Journal of Theoretical and Applied Information Technology
28th February 2018. Vol.96. No 4

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

ENGLISH SENTIMENT CLASSIFICATION USING A GOWER-2 COEFFICIENT AND A GENETIC ALGORITHM WITH A FITNESS-PROPORTIONATE SELECTION IN A PARALLEL NETWORK ENVIRONMENT

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

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

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

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

ABSTRACT

We have already studied a data mining field and a natural language processing field for many years. There are many significant relationships between the data mining and the natural language processing. Sentiment classification Has HAd many crucial contributions to many different fields in everyday life, such as in political Activities, commodity production, and commercial Activities. A new model using a Gower-2 Coefficient (HA) and a Genetic Algorithm (GA) with a fitness function (FF) which is a Fitness-proportionate Selection (FPS) has been proposed for the sentiment classification. This can be applied to a big data. The GA can process many bit arrays. Thus, it saves a lot of storage spaces. We do not need lots of storage spaces to store a big data. Firstly, we create many sentiment lexicons of our basis English sentiment dictionary (bESD) by using the HA through a Google search engine with AND operator and OR operator. Next, According to the sentiment lexicons of the bESD, we encode 7,000,000 sentences of our training data set including the 3,500,000 negative and the 3,500,000 positive in English successfully into the bit arrays in a small storage space. We also encrypt all sentences of 8,000,000 documents of our testing data set comprising the 4,000,000 positive and the 4,000,000 negative in English successfully into the bit arrays in the small storage space. We use the GA with the FPS to cluster one bit array (corresponding to one sentence) of one document of the testing data set into either the bit arrays of the negative sentences or the bit arrays of the positive sentences of the training data set. The sentiment classification of one document is based on the results of the sentiment classification of the sentences of this document of the testing data set. We tested the proposed model in both a sequential environment and a distributed network system. We achieved 88.12% accuracy of the testing data set. The execution time of the model in the parallel network environment is faster than the execution time of the model in the sequential system. The results of this work can be widely used in applications and research of the English sentiment classification.

Keywords: *English Sentiment Classification; Distributed System; Gower-2 Similarity Coefficient; Cloudera; Hadoop Map And Hadoop Reduce; Genetic Algorithm; Fitness-Proportionate Selection*

1. INTRODUCTION

Many machine-learning methods or methods based on lexicons; or a combination of both Have been studied for sentiment classification for many years. Sentiment analysis Has a wide range of applications in the fields of business, organizations, governments and individuals.

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

A genetic algorithm (GA) is a technology of the data mining which is a metaheuristic inspired by the process of natural selection the belongs to the larger class of evolutionary algorithms (EA). The GA is commonly used to generate high-quality solutions

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

to optimization and search problems by relying on bio-inspired operators such as mutation, crossover and selection

The genetic algorithm differs from a classical, derivative-based, optimization algorithm in two main ways, as summarized as follows: (1) Genetic Algorithm : (a) Generates a population of points at each iteration. The best point in the population approaches an optimal solution. (b) Selects the next population by computation which uses random number generators. (2) Classical Algorithm : (a) Generates a single point at each iteration. The sequence of points approaches an optimal solution. (b) Selects the next point in the sequence by a deterministic computation.

With the purpose of this survey, we always try to find a new approach to reform the Accuracy of the sentiment classification results and to shorten the execution time of the proposed model with a low cost. We also try to find a new approach to save a lot of storage spaces of many big data sets and the results of the sentiment classification.

The motivation of this new model is as follows: Many algorithms in the data mining field can be applied to natural language processing, specifically semantic classification for processing millions of English documents. A Gower-2 similarity measure (HA) and a genetic algorithm (GA) of the clustering technologies of the data mining filed can be applied to the sentiment classification in both a sequential environment and a parallel network system. This will result in many discoveries in scientific research, hence the motivation for this study.

The novelty of the proposed approach is as follows: the Gower-2 similarity measure (HA) and the GA are applied to sentiment analysis. This can also be applied to identify the emotions of millions of documents. This survey can be applied to other parallel network systems. Hadoop Map (M) and Hadoop Reduce (R) are used in the proposed model. Therefore, we will study this model in more detail.

To get higher Accuracy of the results of the sentiment classification, to shorten execution times of the sentiment classification and to save lots of storage spaces, we use the GA with a fitness function (FF) which is a Fitness-proportionate Selection (FPS) because as known, the GA processes many bit arrays and the bit arrays always take many small spaces to be run and saved. Unsurprisingly, a storage space of the bit arrays of the training data set is much less than a storage space of all the sentences of the training data set. The HA is used to identify many sentiment values and polarities of many sentiment lexicons of our basis English sentiment dictionary (bESD) through a Google search engine with AND operator and OR operator.

We perform the proposed model as follows: Firstly, the valences and the polarities of the sentiment lexicons of the bESD are identified by using the HA through the Google search engine with AND operator and OR operator. We label all the sentiment lexicons of the bESD by using many binary bits. Therefore, each term (meaningful word or meaningful phrase) in the sentiment lexicons are shown by one bit array. This bit array provides the information of this term about a content of this term (example as “good”, “bad”, “very”, etc.), a valence of the term. Next, we encrypt all the sentences of the training data set to the bit arrays which are stored in a small storage space. All the positive sentences of the training data set are encoded to the positive bit arrays, called the positive bit array group. All the negative sentences of the training data set are encrypted to the negative bit arrays, called the negative bit array group. All the sentences of one document of the testing data set are encoded to the bit arrays of this document. We use the GA with FPS to cluster one bit array (corresponding to one sentence) of one document of the testing data set into either the positive bit array or the negative bit array of the training data set. This document is clustered into the positive polarity if the number of the bit arrays (corresponding to the sentences) clustered into the positive is greater than the number of the bit arrays (corresponding to the sentences) clustered into the negative in the document. This document is clustered into the negative polarity if the number of the bit arrays (corresponding to the sentences) clustered into the positive is less than the number of the bit arrays (corresponding to the sentences) clustered into the negative in the document. This document is clustered into the neutral polarity if the number of the bit arrays (corresponding to the sentences) clustered into the positive is as equal as the number of the bit arrays (corresponding to the sentences) clustered into the negative in the document. Finally, the sentiment classification of all the documents of the testing data set is implemented completely.

All the above things are firstly implemented in a sequential environment to get an accuracy and an execution time of the proposed model. Then, all the above things are performed in a parallel network system to get the Accuracy and the execution times

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

of our proposed model with a purpose which is to shorten the execution times of the model.

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

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

3)The algorithms are built in the proposed model.

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

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

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

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

8)This study also proves the different fields of scientific research can be related in many ways.

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

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

11)The principles are proposed in the research.

12)The Cloudera distributed environment is used in this study.

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

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

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

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

17)The GA – related algorithms are built in this survey.

18)The HA – related algorithms are proposed in this paper.

This study contains 6 sections. Section 1 introduces the study; Section 2 discusses the related works about the genetic algorithm (GA), Fitness-proportionate Selection (FPS), Gower-2 similarity measure (HA), etc.; Section 3 is about the English data set; Section 4 represents the methodology of our proposed model; Section 5 represents the experiment. Section 6 provides the conclusion. The References section comprises all the reference documents; all tables are shown in the Appendices section.

2. RELATED WORK

We summarize many researches which are related to our research. By far, we know the 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 the GOWER-2 coefficient (HA) 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 PMI measure in [1-13]. In the research [1], the authors generate several Norwegian sentiment lexicons by exta sentiment information from two different types of Norwegian text corpus, namely, news corpus and discussion forums. The methodology is based on the Point wise Mutual Information (PMI). The authors introduce a modification of the PMI the considers small "blocks" of the text instead of the text as a whole. The study in [2] introduces a simple algorithm for unsupervised learning of semantic orientation from extremely large corpora, etc.

Two studies related to the PMI measure and Jaccard measure are in [14, 15]. In the survey [14], the authors empirically evaluate the performance of different corpora in sentiment similarity measurement, which is the fundamental task for word polarity classification. The research in [15] proposes a new method to estimate impression of short sentences considering adjectives. In the proposed system, first, an input sentence is analyzed and preprocessed to obtain keywords. Next, adjectives are taken out from the data which is queried from Google N-gram corpus using keywords-based templates.

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

The works related to the Jaccard measure are in [16-22]. The survey in [16] investigates the problem of sentiment analysis of the online review. In the study [17], the authors are addressing the issue of spreading public concern about epidemics. Public concern about a communicable disease can be seen as a problem of its own, etc.

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

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

There are the works related to the GOWER-2 coefficient (HA) 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. In [40], the kernel method converts information – perhaps complex or high-dimensional information – for a pair of subjects to a quantitative value representing either similarity or dissimilarity, with the requirement that it must create a positive semidefinite matrix when applied to all pairs of subjects. This approach provides enormous opportunities to enhance genetic analyses by including a wide range of publically-available data as structured kernel 'prior' information. Kernel methods are appealing for their generality, yet this generality can make it challenging to formulate measures of similarity that directly address a specific scientific aim, or that are most powerful to detect a specific genetic mechanism, etc.

The surveys related to the genetic algorithm (GA) in [45-49]. The survey in [45] sets out to explain wHAt genetic algorithms are and how they can be used to solve real-world problems. In the study [46], Differential Evolution (DE) can be efficiently used to detect the changes in the ECG using optimized features from the ECG beats. For the detection of normal and BBB beats, these DE feature values are given as the input for the LMNN classifier, etc.

There are the researches related to the Fitness-proportionate Selection (FPS) in [50-54]. The authors in [50] report the results of a combined computational and experimental approach in which simple electromechanical systems are evolved through simulations from basic building blocks (bars, actuators and artificial neurons); the 'fittest' machines (defined by their locomotive ability) are then fabricated robotically using rapid manufacturing technology. In the survey [51] an extensive, quantitative comparison is presented, etc.

The latest researches of the sentiment classification are [55-65]. In the research [55], the authors present their machine learning experiments with regard to sentiment analysis in blog, review and forum texts found on the World Wide Web and written in English, Dutch and French. The survey in [56] discusses an approach where an exposed stream of tweets from the Twitter micro blogging site are preprocessed and classified based on their sentiments. In sentiment classification system the concept of opinion subjectivity has been accounted. In the study, the authors present opinion detection and organization subsystem, which have already been integrated into our larger question-answering system, etc.

The surveys related to the binary code of letters in English are shown in [66-71]. The researches in [66-71] show all the binary codes of all the letters in English completely.

There are the researches related to transferring a decimal to a binary code in [72-77]. The surveys in [72-77] show how to transfer one decimal to one binary code.

3. DATA SET

We built the testing data set including 8,000,000 documents in the movie field, which contains 4,000,000 positive and 4,000,000 negative in English in Figure 1. All the documents in our testing data set are automatically extracted from English Facebook, English websites and social networks; then we labeled positive and negative for them.

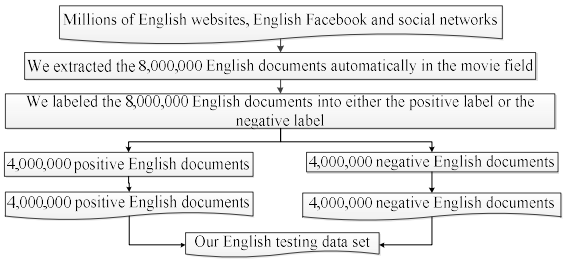


Figure 1: Our English testing data set.

In Figure 2 below, we built the training data set comprising 7,000,000 sentences in the movie field, which containing the 3,500,000 positive and the 3,500,000 negative in English. All the sentences in our training data set are automatically extracted from English Facebook, English websites and

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

social networks; then we labeled positive and negative for them.

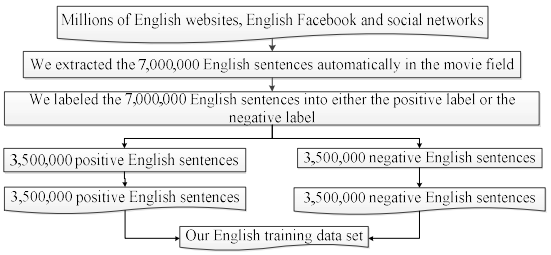


Figure 2: Our English training data set.

4. METHODOLOGY

There are two parts in this section as follows: The first part is the sub-section (4.1) which we create the sentiment lexicons in English in both a sequential environment and a distributed system. The second part is the sub-section (4.2) which we use the Gower-2 Coefficient (HA) and the Genetic Algorithm (GA) with the Fitness-proportionate Selection (FPS) - the fitness function (FF) to cluster 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 sub-section comprises three parts. The first sub-section of this section is the sub-section (4.1.1) which we identify a sentiment value of one word (or one phrase) in English. The second part of this section is the sub-section (4.1.2) which we create a basis English sentiment dictionary (bESD) in a sequential system. The third sub-section of this section is the sub-section (4.1.3) which we create a basis English sentiment dictionary (bESD) in a parallel environment.

The section comprises two parts in the sub-section (4.2). The first part of this section is the sub-section (4.2.1) which we use the Gower-2 Coefficient (HA) and the Genetic Algorithm (GA) with the Fitness-proportionate Selection (FPS) - the fitness function (FF) to classify the documents of the testing data set into either the positive vector group or the negative vector group in a sequential environment. The second part of this section is the sub-section (4.2.2) which we use the Gower-2 Coefficient (HA) and the Genetic Algorithm (GA) with the Fitness-proportionate Selection (FPS) - the fitness function (FF) to cluster the documents of the testing data set into either the positive vector group or the negative vector group in a distributed system.

4.1 The sentiment lexicons in English

There are three parts in this section. The sub-section (4.1.1) is the first sub-section of this section which we identify a sentiment value of one word (or one phrase) in English. The sub-section (4.1.2) is the second part of this section which we create a basis English sentiment dictionary (bESD) in a sequential system. The sub-section (4.1.3) is the third sub-section of this section which we create a basis English sentiment dictionary (bESD) in a parallel environment.

4.1.1 A valence of one word (or one phrase) in English

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

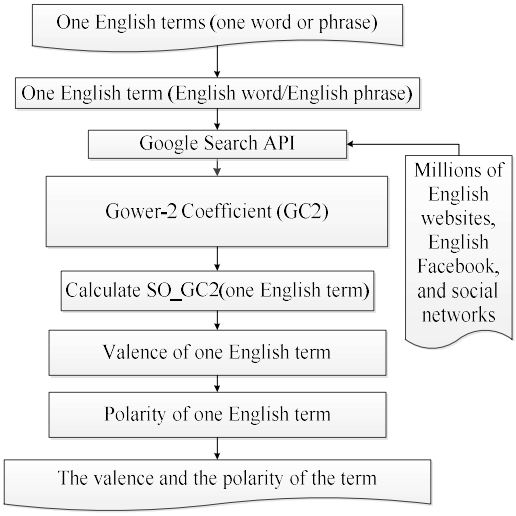


Figure 3: Overview of identifying the valence and the polarity of one term in English using a GOWER-2 coefficient (HA)

The authors of the surveys in [1-15] use an equation about Pointwise Mutual Information (PMI) between two words wi and wj as follows:

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

and an equation about SO (sentiment orientation) of word wi as follow:

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

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

The authors of the works in [1-8] use the positive and the negative of eq. (2) in English as follows: positive = {good, nice, excellent, positive, fortunate, correct, superior} and negative = {bad, nasty, poor, negative, unfortunate, wrong, inferior}.
The authors of the studies in [2, 3, 5] use the PMI equations with the AltaVista search engine and the authors of the researches in [4, 6, 8] use the PMI equations with the Google search engine.
Besides, the authors of the study [4] also use German, the authors of the work [5] also use Macedonian, the authors of the survey [6] also use Arabic, the authors of the work [7] also use Chinese, and the authors of the research [8] also use Spanish. In addition, the Bing search engine is also used in [6].
The authors of the surveys [9-12] use the PMI equations in Chinese, not English, and Tibetan is also added in [9].
About the search engine, the authors of the researches in [11, 12] use the AltaVista search engine, and the authors in [12] use three search engines, such as the Google search engine, the Yahoo search engine and the Baidu search engine. The survey [13] uses the PMI equations in Japanese with the Google search engine. The researches in [14, 15] also use the PMI equations and Jaccard equations with the Google search engine in English.
The authors of the works in [14-22] use the equations about Jaccard between two words wi and wj as follows:

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

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

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

and an equation about SO (sentiment orientation) of word wi as follows:

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

The authors of the surveys in [14-21] use the positive and the negative of eq. (5) in English as follows: positive = {good, nice, excellent, positive, fortunate, correct, superior} and negative = {bad, nasty, poor, negative, unfortunate, wrong, inferior}.
The studies in [14, 15, 17] use the Jaccard equations with the Google search engine in English. The surveys in [16, 21] use the Jaccard equations in English. The authors in [20, 22] use the Jaccard equations in Chinese. The authors in [18] use 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.
According to 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.
PMI, Jaccard, Cosine, Ochiai, Sorensen, Tanimoto and GOWER-2 coefficient (HA) are the similarity measures between two words in [1-32], and they can perform the same functions and with the same chracteristics; so the HA is used in calculating the valence of the words. In addition, we prove the HA can be used in identifying the valence of the English word through the Google search with the AND operator and OR operator.
We Have an equation about the GOWER-2 coefficient (HA) in [39-44] as follows:

$$\text{GOWER} - 2\ \text{Coefficient}(a, b) = \text{GOWER} - 2\ \text{Measure}(a, b) = \text{HA}(a, b) = \frac{(a \cap b) * (\neg a \cap \neg b)}{[(a \cap b) + (\neg a \cap b)] * [(a \cap b) + (a \cap \neg b)] * [(\neg a \cap b) + (\neg a \cap \neg b)] * [(a \cap \neg b) + (\neg a \cap \neg b)]} \quad (6)$$

with a and b are the vectors.

Based on the eq. (1), (2), (3), (4), (5), (6), we propose many new equations of the HA to calculate

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

the valence and the polarity of the English words (or the English phrases) through the Google search engine as the following equations below.

