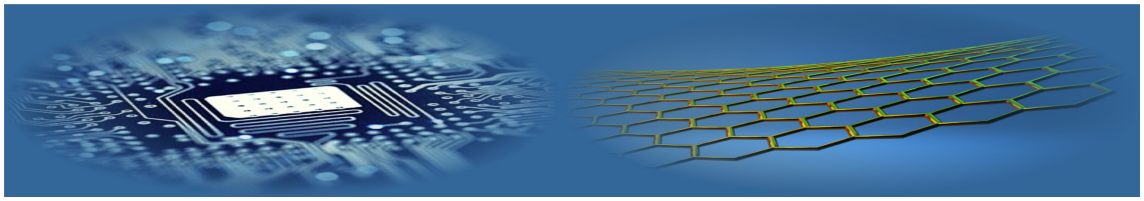

K-Medoids algorithm used for english sentiment classification in a distributed system

Vo Ngoc Phu[1]*, Vo Thi Ngoc Tran[2]

[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

**Corresponding author e-mail: vongocphu03hca@gmail.com; vongocphu@ntt.edu.vn*

Received 30 January 2018, www.cmnt.lv

Abstract

Sentiment classification is significant in everyday life, such as in political activities, commodity production, and commercial activities. Finding a fast, highly accurate solution to classify emotion has been a challenge for scientists. In this research, we have proposed a new model for Big Data sentiment classification in the parallel network environment – a Cloudera system with Hadoop Map (M) and Hadoop Reduce (R). Our new model has used a K-Medoids Algorithm (PAM) with multi-dimensional vector and 2,000,000 English documents of our English training data set for English document-level sentiment classification. Our new model can classify sentiment of millions of English documents based on many English documents in the parallel network environment. However, we tested our new model on our testing data set (including 1,000,000 English reviews, 500,000 positive and 500,000 negative) and achieved 85.98% accuracy.

Key words

computer and information technologies,
natural and engineering sciences,
operation research and decision making,
mathematical and computer modelling

1 Introduction

Sentiment classification is significant in everyday life, such as in political activities, commodity production, and commercial activities. Finding a fast, highly accurate solution to classify emotion has been a challenge for scientists.

Clustering data is to process a set of objects into classes of similar objects. One cluster is a set of data objects which are similar to each other and are not similar to objects in other

clusters. A number of data clusters can be clustered, which can be identified following experience or can be automatically identified as part of clustering method.

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

- Assuming that each English sentence has m English words (or English phrases).
- Assuming that the maximum number of one English sentence is m_max; it means that m is less than m_max or m is equal to m_max.
- Assuming that each English document has n English sentences.
- Assuming that the maximum number of one English document is n_max; it means that n is less than n_max or n is equal to n_max.
- Each English sentence is transferred into one vector (one-dimensional). Thus, the length of the vector is m. If m is less than m_max then each element of the vector from m to m_max-1 is 0 (zero).
- Each English document is transferred into one multi-dimensional vector. Therefore, the multi-dimensional vector has n rows and m columns. If n is less than n_max then each element of the multi-dimensional vector from n to n_max-1 is 0 (zero vector).
- All English documents (2,000,000) of the English training data set are transferred into the multi-dimensional vectors. The 1,000,000 positive English documents of the English training data set are transferred into the 1,000,000 positive multi-dimensional vectors, the positive vector group. The 1,000,000 negative English documents of the English training data set are transferred into the 1,000,000 negative multi-dimensional vectors, the negative vector group.
- All English documents of the English testing data set are transferred into the multi-dimensional vectors.
- One multi-dimensional vector (corresponding to one English document in the English testing data set) is the positive polarity if the vector is clustered into the positive vector group. One multi-dimensional vector (corresponding to one English document in the English testing data set) is the negative polarity if the vector is clustered into the negative vector group. One multi-dimensional vector (corresponding to one English document in the English testing data set) is the neutral polarity if the vector is not clustered into either the positive vector group or the negative vector group.

In this study, we propose a new model by using the K-Medoids Algorithm (PAM) to classify emotions (positive, negative, neutral) of English documents in the parallel system. A study of semantic classification (emotional analysis) using the PAM does not currently exist in the world.

According to the K-Medoids works in the world and in [7-21], there is not any research related to the K-Medoids that is similar to our work. With the research related to the K-Medoids in the parallel system (or the K-Medoids in the distributed system) in the world, there are not any K-Medoids-related studies in the parallel system that are similar to our work.

According to the latest research on sentiment classification in the world and in [22- 39], there is no semantic classification work similar to our model.

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. This will result in many discoveries in scientific research, hence the motivation for this study.

The novelty of the proposed approach is that a K-Medoids Algorithm (PAM) from the data mining field is applied to sentiment analysis. A K-Medoids Algorithm (PAM) is applied to classify semantics of English documents based on many sentences. This algorithm can also be applied to identify the emotions of millions of documents. These above principles are proposed to classify the semantics of a document, and data mining is used in natural language processing. This survey can be applied to other parallel network systems. Hadoop Map (M) and Hadoop Reduce (R) are used in the proposed model.

