficients weight each attribute of an object by that attribute's magnitude. A related set of similarity indices termed "mean ratios" is introduced; these indices	No mention

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

		give each attribute equal weight in all situations. Both quantified coefficients of association and mean ratios are related to a number of measures of similarity introduced to various fields of scientific research during the past 50 years. A review of this literature is included in an attempt to consolidate methodology and simplify nomenclature.	
Our work	-Our novel model, the advantages and disadvantages of this survey are shown in the Conclusion section.		

Table 3: The results of the documents in the testing data set and the accuracy of our novel model for the documents in the testing data set.

	Testing Dataset	**Correct Classification**	**Incorrect Classification**	**Accuracy**
Negative	5,250,000	4,680,124	569,876	89.25 %
Positive	5,250,000	4,691,126	558,874	
Summary	10,500,000	9,371,250	1,128,750	

Table 4: Average time of the classification of our new model for the documents in testing data set.

	Average time of the classification /10,500,000 documents.
The novel model in the sequential environment	44,206,310 seconds
The novel model in the Cloudera distributed system with 3 nodes	13,402,103 seconds
The novel model in the Cloudera distributed system with 6 nodes	7,401,051 seconds
The novel model in the Cloudera distributed system with 9 nodes	4,934,034 seconds

Table 5: Comparisons of our model's benefits and drawbacks with the studies related to the Fuzzy C-Means algorithm (FCM) in [45-49]

Surveys	**Approach**	**Benefits**	**Drawbacks**
[45]	FCM: The fuzzy c-means clustering algorithm	This program generates fuzzy partitions and prototypes for any set of numerical data. 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
[46]	A modified fuzzy c-means algorithm for bias field estimation and segmentat	The result is a slowly varying shading artifact over the image that can produce errors with conventional intensity-based classification. The authors' algorithm is formulated by	No mention

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

	ion of MRI data	modifying the objective function of the standard fuzzy c-means (FCM) algorithm to compensate for such inhomogeneities and to allow the labeling of a pixel (voxel) to be influenced by the labels in its immediate neighborhood. The neighborhood effect acts as a regularizer and biases the solution toward piecewise-homogeneous labelings. Such a regularization is useful in segmenting scans corrupted by salt and pepper noise. Experimental results on both synthetic images and MR data are given to demonstrate the effectiveness and efficiency of the proposed algorithm.	
Our work	-Our novel model, the advantages and disadvantages of this survey are shown in the Conclusion section.		

Table 6: Comparisons of our model's benefits and drawbacks with the studies related to the K-Means algorithm (KM) in [50-54]

Surveys	Approach	Benefits	Drawbacks
[50]	Genetic K-means algorithm	The authors define K-means operator, one-step of K-means algorithm, and use it in GKA as a search operator instead of crossover. The authors also define a biased mutation operator specific to clustering called distance-based-mutation. Using finite Markov chain theory, the authors prove that the GKA converges to the global optimum. It is observed in the simulations that GKA converges to the best known optimum corresponding to the given data in concurrence with the convergence result. It is also observed that GKA searches faster than some of the other evolutionary algorithms used for clustering.	No mention
[51]	Extensions to the k-Means Algorithm for Clustering Large Data Sets with Categorical Values	The authors use the well known soybean disease and credit approval data sets to demonstrate the clustering performance of the two algorithms. The authors' experiments on two real world data sets with half a million objects each show that the two algorithms are efficient when clustering large data sets, which is critical to data mining applications.	No mention
Our work	-Our novel model, the advantages and disadvantages of this survey are shown in the Conclusion section.		

Table 7: Comparisons of our model's benefits and drawbacks with the studies related to the Co-Training algorithm in [55-59]

Surveys	Approach	Benefits	Drawbacks
[55]	Applying co-training methods to statistical parsing	The algorithm iteratively labels the entire data set with parse trees. Using empirical results based on parsing the Wall Street Journal corpus the authors show that training a statistical parser on the combined labeled and unlabeled data strongly out-performs training only on the labeled data.	No mention

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

[56]	Speech Emotion Recogniti on using an Enhanced Co-Training Algorithm	The authors' experimental results demonstrate that compared with the method based on the supervised training, the proposed system makes 9.0% absolute improvement on female model and 7.4% on male model in terms of average accuracy. Moreover, the enhanced co-training algorithm achieves comparable performance to the co-training prototype, while it can reduce the classification noise which is produced by error labeling in the process of semi-supervised learning.	No mention
Our work	-Our novel model, the advantages and disadvantages of this survey are shown in the Conclusion section.		

Table 8: Comparisons of our model's advantages and disadvantages with the works in [60-62]

Resea rches	Approach	Advantages	Disad vanta ges
[60]	Examinin g the vector space model, an informatio n retrieval technique and its variation	In this work, the authors have given an insider to the workingof vector space model techniques used for efficientretrieval techniques. It is the bare fact that each systemhas its own strengths and weaknesses.What we havesorted out in the authors' work for vector space modeling is that themodel is easy to understand and cheaper to implement, considering the fact that the system should be costeffective (i.e., should follow the space/timeconstraint. It is also very popular. Although the system has all these properties, it is facing some major drawbacks.	No mentio n
[61]	+Latent Dirichlet allocation (LDA). +Multi-label text classificati on tasks and apply various feature sets. +Several combinati ons 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 mentio n
Our work	-Our novel model, the advantages and disadvantages of this survey are shown in the Conclusion section.		

Table 9: Comparisons of our model's positives and negatives the latest sentiment classification models (or the latest sentiment classification methods) in [63-65]

Stu dies	Approach	Positives	Negati ves
[63]	The Machine Learning Approaches Applied to Sentiment Analysis-Based Application s	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	No mentio n

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

		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.	
[64]	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," "nicecinematography," "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
Our work	-Our novel model, the advantages and disadvantages of this survey are shown in the Conclusion section.		