In eq. (6), when a has only one element, a is a word. When b has only one element, b is a word. In eq. (6), a is replaced by w1 and b is replaced by w2.

$$\begin{aligned}\text{GOWER} - 2\,\text{Measure}(w1, w2) &= \text{GOWER} \\ &\quad - 2\,\text{Coefficient}(w1, w2) = \\ \text{HA}\,(w1, w2) &= \frac{\text{P}(w1, w2) * \text{P}(\neg w1, \neg w2)}{\text{A}} \end{aligned} \quad (7)$$

with

$$\begin{aligned}\text{A} = [&\text{P}(w1, w2) + \text{P}(\neg w1, w2)] \\ &* [\text{P}(w1, w2) + \text{P}(w1, \neg w2)] \\ &* [\,\text{P}(\neg w1, w2) \\ &+ \text{P}(\neg w1, \neg w2)] \\ &* [\text{P}(w1, \neg w2) \\ &+ \text{P}(\neg w1, \neg w2)\,]\end{aligned}$$

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

$$\begin{aligned}\text{Valence}(w) &= \text{SO\_HA}(w) \\ &= \text{HA}(w, \text{positive\_query}) \\ &- \text{HA}(w, \text{negative\_query})\end{aligned} \quad (8)$$

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

$$\begin{aligned}&\text{HA}(w, \text{positive\_query}) \\ &= \frac{\text{P}(w, \text{positive\_query}) * \text{P}(\neg w, \neg \text{positive\_query})}{\text{A9}}\end{aligned} \quad (9)$$

with

$$\begin{aligned}\text{A9} = [&\text{P}(w, \text{positive\_query}) \\ &+ \text{P}(\neg w, \text{positive\_query})] \\ &* [\text{P}(w, \text{positive\_query}) \\ &+ \text{P}(w, \neg \text{positive\_query})] \\ &* [\,\text{P}(\neg w, \text{positive\_query}) \\ &+ \text{P}(\neg w, \neg \text{positive\_query})] \\ &* [\text{P}(w, \neg \text{positive\_query}) \\ &+ \text{P}(\neg w, \neg \text{positive\_query})\,]\end{aligned}$$

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

$$\begin{aligned}&\text{HA}(w, \text{negative\_query}) \\ &= \frac{\text{P}(w, \text{negative\_query}) * \text{P}(\neg w, \neg \text{negative\_query})}{\text{A10}}\end{aligned} \quad (10)$$

with

$$\begin{aligned}\text{A10} = [&\text{P}(w, \text{negative\_query}) \\ &+ \text{P}(\neg w, \text{negative\_query})] \\ &* [\text{P}(w, \text{negative\_query}) \\ &+ \text{P}(w, \neg \text{negative\_query})] \\ &* [\,\text{P}(\neg w, \text{negative\_query}) \\ &+ \text{P}(\neg w, \neg \text{negative\_query})] \\ &* [\text{P}(w, \neg \text{negative\_query}) \\ &+ \text{P}(\neg w, \neg \text{negative\_query})\,]\end{aligned}$$

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

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

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

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

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

5)Valence(W) = SO_HA(w): valence of English word (or English phrase) w; is SO of word (or phrase) by using the GOWER-2 coefficient (HA)

6)positive_query: { creative 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

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

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))).
According to 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.
If HA (w, positive_query) is as equal as 1, the English word w is the nearest of positive_query.
If HA(w, positive_query) is as equal as 0, the English word w is the farthest of positive_query.
If HA(w, positive_query) > 0 and HA(w, positive_query) ≤ 1, the English word w belongs to positive_query being the positive group of the English words.
If HA(w, negative_query) is as equal as 1, the English word w is the nearest of negative_query.
If HA(w, negative_query) is as equal as 0, the English word w is the farthest of negative_query.
If HA(w, negative_query) > 0 and HA(w, negative_query) ≤ 1, the English word w belongs to negative_query being the negative group of the English words.
So, the valence of the English word w is the value of HA(w, positive_query) substracting the value of HA(w, negative_query) and the eq. (8) is the equation of identifying the valence of the English word w.
We have the information about HA as follows:
1)HA(w, positive_query) ≥ 0 and HA(w, positive_query) ≤ 1.
2)HA(w, negative_query) ≥ 0 and HA (w, negative_query) ≤ 1
3)If HA (w, positive_query) = 0 and HA (w, negative_query) = 0 then SO_HA (w) = 0.
4)If HA (w, positive_query) = 1 and HA (w, negative_query) = 0 then SO_HA (w) = 0.
5)If HA (w, positive_query) = 0 and HA (w, negative_query) = 1 then SO_HA (w) = -1.
6)If HA (w, positive_query) = 1 and HA (w, negative_query) = 1 then SO_HA(w) = 0.
So, SO_HA (w) ≥ -1 and SO_HA (w) ≤ 1.
If SO_HA (w) > 0, the polarity of the English word w is positive polarity. If SO_HA (w) < 0, the polarity of the English word w is negative polarity. If SO_HA (w) = 0, the polarity of the English word w is neutral polarity. In addition, the semantic value of the English word w is SO_HA (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 the 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 the 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. (8), eq. (9), and eq. (10), we identify the valence and the polarity of one word (or one phrase) in English by using the HA through the Google search engine with AND operator and OR operator.
We show the comparisons of advantages of the results of our new model with the researches in the tables as follows: Table 1, Table 2, Table 3, and Table 4.
In Table 1 and Table 2 below of the Appendices section, we compare our model's results with the surveys in [1- 22].

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

In Table 3 and Table 4 below, we compare our model's results with the researches related to the GOWER-2 coefficient (HA) in [39,40].

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

In this part, the valences and the polarities of the English words or phrases for our basis English sentiment dictionary (bESD) are calculated by using the HA in a sequential system from at least 55,000 English terms, including nouns, verbs, adjectives, etc. according to [33-38], as the following diagram in Figure 4 below shows.

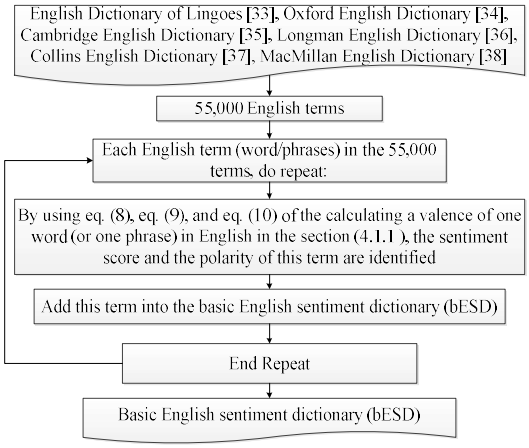


Figure 4: Overview of creating a basis English sentiment dictionary (bESD) in a sequential environment

The algorithm 1 is proposed to perform this section. The algorithm 1 has the main ideas as follows:

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

Output: a basis English sentiment dictionary (bESD)

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

Step 2: By using eq. (8), eq. (9), and eq. (10) 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 HA through the Google search engine with AND operator and OR operator.

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

Step 4: End Repeat – End Step 1;

Step 5: Return bESD;

We store more 55,000 English words (or English phrases) of our basis English sentiment dictionary (bESD) in Microsoft SQL Server 2008 R2.

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

In this part, the valences and the polarities of the English words or phrases for our basis English sentiment dictionary (bESD) are identified by using the HA in a parallel network environment from at least 55,000 English terms, including nouns, verbs, adjectives, etc. based on [33-38], as the following diagram in Figure 5 below shows.

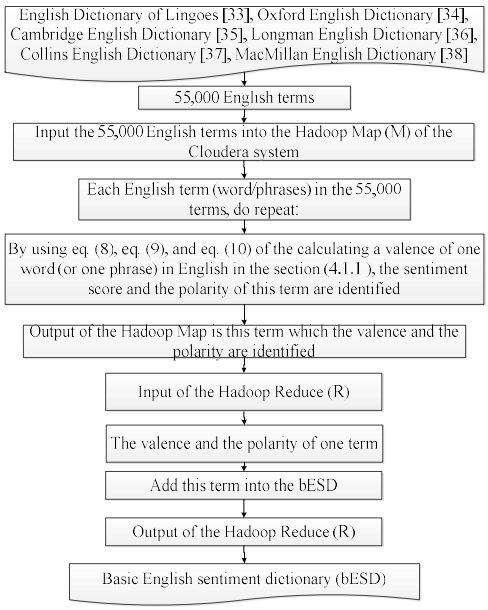


Figure 5: Overview of creating a basis English sentiment dictionary (bESD) in a distributed environment

This section in Figure 5 comprises two phases as follows: the Hadoop Map (M) phase and the Hadoop Reduce (R) phase. The input of the Hadoop Map phase is the 55,000 terms in English in [33-38]. The output of the Hadoop Map phase is one term which the sentiment score and the polarity are identified. The output of the Hadoop Map phase is the input of the Hadoop Reduce phase. Thus, the input of the Hadoop Reduce phase is one term which the sentiment score and the polarity are identified. The output of the Hadoop Reduce phase is the basis English sentiment dictionary (bESD).

The algorithm 2 is built to implement the Hadoop Map phase of creating a basis English sentiment dictionary (bESD) in a distributed environment. The algorithm 2 Has its main ideas as follows:

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:

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

Step 2: By using eq. (8), eq. (9), and eq. (10) 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 HA through the Google search engine with AND operator and OR operator.

Step 3: Return this term;

The algorithm 3 is proposed to perform the Hadoop Reduce phase of creating a basis English sentiment dictionary (bESD) in a distributed environment. The algorithm 3 comprises the main ideas as follows:

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

Output: a basis English sentiment dictionary (bESD)

Step 1: Receive this term;

Step 2: Add this term into the basis English sentiment dictionary (bESD);

Step 3: Return bESD;

At least 55,000 English words (or English phrases) of our basis English sentiment dictionary (bESD) are stored in Microsoft SQL Server 2008 R2.

4.2 Implementing the Gower-2 Coefficient (HA) and the Genetic Algorithm (GA) with the Fitness-proportionate Selection (FPS) - the fitness function (FF) in both a sequential environment and a distributed network system

In Figure 6, this section (4.2) comprises two parts. The fist part of this section is the sub-section (4.2.1) which we use the Gower-2 Coefficient (HA) and the Genetic Algorithm (GA) with the Fitness-proportionate Selection (FPS) - the fitness function (FF) to classify the documents of the testing data set into either the positive vector group or the negative vector group in a sequential environment. The second part of this section is the sub-section (4.2.2) which we use the Gower-2 Coefficient (HA) and the Genetic Algorithm (GA) with the Fitness-proportionate Selection (FPS) - the fitness function (FF) to cluster the documents of the testing data set into either the positive vector group or the negative vector group in a distributed system.

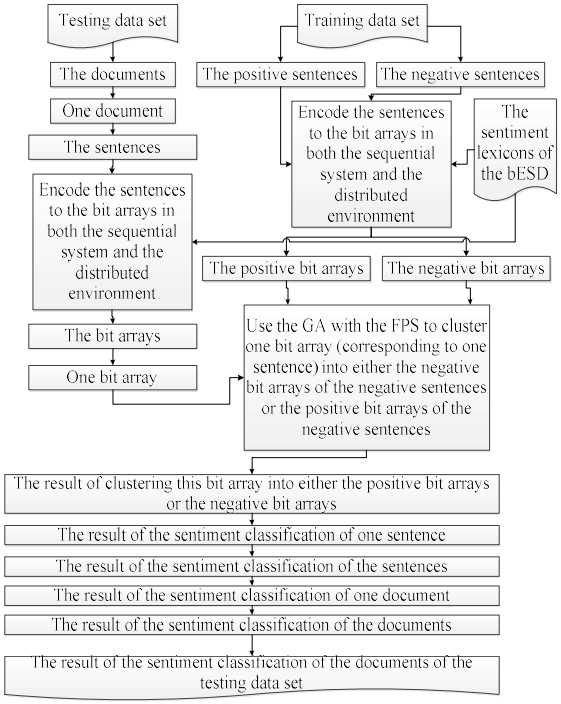


Figure 6: Overview of implementing the Gower-2 Coefficient (HA) and the Genetic Algorithm (GA) with the Fitness-proportionate Selection (FPS) - the fitness function (FF) in both a sequential environment and a distributed network system

4.2.1 Performing the Gower-2 Coefficient (HA) and the Genetic Algorithm (GA) with the Fitness-proportionate Selection (FPS) - the fitness function (FF) in a sequential environment

In Figure 7, The genetic algorithm (GA) is used with the Fitness-proportionate Selection (FPS) - the fitness function (FF) to cluster the documents of the testing data set into either the positive polarity or the negative polarity in the sequential environment. In this section, we perform the proposed model in the sequential system as follows: Firstly, we build the sentiment lexicons of the bESD based on a basis English sentiment dictionary (bESD) in a sequential environment (4.1.2). We encrypt the sentiment lexicons of the bESD to the bit arrays and each bit array in the bit arrays presents each term in the sentiment lexicons with the information as follows: a content of this term, a sentiment score of this term. This is called the bit arrays of the bESD which are stored in a small storage space. Then, we encode all the sentences of the training data set to the bit arrays based on the bit arrays of the sentiment lexicons of the bESD. A positive bit array group of the training data set which we encrypt all the positive sentences of the training data set to the positive bit arrays according

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

to the bit arrays of the sentiment lexicons of the bESD. A negative bit array group of the training data set which we encode all the negative sentences of the training data set to the negative bit arrays according to the bit arrays of the sentiment lexicons of the bESD. Each document in the documents of the testing data set is separated to the sentences. Each sentence in the sentences of one document of the testing data set is encrypted to one bit array (corresponding to one sentence) of the document. Next, one bit array of the document is clustered into either the positive bit array group or the negative bit array group of the training data set by using the GA with the FPS. Then, all the bit arrays (corresponding to all the sentences) of the document of the testing data set are clustered into either the positive array group or the negative bit array group of the training data set using the GA with the FPS. Based on the results of the sentiment classification of the bit arrays of the document, the sentiment classification of this document is identified completely. If the number of the bit arrays clustered into the positive polarity is greater than the number of the bit arrays clustered into the negative polarity in the document, this document is clustered into the positive. If the number of the bit arrays clustered into the positive polarity is less than the number of the bit arrays clustered into the negative polarity in the document, this document is clustered into the negative. If the number of the bit arrays clustered into the positive polarity is as equal as the number of the bit arrays clustered into the negative polarity in the document, this document is clustered into the neutral. It means this document does not belong to both the positive polarity and the negative polarity. Finally, the sentiment classification of the documents of the testing data set is identified successfully.

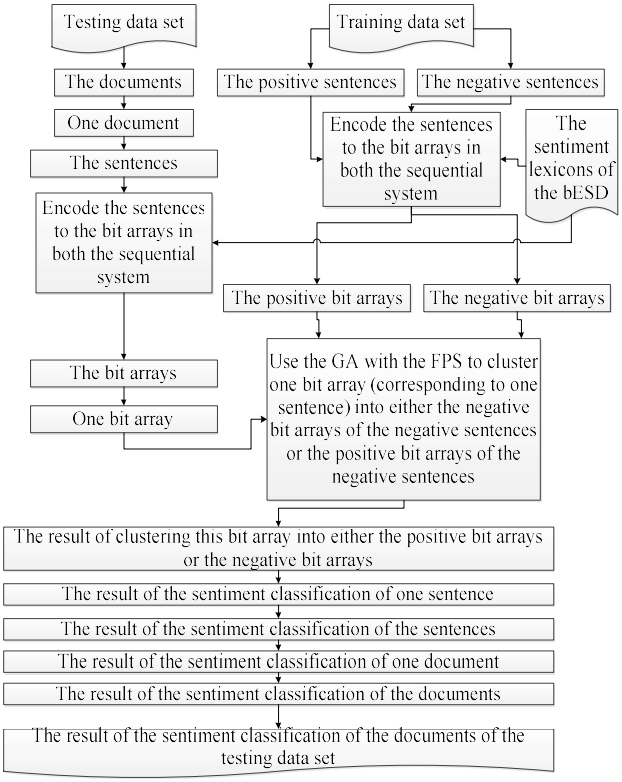


Figure 7: Overview of performing the Gower-2 Coefficient (HA) and the Genetic Algorithm (GA) with the Fitness-proportionate Selection (FPS) - the fitness function (FF) in a sequential environment

Firstly, we build the sentiment lexicons of the bESD based on a basis English sentiment dictionary (bESD) in a sequential environment (4.1.2).

We encrypt the sentiment lexicons of the bESD to the bit arrays and each bit array in the bit arrays presents each term in the sentiment lexicons with the information as follows: a content of this term, a sentiment score of this term. This is called the bit arrays of the bESD which are stored in a small storage space.

We assume the sentiment lexicons of the bESD are stored in the table as follows:

Ordering number	Lexicons	Valence
1	Good	+1
2	Very good	+2
3	Bad	-1
4	Very bad	-2
5	Terrible	-1.2
6	Very terrible	-2.3
…	…	…
55,000	…	…
…	…	…

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

According to the sentiment lexicons of the bESD, we see the valences of the sentiment lexicons are from -10 to +10. Thus, a natural part of one valence is presented by the 4 binary bits and we also use the 4 binary bits of a surplus part of this valence. So, the 8 binary bits are used for presenting one valence of one sentiment lexicons in a binary code.

Based on the English dictionaries [33-38], the longest word in English has 189,819 letters. According to the binary code of letters in English in [66-71], we see the 7 binary bits are used in encode one letter in all the letters in English. Therefore, we need 189,819 (letters) x 7 (bits) = 1,328,733 (bits) to present one word in English.

So, we need (1,328,733 bits of the content + 8 bits of the valence) = 1,328,741 bits to show fully one sentiment lexicon of the bESD in Figure 8 as follows:

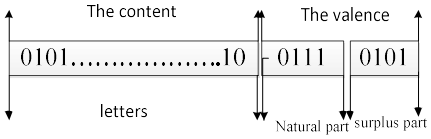


Figure 8: Overview of presenting one sentiment lexicon of the bESD in a binary code

We transfer the valence of one sentiment lexicon of the bESD to a binary code based on the transferring a decimal to a binary code in [72-77].

We build the algorithm 4 to encrypt one sentiment lexicon (comprising the content and the valence) to a binary array in the sequential environment. The main ideas of the algorithm 4 are as follows:

Input: one sentiment lexicon of the bESD

Output: a bit array

Step 1: Split this term into the letters.

Step 2: Set ABitArray := null;

Step 3: Set Valence := Get a valence of this term based on the bESD;

Step 4: Each letter in the letters, do repeat:

Step 5: Based on the binary code of letters in English in [66-71], we get a bit array of this letter;

Step 6: Add the bit array of this letter into ABitArray;

Step 7: End Repeat – End Step 3;

Step 8: Based on on the transferring a decimal to a binary code in [72-77], we transfer the valence to a bit array;

Step 9: Add this bit array into ABitArray;

Step 10: Return ABitArray;

We propose the algorithm 5 to encode one sentence in English to a binary array in the sequential system. The main ideas of the algorithm 5 are as follows:

Input: one sentence;

Output: a bit array;

Step 1: Set ABitArrayOfSentence := null;

Step 2: Split this sentence into the meaningful terms (meaningful word or meaningful phrase);

Step 3: Each term in the terms, do repeat:

Step 4: ABitArray := The algorithm 4 to encrypt one sentiment lexicon (comprising the content and the valence) to a binary array in the sequential environment with the input is this term;

Step 5: Add ABitArray into ABitArrayOfSentence;

Step 6: End Repeat – End Step 3;

Step 7: Return ABitArrayOfSentence;

We encrypt all the positive sentences of the training data set to the positive bit arrays based on the bit arrays of the sentiment lexicons of the bESD in the algorithm 6, called the positive bit array group. The main ideas of the algorithm 6 are as follows:

Input: all the positive sentences of the training data set

Output:a positive bit array group;

Step 1: Set APositiveBitArrayGroup := null;

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

Step 3: ABitArray := the algorithm 5 to encode one sentence in English to a binary array in the sequential system with the input is this sentence;

Step 4: Add ABitArray into APositiveBitArrayGroup;

Step 5: End Repeat – End Step 2;

Step 6: Return APositiveBitArrayGroup;

We encode all the negative sentences of the training data set to the negative bit arrays based on the bit arrays of the sentiment lexicons of the bESD in the algorithm 7, called the negative bit array group. The main ideas of the algorithm 7 are as follows:

Input: all the negative sentences of the training data set

Output:a negative bit array group;

Step 1: Set ANegativeBitArrayGroup := null;

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

Step 3: ABitArray := the algorithm 5 to encode one sentence in English to a binary array in the sequential system with the input is this sentence;

Step 4: Add ABitArray into ANegativeBitArrayGroup;

Step 5: End Repeat – End Step 2;

Step 6: Return ANegativeBitArrayGroup;

We propose the algorithm 8 to transfer one document of the testing data set into the bit arrays

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

of the document in the sequential system. The main ideas of the algorithm 8 are as follows:

Input: one document of the testing data set

Output: the bit arrays of the document;

Step 1: Set TheBitArraysOfTheDocument := null;

Step 2: Split this document into the sentences;

Step 3: Each sentence in the sentences, do repeat:

Step 4: ABitArray := the algorithm 5 to encode one sentence in English to a binary array in the sequential system with the input is this sentence;

Step 5: Add ABitArray into TheBitArraysOfTheDocument;

Step 6: End Repeat- End Step 3;

Step 7: Return TheBitArraysOfTheDocument;

We present the information about the GA briefly as follows:

1)According to [45-49], we show the basic operations of the genetic algorithm, at the same time, also used for the GA in our sequential environment and our parallel network environment. The genetic algorithm (GA: Genetic Algorithms) and other evolutionary algorithms based on forming the notion the natural evolutionary process is reasonable, perfect. It stems from the evolved idea to survive and grow in the wild. GA is a problem-solving method to mimic the behavior of humans in order to survive and develop. It helps to find the optimal solution and the best in terms of time and space allow. GA considers all solutions, by at least some solutions, then eliminates the irrelevant components and select the relevant components more adapted to create birth and evolution aimed at creating solutions which Have a new adaptive coefficient increasing. The adaptive coefficient is used as a gauge of the solution. The main steps of the GA:

Step 1: Select models to symbolize the solutions. The models can be sequence (string) of the binary number: 1 and 0, decimal and can be letters or mixture letters and numbers.