Our model has many significant applications to many areas of research as well as commercial applications:

- The algorithm of data mining is applicable to semantic analysis of natural language processing.
- This study also proves that different fields of scientific research can be related in many ways.

- Millions of English documents are successfully processed for emotional analysis.
- Many studies and commercial applications can use the results of this survey.
- The semantic classification is implemented in the parallel network environment.
- The principles are proposed in the research.
- The opinion classification of English documents is performed on English sentences.
- The proposed model can be applied to other languages easily.
- The Cloudera distributed environment is used in this study.
- The proposed work can be applied to other distributed systems.
- This survey uses Hadoop Map (M) and Hadoop Reduce (R).
- Our proposed model can be applied to many different parallel network environments such as a Cloudera system
- This study can be applied to many different distributed functions such as Hadoop Map (M) and Hadoop Reduce (R).
- The PAM – related algorithms are proposed in this study.

This study contains 6 sections. Section 1 introduces the study; Section 2 discusses the related works about the K-Medoids Algorithm (PAM), 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

In this section, we describe summaries of many studies related to a K-Medoids Algorithm (PAM), vector space model (VSM), Hadoop, Cloudera, etc.

There are the works related to vector space modelling in [1, 2, 3]. First, we transfer all English sentences into many vectors, which are used in the VSM algorithm. In this research [1], the authors will examine the vector space model, an information retrieval technique, and its variation. The rapid growth of the Internet and the abundance of documents and different forms of information available underscores the need for good information retrieval technique. The vector space model is an algebraic model used for information retrieval. It represents natural language documents in a formal manner using of vectors in a multi-dimensional space and allows decisions to be made as to which documents are similar to each other and to the queries fired. This research attempts to examine the vector space model, an information retrieval technique that is widely used today. It also explains the existing variations of VSM and proposes the new variation that should be considered. In text classification tasks, one of the main problems [2] is to choose which features give the best results. Various features can be used like words, n-grams, syntactic n-grams of various types (POS tags, dependency relations, mixed, etc.); or a combination of these features can be considered. Also, algorithms for dimensionality reduction of these sets of features can be applied, such as latent Dirichlet allocation (LDA). In this research, the authors consider multi-label text classification tasks and apply various feature sets. The authors consider a subset of multi-labeled files of the Reuters-21578 corpus. The authors use traditional TF-IDF values of the features and tried both considering and ignoring stop words. The authors also tried several combinations of features, like bi-grams and uni-grams. The authors also experimented with adding LDA results into vector space models as new features. These last experiments obtained the best results. KNN and SVM [3] are two machine learning approaches to text categorization (TC) based on the vector space model. In this model, borrowed from information retrieval, documents are represented as a vector where each component is associated with a particular word from the vocabulary. Traditionally, each component value is assigned using the information retrieval TFIDF measure. While this weighting method seems very appropriate for IR, it is not clear that it is the best choice for TC problems. Actually, this weighting method does not leverage the information implicitly contained in the categorization task to represent documents. In this research, the authors introduce a new weighting method based on statistical estimation of the importance of a word for a specific categorization problem. This method also has the benefit to make feature selection implicit, since useless features of the categorization problem considered get a very small weight. Extensive experiments reported in the research show that this new weighting method improves significantly the classification accuracy as measured on many

categorization tasks.

Many research projects related to implementing algorithms, applications, studies in parallel network environment in [4, 5, 6]. In [4, 5], Hadoop is an Apache-based framework used to handle large data sets on clusters consisting of multiple computers, using the Map and Reduce programming model. The two main projects of the Hadoop are Hadoop Distributed File System (HDFS) and Hadoop M/R (Hadoop Map /Reduce). Hadoop M/R allows engineers to program for writing applications for parallel processing of large data sets on clusters consisting of multiple computers. A M/R task has two main components: (1) Map and (2) Reduce. This framework splits inputting data into chunks which multiple Map tasks can handle with a separate data partition in parallel. The outputs of the map tasks are gathered and processed by the Reduce task ordered. The input and output of each M/R task are stored in HDFS because the Map tasks and the Reduce tasks perform on the pair (key, value), and formatted input and output formats will be the pair (key, value). Cloudera [6], the global provider of the fastest, easiest, and most secure data management and analytics platform built on Apache™ Hadoop® and the latest open source technologies, announced today that it will submit proposals for Impala and Kudu to join the Apache Software Foundation (ASF). By donating its leading analytic database and columnar storage projects to the ASF, Cloudera aims to accelerate the growth and diversity of their respective developer communities. Cloudera delivers the modern data management and analytics platform built on Apache Hadoop and the latest open source technologies. The world's leading organizations trust Cloudera to help solve their most challenging business problems with Cloudera Enterprise, the fastest, easiest and most secure data platform available to the modern world. Cloudera's customers efficiently capture, store, process, and analyze vast amounts of data, empowering them to use advanced analytics to drive business decisions quickly, flexibly, and at lower cost than has been possible before. To ensure Cloudera's customers are successful, it offers comprehensive support, training and professional services.

There are the works related to the K-Medoids Algorithm (PAM) in [7- 21].

The latest researches of the sentiment classification are [22-39]. In the research [22], 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 [23] discusses an approach where an exposed stream of tweets from the Twitter micro blogging site are pre-processed 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.

3 Data set

In Figure 1 below, the English training data set includes 2,000,000 English documents in the movie field, which contains 1,000,000 positive English documents and 1,000,000 negative English documents. All English sentences in our English training data set are automatically extracted from English Facebook, English websites and social networks; then we labelled positive and negative for them.

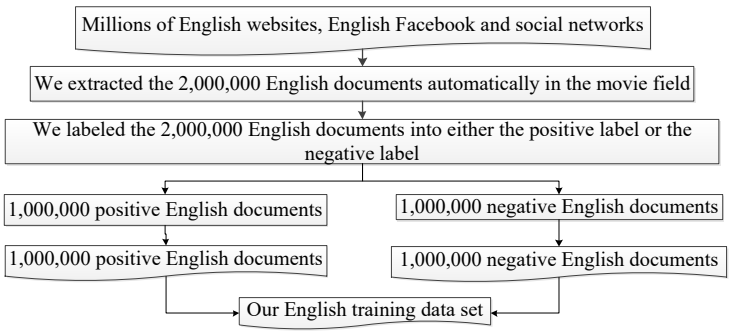

FIGURE 1 Our English training data set

In Figure 2 below, the English testing data set includes 1,000,000 English documents in the movie field, which contains 500,000 positive English documents and 500,000 negative English documents. All English sentences in our English training data set are automatically extracted from English Facebook, English websites and social networks; then we labelled positive and negative for them.

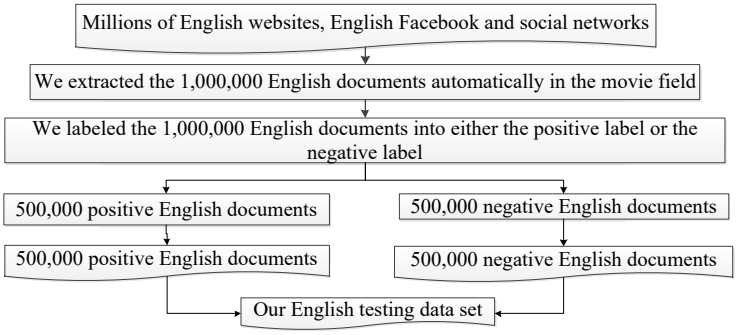

FIGURE 2 Our English testing data set

4 Methodology

This section has two parts: semantic classification for the 1,000,000 English documents of the testing in the sequential environment is presented in the first part. In the second part, sentiment classification for the 1,000,000 English reviews of the testing in the parallel network environment is displayed.

With the English training data set, there are two groups. The first group includes 1,000,000 positive documents and the second group is 1,000,000 negative documents. The first group is called the positive cluster. The second group is called the negative cluster. All documents in both the first group and the second group go through the segmentation of words and stop-words removal; then, they are transferred into the multi-dimensional vectors (vector representation). The 1,000,000 positive documents of the positive cluster are transferred into the 1,000,000 positive multi-dimensional vectors which are called the positive vector group (or positive vector cluster). The 1,000,000 negative documents of the negative cluster are transferred into the 1,000,000 negative multi-dimensional vectors which are called the negative vector group (or negative vector cluster). Therefore, the training data set includes the positive vector group (or positive vector cluster) and the negative vector group (or negative vector cluster).

In [1, 2, 3], the VSM is an algebraic model used for information retrieval. It represents a natural language document in a formal manner by the use of vectors in a multidimensional space. The vector space model (VSM) is a way of representing documents through the words they contain. The concepts behind vector space modelling are that by placing terms, documents, and queries in a term-document space, it is possible to compute the similarities between queries and the terms or documents and allow the results of the computation to be ranked according to the similarity measure between them. The VSM allows decisions to be made about which documents are similar to each other and to queries.

We have transferred all English sentences into one-dimensional vectors similar to VSM [1, 2, 3].

4.1 A K-MEDOIDS ALGORITHM (PAM) IN A SEQUENTIAL ENVIRONMENT

In Figure 3, in the sequential environment, the 1,000,000 documents of the English testing data set are transferred to the 1,000,000 multi-dimensional vectors: each document of the testing data set is transferred to each multi-dimensional vector (each sentence of one document in the testing data set is transferred to the one-dimensional vector similar to VSM [1, 2, 3]). The 1,000,000 positive documents in the training data set are transferred to the 1,000,000 positive multi-dimensional vectors, called the positive vector group in the sequential environment: each document of the 1,000,000 positive documents is transferred to each multi-dimensional vector (each sentence, of one document in the 1,000,000

positive documents, is transferred to the one-dimensional vector similar to VSM [1, 2, 3] in the sequential environment). The 1,000,000 negative documents in the training data set are transferred to the 1,000,000 negative multi-dimensional vectors, called the negative vector group in the sequential environment: each document of the 1,000,000 negative documents is transferred to each multi-dimensional vector (each sentence, of one English document in the 1,000,000 negative documents, is transferred to the one-dimensional vector similar to VSM [1, 2, 3] in the sequential environment).