Step 2: Select the adaptive function (or the Fitness function) to use as a gauge of the solution.

Step 3: Continue the transformation form until achieving the best solution, or until the termination of the time.

2)Genetic Operators and Genetic Operations

Reproductive Operator

Reproductive operator includes two processes: the reproduction process (allowing regeneration), the selection process (selection).

Allowing Regeneration

Allowing regeneration is the process which allows chromosomes to copy on the basis of the adaptive coefficient. The adaptive coefficient is a function which is assigned the real value, corresponding to each chromosome in the population.

This process is described as follows:

- Determine the adaptive coefficient of each chromosome in the population at generation t, tabulate cumulative adaptive values (in order assigned to each chromosome).
- Suppose, the population Has n individuals. Call the adaptive coefficient of the corresponding chromosome is fi, cumulative total is fti which is defined by

$$\text{fti} = \sum_{j=1}^{i} \text{fi}$$

- Call Fn is the sum of the adaptive coefficient in all the population. Pick a random number f between 0 and Fn. Select the first instance correspond f ≥ ftk into new population.

Selection Process (Selection)

The selection process is the process of removing the poor adaptive chromosomes in the population. This process is described as follows:

- Arrange population in order of descending degree of adaptation.
- Remove the chromosome in the last of the sequence. Keep n in the best individuals.

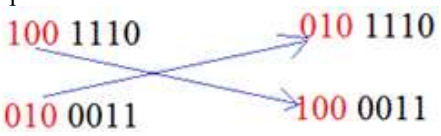
Crossover Operator (Crossover)

Crossover is the process of creating the new chromosomes based on the father-mother chromosomes by grafting a segment on the father-mother chromosomes together. The crossover operator is assigned with a probability pc. This process is described as follows:

- Randomly select a pair of chromosomes (father-mother) in the population. Suppose, the father-mother chromosomes Have the same length m.
- Create a random number in the range from 1 to m-1 (called as cross coupling point). The cross coupling point divides the father-mother chromosomes into two sub-strings which Have lengths m1, m2. The two new sub-strings created, is: m11 + m22 and m21 + m12.
- Put the two new chromosomes into the population.
-

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

Example:

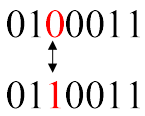


Mutation Operator (Mutation)

Mutation is a phenomenon which the child chromosomes carry some features, not in the genetic code of the father-mother chromosomes.

- Choose a random chromosome in the population;
- Create a random number k between 1 and m, $1 \leq k \leq m$;
- Change bit k. Put this chromosome in the population to participate in the evolution of the next generation.

Example:



Each pair of parents bears two children in one of the following two methods

Asexual Reproduction

Each child is an exact copy of each father or each mother.

Example:

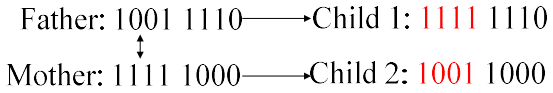
Father: 01101100 ⟶ Child 1: 01101100

Mother: 11001110 ⟶ Child 2: 11001110

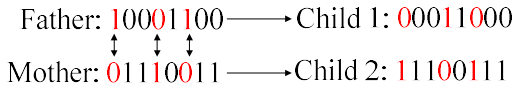
Sexual Reproduction (crossover)

Some bits are copied from the mother or a few bits are copied from the father.

Example of the sexual reproduction intersecting half



Example of the sexual reproduction intersecting 3 bits



In Figure 9, we show the diagram of the GA in the sequential environment as follows:

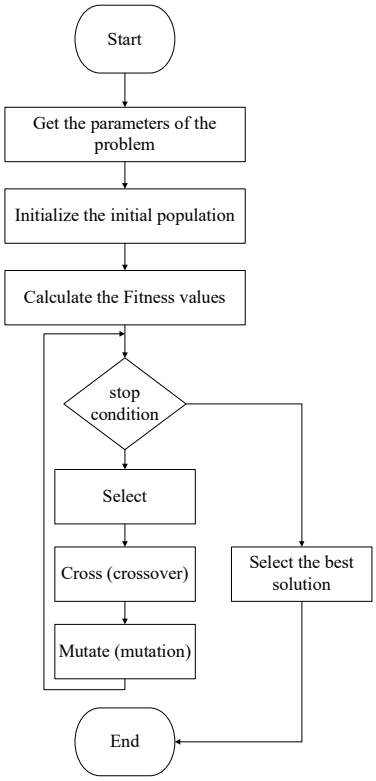


Figure 9: The diagram of the GA in the sequential environment.

We build the algorithm 9 to cluster one bit array (corresponding to one sentence) of the document into either the positive bit array group or the negative bit array group of the training data set by using the GA with the FPS in the sequential system. The main ideas of the algorithm 9 are as follows:

Input: one bit array (corresponding to one sentence) of the document; the positive bit array group and the negative bit array group the training data set;

Output: the sentiments (positive, negative, or neutral)

Step 1: randomly initialize population(t)

Step 2: determine fitness of population(t)

Step 3: repeat

Step 4: select parents from population(t)

Step 5: perform crossover on parents creating population(t+1)

Step 6: perform mutation of population(t+1)

Step 7: determine fitness of population(t+1)

Step 8: until best individual is good enough

Step 9: Return this bit array clustered into either the positive bit array group or the negative bit array group of the training data set.

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

Fitness is defined as an objective function, the quantifies, the optimality of a solution (chromosome) to the target problem. How to choose Fitness is dependent on the problem that we study. Choosing the different Fitness function will give the different results. In this survey, we use the Fitness-proportionate selection (FPS).

According to the researches related to the Fitness-proportionate Selection (FPS) in [50-54], Fitness-proportionate selection is detailed. Fitness proportionate selection is a genetic operator used in genetic algorithms for selecting potentially useful solutions for recombination. In fitness proportionate selection, as in all selection methods, the fitness function assigns a fitness to possible solutions or chromosomes. This fitness level is used to associate a probability of selection with each individual chromosome. If f_i is the fitness of individual i in the population, its probability of being selected is

$$p_i = \frac{f_i}{\Sigma_{j=1}^{N} f_j},$$

where N is the number of individuals in the population.

We propose the algorithm 10 to cluster one document of the testing data set into either the positive bit array group or the negative bit array group of the training data set using the GA with the FPS in the sequential environment. The main ideas of the algorithm 10 are as follows:

Input: one document of the testing data set; the positive bit array group and the negative bit array group of the training data set;

Output: the sentiments (positive, negative, or neutral)

Step 1: ABitArrayGroupOfOneDocument := the algorithm 8 to transfer one document of the testing data set into the bit arrays of the document in the sequential system with the input is this document;

Step 2: Set count_positive := 0 and count_negative := 0;

Step 3: Each bit array in ABitArrayGroupOfOneDocument, do repeat:

Step 4: OneResult := the algorithm 9 to cluster one bit array (corresponding to one sentence) of the document into either the positive bit array group or the negative bit array group of the training data set by using the GA with the FPS in the sequential system with the input is this bit array, the positive bit array group and the negative bit array group of the training data set;

Step 5: If OneResult is the positive Then count_positive := count_positive + 1;

Step 6: Else If OneResult is the negative Then count_negative := count_negative + 1;

Step 7: End Repeat – End Step 3;

Step 8: If count_positive is greater than count_negative Then Return positive;

Step 9: Else If count_positive is less than count_negative Then Return negative;

Step 10: Return neutral;

We build the algorithm 11 to cluster the documents of the testing data set into either the positive or the negative in the sequential environment. The main ideas of the algorithm 11 area as follows:

Input: the testing data set and the training data set;

Output: the results of the sentiment classification of the testing data set;

Step 1: The valences and the polarities of the sentiment lexicons of the bESD are calculated based on a basis English sentiment dictionary (bESD) in a sequential environment (4.1.2);

Step 2: A positive bit array group := encrypt all the positive sentences of the training data set to the bit arrays based on the bit arrays of the sentiment lexicons of the bESD in the algorithm 6 with the input is the positive sentences of the training data set;

Step 3: A negative bit array group := encode all the negative sentences of the training data set to the bit arrays based on the bit arrays of the sentiment lexicons of the bESD in the algorithm 7 with the input is the positive sentences of the training data set;

Step 4: Set TheResultsOfTheTestingDataSet := null;

Step 5: Each document in the documents of the testing data set, do repeat:

Step 6: OneResult := the algorithm 10 to cluster one document of the testing data set into either the positive bit array group or the negative bit array group of the training data set using the GA with the FPS in the sequential environment with the input is this document, the positive bit array group and the negative bit array group;

Step 7: Add OneResult into TheResultsOfTheTestingDataSet;

Step 8: End Repeat – End Step 5;

Step 9: Return TheResultsOfTheTestingDataSet;

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

4.2.2 Implementing the Gower-2 Coefficient (HA) and the Genetic Algorithm (GA) with the Fitness-proportionate Selection (FPS) - the fitness function (FF) in both a distributed network system

In Figure 10, The genetic algorithm (GA) is used with the Fitness-proportionate Selection (FPS) - the fitness function (FF) to cluster the documents of the testing data set into either the positive polarity or the negative polarity in the Cloudera parallel network environment. In this section, we perform the proposed model in the distributed system as follows: Firstly, we build the sentiment lexicons of the bESD based on a basis English sentiment dictionary (bESD) in a distributed system (4.1.3). We encrypt the sentiment lexicons of the bESD to the bit arrays and each bit array in the bit arrays presents each term in the sentiment lexicons with the information as follows: a content of this term, a sentiment score of this term. This is called the bit arrays of the bESD which are stored in a small storage space. Then, we encode all the sentences of the training data set to the bit arrays based on the bit arrays of the sentiment lexicons of the bESD. A positive bit array group of the training data set which we encrypt all the positive sentences of the training data set to the positive bit arrays according to the bit arrays of the sentiment lexicons of the bESD. A negative bit array group of the training data set which we encode all the negative sentences of the training data set to the negative bit arrays according to the bit arrays of the sentiment lexicons of the bESD. Each document in the documents of the testing data set is separated to the sentences. Each sentence in the sentences of one document of the testing data set is encrypted to one bit array (corresponding to one sentence) of the document. Next, one bit array of the document is clustered into either the positive bit array group or the negative bit array group of the training data set by using the GA with the FPS. Then, all the bit arrays (corresponding to all the sentences) of the document of the testing data set are clustered into either the positive array group or the negative bit array group of the training data set using the GA with the FPS. Based on the results of the sentiment classification of the bit arrays of the document, the sentiment classification of this document is identified completely. If the number of the bit arrays clustered into the positive polarity is greater than the number of the bit arrays clustered into the negative polarity in the document, this document is clustered into the positive. If the number of the bit arrays clustered into the positive polarity is less than the number of the bit arrays clustered into the negative polarity in the document, this document is clustered into the negative. If the number of the bit arrays clustered into the positive polarity is as equal as the number of the bit arrays clustered into the negative polarity in the document, this document is clustered into the neutral. It means this document does not belong to both the positive polarity and the negative polarity. Finally, the sentiment classification of the documents of the testing data set is identified successfully.

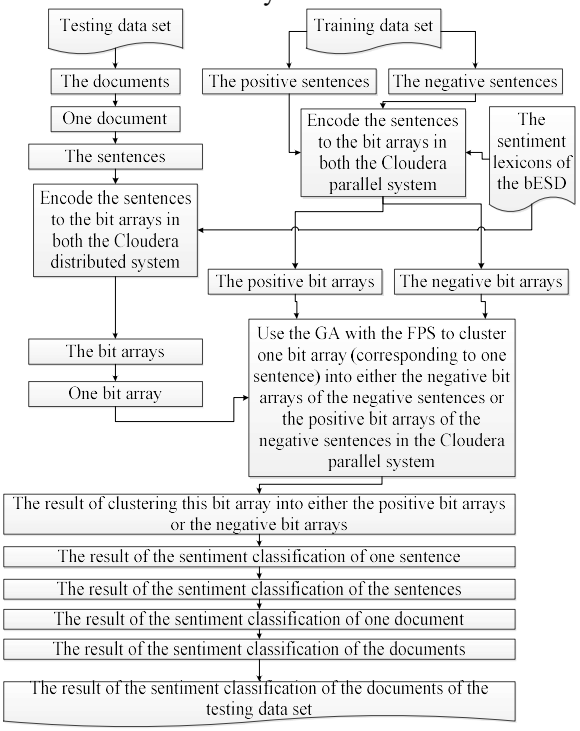


Figure 10: Overview of performing the Gower-2 Coefficient (HA) and the Genetic Algorithm (GA) with the Fitness-proportionate Selection (FPS) - the fitness function (FF) in a distributed environment

Firstly, we build the sentiment lexicons of the bESD based on a basis English sentiment dictionary (bESD) in a distributed system (4.1.3)

We encrypt the sentiment lexicons of the bESD to the bit arrays and each bit array in the bit arrays presents each term in the sentiment lexicons with the information as follows: a content of this term, a sentiment score of this term. This is called the bit arrays of the bESD which are stored in a small storage space.

We assume the sentiment lexicons of the bESD are stored in the table as follows:

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

Ordering number	Lexicons	Valence
1	Good	+1
2	Very good	+2
3	Bad	-1
4	Very bad	-2
5	Terrible	-1.2
6	Very terrible	-2.3
…	…	…
55,000	…	…
…	…	…

According to the sentiment lexicons of the bESD, we see the valences of the sentiment lexicons are from -10 to +10. Thus, a natural part of one valence is presented by the 4 binary bits and we also use the 4 binary bits of a surplus part of this valence. So, the 8 binary bits are used for presenting one valence of one sentiment lexicons in a binary code.

Based on the English dictionaries [33-38], the longest word in English has 189,819 letters. According to the binary code of letters in English in [66-71], we see the 7 binary bits are used in encode one letter in all the letters in English. Therefore, we need 189,819 (letters) x 7 (bits) = 1,328,733 (bits) to present one word in English.

So, we need (1,328,733 bits of the content + 8 bits of the valence) = 1,328,741 bits to show fully one sentiment lexicon of the bESD in Figure 11 as follows:

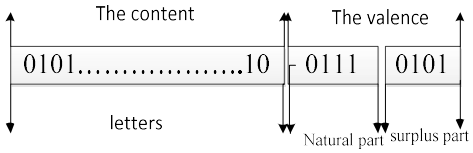


Figure 11: Overview of presenting one sentiment lexicon of the bESD in a binary code

We transfer the valence of one sentiment lexicon of the bESD to a binary code based on the transferring a decimal to a binary code in [72-77].

In Figure 12, we build the algorithm 12 and the algorithm 13 to encrypt one sentiment lexicon (comprising the content and the valence) to a binary array in the distributed network environment. This stage in Figure 12 comprises two phases as follows: Hadoop Map phase and Hadoop Reduce phase. The input of the Hadoop Map is one sentiment lexicon of the bESD. The output of the Hadoop Map is a bit array of one letter. The input of the Hadoop Reduce is the output of the Hadoop Map, thus, the input of the Hadoop Reduce is a bit array of one letter. The output of the Hadoop Reduce is a bit array of the term.

We propose the algorithm 12 to implement the Hadoop Map phase of encrypting one sentiment lexicon (comprising the content and the valence) to a binary array in the distributed network environment. The main ideas of the algorithm 12 are as follows:

Input: one sentiment lexicon of the bESD

Output: a bit array of one letter;

Step 1: Input this term and the bESD into the Hadoop Map in the Cloudera system;

Step 2: Split this term into the letters.

Step 3: Set Valence := Get a valence of this term based on the bESD;

Step 4: Each letter in the letters, do repeat:

Step 5: Based on the binary code of letters in English in [66-71], we get a bit array of this letter;

Step 6: Return the bit array of this letter; //the output of the Hadoop Map

We build the algorithm 13 to perform the Hadoop Reduce phase of encrypting one sentiment lexicon (comprising the content and the valence) to a binary array in the distributed network environment. The main ideas of the algorithm 13 are as follows:

Input: the bit array of this letter; //the output of the Hadoop Map

Output: a bit array of the term - ABitArray;

Step 1: Receive the bit array of this letter;

Step 2: Add the bit array of this letter into ABitArray;

Step 3: If this term is full Then

Step 4: Based on on the transferring a decimal to a binary code in [72-77], we transfer the valence to a bit array;

Step 5: Add this bit array into ABitArray;

Step 6: End If – End Step 3;

Step 7: Return ABitArray;

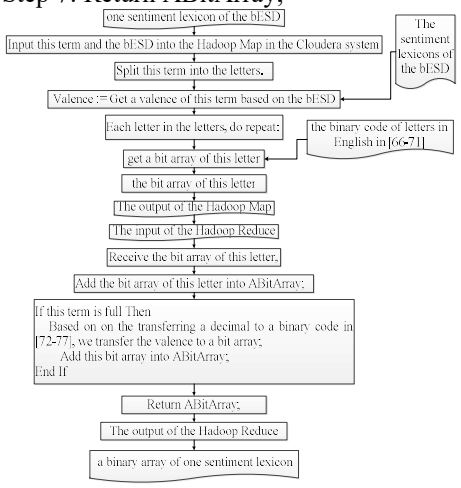


Figure 12: Overview of encrypting one sentiment lexicon (comprising the content and the valence) to a binary array in the distributed network environment

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

In Figure 13, we build the algorithm 14 and the algorithm 15 to encode one sentence in English to a binary array in the distributed system. This stage in Figure 13 comprises two phases as follows: Hadoop Map phase and Hadoop Reduce phase. The input of the Hadoop Map is one sentence. The output of the Hadoop Map is a bit array of one term - ABitArray. The input of the Hadoop Reduce is the output of the Hadoop Map, thus, the input of the Hadoop Reduce is a bit array of one term - ABitArray. The output of the Hadoop Reduce is a bit array of the sentence – AbitArrayOfSentence.