This part is done as follows in Figure 4

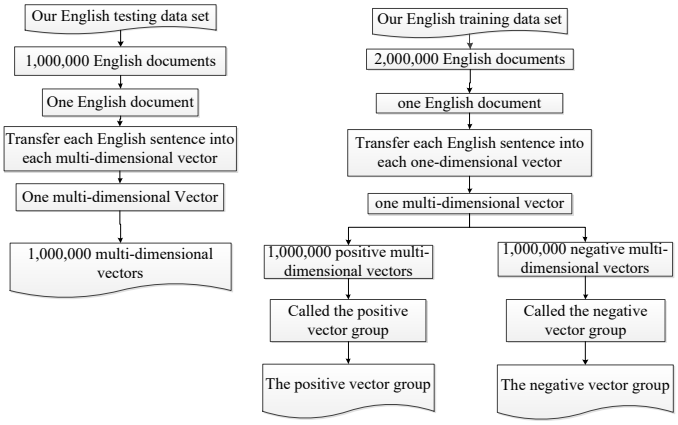

FIGURE 3 Overview of transferring all English documents into the multi-dimensional vectors

below: In the sequential environment, the PAM is implemented to cluster one multi-dimensional vector (called A) of the English testing data set to the positive vector group or the negative vector group. The document (corresponding to A) is the positive polarity if A is clustered to the positive vector group. The document (corresponding to A) is the negative polarity if A is clustered to the negative vector group. The document (corresponding to A) is the neutral polarity if A is not clustered to both the positive vector group and the negative vector group.

We built many algorithms to perform

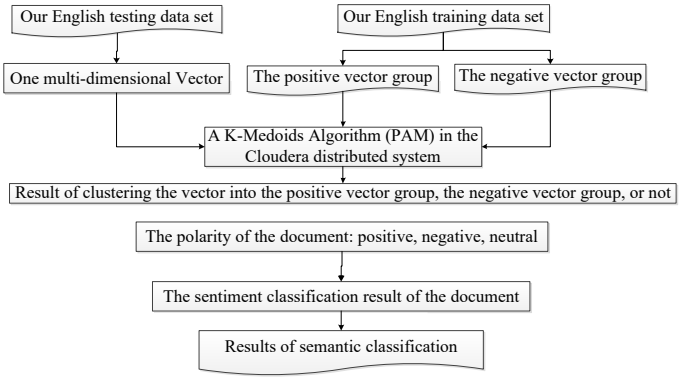

FIGURE 4 A K-Medoids Algorithm (PAM) in the Sequential Environment

the PAM in the sequential environment. We built the algorithm 1 to transfer one English document into one multi-dimensional vector. Each document is split into many sentences. Each sentence in each document is transferred to one one-dimensional vector based on VSM [1, 2, 3] in the sequential environment. We insert all the one-dimensional vectors of the sentences into one multi-dimensional vector of one document

The main ideas of the algorithm 1 are as follows:

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

We built the algorithm 2 to create the positive vector group. Each document in the 1,000,000 positive documents in the English training data set is split into many sentences. Each sentence of the document is transferred to one one-dimensional vector based on VSM [1, 2, 3] in the sequential environment. We insert all the one-dimensional vectors of the sentences of the document into one one-dimensional vector of the document. Then, the 1,000,000 positive documents in the English training data are transferred to the 1,000,000 positive multi-dimensional vectors.

The main ideas of the algorithm 2 are as follows:

- Input: the 1,000,000 positive English documents of the English training data set
- Output: the positive vector group PositiveVectorGroup
- Step 1: Each document in the 1,000,000 positive document of the training data set, do repeat:
- Step 2: OneMultiDimensionalVector := Call Algorithm 1 with the positive English document in the English training data set;
- Step 3: Add OneMultiDimensionalVector into PositiveVectorGroup;
- Step 4: End Repeat – End Step 1
- Step 5: Return PositiveVectorGroup;

We built the algorithm 3 to create the negative vector group. Each document in the 1,000,000 negative documents in the English training data set is split into many sentences. Each sentence of the document is transferred to one one-dimensional vector based on VSM [1, 2, 3] in the sequential environment. We insert all the one-dimensional vectors of the sentences of the document into one one-dimensional vector of the document. Then, the 1,000,000 negative documents in the English training data set are transferred to the 1,000,000 negative multi-dimensional vectors.

The main ideas of the algorithm 3 are as follows:

- Input: the 1,000,000 negative English documents of the English training data set.
- Output: the negative vector group PositiveVectorGroup
- Step 1:Each document in the 1,000,000 negative document of the training data set, do repeat:
- Step 2: OneMultiDimensionalVector := Call Algorithm 1 with the negative English document in the English training data set;
- Step 3: Add OneMultiDimensionalVector into NegativeVectorGroup;
- Step 4: End Repeat – End Step 1
- Step 5: Return Negative VectorGroup;

We built the algorithm 4 to cluster one multi-dimensional vector (corresponding to one document of the English testing data set) into the positive vector group PositiveVectorGroup, the negative vector group NegativeVectorGroup, or not.

The main ideas of the algorithm 4 are as follows:

- Input: one multi-dimensional vector A (corresponding to one English document of the English testing data set), the positive vector group PositiveVectorGroup, the negative vector group NegativeVectorGroup;
- Output: positive, negative, neutral;
- Step 1: Implement the K-Medoids Algorithm (PAM) based on the K-Medoids Algorithm in [7-21] with input is one multi-dimensional vector (corresponding to one English document of the English testing data set), the positive vector group PositiveVectorGroup, the negative vector group NegativeVectorGroup;
- Step 2: With the results of Step 1, If the vector is clustered into the positive vector group Then Return positive;
- Step 3: Else If the vector is clustered into the negative vector group Then Return negative; End If – End Step 2
- Step 4: Return neutral;

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

4.2 A K-MEDOIDS ALGORITHM (PAM) IN A PARALLEL NETWORK ENVIRONMENT

In Figure 5, all documents of both the English testing data set and the English training data set are transferred into all the multi-dimensional vectors in the Cloudera parallel network environment. With the 2,000,000 documents of the English training data set, we transferred them into the 2,000,000 multi-dimensional vectors by using Hadoop Map (M) /Reduce (R) in the Cloudera parallel network environment with the purpose of shortening the execution time of this task. The 1,000,000 positive documents of the English training data set are transferred into the 1,000,000 positive vectors in the Cloudera parallel system and are called the positive vector group. The 1,000,000 negative documents of the English training data set are transferred into the 1,000,000 negative vectors in the Cloudera parallel system and are called the negative vector group. Besides, the 1,000,000 documents of the English testing data set are transferred to the 1,000,000 multi-dimensional vectors by using Hadoop Map (M) /Reduce (R) in the Cloudera parallel network environment with the purpose of shortening the execution time of this task.

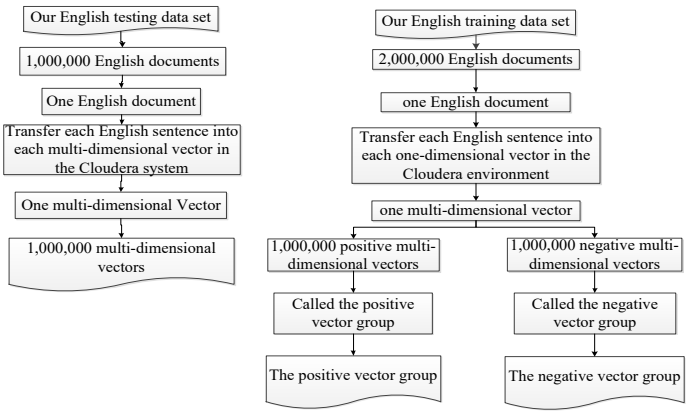

FIGURE 5 Overview of transferring all English documents into the multi-dimensional vectors in the Cloudera distributed system

This part is done as follows in Figure 6, below. In the Cloudera distributed network environment, by using the PAM, one multi-dimensional vector (called A) of one document in the English testing data set is clustered into the positive vector group or the negative vector group. The document (corresponding to A) is the positive polarity if A is clustered into the positive vector group. The document (corresponding to A) is the negative polarity if A is clustered into the negative vector group. The document (corresponding to A) is the neutral polarity if A is not clustered into both the positive vector group and the negative vector group.

An overview of transferring each English

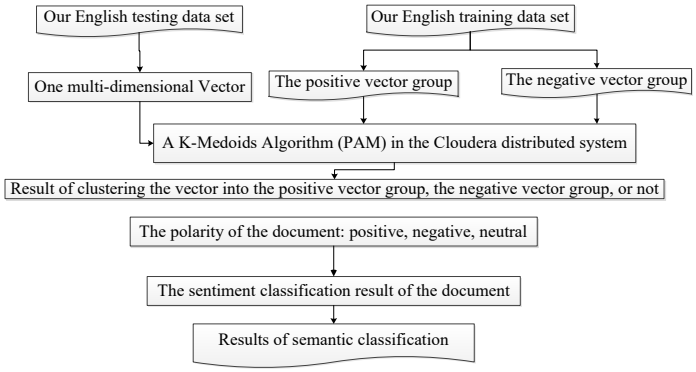

FIGURE 6 A K-Medoids Algorithm (PAM) in the Parallel Network Environment

sentence into one vector in the Cloudera network environment is follows in Figure 7.

In Figure 7, transferring each English document into one vector in the Cloudera network environment includes two phases: Map (M) phases and Reduce (R) phases. Input of the Map phase is one document and Output of the Map phase is many components of a vector which corresponds to the document. One document, input into Hadoop Map (M), is split into many sentences. Each sentence in the English document is transferred into one one-dimensional vector based on VSM [1, 2, 3]. This is repeated for all the sentences of the document until all the sentences are transferred into all the one-dimensional vectors of the document. After finishing to transfer each sentence of the document into one one-dimensional vector, the Map phase of Cloudera automatically transfers the one-dimensional vector into the Reduce phase.

In Figure 7, the input of the Reduce phase is the output of the Map phase, and this input comprises many components (many one-dimensional vector) of a multi-dimensional vector. The output of the Reduce phase is a multi-dimensional vector which corresponds to the document. In the Reduce phase of Cloudera, those components of the vector are built into one multi-dimensional vector.