We propose the algorithm 14 to perform the Hadoop Map phase of encoding one sentence in English to a binary array in the parallel system. The main ideas of the algorithm 14 are as follows:

Input: one sentence;

Output: a bit array of one term - ABitArray;

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

Step 2: Split this sentence into the meaningful terms (meaningful word or meaningful phrase);

Step 3: Each term in the terms, do repeat:

Step 4: ABitArray := the encrypting one sentiment lexicon (comprising the content and the valence) to a binary array in the distributed network environment in Figure 12 with the input is this term;

Step 5: Return ABitArray;

We propose the algorithm 15 to implement the Hadoop Reduce of encoding one sentence in English to a binary array in the parallel system. The main ideas of the algorithm 15 are as follows:

Input: a bit array of one term – AbitArray – the output of the Hadoop Map;

Output: a bit array of the sentence - ABitArrayOfSentence;

Step 1: Receive AbitArray;

Step 2: Add ABitArray into ABitArrayOfSentence;

Step 3: Return ABitArrayOfSentence;

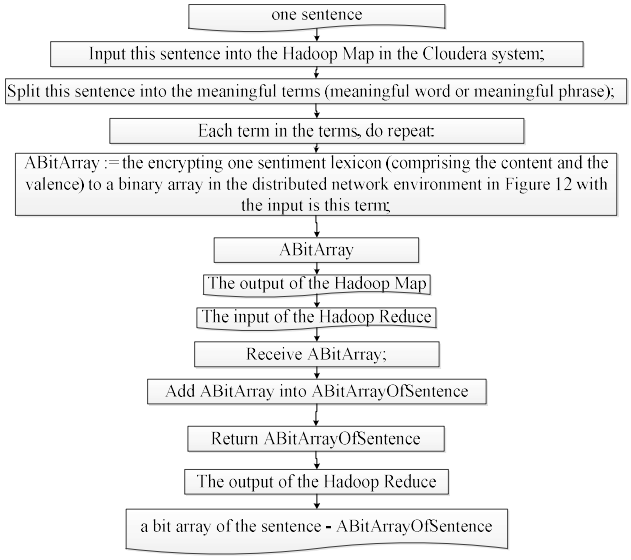


Figure 13: Overview of encoding one sentence in English to a binary array in the parallel system

In Figure 14, we build the algorithm 16 and the algorithm 17 to encrypt all the positive sentences of the training data set to the bit arrays based on the bit arrays of the sentiment lexicons of the bESD in the distributed environment, called the positive bit array group. This stage in Figure 14 comprises two phases as follows: Hadoop Map phase and Hadoop Reduce phase. The input of the Hadoop Map is all the positive sentences of the training data set. The output of the Hadoop Map is ABitArray – a bit array of one sentence. The input of the Hadoop Reduce is the output of the Hadoop Map, thus, the input of the Hadoop Reduce is ABitArray – a bit array of one sentence. The output of the Hadoop Reduce is a positive bit array group – APositiveBitArrayGroup.

We encrypt all the positive sentences of the training data set to the bit arrays based on the bit arrays of the sentiment lexicons of the bESD in the algorithm 16 to perform the Hadoop Map phase of this stage in the distributed environment, called the positive bit array group. The main ideas of the algorithm 16 are as follows:

Input: all the positive sentences of the training data set

Output: ABitArray – a bit array of one sentence;

Step 1: Input all the positive sentences of the training data set into the Hadoop Map in the Cloudera system;

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

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

Step 3: ABitArray := the encoding one sentence in English to a binary array in the parallel system in Figure 13 with the input is this sentence;

Step 4: Return ABitArray;

We encrypt all the positive sentences of the training data set to the bit arrays based on the bit arrays of the sentiment lexicons of the bESD in the algorithm 17 perform the Hadoop Reduce phase of this stage in the parallel system, called the positive bit array group. The main ideas of the algorithm 17 are as follows:

Input: ABitArray – a bit array of one sentence;

Output:a positive bit array group - APositiveBitArrayGroup;

Step 1: Receive ABitArray;

Step 2: Add ABitArray into APositiveBitArrayGroup;

Step 3: Return APositiveBitArrayGroup;

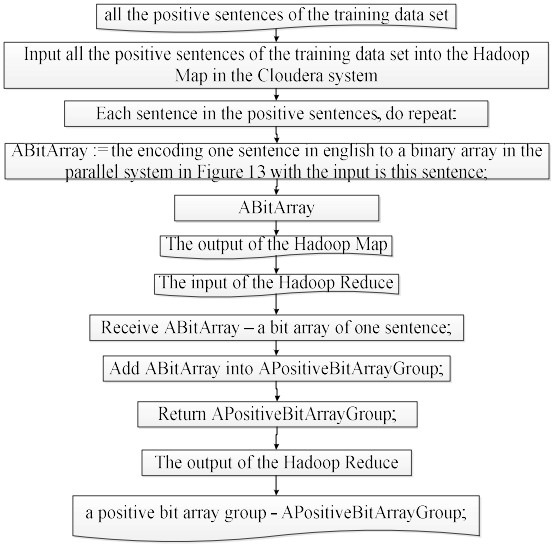


Figure 14: Overview of encrypting all the positive sentences of the training data set to the bit arrays based on the bit arrays of the sentiment lexicons of the bESD in the distributed environment, called the positive bit array group

In Figure 15, we build the algorithm 18 and the algorithm 19 to encrypt all the negative sentences of the training data set to the bit arrays based on the bit arrays of the sentiment lexicons of the bESD in the distributed environment, called the negative bit array group. This stage in Figure 15 comprises two phases as follows: Hadoop Map phase and Hadoop Reduce phase. The input of the Hadoop Map is all the negative sentences of the training data set. The output of the Hadoop Map is ABitArray – a bit array of one sentence. The input of the Hadoop Reduce is the output of the Hadoop Map, thus, the input of the Hadoop Reduce is ABitArray – a bit array of one sentence. The output of the Hadoop Reduce is a negative bit array group – ANegativeBitArrayGroup.

We encrypt all the negative sentences of the training data set to negative the bit arrays based on the bit arrays of the sentiment lexicons of the bESD in the algorithm 18 to perform the Hadoop Map phase of this stage in the distributed environment, called the negative bit array group. The main ideas of the algorithm 18 are as follows:

Input: all the negative sentences of the training data set

Output: ABitArray – a bit array of one sentence;

Step 1: Input all the negative sentences of the training data set into the Hadoop Map in the Cloudera system;

Step 2: Each sentence in the negative sentences, do repeat:

Step 3: ABitArray := the encoding one sentence in English to a binary array in the parallel system in Figure 13 with the input is this sentence;

Step 4: Return ABitArray;

We encrypt all the negative sentences of the training data set to the bit arrays based on the bit arrays of the sentiment lexicons of the bESD in the algorithm 19 perform the Hadoop Reduce phase of this stage in the parallel system, called the negative bit array group. The main ideas of the algorithm 19 are as follows:

Input: ABitArray – a bit array of one sentence;

Output:a negative bit array group - ANegativeBitArrayGroup;

Step 1: Receive ABitArray;

Step 2: Add ABitArray into ANegativeBitArrayGroup;

Step 3: Return ANegativeBitArrayGroup;

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

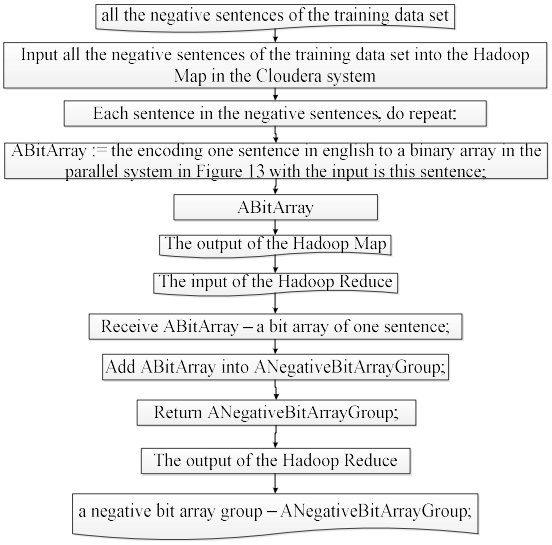


Figure 15: Overview of encrypting all the negative sentences of the training data set to the bit arrays based on the bit arrays of the sentiment lexicons of the bESD in the distributed environment, called the negative bit array group

In Figure 16, we build the algorithm 20 and the algorithm 21 to transfer one document of the testing data set into the bit arrays of the document in the parallel system. This stage in Figure 16 comprises two phases as follows: Hadoop Map phase and Hadoop Reduce phase. The input of the Hadoop Map is one document of the testing data set. The output of the Hadoop Map is one bit array of one sentence of the document – the output of the Hadoop Map. The input of the Hadoop Reduce is the output of the Hadoop Map, thus, the input of the Hadoop Reduce is one bit array of one sentence of the document – the output of the Hadoop Map. The output of the Hadoop Reduce is the bit arrays of the document.

We propose the algorithm 20 to perform the Hadoop Map phase of transferring one document of the testing data set into the bit arrays of the document in the parallel system. The main ideas of the algorithm 20 are as follows:

Input: one document of the testing data set

Output: ABitArray - one bit array of one sentence of the document – the output of the Hadoop Map;

Step 1: Input one document of the testing data set 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: ABitArray := the encoding one sentence in English to a binary array in the parallel system in Figure 13 with the input is this sentence;

Step 5: Return ABitArray;

We propose the algorithm 21 to perform the Hadoop Reduce phase of transferring one document of the testing data set into the bit arrays of the document in the parallel system. The main ideas of the algorithm 21 are as follows:

Input: ABitArray - one bit array of one sentence of the document – the output of the Hadoop Map;

Output: the bit arrays of the document - TheBitArraysOfTheDocument;

Step 1:Receive ABitArray;

Step 2: Add ABitArray into TheBitArraysOfTheDocument;

Step 3: Return TheBitArraysOfTheDocument;

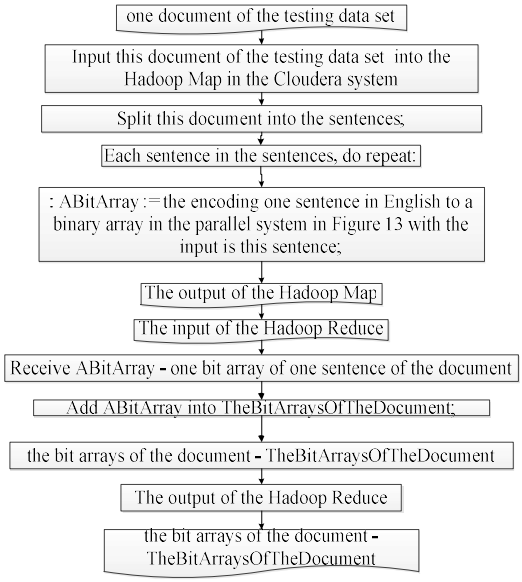


Figure 16: Overview of transferring one document of the testing data set into the bit arrays of the document in the parallel system

We present the information about the GA briefly as follows:

1)According to [45-49], we show the basic operations of the genetic algorithm, at the same time, also used for the GA in our sequential environment and our parallel network environment. The genetic algorithm (GA: Genetic Algorithms) and other evolutionary algorithms based on forming the notion the natural evolutionary process is reasonable, perfect. It stems from the evolved idea to survive and grow in the wild. GA is a problem-solving method to mimic the behavior of humans in order to survive and develop. It helps to find the optimal solution and the best in terms of time and space allow. GA considers all solutions, by at least some solutions, then eliminates the irrelevant

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

components and select the relevant components more adapted to create birth and evolution aimed at creating solutions which have a new adaptive coefficient increasing. The adaptive coefficient is used as a gauge of the solution. The main steps of the GA:

Step 1: Select models to symbolize the solutions. The models can be sequence (string) of the binary number: 1 and 0, decimal and can be letters or mixture letters and numbers.

Step 2: Select the adaptive function (or the Fitness function) to use as a gauge of the solution.

Step 3: Continue the transformation form until achieving the best solution, or until the termination of the time.

2)Genetic Operators and Genetic Operations

Reproductive Operator

Reproductive operator includes two processes: the reproduction process (allowing regeneration), the selection process (selection).

Allowing Regeneration

Allowing regeneration is the process which allows chromosomes to copy on the basis of the adaptive coefficient. The adaptive coefficient is a function which is assigned the real value, corresponding to each chromosome in the population.

This process is described as follows:

- Determine the adaptive coefficient of each chromosome in the population at generation t, tabulate cumulative adaptive values (in order assigned to each chromosome).
- Suppose, the population Has n individuals. Call the adaptive coefficient of the corresponding chromosome is fi, cumulative total is fti which is defined by

$$\text{fti} = \sum_{j=1}^{i} \text{fi}$$

- Call Fn is the sum of the adaptive coefficient in all the population. Pick a random number f between 0 and Fn. Select the first instance correspond $f \geq ftk$ into new population.

Selection Process (Selection)

The selection process is the process of removing the poor adaptive chromosomes in the population. This process is described as follows:

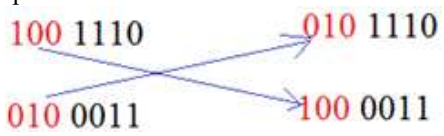
- Arrange population in order of descending degree of adaptation.
- Remove the chromosome in the last of the sequence. Keep n in the best individuals.

Crossover Operator (Crossover)

Crossover is the process of creating the new chromosomes based on the father-mother chromosomes by grafting a segment on the father-mother chromosomes together. The crossover operator is assigned with a probability pc. This process is described as follows:

- Randomly select a pair of chromosomes (father-mother) in the population. Suppose, the father-mother chromosomes Have the same length m.
- Create a random number in the range from 1 to m-1 (called as cross coupling point). The cross coupling point divides the father-mother chromosomes into two sub-strings which Have lengths m1, m2. The two new sub-strings created, is: m11 + m22 and m21 + m12.
- Put the two new chromosomes into the population.

Example:

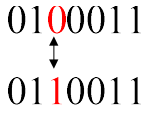


Mutation Operator (Mutation)

Mutation is a phenomenon which the child chromosomes carry some features, not in the genetic code of the father-mother chromosomes.

- Choose a random chromosome in the population;
- Create a random number k between 1 and m, $1 \leq k \leq m$;
- Change bit k. Put this chromosome in the population to participate in the evolution of the next generation.

Example:



Each pair of parents bears two children in one of the following two methods

Asexual Reproduction

Each child is an exact copy of each father or each mother.

Example:

Father: 01101100 ⟶ Child 1: 01101100

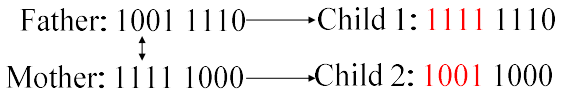
Mother: 11001110 ⟶ Child 2: 11001110

Sexual Reproduction (crossover)

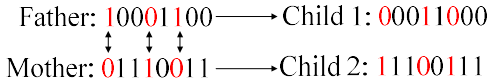
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Some bits are copied from the mother or a few bits are copied from the father.

Example of the sexual reproduction intersecting half



Example of the sexual reproduction intersecting 3 bits



In Figure 17, we show the diagram of the GA in the sequential environment as follows:

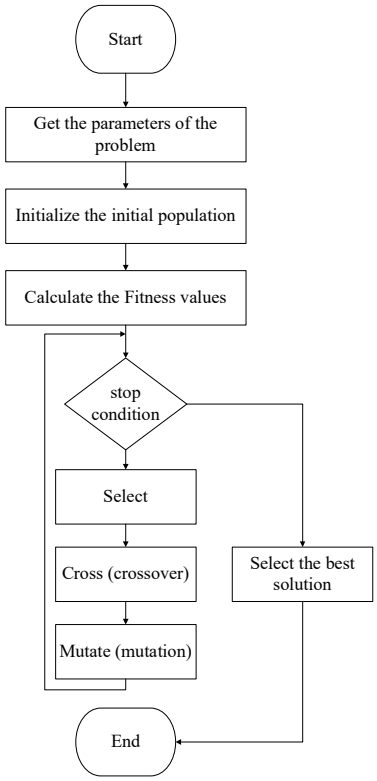


Figure 17: The diagram of the GA in the sequential environment.

In Figure 18, we build the algorithm 22 and the algorithm 23 to cluster one bit array (corresponding to one sentence) of the document into either the positive bit array group or the negative bit array group of the training data set by using the GA with the FPS in the distributed system. This stage in Figure 18 comprises two phases as follows: Hadoop Map phase and Hadoop Reduce phase. The input of the Hadoop Map is one bit array (corresponding to one sentence) of the document; the positive bit array group and the negative bit array group the training data set. The output of the Hadoop Map is the bit array clustered into either the positive bit array group or the negative bit array group of the training data set. The input of the Hadoop Reduce is the output of the Hadoop Map, thus, the input of the Hadoop Reduce is the bit array clustered into either the positive bit array group or the negative bit array group of the training data set. The output of the Hadoop Reduce is the bit array clustered into either the positive bit array group or the negative bit array group of the training data set.

We build the algorithm 22 to perform the Hadoop Map phase of clustering one bit array (corresponding to one sentence) of the document into either the positive bit array group or the negative bit array group of the training data set by using the GA with the FPS in the distributed system. The main ideas of the algorithm 22 are as follows:

Input: one bit array (corresponding to one sentence) of the document; the positive bit array group and the negative bit array group the training data set;

Output: the bit array clustered into either the positive bit array group or the negative bit array group of the training data set.

Step 1: Input the bit array (corresponding to one sentence) of the document; the positive bit array group and the negative bit array group the training data set into the Hadoop Map in the Cloudera system;

Step 2: randomly initialize population(t)

Step 3: determine fitness of population(t)

Step 4: repeat

Step 5: select parents from population(t)

Step 6: perform crossover on parents creating population(t+1)

Step 7: perform mutation of population(t+1)

Step 8: determine fitness of population(t+1)

Step 9: until best individual is good enough

Step 10: Return this bit array clustered into either the positive bit array group or the negative bit array group of the training data set.

We build the algorithm 23 to perform the Hadoop Reduce phase of clustering one bit array (corresponding to one sentence) of the document into either the positive bit array group or the negative bit array group of the training data set by using the GA with the FPS in the parallel system. The main ideas of the algorithm 23 are as follows:

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

Input: the bit array clustered into either the positive bit array group or the negative bit array group of the training data set.

Output: the sentiments (positive, negative, or neutral)

Step 1: Receive the bit array clustered into either the positive bit array group or the negative bit array group of the training data set;

Step 2: If this bit array clustered into the positive bit array group Then Return positive;

Step 3: If this bit array clustered into the negative bit array group Then Return negative;

Step 4: Return neutral;

Fitness is defined as an objective function the quantifies the optimality of a solution (chromosome) to the target problem. How to choose Fitness is dependent on the problem the we study. Choosing the different Fitness function will give the different results. In this survey, we use the Fitness-proportionate selection (FPS).