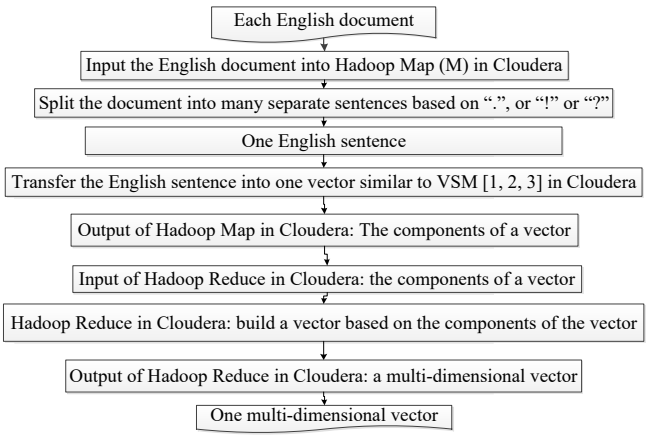

FIGURE 7 Overview of transforming each English sentence into one vector in Cloudera

The 1,000,000 documents of the English testing data set are transferred into the 1,000,000 multi-dimensional vectors based on Figure 7.

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

4.2.1 Hadoop map (M)

This phase is done as illustrated in Figure 8, below. The K-Medoids Algorithm (PAM) in Cloudera is based on the K-Medoids Algorithm in [7- 18]. The input is one multi-dimensional vector in the English testing data set, the positive

vector group and the negative vector group of the English training data set. The output of the K-NN is the clustering results of the multi-dimensional vector into the positive vector group or the negative vector group, or not.

The main ideas of the PAM (k: the number of clusters; D: the training data set) are as follows:

1. Arbitrarily choose k objects in D as the initial representative objects or seeds;
2. repeat:
3. assign each remaining object to the cluster with the nearest representative object;
4. randomly select a non-representative object, Orandomm;
5. compute the total cost, S, of swapping representative object, Oj, with Orandom;
6. if S < 0 Then swap Oj with Orandom to form the new set of k representative objects;
7. until no change;

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

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

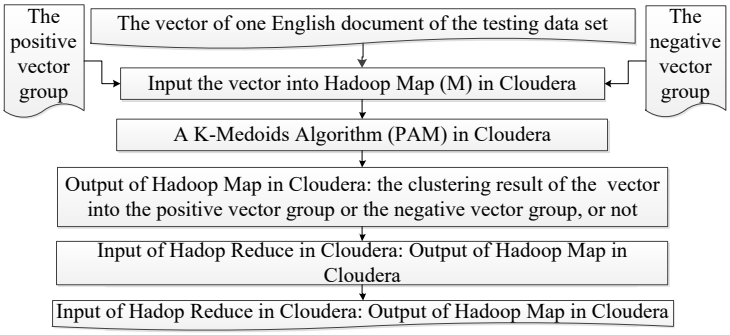

FIGURE 8 Overview of the PAM in Hadoop Map (M) in Cloudera

4.2.2 Hadoop reduce (R)

This phase is done as illustrated below in Figure 9. After receiving the clustering result of Hadoop Map, Hadoop Reduce labels the semantics polarity for the multi-dimensional vector which is classified. Then, the output of Hadoop Reduce will return the semantics polarity of one document (corresponding to the multi-dimensional vector) in the English testing data set. One document is the positive polarity if the multi-dimensional vector is clustered into the positive vector group. One document is the negative polarity if the multi-dimensional vector is clustered into the negative vector group. One document is the neutral polarity if the multi-dimensional vector is not clustered into both the positive vector group and the negative vector group.

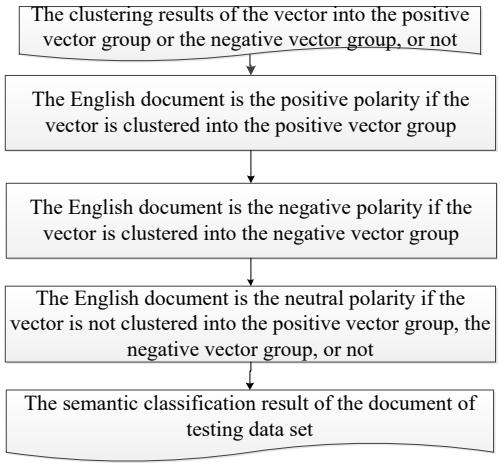

FIGURE 9 Overview of Hadoop Reduce (R) in Cloudera

5 Experiment

We have measured an Accuracy (A) to calculate the accuracy of the results of emotion classification. A Java programming language is used for programming to save data sets, implementing our proposed model to classify the 1,000,000 documents of the testing data set. To implement the proposed model, we have already used Java programming language to save the English training data set and the English testing data set and to save the results of emotion classification.

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

The results of the 1,000,000 documents of the English testing data set to test are presented in Table 1 below.