According to the researches related to the Fitness-proportionate Selection (FPS) in [50-54], Fitness-proportionate selection is detailed. Fitness proportionate selection is a genetic operator used in genetic algorithms for selecting potentially useful solutions for recombination. In fitness proportionate selection, as in all selection methods, the fitness function assigns a fitness to possible solutions or chromosomes. This fitness level is used to associate a probability of selection with each individual chromosome. If f_i is the fitness of individual i in the population, its probability of being selected is

$$p_i = \frac{f_i}{\sum_{j=1}^{N} f_j},$$

where N is the number of individuals in the population.

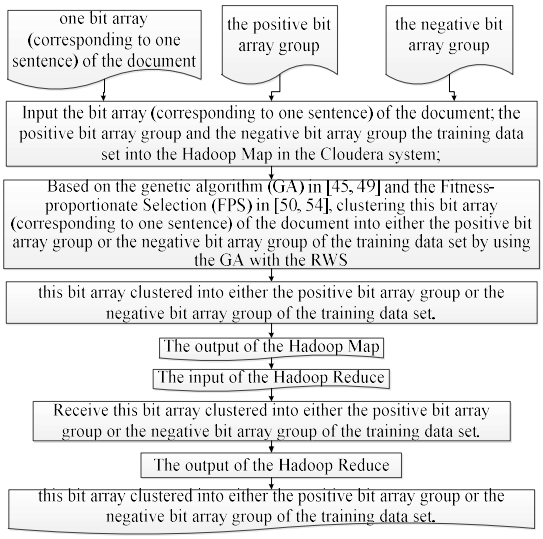


Figure 18: Overview of clustering one bit array (corresponding to one sentence) of the document into either the positive bit array group or the negative bit array group of the training data set by using the GA with the FPS in the distributed system

In Figure 19, we build the algorithm 24 and the algorithm 25 to cluster one document of the testing data set into either the positive bit array group or the negative bit array group of the training data set using the GA with the FPS in the distributed environment. This stage in Figure 19 comprises two phases as follows: Hadoop Map phase and Hadoop Reduce phase. The input of the Hadoop Map is one document of the testing data set; the positive bit array group and the negative bit array group of the training data set. The output of the Hadoop Map is OneResult – the sentiment classification of one bit array of the document. The input of the Hadoop Reduce is the output of the Hadoop Map, thus, the input of the Hadoop Reduce is OneResult – the sentiment classification of one bit array of the document. The output of the Hadoop Reduce is the sentiments (positive, negative, or neutral) of the document

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

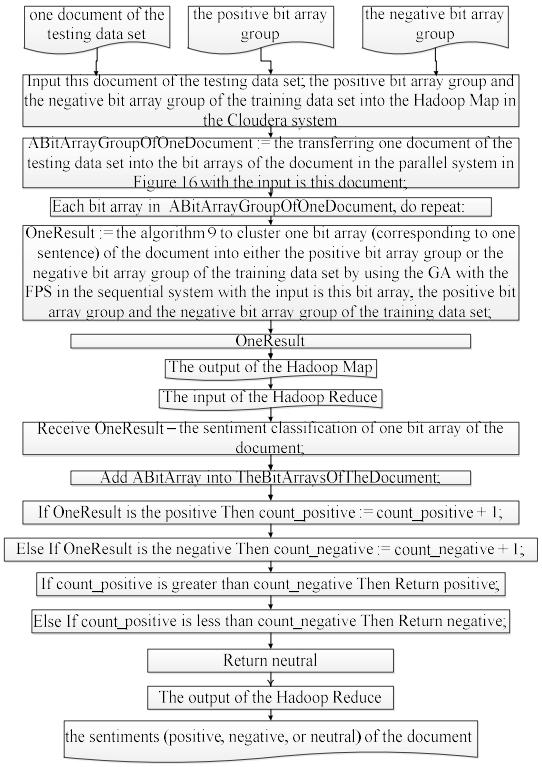


Figure 19: Overview of clustering one document of the testing data set into either the positive bit array group or the negative bit array group of the training data set using the GA with the FPS in the distributed environment

We propose the algorithm 24 to perform the Hadoop Map phase of clustering one document of the testing data set into either the positive bit array group or the negative bit array group of the training data set using the GA with the FPS in the distributed environment. The main ideas of the algorithm 24 are as follows:

Input: one document of the testing data set; the positive bit array group and the negative bit array group of the training data set;

Output: OneResult – the sentiment classification of one bit array of the document – the output of the Hadoop Map;

Step 1: Input the document of the testing data set; the positive bit array group and the negative bit array group of the training data set into the Hadoop Map in the Cloudera system;

Step 2: ABitArrayGroupOfOneDocument := the transferring one document of the testing data set into the bit arrays of the document in the parallel system in Figure 16 with the input is this document;

Step 3: Each bit array in ABitArrayGroupOfOneDocument, do repeat:

Step 4: OneResult := the algorithm 9 to cluster one bit array (corresponding to one sentence) of the document into either the positive bit array group or the negative bit array group of the training data set by using the GA with the FPS in the sequential system with the input is this bit array, the positive bit array group and the negative bit array group of the training data set;

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

We propose the algorithm 25 to perform the Hadoop Reduce phase of clustering one document of the testing data set into either the positive bit array group or the negative bit array group of the training data set using the GA with the FPS in the parallel environment. The main ideas of the algorithm 25 are as follows:

Input: OneResult – the sentiment classification of one bit array of the document – the output of the Hadoop Map;

Output: the sentiments (positive, negative, or neutral) of the document

Step 1: Receive OneResult – the sentiment classification of one bit array of the document;

Step 2: If OneResult is the positive Then count_positive := count_positive + 1;

Step 3: Else If OneResult is the negative Then count_negative := count_negative + 1;

Step 4: If count_positive is greater than count_negative Then Return positive;

Step 5: Else If count_positive is less than count_negative Then Return negative;

Step 6: Return neutral;

In Figure 20, we build the algorithm 26 and the algorithm 27 to perform the Hadoop Map phase of clustering the documents of the testing data set into either the positive or the negative in the distributed environment. This stage in Figure 20 comprises two phases as follows: Hadoop Map phase and Hadoop Reduce phase. The input of the Hadoop Map is the testing data set and the training data set. The output of the Hadoop Map is the result of the sentiment classification of one document the testing data set. The input of the Hadoop Reduce is the output of the Hadoop Map, thus, the input of the Hadoop Reduce is the result of the sentiment classification of one document the testing data set. The output of the Hadoop Reduce is the results of the sentiment classification of the testing data set;

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

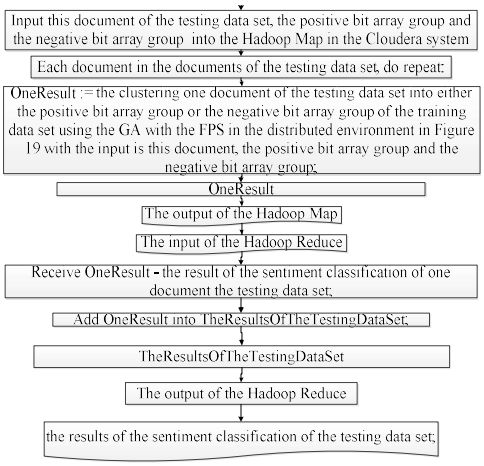


Figure 20: Overview of to performing the Hadoop Map phase of clustering the documents of the testing data set into either the positive or the negative in the distributed environment

We build the algorithm 26 to perform the Hadoop Map phase of clustering the documents of the testing data set into either the positive or the negative in the distributed environment. The main ideas of the algorithm 26 area as follows:

Input: the testing data set and the training data set;

Output: OneResult - the result of the sentiment classification of one document the testing data set – the output of the Hadoop Map ;

Step 1: The valences and the polarities of the sentiment lexicons of the bESD are calculated based on a basis English sentiment dictionary (bESD) in a distributed system (4.1.3);

Step 2: A positive bit array group := the encrypting all the positive sentences of the training data set to the bit arrays based on the bit arrays of the sentiment lexicons of the bESD in the distributed environment, called the positive bit array group in Figure 14 with the input is the positive sentences of the training data set;

Step 3: A negative bit array group := the encrypting all the negative sentences of the training data set to the bit arrays based on the bit arrays of the sentiment lexicons of the bESD in the distributed environment, called the negative bit array group in Figure 15 with the input is the positive sentences of the training data set;

Step 4: Input the documents of the testing data set, the positive bit array group and the negative bit array group into the Hadoop Map in the Cloudera system;

Step 5: Each document in the documents of the testing data set, do repeat:

Step 6: OneResult := the clustering one document of the testing data set into either the positive bit array group or the negative bit array group of the training data set using the GA with the FPS in the distributed environment in Figure 19 with the input is this document, the positive bit array group and the negative bit array group;

Step 7: Return OneResult;

We build the algorithm 27 to perform the Hadoop Reduce phase of clustering the documents of the testing data set into either the positive or the negative in the parallel environment. The main ideas of the algorithm 27 area as follows:

Input: OneResult - the result of the sentiment classification of one document the testing data set;

Output: the results of the sentiment classification of the testing data set;

Step 1: Receive OneResult - the result of the sentiment classification of one document the testing data set;

Step 2: Add OneResult into TheResultsOfTheTestingDataSet;

Step 3: Return TheResultsOfTheTestingDataSet;

5. EXPERIMENT

An Accuracy (A) is identified to calculate the accuracy of the results of the sentiment classification in this survey. We use a Java programming language programming to save data sets, implementing our proposed model to classify the 8,000,000 documents of the testing data set. To implement the proposed model, we have already used Java programming language to save the English testing data set and to save the results of the sentiment classification.

Our new model is performed 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 CHAhe, 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 Gower-2 Coefficient (HA) and the Genetic Algorithm (GA) with the Fitness-proportionate Selection (FPS) - the fitness function (FF).

We implement the proposed model related to the Gower-2 Coefficient (HA) and the Genetic Algorithm (GA) with the Fitness-proportionate Selection (FPS) - the fitness function (FF) in the Cloudera parallel network environment as follows:

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

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 CHAhe, 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 Gower-2 Coefficient (HA) and the Genetic Algorithm (GA) with the Fitness-proportionate Selection (FPS) - the fitness function (FF) in the Cloudera.

We show the significant information about this experiment of the proposed model in the tables as follows: Table 1, Table 2, and Table 3.

The results of the documents of the English testing data set to test are presented in Table 1 below.

The Accuracy of the sentiment classification of the documents in the English testing data set is shown in Table 2 below.

In Table 3 below, the average time of the classification of our new model for the English documents in testing data set are displayed

6. CONCLUSION

A new model using the Gower-2 Coefficient (HA) and the Genetic Algorithm (GA) with the Fitness-proportionate Selection (FPS) - the fitness function (FF) Has been built to cluster the documents in English with Hadoop Map (M) /Reduce (R) in the Cloudera parallel network environment in this survey. With our proposed new model, we have achieved 88.12% accuracy of the testing data set in Table 2. Until now, not many studies have shown the clustering methods can be used to classify data. Our research shows the clustering methods are used to classify data and, in particular, can be used to classify emotion in text.

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

We show the significant information about the average times of the sentiment classification of the proposed model in Table 3 as follows:

1)The average time of the semantic classification of using the Gower-2 Coefficient (HA) and the Genetic Algorithm (GA) with the Fitness-proportionate Selection (FPS) - the fitness function (FF) in the sequential environment is 41,648,984 seconds / 8,000,000 English documents and it is greater than the average time of the emotion classification of using the Gower-2 Coefficient (HA) and the Genetic Algorithm (GA) with the Fitness-proportionate Selection (FPS) - the fitness function (FF) in the Cloudera parallel network environment with 3 nodes which is 12,549,661 seconds / 8,000,000 English documents.

2)The average time of the semantic classification of using the Gower-2 Coefficient (HA) and the Genetic Algorithm (GA) with the Fitness-proportionate Selection (FPS) - the fitness function (FF) in the sequential environment is the longest time in the table.

3)The average time of the emotion classification of using the Gower-2 Coefficient (HA) and the Genetic Algorithm (GA) with the Fitness-proportionate Selection (FPS) - the fitness function (FF) in the Cloudera parallel network environment with 9 nodes, which is 4,649,887 seconds / 8,000,000 English documents, is the shortest time in the table.

4)Besides, The average time of the emotion classification of using the Gower-2 Coefficient (HA) and the Genetic Algorithm (GA) with the Fitness-proportionate Selection (FPS) - the fitness function (FF) in the Cloudera parallel network environment with 6 nodes is 7,074,830 seconds / 8,000,000 English documents

5)The average time of the emotion classification of using the Gower-2 Coefficient (HA) and the Genetic Algorithm (GA) with the Fitness-proportionate Selection (FPS) - the fitness function (FF) in the Cloudera parallel network environment with 3 nodes is greater than the average time of the emotion classification of using the Gower-2 Coefficient (HA) and the Genetic Algorithm (GA) with the Fitness-proportionate Selection (FPS) - the fitness function (FF) in the Cloudera parallel network environment with 6 nodes.

The execution time of using the Gower-2 Coefficient (HA) and the Genetic Algorithm (GA) with the Fitness-proportionate Selection (FPS) - the fitness function (FF) in the Cloudera is dependent on the performance of the Cloudera parallel system and also dependent on the performance of each server on the Cloudera system.

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

1)The GA – related algorithms.

2)The Gower-2 Coefficient – related algorithms.

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

3)The performance of the distributed network environment.

4)The performance of each node of the distributed environment.

5)The performance of each server of the parallel system.

6)The number of the nodes of the parallel environment.

7)The testing data set and the training data set.

8)The sizes of the data sets.

9)The parallel functions such as Hadoop Map and Hadoop Reduce.

10)The operating system of the parallel network such as the Cloudera.

The proposed model has many advantages and disadvantages. Its positives are as follows: It the Gower-2 Coefficient (HA) and the Genetic Algorithm (GA) with the Fitness-proportionate Selection (FPS) - the fitness function (FF) to classify semantics of English documents based on sentences. The proposed model can process millions of documents in the shortest time. This study can be performed in distributed systems to shorten the execution time of the proposed model. It can be applied to other languages. It can save a lot of the storage spaces. 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 as follows: Table 8, Table 9, Table 10, Table 11, Table 12, and Table 13.

In Table 8, we show the comparisons of our model's results with the genetic algorithm (GA) in [45-49].

The comparisons of our model's benefits and drawbacks with the works related to the genetic algorithm (GA) in [45-49] are displayed in Table 9.

In Table 10, we display the comparisons of our model's results with the Fitness-proportionate Selection (FPS) in [50-54].

The comparisons of our model's advantages and disadvantages with the Fitness-proportionate Selection (FPS) in [50-54] are presented in Table 11.

In Table 12, we show the comparisons of our model with the latest sentiment classification models (or the latest sentiment classification methods) in [55-65].

The comparisons of our model's positives and negatives with the latest sentiment classification models (or the latest sentiment classification methods) in [55-65] are displayed in Table 13.

FUTURE WORK

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

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 HAL 2010 Student Research Workshop, Sweden, 2010, pp 25–30.

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

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

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

[8] Julian Brooke, Milan Tofiloski, Maite Taboada, "Cross-Linguistic Sentiment Analysis: From English to Spanish", International Conference

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

RANLP 2009 - Borovets, Bulgaria, 2009, pp 50–54.

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

[10]Tan, S.; ZHAng, J., "An empirical study of sentiment analysis for Chinese documents", Expert Systems with Applications (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 NAHAL HLT 2007, Companion Volume, NY, 2007, pp 189–192.

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

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

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

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

[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(IJHAT), 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 BHAhelier, 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, JHAob Goldenberg, Moshe Fresko, "Mine Your Own Business: Market-Structure Surveillance Through Text Mining", Marketing Science, Vol. 31, No. 3, 2012, pp 521-543.

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

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

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

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

[26]Snežana MLADENOVIĆ DRINIĆ, Ana NIKOLIĆ, Vesna PERIĆ, "Cluster Analysis of Soybean Genotypes Based on RAPD Markers",

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

Proceedings 43rd Croatian and 3rd International Symposium on Agriculture. Opatija. Croatia, 2008, 367- 370.

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

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

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

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

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

[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, 2017, 69 pages.

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

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

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

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

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

[38]MHAMillan English Dictionary, http://www.mHAmillandictionary.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]Schaid D.J, "Genomic Similarity and Kernel Methods II: Methods for Genomic Information" Hum Hered 2010;70:132–140, https://doi.org/10.1159/000312643, 2010

[41]Yangjian Zhang, Ming Xu, Hua Chen, Jonathan Adams, "Global pattern of NPP to GPP ratio derived from MODIS data: effects of ecosystem type, geographical location and climate", Global Ecology and Biogeography, Volume 18, Issue 3, , 2009, Pages 280–290 , DOI: 10.1111/j.1466-8238.2008.00442.x

[42]Dr.rer.pol., MS U. Helmert (Research Fellow) 1, MD MS S. Shea, "Social inequalities and health status in western Germany", Public Health, Volume 108, Issue 5, 1994, Pages 341-356, https://doi.org/10.1016/S0033-3506(05)80070-8

[43]Meilan M. Rutter, James Collins, Susan R. Rose, Jessica G. Woo, Heidi Sucharew, Hemant Sawnani, Kan N. Hor, Linda H. Cripe, Brenda L. Wong, "Growth hormone treatment in boys with Duchenne muscular dystrophy and glucocorticoid-induced growth failure", Neuromuscular Disorders, Volume 22, Issue 12, 2012, Pages 1046-1056, https://doi.org/10.1016/j.nmd.2012.07.009

[44]John Yu, Victor Lemas, Theodore Page, James D. Connor, Alice L. Yu, "Induction of Erythroid Differentiation in K562 Cells by Inhibitors of Inosine Monophosphate Dehydrogenase", Cancer Research, Volume 49, Issue 20, 1989

[45]Davis, L. (ed.), "HAndbook of genetic algorithms", Van Nostrand Reinhold, New York., 1991.

[46]Padmavathi Kora, K. Sri Rama Krishna, "Bundle Block Detection Using Genetic Neural Network", Information Systems Design and Intelligent Applications, Volume 434 of the series Advances in Intelligent Systems and Computing, 2016, 309-317.

[47]Gang Yang, SHAohui Wu, Qin Jin, Jieping Xu, "A hybrid approach based on stocHastic competitive Hopfield neural network and efficient genetic algorithm for frequency assignment problem", Applied Soft Computing, Volume 39, 2016, 104–116.

[48]Selçuk Erkaya, İbrahim Uzmay, "Balancing of Planar MecHAnisms HAving Imperfect Joints Using Neural Network-Genetic Algorithm (NN-GA) Approach", Dynamic Balancing of MecHAnisms and Synthesizing of Parallel Robots, 2016, 299-317.

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

[49]Jing Wu, Huapeng Wu, Yuntao Song, Yong Cheng, Wenglong ZHAo, Yongbo Wang, "Genetic algorithm trajectory plan optimization for EAMA: EAST Articulated Maintenance Arm", Fusion Engineering and Design, 2016

[50]Hod Lipson, Jordan B. Pollack, "Automatic design and manufacture of robotic lifeforms", Nature 406, 974-978,31 August 2000, doi:10.1038/35023115;

[51]Eckart Zitzler, Lothar Thiele, "Multiobjective optimization using evolutionary algorithms — A comparative case study", International Conference on Parallel Problem Solving from Nature PPSN 1998: Parallel Problem Solving from Nature — PPSN V , 1998, pp 292-301

[52]Peter J. B. Hancock, "An empirical comparison of selection methods in evolutionary algorithms", AISB Workshop on Evolutionary Computing AISB EC 1994: Evolutionary Computing, 1994, pp 80-94

[53]Brad L. Miller, David E. Goldberg, "Genetic Algorithms, Selection Schemes, and the Varying Effects of Noise", Evolutionary Computation, Volume 4 | Issue 2, 2007, p.113-131

[54]Pengfei Guo; Xuezhi Wang; Yingshi Han, "The enhanced genetic algorithms for the optimization design", 2010 3rd International Conference on Biomedical Engineering and Informatics (BMEI), DOI: 10.1109/BMEI.2010.5639829 , Yantai, China, 2010

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

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

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

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

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

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

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

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

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

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

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

[66]ASCII of Wikipedia, https://en.wikipedia.org/wiki/ASCII, 2017.