The accuracy of the emotional classification 1,000,000 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 1,000,000 English documents in testing data set are displayed

6 Conclusion

Although our new model has been tested on our English data set, it can be applied to many other languages. In this paper, our model has been tested on the 1,000,000 English documents of the testing data set in which the data sets are small. However, our model can be applied to larger data sets with millions of English documents in the shortest time.

In this work, we have proposed a new model to classify sentiment of English documents using the K-Medoids Algorithm (PAM) with Hadoop Map (M) /Reduce (R) in the Cloudera parallel network environment. With our proposed new model, we have achieved 85.98% accuracy of the testing data set. Until now, not many studies have shown that the clustering methods can be used to classify data. Our research shows that clustering methods are used to classify data and, in particular, can be used to classify emotion in text.

In Table 3, the average time of the semantic classification of the K-Medoids Algorithm (PAM) in the sequential environment is 5,267,195 seconds /1,000,000 English documents and it is greater than the average time of the emotion classification of the PAM in the Cloudera parallel network environment – 3 nodes which is 1,688,939 seconds /1,000,000 English documents. The average time of the emotion classification of the PAM in the Cloudera parallel network environment – 9 nodes, which is 584,647 seconds /1,000,000 English documents, is the shortest time. Besides, The average time of the emotion classification of the PAM in the Cloudera parallel network environment – 6 nodes is 868,384 seconds /1,000,000 English documents

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

The proposed model has many advantages and disadvantages. Its positives are as follows: It uses the K-Medoids Algorithm (PAM) to classify semantics of English documents based on sentences. The proposed model can process millions of documents in the shortest time. This study can be performed in distributed systems. It can be applied to other languages. Its negatives are as follows: It has a low rate of accuracy. It costs too much and takes too much time to implement this proposed model.

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

In Table 4 and Table 5 below, we compare our model's results with the studies in [1, 2, 3].

In Table 6 and Table 7 below, we compare our model's results with the works related to the K-Medoids Algorithm (PAM) in [7 - 21].

In Table 8 and Table 9 below, we compare our model's results with the latest research on sentiment classification (or sentiment analysis or opinion mining) in [22 - 27].

References

[1] Singh Vaibhav Kant, Singh Vinay Kumar 2015 Vector Space Model: An Information Retrieval System *Int. J. Adv. Engg. Res. Studies/IV/II/Jan.-March, 2015* 141-3

[2] Carrera-Trejo V, Sidorov G, Miranda-Jiménez S, Moreno Ibarra M, Martínez R C 2015 Latent Dirichlet Allocation complement in the vector space model for Multi-Label Text Classification *International Journal of Combinatorial Optimization Problems and Informatics* **6**(1) 7-19

[3] Soucy P, Mineau G W 2015 Beyond TFIDF Weighting for Text Categorization in the Vector Space Model *Proceedings of the 19th International Joint Conference on Artificial Intelligence, USA* 1130-5

[4] *Hadoop* 2017 http://hadoop.apache.org

[5] *Apache* 2017 http://apache.org

[6] *Cloudera* 2017 http://www.cloudera.com

[7] Park Hae-Sang, Jun Chi-Hyuck 2009 A simple and fast algorithm for K-medoids clustering *Expert Systems with Applications* **36**(2): 2 3336-41 https://doi.org/10.1016/j.eswa.2008.01.039

[8] Krishnapuram R, Joshi A, Yi Liyu 1999 A fuzzy relative of the k-medoids algorithm with application to web document and snippet clustering. *Fuzzy Systems Conference Proceedings, 1999. FUZZ-IEEE '99. 1999 IEEE International* DOI: 10.1109/FUZZY.1999.790086, Seoul, South Korea, South Korea

[9] Velmurugan T, Santhanam T 2010 Computational Complexity between K-Means and K-Medoids Clustering Algorithms for Normal and Uniform Distributions of Data Points *Journal of Computer Science* **6**(3) 363-8 ISSN 1549-3636

[10] Zhang Q, Couloigner I 2005 A New and Efficient K-Medoid Algorithm for Spatial Clustering *International Conference on Computational Science and Its Applications (ICCSA 2005): Computational Science and Its Applications – ICCSA 2005* 181-9

[11] Sheng W, Liu X 2006 A genetic k-medoids clustering algorithm *Journal of Heuristics* **12**(6) 447–66

[12] Cao D, Yang B 2010 An improved k-medoids clustering algorithm *The 2nd International Conference on Computer and Automation Engineering (ICCAE)* DOI: 10.1109/ICCAE.2010.5452085, Singapore

[13] Reynolds A P, Richards G, de la Iglesia B, Rayward-Smith V J 2006 Clustering Rules: A Comparison of Partitioning and Hierarchical Clustering Algorithms *Journal of Mathematical Modelling and Algorithms* **5**(4) 475–504

[14] Vijaya P A, Narasimha Murty M, Subramanian D K 2004 Leaders–Subleaders: An efficient hierarchical clustering algorithm for large data sets. *Pattern Recognition Letters* **25**(4) 505-13 https://doi.org/10.1016/j.patrec.2003.12.013