[67]ASCII Codes Table, http://ascii.cl/, 2017.

[68]ASCII Table, http://www.theasciicode.com.ar/, 2017

[69]ASCII CHAracter Set, http://ee.HAwaii.edu/~tep/EE160/Book/cHAp4/subsection2.1.1.1.html, 2017.

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

[70]ASCII AlpHAbet CHAracters, http://www.kerryr.net/pioneers/ascii2.htm, 2017.

[71]ASCII Code, http://www.ascii-code.net/, 2017

[72]Decimal to Binary Converter, http://www.binaryhexconverter.com/decimal-to-binary-converter, 2017

[73]Decimal to binary, http://www.rapidtables.com/convert/number/decimal-to-binary.htm, 2017.

[74]Converting from decimal to binary, https://www.kHAnacademy.org/math/algebra-home/alg-intro-to-algebra/algebra-alternate-number-bases/v/decimal-to-binary, 2017

[75]wikiHow to Convert from Decimal to Binary, http://www.wikihow.com/Convert-from-Decimal-to-Binary, 2017.

[76]Converting Decimal Numbers to Binary Numbers, http://interactivepython.org/runestone/static/pythonds/BasicDS/ConvertingDecimalNumberstoBinaryNumbers.html, 2017

[77]Binary to Decimal Conversion, http://www.electronics-tutorials.ws/binary/bin_2.html, 2017.

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

APPENDICES

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

Table 2: Comparisons of our model's advantages and disadvantages with the works related to [1-32].

Table 3: Comparisons of our model's results with the works related to the GOWER-2 coefficient (HA) in [39-44].

Table 4: Comparisons of our model's benefits and drawbacks with the works related to the GOWER-2 coefficient (HA) in [39-44].

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

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

Table 7: Average time of the classification of our new model for the English documents in testing data set.

Table 8: Comparisons of our model's results with the genetic algorithm (GA) in [45-49].

Table 9: Comparisons of our model's benefits and drawbacks with the works related to the genetic algorithm (GA) in [45-49].

Table 10: Comparisons of our model's results with the Fitness-proportionate Selection (FPS) in [50-54].

Table 11: Comparisons of our model's advantages and disadvantages with the Fitness-proportionate Selection (FPS) in [50-54].

Table 12: Comparisons of our model with the latest sentiment classification models (or the latest sentiment classification methods) in [55-65]

Table 13: Comparisons of our model's positives and negatives with the latest sentiment classification models (or the latest sentiment classification methods) in [55-65]

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

GOWER-2 coefficient (HA)

Semantic classification, sentiment classification: SC

Studies	PMI	JM	Language	SD	DT	HA	SC	Other measures	Search engines
[1]	Yes	No	English	Yes	Yes	No	Yes	No	No Mention
[2]	Yes	No	English	Yes	No	No	Yes	Latent Semantic Analysis (LSA)	AltaVista
[3]	Yes	No	Englis	Yes	Ye	No	Yes	Baseline; Turne	AltaVista
			h		s			y-inspired; NB; Cluster+NB; Human	
[4]	Yes	No	English German	Yes	Yes	No	Yes	SimRank	Google search engine
[5]	Yes	No	English Macedonian	Yes	Yes	No	Yes	No Mention	AltaVista search engine
[6]	Yes	No	English Arabic	Yes	No	No	Yes	No Mention	Google search engine Bing search engine
[7]	Yes	No	English Chinese	Yes	Yes	No	Yes	SVM(CN); SVM(EN); SVM(ENCN1); SVM(ENCN2); TSVM(CN); TSVM(EN); TSVM(ENCN1); TSVM(ENCN2); CoTrain	No Mention
[8]	Yes	No	English Spanish	Yes	Yes	No	Yes	SO Calculation SVM	Google
[9]	Yes	No	Chinese Tibeta	Yes	Yes	No	Yes	- Feature selection - Expec	No Mention

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

			n					tation Cross Entropy - Information Gain	
[10]	Yes	No	Chinese	Yes	Yes	No	Yes	DF, CHI, MI andIG	No Mention
[11]	Yes	No	Chinese	Yes	No	No	Yes	Information Bottleneck Method (IB); LE	AltaVista
[12]	Yes	No	Chinese	Yes	Yes	No	Yes	SVM	Google Yahoo Baidu
[13]	Yes	No	Japanese	No	No	No	Yes	HArmonic–Mean	Google and replaced the NEAR operator with the AND operator inthe SO formula.
[14]	Yes	Yes	English	Yes	Yes	No	Yes	Dice; NGD	Google search engine
[15]	Yes	Yes	English	Yes	No	No	Yes	Dice; Overlap	Google
[16]	No	Yes	English	Yes	Yes	No	Yes	A Jaccard index based clustering algorithm (JIBCA)	No Mention
[17]	No	Yes	English	Yes	Yes	No	Yes	Naıve Bayes, Two-Step Multinomial Naıve Bayes	Google
[18]	No	Yes	Arabic	No	No	No	Yes	Naive Bayes (NB); Support Vector Machines (SVM); Rocchio; Cosine	No Mention
[19]	No	Yes	Chinese	Yes	Yes	No	Yes	A new score – Economic Value (EV), etc.	Chinese search
[20]	No	Yes	Chinese	Yes	Yes	No	Yes	Cosine	No Mention
[21]	No	Yes	English	No	Yes	No	Yes	Cosine	No Mention
[22]	No	Yes	Chinese	No	Yes	No	Yes	Dice; overlap; Cosine	No Mention
[28]	No	No	Vietnamese	No	No	No	Yes	Ochiai Measure	Google
[29]	No	No	English	No	No	No	Yes	Cosine coefficient	Google
[30]	No	No	English	No	No	No	Yes	Sorensen measure	Google
[31]	No	Yes	Vietnamese	No	No	No	Yes	Jaccard	Google

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

[32]	No	No	English	No	No	No	Yes	Tanimoto coefficient	Google
Our work	No	No	English Language	No	No	Yes	Yes	No	Google search engine

Table 2: Comparisons of our model's advantages and disadvantages with the works related to [1-32].

Surveys	Approach	Advantages	Disadvantages
[1]	Constructing sentiment lexicons in Norwegian from a large text corpus	Through the authors' PMI computations in this survey they used a distance of 100 words from the seed word, but it might be the other lengths the generate better sentiment lexicons. Some of the authors' preliminary research showed the 100 gave a better result.	The authors need to investigate this more closely to find the optimal distance. Another factor the Has not been investigated much in the literature is the selection of seed words. Since they are the basis for PMI calculation, it might be a lot to gain by finding better seed words. The authors would like to explore the impact the different approaches to seed word selection Have on the performance of the developed sentiment lexicons.
[2]	Unsupervised Learning of Semantic Orientation from a Hundred-Billion-Word Corpus.	This survey Has presented a general strategy for learning semantic orientation from semantic association, SO-A. Two instances of this strategy Have been empirically evaluated, SO-PMI-IR and SO-LSA. The Accuracy of SO-PMI-IR is comparable to the Accuracy of HM, the algorithm of HAtzivassiloglou and McKeown (1997). SO-PMI-IR requires a large corpus, but it is simple, easy to implement, unsupervised, and it is not restricted to adjectives.	No Mention
[3]	Graph-based user classification for informal online political discourse	The authors describe several experiments in identifying the political orientation of posters in an informal environment. The authors' results indicate the most promising approach is to augment text classification methods by exploiting information about how posters interact with each other	There is still much left to investigate in terms of optimizing the linguistic analysis, beginning with spelling correction and working up to sHAllow parsing and co-reference identification. Likewise, it will also be worthwhile to further investigate exploiting sentiment values of phrases and clauses, taking cues from methods
[4]	A novel, graph-based approach using SimRank.	The authors presented a novel approach to the translation of sentiment information the outperforms SOPMI, an established method. In particular, the authors could show the SimRank outperforms SO-PMI for values of the threshold x in an interval the most likely leads to the correct separation of positive, neutral, and negative adjectives.	The authors' future work will include a further examination of the merits of its application for knowledge-sparse languages.
[5]	Analysis in Twitter for Macedonian	The authors' experimental results show an F1-score of 92.16, which is very strong and is on par with the best results for English, which were Achieved in recent SemEval competitions.	In future work, the authors are interested in studying the impact of the raw corpus size, e.g., the authors could only collect Half a million tweets for creating lexicons and analyzing/evaluating the system, while Kiritchenko et al. (2014) built their lexicon on million tweets and evaluated their system on 135 million English tweets. Moreover, the authors are interested not only in quantity but also in quality, i.e., in studying the quality of the individual words and phrases used as seeds.
[6]	Using Web Search Engines	- For the General English sub-task, the authors' system Has modest but	Although the results are encouraging, further investigation is required, in both

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

	for English and Arabic Unsupervised Sentiment Intensity Prediction	interesting results. - For the Mixed Polarity English sub-task, the authors' system results achieve the second place. - For the Arabic phrases sub-task, the authors' system Has very interesting results since they applied the unsupervised method only	languages, concerning the choice of positive and negative words which once associated to a phrase, they make it more negative or more positive.
[7]	Co-Training for Cross-Lingual Sentiment Classification	The authors propose a co-training approach to making use of unlabeled Chinese data. Experimental results show the effectiveness of the proposed approach, which can outperform the standard inductive classifiers and the transductive classifiers.	In future work, the authors will improve the sentiment classification Accuracy in the following two ways: 1) The smoothed co-training approach used in (MiHAlcea, 2004) will be adopted for sentiment classification. 2) The authors will employ the structural correspondence learning (SCL) domain adaption algorithm used in (Blitzer et al., 2007) for linking the translated text and the natural text.
[8]	Cross-Linguistic Sentiment Analysis: From English to Spanish	Our Spanish SO calculator (SOCAL) is clearly inferior to the authors' English SO-CAL, probably the result of a number of factors, including a small, preliminary dictionary, and a need for additional adaptation to a new language. Translating our English dictionary also seems to result in significant semantic loss, at least for original Spanish texts.	No Mention
[9]	Micro–blog Emotion Orientation Analysis Algorithm Based on Tibetan and Chinese Mixed Text	By emotion orientation analyzing and studying of Tibetan microblog which is concerned in Sina, making Tibetan Chinese emotion dictionary, Chinese sentences, Tibetan part of speech sequence and emotion symbol as emotion factors and using expected cross entropy combined fuzzy set to do feature selection to realize a kind of microblog emotion orientation analyzing algorithm based on Tibetan and Chinese mixed text. The experimental results showed the method can obtain better performance in Tibetan and Chinese mixed Microblog orientation analysis.	No Mention
[10]	An empirical study of sentiment analysis for Chinese documents	Four feature selection methods (MI, IG, CHI and DF) and five learning methods (centroid classifier, K-nearest neighbor, winnow classifier, Naïve Bayes and SVM) are investigated on a Chinese sentiment corpus with a size of 1021 documents. The experimental results indicate the IG performs the best for sentimental terms selection and SVM exhibits the best performance for sentiment classification. Furthermore, the authors found the sentiment classifiers are severely dependent on domains or topics.	No Mention
[11]	Adapting Information Bottleneck Method for Automatic Construction of Domain-oriented Sentiment Lexicon	The authors' theory verifies the convergence property of the proposed method. The empirical results also support the authors' theoretical analysis. In their experiment, it is shown the proposed method greatly outperforms the baseline methods in the task of building out-of-domain sentiment lexicon.	In this study, only the mutual information measure is employed to measure the three kinds of relationship. In order to show the robustness of the framework, the authors' future effort is to investigate how to integrate more measures into this framework.
[12]	Sentiment Classification for Consum	This study adopts three supervised learning approaches and a web-based semantic orientation	No Mention

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

	er Word-of-Mouth in Chinese: Comparison between Supervised and Unsupervised Approaches	approach, PMI-IR, to Chinese reviews. The results show the SVM outperforms naive bayes and N-gram model on various sizes of training examples, but does not obviously exceeds the semantic orientation approach when the number of training examples is smaller than 300.	
[13]	Modifying SO-PMI for Japanese Weblog Opinion Mining by Using a Balancing Factor and Detecting Neutral Expressions	After these modifications, the authors Achieved a well-balanced result: both positive and negative Accuracy exceeded 70%. This shows the authors' proposed approach not only adapted the SO-PMI for Japanese, but also modified it to analyze Japanese opinions more effectively.	In the future, the authors will evaluate different choices of words for the sets of positive and negative reference words. The authors also plan to appraise their proposal on other languages.
[14]	In this survey, the authors empirically evaluate the performance of different corpora in sentiment similarity measurement, which is the fundamental task for word polarity classification.	Experiment results show the Twitter data can Achieve a much better performance than the Google, Web1T and Wikipedia based methods.	No Mention
[15]	Adjective-Based Estimation of Short Sentence 's Impressi	The adjectives are ranked and top na adjectives are considered as an output of system. For example, the experiments were carried out and got	In the authors' future work, they will improve more in the tasks of keyword extraction and semantic similarity methods to make the proposed
	on	fairly good results. With the input "it is snowy", the results are white (0.70), light (0.49), cold (0.43), solid (0.38), and scenic (0.37)	system working well with complex inputs.
[16]	Jaccard Index based Clustering Algorithm for Mining Online Review	In this work, the problem of predicting sales performance using sentiment information mined from reviews is studied and a novel JIBCA Algorithm is proposed and mathematically modeled. The outcome of this generates knowledge from mined data the can be useful for forecasting sales.	For future work, by using this framework, it can extend it to predicting sales performance in the other domains like customer electronics, mobile phones, computers based on the user reviews posted on the websites, etc.
[17]	Twitter sentiment classification for measuring public health concerns	Based on the number of tweets classified as Personal Negative, the authors compute a Measure of Concern (MOC) and a timeline of the MOC. We attempt to correlate peaks of the MOC timeline to the peaks of the News (Non-Personal) timeline. The authors' best Accuracy results are Achieved using the two-step method with a Naïve Bayes classifier for the Epidemic domain (six datasets) and the Mental Health domain (three datasets).	No Mention
[18]	Ensemble of Classification algorithms for Subjectivity and Sentiment Analysis of Arabic Customers' Reviews	The experimental results show the ensemble of the classifiers improves the classification effectiveness in terms of macro-F1 for both levels. The best results obtained from the subjectivity analysis and the sentiment classification in terms of macro-F1 are 97.13% and 90.95% respectively.	No Mention

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

[19]	Automatic Construction of Financial Semantic Orientation Lexicon from Large-Scale Chinese News Corpus	Semantic orientation lexicon of positive and negative words is indispensable for sentiment analysis. However, many lexicons are manually created by a small number of human subjects, which are susceptible to high cost and bias. In this survey, the authors propose a novel idea to construct a financial semantic orientation lexicon from large-scale Chinese news corpus automatically ...	No Mention			hyper-parameter settings. Considering the difficulty of tuning hyper-parameters in a resourcescarce setting, the stable performance of parameter-free label propagation is promising.	
[20]	Sentiment Classification in Under-Resourced Languages Using Graph-based Semi-supervised Learning Methods	In particular, the authors found the choosing initially labeled vertices in HAcordance with their degree and PageRank score can improve the performance. However, pruning unreliable edges will make things more difficult to predict. The authors believe the other people who are interested in this field can benefit from their empirical findings.	As future work, first, the authors will attempt to use a sophisticated approach to induce better sentiment features. The authors consider such elaborated features improve the classification performance, especially in the book domain. The authors also plan to exploit a much larger amount of unlabeled data to fully take advantage of SSL algorithms	[28]	A Vietnamese adjective emotion dictionary based on exploitation of Vietnamese language chracteristics	The Vietnamese adjectives often bear emotion which values (or semantic scores) are not fixed and are changed when they appear in different contexts of these phrases. Therefore, if the Vietnamese adjectives bring sentiment and their semantic values (or their sentiment scores) are not changed in any context, then the results of the emotion classification are not high Accuracy. The authors propose many rules based on Vietnamese language chracteristics to determine the emotional values of the Vietnamese adjective phrases bearing sentiment in specific contexts. The authors' Vietnamese sentiment adjective dictionary is widely used in applications and researches of the Vietnamese semantic classification.	not calculating all Vietnamese words completely; not identifying all Vietnamese adjective phrases fully, etc.
[21]	A text-mining approach and combine it with semantic network analysis tools	In summary, the authors hope the text-mining and derived market-structure analysis presented in this paper provides a first step in exploring the extremely large, rich, and useful body of consumer data readily available on Web 2.0.	No Mention				
[22]	Sentiment Classification in Resource-Scarce Languages by using Label Propagation	The authors compared our method with supervised learning and semi-supervised learning methods on real Chinese reviews classification in three domains. Experimental results demonstrated the label propagation showed a competitive performance against SVM or Transductive SVM with best	The authors plan to further improve the performance of LP in sentiment classification, especially when the authors only Have a small number of labeled seeds. The authors will exploit the idea of restricting the label propagating steps when the available labeled data is quite small.	[29]	A Valences - Totaling Model for English Sentiment Classification	The authors present a full range of English sentences; thus, the emotion expressed in the English text is classified with more precision. The authors new model is not dependent on a special domain and training data set—it is a domain-independent classifier. The authors test our new model on the Internet data in English. The calculated valence (and polarity) of	It Has low Accuracy; it misses many sentiment-bearing English words; it misses many sentiment-bearing English phrases because sometimes the valence of a English phrase is not the total of the valences of the English words in this phrase; it misses many English sentences which are not processed fully; and it misses

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

		English semantic words in this model is based on many documents on millions of English Web sites and English social networks.	many English documents which are not processed fully.
[30]	Shifting Semantic Values of English Phrases for Classification	The results of the sentiment classification are not high Accuracy if the English phrases bring the emotions and their semantic values (or their sentiment scores) are not changed in any context. For those reasons, the authors propose many rules based on English language grammars to calculate the sentimental values of the English phrases bearing emotion in their specific contexts. The results of this work are widely used in applications and researches of the English semantic classification.	This survey is only applied to the English adverb phrases. The proposed model is needed to research more and more for the different types of the English words such as English noun, English adverbs, etc
[31]	A Valence-Totaling Model for Vietnamese Sentiment Classification	The authors Have used the VTMfV to classify 30,000 Vietnamese documents which include the 15,000 positive Vietnamese documents and the 15,000 negative Vietnamese documents. The authors Have Achieved Accuracy in 63.9% of the authors' Vietnamese testing data set. VTMfV is not dependent on the special domain. VTMfV is also not dependent on the training data set and there is no training stage in this VTMfV. From the authors' results in this work, our VTMfV can be applied in the different fields of the Vietnamese natural language processing. In addition, the authors' TCMfV can	it Has a low Accuracy.
		be applied to many other languages such as Spanish, Korean, etc. It can also be applied to the big data set sentiment classification in Vietnamese and can classify millions of the Vietnamese documents	
[32]	Semantic Lexicons of English Nouns for Classification	The proposed rules based on English language grammars to calculate the sentimental values of the English phrases bearing emotion in their specific contexts. The results of the sentiment classification are not high Accuracy if the English phrases bring the emotions and their semantic values (or their sentiment scores) are not changed in any context. The valences of the English words (or the English phrases) are identified by using Tanimoto Coefficient (TC) through the Google search engine with AND operator and OR operator. The emotional values of the English noun phrases are based on the English grammars (English language chracteristics)	This survey is only applied in the English noun phrases. The proposed model is needed to research more and more about the different types of the English words such as English English adverbs, etc.
Our work	-GOWER-2 coefficient (HA) through the Google search engine with AND operator and OR operator. -We use the Gower-2 Coefficient (HA) and the Genetic Algorithm (GA) with the fitness function (FF) which is the Fitness-proportionate Selection (FPS) to classify one document of the testing data set into either the positive polarity or the negative polarity in both the sequential environment and the distributed system. The advantages and disadvantages of this survey are shown in the Conclusion section.		

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

Table 3: Comparisons of our model's results with the works related to the GOWER-2 coefficient (HA) in [39-44].

Studies	PMI	JM	GOWER-2 coefficient (HA)	Language	SD	DT	Sentiment Classification
[39]	Yes	Yes	Yes	English	NM	NM	No mention
[40]	No	No	Yes	NM	NM	NM	No mention
[41]	No	No	Yes	NM	NM	NM	No mention
[42]	No	No	Yes	NM	NM	NM	No mention
[43]	No	No	Yes	NM	NM	NM	No mention
[44]	No	No	Yes	NM	NM	NM	No mention
Our work	No	No	Yes	English Language	Yes	Yes	Yes

Table 4: Comparisons of our model's benefits and drawbacks with the studies related to the GOWER-2 coefficient (HA) 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]	Genomic Similarity and Kernel Methods II: Methods for Genomic Information	Although it is difficult to create a cook book of kernels for genetic studies, useful guidelines can be gleaned from a variety of novel published approaches. The authors review some novel developments of kernels for specific analyses and speculate on how to build kernels for complex genomic attributes based on publically available data. The creativity of analysts, with rigorous evaluations by applications to real and simulated data, will ultimately provide a much stronger array of kernel 'tools' for genetic analyses.	No mention
[41]	Global pattern of NPP to GPP ratio derived from MODIS data: effects of ecosystem type, geographical location and climate	The NPP/GPP ratio exhibited a pattern depending on the main climatic characteristics such as temperature and precipitation and geographical factors such as latitude and altitude. The findings of this research challenge the widely held assumption that the NPP/GPP ratio is consistent regardless of ecosystem type	No mention
[42]	Social inequalities and health status in western Germany	In western Germany, despite a health system with almost free access for the general population, strong social class inequalities exist for many diseases. These inequalities cannot be explained by social class differences in smoking, obesity or Pattern A behaviour. More research is needed to identify underlying causes for these persistent social inequalities in health status	No mention
[43]	Growth hormone treatment in boys with Duchenne muscular dystrophy and glucocorticoid-induced growth failure	The rate of weight gain was unchanged, at 2.8 ± 0.6 kg/year pre-growth hormone and 2.6 ± 0.7 kg/year at 1 year. Motor function decline was similar pre-growth hormone and at 1 year. Cardiopulmonary function was unchanged. Three experienced side effects. In this first comprehensive report of growth hormone in Duchenne muscular dystrophy, growth hormone improved growth at 1 year, without detrimental effects observed on neuromuscular and cardiopulmonary	No mention

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

		function.	
[44]	Induction of Erythroid Differentiation in K562 Cells by Inhibitors of Inosine Monophosphate Dehydrogenase	Studies with isoelectric focusing, globin chain analyses, and immunochemical assays indicated that both Aγ and Gγ were detected and that the hemoglobin produced in the ribavirin-treated cells consisted of approximately 60% fetal hemoglobin and its acetylated equivalents. The adult-type α globin was found, while no β globin chains were demonstrated. Thus, accumulation of fetal hemoglobin and production of α globin chain in ribavirin-treated cells are different from the pattern of hemoglobins induced by hemin.	No mention
Our work	-GOWER-2 coefficient (HA) through the Google search engine with AND operator and OR operator. -We use the Gower-2 Coefficient (HA) and the Genetic Algorithm (GA) with the fitness function (FF) which is the Fitness-proportionate Selection (FPS) to classify one document of the testing data set into either the positive polarity or the negative polarity in both the sequential environment and the distributed system. The advantages and disadvantages of this survey are shown in the Conclusion section.		

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

	Testing Dataset	**Correct Classification**	**Incorrect Classification**
Negative	4,000,000	3,534,727	465,273
Positive	4,000,000	3,514,873	485,127
Summary	8,000,000	7,049,600	950,400

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

Proposed Model	Class	Accuracy
Our new model	Negative	87.12 %
	Positive	

Table 7: Average time of the classification of our new model for the English documents in testing data set.

	Average time of the classification /8,000,000 English documents.
The Gower-2 Coefficient (HA) and the Genetic Algorithm (GA) with the fitness function (FF) which is the Fitness-proportionate Selection (FPS) in the sequential environment	41,648,984 seconds
The Gower-2 Coefficient (HA) and the Genetic Algorithm (GA) with the fitness function (FF) which is the Fitness-proportionate Selection (FPS) in the Cloudera distributed system with 3 nodes	12,549,661 seconds
The Gower-2 Coefficient (HA) and the Genetic Algorithm (GA) with the fitness function (FF) which is the Fitness-proportionate Selection (FPS) in the Cloudera distributed system with 6 nodes	7,074,830 seconds
The Gower-2 Coefficient (HA) and the Genetic Algorithm (GA) with the fitness function (FF) which is the Fitness-proportionate Selection (FPS) in the Cloudera distributed system with 9 nodes	4,649,887 seconds

Table 8: Comparisons of our model's results with the works related the genetic algorithm (GA) in [45-49].

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

Studies	**HA**	**CT**	**Sentiment Classification**	**PNS**	**SD**	**DT**	**Language**
[45]	No	No	No	No	Yes	No	EL
[46]	No	No	Yes	No	Yes	No	EL
[47]	No	No	Yes	No	Yes	Yes	EL
[48]	No	No	Yes	No	Yes	Yes	EL
[49]	No	No	Yes	No	Yes	Yes	EL
Our work	Yes	Yes	Yes	Yes	Yes	Yes	EL

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

Table 9: Comparisons of our model's advantages and disadvantages with the works related to the genetic algorithm (GA) in [45-49].

Researches	Approach	Advantages	Disadvantages
[45]	Genetic algorithms	This study is meant to be a practical guide for practitioners, not, say. a textbook for a Machine Learning course. As a high-level introduction, the tutorial serves this purpose well, and is strongly supplemented by the case studies. The survey is clearly written and enjoyable to read, and, short of hiring Davis himself as a consultant, reading his work is probably the quickest and easiest way to get off the ground for a first real GA application.	No mention
[46]	Bundle block detection using differential evolution and levenberg marquardt neural network	The classification accuracy by the DE with LMNN was 99.1% for the detection of BBB. The proposed results Have shown the DE method can extract more relevant features than the other methods in the literature with highest classification accuracy for the detection of BBB.	No mention
[47]	A hybrid approach based on stocHastic competitive Hopfield neural network and efficient genetic algorithm for frequency assignment problem	In this survey, the authors first propose five optimal strategies to build an efficient genetic algorithm. Then the authors explore three hybridizations between SCHNN and EGA to discover the best hybrid algorithm. The authors believe the comparison can also be helpful for hybridizations between neural networks and other evolutionary algorithms such as the particle swarm optimization algorithm, the artificial bee colony algorithm, etc. In the experiments, the authors' hybrid algorithm obtains better or comparable performance than other algorithms on 5 benchmark problems and 12 large problems randomly generated. Finally, the authors show the authors' hybrid algorithm can obtain good results with a small size population.	No mention
[48]	Balancing of Planar MecHAnisms HAving Imperfect Joints Using Neural Network-Genetic Algorithm (NN-GA) Approach	In this study, dynamic response of mecHAnism HAving revolute joints with clearance is investigated. A four-bar mecHAnism HAving two revolute joints with clearance is considered as a model mecHAnism. A neural network was used to model several chracteristics of joint clearance. Kinematic and dynamic analyses were achieved using continuous contact mode between journal and bearing. A genetic algorithm was also used to determine the appropriate values of design variables for reducing the additional vibration effects due primarily to the joint clearance. The results show the optimum adjusting of suitable design variables gives a certain decrease in sHAking forces and their moments on the mecHAnism frame.	No mention
[49]	Genetic algorithm trajectory plan optimization for EAMA: EAST Articulated Maintenance Arm. Fusion Engineering and Design	This survey presents a trajectory optimization method which aims to pursue the 7-DOF articulated arm a stable movement, which keeps the mounted inspection camera anti-vibration. Based on dynamics analysis, trajectory optimization algorithm adopts multi-order polynomial interpolation in joint space and high order geometry Jacobian transform. The object of optimization algorithm is to suppress end-effector movement vibration by minimizing jerk RMS (root mean square) value. The proposed solution Has such chracteristics which can satisfy kinematic constraints of EAMA's motion and ensure the arm running under the absolute values of velocity, acceleration and jerk boundaries. GA (genetic algorithm) is employed to find global and robust solution for this problem.	No mention

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

Our work	-GOWER-2 coefficient (HA) through the Google search engine with AND operator and OR operator. -We use the Gower-2 Coefficient (HA) and the Genetic Algorithm (GA) with the fitness function (FF) which is the Fitness-proportionate Selection (FPS) to classify one document of the testing data set into either the positive polarity or the negative polarity in both the sequential environment and the distributed system. The advantages and disadvantages of this survey are shown in the Conclusion section.

Table 10: Comparisons of our model's results with the Fitness-proportionate Selection (FPS) in [50-54].

Studies	**HA**	**CT**	**Sentiment Classification**	**PNS**	**SD**	**DT**	**Language**
[50]	No	No	No	No	Yes	No	EL
[51]	No	No	Yes	No	Yes	No	EL
[52]	No	No	Yes	No	Yes	Yes	EL
[53]	No	No	Yes	No	Yes	Yes	EL
[54]	No	No	Yes	No	Yes	Yes	EL
Our work	Yes	Yes	Yes	Yes	Yes	Yes	EL

Table 11: Comparisons of our model's advantages and disadvantages with the Fitness-proportionate Selection (FPS) in [50-54].

Researches	**Approach**	**Advantages**	**Disadvantages**
[50]	Automatic design and manufacture of robotic lifeforms	Few robots are available because these costs must be absorbed through mass production, which is justified only for toys, weapons and industrial systems such as automatic teller machines. Here the authors report the results of a combined computational and experimental approach in which simple electromechanical systems are evolved through simulations from basic building blocks (bars, actuators and artificial neurons); the 'fittest' machines (defined by their locomotive ability) are then fabricated robotically using rapid manufacturing technology. The authors thus achieve autonomy of design and construction using evolution in a 'limited universe' physical simulation coupled to automatic fabrication.	No mention
[51]	Multiobjective optimization using evolutionary algorithms — A comparative case study	In this survey an extensive, quantitative comparison is presented, applying four multiobjective evolutionary algorithms to an extended 0/1 knapsack problem.	No mention
[52]	An empirical comparison of selection methods in evolutionary algorithms	Fitness proportionate selection suffers from scaling problems: a number of techniques to reduce these are illustrated. The sampling errors caused by roulette wheel and tournament selection are demonstrated. The EP selection model is shown to be equivalent to an ES model in one form, and surprisingly similar to fitness proportionate selection in another. Generational models are shown to be remarkably immune to evaluation noise, models that retain parents much less so.	No mention
[53]	Genetic Algorithms, Selection Schemes, and the Varying Effects of Noise	The selection schemes modeled in this paper include proportionate selection, tournament selection, (μ, λ) selection, and linear ranking selection. An allele-wise model for convergence in the presence of noise is developed for the OneMax domain, and then extended to more complex domains where the building blocks are uniformly scaled. These models are shown to accurately predict the convergence rate of GAs for a wide range of noise levels.	No mention
[54]	The enhanced genetic algorithms for the optimization design	As the results of the proven systems show, the hybrid genetic algorithm can determines the better optimum design than the traditional optimization algorithms and genetic algorithm. The interval genetic algorithm and hybrid interval genetic algorithm can avoid calculating system slope in traditional interval analysis and determines the	No mention

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

		optimum interval range of the parameters under allowable corresponding objective error boundary. It is the first time that genetic algorithm has been applied to interval optimization process.	
Our work	-GOWER-2 coefficient (HA) through the Google search engine with AND operator and OR operator. -We use the Gower-2 Coefficient (HA) and the Genetic Algorithm (GA) with the fitness function (FF) which is the Fitness-proportionate Selection (FPS) to classify one document of the testing data set into either the positive polarity or the negative polarity in both the sequential environment and the distributed system. The advantages and disadvantages of this survey are shown in the Conclusion section.		

Table 12: Comparisons of our model with the latest sentiment classification models (or the latest sentiment classification methods) in [55-65]

Studies	**HA**	**CT**	**Sentiment Classification**	**PNS**	**SD**	**DT**	**Language**
[55]	No	No	Yes	NM	Yes	Yes	Yes
[56]	No	No	Yes	NM	Yes	Yes	NM
[57]	No	No	Yes	NM	Yes	Yes	EL
[58]	No	No	Yes	NM	Yes	Yes	NM
[59]	No	No	Yes	No	No	No	EL
[60]	No	No	Yes	No	No	No	EL
Our work	Yes	Yes	Yes	Yes	Yes	Yes	Yes

Table 13: Comparisons of our model's positives and negatives the latest sentiment classification models (or the latest sentiment classification methods) in [55-65]

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

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

		sentiment analysis. The authors' proposal is also largely superior to the best lexicon-based methods as well as to supervised combinations of them. In fact, the proposed approach is the only one to produce the best results in all tested datasets in all scenarios.	t analysis tasks such as scoring movies or products according to their related reviews.
[58]	Rule-Based Machine Learning Algorithms	The proposed approach is tested by experimenting with online books and political reviews and demonstrates the efficiency through Kappa measures, which Have a higher Accuracy of 97.4% and a lower error rate. The weighted average of different Accuracy measures like Precision, Recall, and TP-Rate depicts higher efficiency rate and lower FP-Rate. Comparative experiments on various rule-based machine learning algorithms Have been performed through a ten-fold cross validation training model for sentiment classification.	No mention
[59]	The Combination of Term-Counting Method and EnHAnced Contextual Valence Shifters Method	The authors Have explored different methods of improving the Accuracy of sentiment classification. The sentiment orientation of a document can be positive (+), negative (-), or neutral (0). The authors combine five dictionaries into a new one with 21,137 entries. The new dictionary Has many verbs, adverbs, phrases and idioms the were not in five dictionaries before. The study shows the authors' proposed method based on the combination of Term-Counting method and EnHAnced Contextual Valence Shifters method Has improved the accuracy of sentiment classification. The combined method Has accuracy 68.984% on the testing dataset, and 69.224% on the training dataset. All of these methods are implemented to classify the reviews based on our new dictionary and the Internet Movie Database data set.	
[60]	Naive Bayes Model with N-GRAM Method, Negation HAndling Method, Chi-Square Method and Good-Turing Discounting, etc.	The authors Have explored the Naive Bayes model with N-GRAM method, Negation HAndling method, Chi-Square method and Good-Turing Discounting by selecting different thresholds of Good-Turing Discounting method and different minimum frequencies of Chi-Square method to improve the Accuracy of sentiment classification.	No Mention
Our work	-GOWER-2 coefficient (HA) through the Google search engine with AND operator and OR operator. -We use the Gower-2 Coefficient (HA) and the Genetic Algorithm (GA) with the fitness function (FF) which is the Fitness-proportionate Selection (FPS) to classify one document of the testing data set into either the positive polarity or the negative polarity in both the sequential environment and the distributed system. The positives and negatives of the proposed model are given in the Conclusion section.		

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

APPENDIX OF CODE

ALGORITHM 1: Creating a basis English sentiment dictionary (bESD) in a sequential environment
ALGORITHM 2: implementing the Hadoop map phase of creating a basis english sentiment dictionary (besd) in a distributed environment
ALGORITHM 3: performing the Hadoop reduce phase of creating a basis english sentiment dictionary (besd) in a distributed environment
ALGORITHM 4: encrypting one sentiment lexicon (comprising the content and the valence) to a binary array in the sequential environment
ALGORITHM 5: encoding one sentence in English to a binary array in the sequential system
ALGORITHM 6: encrypting all the positive sentences of the training data set to the bit arrays based on the bit arrays of the sentiment lexicons of the besd in the sequential system
ALGORITHM 7: encoding all the negative sentences of the training data set to the bit arrays based on the bit arrays of the sentiment lexicons of the besd in the sequential system
ALGORITHM 8: transferring one document of the testing data set into the bit arrays of the document in the sequential system
ALGORITHM 9: Genetic Algorithm in the sequential environment.
ALGORITHM 10: clustering one document of the testing data set into either the positive bit array group or the negative bit array group of the training data set using the ga with the FPS in the sequential environment
ALGORITHM 11: clustering the documents of the testing data set into either the positive or the negative in the sequential environment
ALGORITHM 12: implementing the Hadoop map phase of encrypting one sentiment lexicon (comprising the content and the valence) to a binary array in the distributed network environment
ALGORITHM 13: performing the Hadoop reduce phase of encrypting one sentiment lexicon (comprising the content and the valence) to a binary array in the distributed network environment
ALGORITHM 14: performing the Hadoop map phase of encoding one sentence in english to a binary array in the parallel system
ALGORITHM 15: implementing the Hadoop reduce of encoding one sentence in english to a binary array in the parallel system
ALGORITHM 16: encrypting all the positive sentences of the training data set to the bit arrays based on the bit arrays of the sentiment lexicons of the bESD to perform the Hadoop map phase of this stage in the distributed environment, called the positive bit array group
ALGORITHM 17: encrypting all the positive sentences of the training data set to the bit arrays based on the bit arrays of the sentiment lexicons of the bESD to perform the Hadoop reduce phase in the parallel system, called the positive bit array group
ALGORITHM 18: encrypting all the negative sentences of the training data set to negative the bit arrays based on the bit arrays of the sentiment lexicons of the bESD to perform the Hadoop map phase in the distributed environment, called the negative bit array group
ALGORITHM 19: encrypting all the negative sentences of the training data set to the bit arrays based on the bit arrays of the sentiment lexicons of the bESD to perform the Hadoop reduce phase in the parallel system, called the negative bit array group
ALGORITHM 20: performing the Hadoop map phase of transferring one document of the testing data set into the bit arrays of the document in the parallel system
ALGORITHM 21: performing the Hadoop reduce phase of transferring one document of the testing data set into the bit arrays of the document in the parallel system
ALGORITHM 22: performing the Hadoop map phase of clustering one bit array (corresponding to one sentence) of the document into either the positive bit array group or the negative bit array group of the training data set by using the ga with the FPS in the distributed system
ALGORITHM 23: performing the Hadoop reduce phase of clustering one bit array (corresponding to one sentence) of the document into either the positive bit array group or the negative bit array group of the training data set by using the ga with the FPS in the parallel system
ALGORITHM 24: performing the Hadoop map phase of clustering one document of the testing data set into either the positive bit array group or the negative bit array group of the training data set using the ga with the FPS in the distributed environment
ALGORITHM 25: performing the Hadoop reduce phase of clustering one document of the testing data set into either the positive bit array group or the negative bit array group of the training data set using the ga with the FPS in the parallel environment
ALGORITHM 26: performing the Hadoop map phase of clustering the documents of the testing data set into either the positive or the negative in the distributed environment
ALGORITHM 27: performing the Hadoop reduce phase of clustering the documents of the testing data set into either the positive or the negative in the parallel environment

ALGORITHM 1: Creating a basis English sentiment dictionary (bESD) in a sequential environment

Input: the 55,000 English terms; the Google search engine
Output: a basis English sentiment dictionary (bESD)
Begin
Step 1: Set bESD := null;
Step 2: For i = 1; i < 55,000; i++, do repeat:
Step 3: By using eq. (8), eq. (9), and eq. (10) 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 i are identified. The valence and the polarity are calculated by using the HA

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

through the Google search engine with AND operator and OR operator.
Step 4: Add this term into bESD;
Step 5: End Repeat – End Step 2;
Step 6: Return bESD;
End;

ALGORITHM 2: implementing the Hadoop map phase of creating a basis english sentiment dictionary (besd) in a distributed environment

Input: : the 55,000 English terms; the Google search engine
Output: one term which the sentiment score and the polarity are identified.
Begin
Step 1: Each term in the 55,000 terms, do repeat:
Step 2: By using eq. (8), eq. (9), and eq. (10) 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 KC through the Google search engine with AND operator and OR operator.
Step 3: Return this term;
End;

ALGORITHM 3: performing the Hadoop reduce phase of creating a basis english sentiment dictionary (besd) in a distributed environment

Input: one term which the sentiment score and the polarity are identified – The output of the Hadoop Map phase.
Output: a basis English sentiment dictionary (bESD)
Begin
Step 1: Receive this term;
Step 2: Add this term into the basis English sentiment dictionary (bESD);
Step 3: Return bESD;
End;

ALGORITHM 4: encrypting one sentiment lexicon (comprising the content and the valence) to a binary array in the sequential environment

Input: one sentiment lexicon of the bESD
Output: a bit array
Begin
Step 1: Split this term into the letters.
Step 2: Set ABitArray := null;
Step 3: Set Valence := Get a valence of this term based on the bESD;
Step 4: Each letter in the letters, do repeat:
Step 5: Based on the binary code of letters in English in [66-71], we get a bit array of this letter;
Step 6: Add the bit array of this letter into ABitArray;
Step 7: End Repeat – End Step 3;
Step 8: Based on on the transferring a decimal to a binary code in [72-77], we transfer the valence to a bit array;
Step 9: Add this bit array into ABitArray;
Step 10: Return ABitArray;
End;

ALGORITHM 5: encoding one sentence in English to a binary array in the sequential system

Input: one sentence;
Output: a bit array;
Begin
Step 1: Set ABitArrayOfSentence := null;
Step 2: Split this sentence into the meaningful terms (meaningful word or meaningful phrase);
Step 3: Each term in the terms, do repeat:
Step 4: ABitArray := The algorithm 4 to encrypt one sentiment lexicon (comprising the content and the valence) to a binary array in the sequential environment with the input is this term;
Step 5: Add ABitArray into ABitArrayOfSentence;
Step 6: End Repeat – End Step 3;
Step 7: Return ABitArrayOfSentence;
End;

ALGORITHM 6: encrypting all the positive sentences of the training data set to the bit arrays based on the bit arrays of the sentiment lexicons of the besd in the sequential system

Input: all the positive sentences of the training data set
Output: a positive bit array group;
Begin
Step 1: Set APositiveBitArrayGroup := null;
Step 2: Each sentence in the positive sentences, do repeat:
Step 3: ABitArray := the algorithm 5 to encode one sentence in English to a binary array in the sequential system with the input is this sentence;
Step 4: Add ABitArray into APositiveBitArrayGroup;
Step 5: End Repeat – End Step 2;
Step 6: Return APositiveBitArrayGroup;
End;

ALGORITHM 7: encoding all the negative sentences of the training data set to the bit arrays based on the bit arrays of the sentiment lexicons of the besd in the sequential system

Input: all the negative sentences of the training data set
Output: a negative bit array group;
Begin
Step 1: Set ANegativeBitArrayGroup := null;
Step 2: Each sentence in the positive sentences, do repeat:
Step 3: ABitArray := the algorithm 5 to encode one sentence in English to a binary array in the sequential system with the input is this sentence;
Step 4: Add ABitArray into ANegativeBitArrayGroup;
Step 5: End Repeat – End Step 2;
Step 6: Return ANegativeBitArrayGroup;
End;

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

ALGORITHM 8: transferring one document of the testing data set into the bit arrays of the document in the sequential system

Input: one document of the testing data set
Output: the bit arrays of the document;
Begin
Step 1: Set TheBitArraysOfTheDocument := null;
Step 2: Split this document into the sentences;
Step 3: Each sentence in the sentences, do repeat:
Step 4: ABitArray := the algorithm 5 to encode one sentence in English to a binary array in the sequential system with the input is this sentence;
Step 5: Add ABitArray into TheBitArraysOfTheDocument;
Step 6: End Repeat- End Step 3;
Step 7: Return TheBitArraysOfTheDocument;
End;

ALGORITHM 9: Genetic Algorithm in the sequential environment.

Input:
P: the input data set includes the binary bit sequences of the Algorithm 3 (the binary data set table) //initial //population
Output:
P' //improved population and it is the information to build decision tree
Begin
1. Repeat:
2. N = |P|;
3. P' = {};
4. Repeat:
5. i1, i2 = select (P, Fitness);
6. o1, o2 = cross (i1, i2, Fitness);
7. o1 = mutate (o1, Fitness);
8. o2 = mutate (o2, Fitness);
9. P' = P' ∪ {o1, o2};
10. until |P'| = N;
11. P = P';
12. Until termination criteria satisfied; // the best individual (cá thể) in P, according to Fitness (the individual Has the //highest Fitness)
End;

ALGORITHM 10: clustering one document of the testing data set into either the positive bit array group or the negative bit array group of the training data set using the ga with the FPS in the sequential environment

Input: one document of the testing data set; the positive bit array group and the negative bit array group of the training data set;
Output: the sentiments (positive, negative, or neutral)
Begin
Step 1: ABitArrayGroupOfOneDocument := the algorithm 8 to transfer one document of the testing data set into the bit arrays of the document in the sequential system with the input is this document;
Step 2: Set count_positive := 0 and count_negative := 0;
Step 3: Each bit array in ABitArrayGroupOfOneDocument, do repeat:
Step 4: OneResult := the algorithm 9 to cluster one bit array (corresponding to one sentence) of the document into either the positive bit array group or the negative bit array group of the training data set by using the GA with the FPS in the sequential system with the input is this bit array, the positive bit array group and the negative bit array group of the training data set;
Step 5: If OneResult is the positive Then count_positive := count_positive + 1;
Step 6: Else If OneResult is the negative Then count_negative := count_negative + 1;
Step 7: End Repeat – End Step 3;
Step 8: If count_positive is greater than count_negative Then Return positive;
Step 9: Else If count_positive is less than count_negative Then Return negative;
Step 10: Return neutral;
End;

ALGORITHM 11: clustering the documents of the testing data set into either the positive or the negative in the sequential environment

Input: the testing data set and the training data set;
Output: the results of the sentiment classification of the testing data set;
Begin
Step 1: The valences and the polarities of the sentiment lexicons of the bESD are calculated based on a basis English sentiment dictionary (bESD) in a sequential environment (4.1.2);
Step 2: A positive bit array group := encrypt all the positive sentences of the training data set to the bit arrays based on the bit arrays of the sentiment lexicons of the bESD in the algorithm 6 with the input is the positive sentences of the training data set;
Step 3: A negative bit array group := encode all the negative sentences of the training data set to the bit arrays based on the bit arrays of the sentiment lexicons of the bESD in the algorithm 7 with the input is the positive sentences of the training data set;
Step 4: Set TheResultsOfTheTestingDataSet := null;
Step 5: Each document in the documents of the testing data set, do repeat:
Step 6: OneResult := the algorithm 10 to cluster one document of the testing data set into either the positive bit array group or the negative bit array group of the training data set using the GA with the FPS in the sequential environment with the input is this document, the positive bit array group and the negative bit array group;
Step 7: Add OneResult into TheResultsOfTheTestingDataSet;
Step 8: End Repeat – End Step 5;
Step 9: Return TheResultsOfTheTestingDataSet;
End;

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

ALGORITHM 12: implementing the Hadoop map phase of encrypting one sentiment lexicon (comprising the content and the valence) to a binary array in the distributed network environment

Input: one sentiment lexicon of the bESD
Output: a bit array of one letter;
Begin
Step 1: Input this term and the bESD into the Hadoop Map in the Cloudera system;
Step 2: Split this term into the letters.
Step 3: Set Valence := Get a valence of this term based on the bESD;
Step 4: Each letter in the letters, do repeat:
Step 5: Based on the binary code of letters in English in [66-71], we get a bit array of this letter;
Step 6: Return the bit array of this letter; //the output of the Hadoop Map
End;

ALGORITHM 13: performing the Hadoop reduce phase of encrypting one sentiment lexicon (comprising the content and the valence) to a binary array in the distributed network environment

Input: the bit array of this letter; //the output of the Hadoop Map
Output: a bit array of the term - ABitArray;
Begin
Step 1: Receive the bit array of this letter;
Step 2: Add the bit array of this letter into ABitArray;
Step 3: If this term is full Then
Step 4: Based on on the transferring a decimal to a binary code in [72-77], we transfer the valence to a bit array;
Step 5: Add this bit array into ABitArray;
Step 6: End If – End Step 3;
Step 7: Return ABitArray;
End;

ALGORITHM 14: performing the Hadoop map phase of encoding one sentence in english to a binary array in the parallel system

Input: one sentence;
Output: a bit array of one term - ABitArray;
Begin
Step 1: Input this sentence into the Hadoop Map in the Cloudera system;
Step 2: Split this sentence into the meaningful terms (meaningful word or meaningful phrase);
Step 3: Each term in the terms, do repeat:
Step 4: ABitArray := the encrypting one sentiment lexicon (comprising the content and the valence) to a binary array in the distributed network environment in Figure 12 with the input is this term;
Step 5: Return ABitArray;
End;

ALGORITHM 15: implementing the Hadoop reduce of encoding one sentence in english to a binary array in the parallel system

Input: a bit array of one term – AbitArray – the output of the Hadoop Map;
Output: a bit array of the sentence - ABitArrayOfSentence;
Begin
Step 1: Receive AbitArray;
Step 2: Add ABitArray into ABitArrayOfSentence;
Step 3: Return ABitArrayOfSentence;
End;

ALGORITHM 16: encrypting all the positive sentences of the training data set to the bit arrays based on the bit arrays of the sentiment lexicons of the bESD to perform the Hadoop map phase of this stage in the distributed environment, called the positive bit array group

Input: all the positive sentences of the training data set
Output: ABitArray – a bit array of one sentence;
Begin
Step 1: Input all the positive sentences of the training data set into the Hadoop Map in the Cloudera system;
Step 2: Each sentence in the positive sentences, do repeat:
Step 3: ABitArray := the encoding one sentence in english to a binary array in the parallel system in Figure 13 with the input is this sentence;
Step 4: Return ABitArray;
End;

ALGORITHM 17: encrypting all the positive sentences of the training data set to the bit arrays based on the bit arrays of the sentiment lexicons of the bESD to perform the Hadoop reduce phase in the parallel system, called the positive bit array group

Input: ABitArray – a bit array of one sentence;
Output: a positive bit array group - APositiveBitArrayGroup;
Begin
Step 1: Receive ABitArray;
Step 2: Add ABitArray into APositiveBitArrayGroup;
Step 3: Return APositiveBitArrayGroup;
End;

ALGORITHM 18: encrypting all the negative sentences of the training data set to negative the bit arrays based on the bit arrays of the sentiment lexicons of the bESD to perform the Hadoop map phase in the distributed environment, called the negative bit array group

Input: all the negative sentences of the training data set
Output: ABitArray – a bit array of one sentence;
Begin
Step 1: Input all the negative sentences of the training data set into the Hadoop Map in the Cloudera system;
Step 2: Each sentence in the negative sentences, do repeat:

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

Step 3: ABitArray := the encoding one sentence in English to a binary array in the parallel system in Figure 13 with the input is this sentence;
Step 4: Return ABitArray;
End;

ALGORITHM 19: encrypting all the negative sentences of the training data set to the bit arrays based on the bit arrays of the sentiment lexicons of the bESD to perform the Hadoop reduce phase in the parallel system, called the negative bit array group

Input: ABitArray – a bit array of one sentence;
Output: a negative bit array group - ANegativeBitArrayGroup;
Begin
Step 1: Receive ABitArray;
Step 2: Add ABitArray into ANegativeBitArrayGroup;
Step 3: Return ANegativeBitArrayGroup;
End;

ALGORITHM 20: performing the Hadoop map phase of transferring one document of the testing data set into the bit arrays of the document in the parallel system

Input: one document of the testing data set
Output: ABitArray - one bit array of one sentence of the document – the output of the Hadoop Map;
Begin
Step 1: Input one document of the testing data set 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: ABitArray := the encoding one sentence in English to a binary array in the parallel system in Figure 13 with the input is this sentence;
Step 5: Return ABitArray;
End;

ALGORITHM 21: performing the Hadoop reduce phase of transferring one document of the testing data set into the bit arrays of the document in the parallel system

Input: ABitArray - one bit array of one sentence of the document – the output of the Hadoop Map;
Output: the bit arrays of the document - TheBitArraysOfTheDocument;
Begin
Step 1:Receive ABitArray;
Step 2: Add ABitArray into TheBitArraysOfTheDocument;
Step 3: Return TheBitArraysOfTheDocument;
End;

ALGORITHM 22: performing the Hadoop map phase of clustering one bit array (corresponding to one sentence) of the document into either the positive bit array group or the negative bit array group of the training data set by using the ga with the FPS in the distributed system

Input: one bit array (corresponding to one sentence) of the document; the positive bit array group and the negative bit array group the training data set;
Output: the bit array clustered into either the positive bit array group or the negative bit array group of the training data set.
Begin
Step 1: Input the bit array (corresponding to one sentence) of the document; the positive bit array group and the negative bit array group the training data set into the Hadoop Map in the Cloudera system;
Step 2: randomly initialize population(t)
Step 3: determine fitness of population(t)
Step 4: repeat
Step 5: select parents from population(t)
Step 6: perform crossover on parents creating population(t+1)
Step 7: perform mutation of population(t+1)
Step 8: determine fitness of population(t+1)
Step 9: until best individual is good enough
Step 10: Return this bit array clustered into either the positive bit array group or the negative bit array group of the training data set.
End;

ALGORITHM 23: performing the Hadoop reduce phase of clustering one bit array (corresponding to one sentence) of the document into either the positive bit array group or the negative bit array group of the training data set by using the ga with the FPS in the parallel system

Input: the bit array clustered into either the positive bit array group or the negative bit array group of the training data set.
Output: the sentiments (positive, negative, or neutral)
Begin
Step 1: Receive the bit array clustered into either the positive bit array group or the negative bit array group of the training data set;
Step 2: If this bit array clustered into the positive bit array group Then Return positive;
Step 3: If this bit array clustered into the negative bit array group Then Return negative;
Step 4: Return neutral;
End;

ALGORITHM 24: performing the Hadoop map phase of clustering one document of the testing data set into either the positive bit array group or the negative bit array group of the training data set using the ga with the FPS in the distributed environment

Input: one document of the testing data set; the positive bit array group and the negative bit array group of the training data set
Output: OneResult – the sentiment classification of one bit array of the document – the output of the Hadoop Map;
Begin
Step 1: Input the document of the testing data set; the positive bit array group and the negative bit array group of the training data set into the Hadoop Map in the Cloudera system;
Step 2: ABitArrayGroupOfOneDocument := the transferring one document of the testing data set into

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

the bit arrays of the document in the parallel system in Figure 16 with the input is this document;

Step 3: Each bit array in ABitArrayGroupOfOneDocument, do repeat:

Step 4: OneResult := the algorithm 9 to cluster one bit array (corresponding to one sentence) of the document into either the positive bit array group or the negative bit array group of the training data set by using the GA with the FPS in the sequential system with the input is this bit array, the positive bit array group and the negative bit array group of the training data set;

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

End;

ALGORITHM 25: performing the Hadoop reduce phase of clustering one document of the testing data set into either the positive bit array group or the negative bit array group of the training data set using the ga with the FPS in the parallel environment

Input: OneResult – the sentiment classification of one bit array of the document – the output of the Hadoop Map;

Output: the sentiments (positive, negative, or neutral) of the document

Begin

Step 1: Receive OneResult – the sentiment classification of one bit array of the document;

Step 2: If OneResult is the positive Then count_positive := count_positive + 1;

Step 3: Else If OneResult is the negative Then count_negative := count_negative + 1;

Step 4: If count_positive is greater than count_negative Then Return positive;

Step 5: Else If count_positive is less than count_negative Then Return negative;

Step 6: Return neutral;

End;

ALGORITHM 26: performing the Hadoop map phase of clustering the documents of the testing data set into either the positive or the negative in the distributed environment

Input: the testing data set and the training data set;

Output: OneResult - the result of the sentiment classification of one document the testing data set – the output of the Hadoop Map ;

Begin

Step 1: The valences and the polarities of the sentiment lexicons of the bESD are calculated based on a basis English sentiment dictionary (bESD) in a distributed system (4.1.3);

Step 2: A positive bit array group := the encrypting all the positive sentences of the training data set to the bit arrays based on the bit arrays of the sentiment lexicons of the bESD in the distributed environment, called the positive bit array group in Figure 14 with the input is the positive sentences of the training data set;

Step 3: A negative bit array group := the encrypting all the negative sentences of the training data set to the bit arrays based on the bit arrays of the sentiment lexicons of the bESD in the distributed environment, called the negative bit array group in Figure 15 with the input is the positive sentences of the training data set;

Step 4: Input the documents of the testing data set, the positive bit array group and the negative bit array group into the Hadoop Map in the Cloudera system;

Step 5: Each document in the documents of the testing data set, do repeat:

Step 6: OneResult := the clustering one document of the testing data set into either the positive bit array group or the negative bit array group of the training data set using the GA with the FPS in the distributed environment in Figure 19 with the input is this document, the positive bit array group and the negative bit array group;

Step 7: Return OneResult;

End;

ALGORITHM 27: performing the Hadoop reduce phase of clustering the documents of the testing data set into either the positive or the negative in the parallel environment

Input: OneResult - the result of the sentiment classification of one document the testing data set;

Output: the results of the sentiment classification of the testing data set;

Begin

Step 1: Receive OneResult - the result of the sentiment classification of one document the testing data set;

Step 2: Add OneResult into TheResultsOfTheTestingDataSet;

Step 3: Return TheResultsOfTheTestingDataSet;

End;