[15] Zadegan S M R, Mirzaie M, Sadoughi F 2013 Ranked k-medoids: A fast and accurate rank-based partitioning algorithm for clustering large datasets *Knowledge-Based Systems* **39** 133-43 https://doi.org/10.1016/j.knosys.2012.10.012

[16] Lamiaa Fattouh Ibrahim 2005 Using of clustering algorithm CWSP-PAM for rural network planning *Third International Conference on Information Technology and Alications, 2005. ICITA 2005k* DOI: 10.1109/ICITA.2005.300, Sydney, NSW, Australia

[17] Ibrahim L F 2006 Using of Clustering and Ant-Colony Algorithms CWSP-PAM-ANT in Network Planning *International Conference on Digital Telecommunications, ICDT '06* DOI: 10.1109/ICDT.2006.77, Cote d'Azur, France

[18] Arantes Paterlini A, Nascimento M A, Traina Jr. C 2011 Using Pivots to Speed-Up k-Medoids Clustering *Journal of Information and Data Management* **2**(2)

[19] Chu S-C, Roddick J F, Pan J-S 2002 An Incremental Multi-Centroid, Multi-Run Sampling Scheme for K-medoids-based Algorithms *WIT Transactions on Information and Communication Technologies* 10.2495/DATA020531, 28

[20] Zhu Ying-ting, Wang Fu-zhang, Shan Xing-hua, Lv Xiao-yan 2014 K-medoids clustering based on MapReduce and optimal search of medoids *9th International Conference on Computer Science & Education (ICCSE)* DOI: 10.1109/ICCSE.2014.6926527, Vancouver, BC, Canada

[21] Patel A, Singh P 2013 New Approach for K-mean and K-medoids Algorithm *International Journal of Computer Applications Technology and Research* **2**(1) 1-5 ISSN: 2319–8656

[22] Agarwal B, Mittal N 2016 Machine Learning Approach for Sentiment Analysis *Prominent Feature Extraction for Sentiment Analysis* pISBN 978-3-319-25341-1, 10.1007/978-3-319-25343-5_3, 21-45

[23] Agarwal B, Mittal N 2016 Semantic Orientation-Based Approach for Sentiment Analysis *Prominent Feature Extraction for Sentiment Analysis* pISBN 978-3-319-25341-1, 10.1007/978-3-319-25343-5_6, 77-88

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

[25] Ahmed S, Danti A 2016 Effective Sentimental Analysis and Opinion Mining of Web Reviews Using Rule Based Classifiers *Computational Intelligence in Data Mining* 1 pISBN 978-81-322-2732-8, DOI 10.1007/978-81-322-2734-2_18, 171-9, India

[26] Vo Ngoc Phu, Phan Thi Tuoi 2014 Sentiment classification using Enhanced Contextual Valence Shifters *International Conference on Asian Language Processing (IALP)* 224-9

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

[28] Vo Ngoc Phu, Vo Thi Ngoc Chau, Vo Thi Ngoc Tran, Nguyen Duy Dat 2017 A Vietnamese adjective emotion dictionary based on exploitation of Vietnamese language characteristics *Artificial Intelligence Review* DOI 10.1007/s10462-017-9538-6, 1-69

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

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

[31] Vo Ngoc Phu, Vo Thi Ngoc Chau, Vo Thi Ngoc Tran, Nguyen Duy Dat 2017 A Vietnamese adjective emotion dictionary based on exploitation of

Vietnamese language characteristics *International Journal of Artificial Intelligence Review (AIR)* doi:10.1007/s10462-017-9538-6, 1-67

[32] Vo Ngoc Phu, Vo Thi Ngoc Chau, Vo Thi Ngoc Tran, Nguyen Duy Dat, Tuan A. Nguyen 2017 STING Algorithm used English Sentiment Classification in A Parallel Environment *International Journal of Pattern Recognition and Artificial Intelligence* 31(7) DOI: 10.1142/S0218001417500215, 30 pages

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

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

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

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

[37] Vo Ngoc Phu, Vo Thi Ngoc Chau, Vo Thi Ngoc Tran, Nguy Duy Dat, Khanh Ly Doan Duy 2017 A Valence-Totaling Model for Vietnamese Sentiment Classification *International Journal of Evolving Systems (EVOS)* DOI: 10.1007/s12530-017-9187-7, 49 pages

[38] Vo Ngoc Phu, Vo Thi Ngoc Tran, Vo Thi Ngoc Chau, Dat Nguyen Duy, Khanh Ly Doan Duy 2017 Semantic lexicons of English nouns for classification *Evolving Systems* DOI: 10.1007/s12530-017-9188-6

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

Appendices

Table 1: The results of the 1,000,000 English documents in the testing data set.
Table 2: The accuracy of our new model for the 1,000,000 English documents in the testing data set.
Table 3: Average time of the classification of our new model for the 1,000,000 English documents in testing data set.
Table 4: Comparisons of our model's results with the works in [1-3].
Table 5: Comparisons of our model's advantages and disadvantages with the works in [1-3].
Table 6: Comparisons of our model's results with the works related to the K-Medoids Algorithm (PAM) in [7-18].
Table 7: Comparisons of our model's merits and demerits with the works related to the K-Medoids Algorithm (PAM) in [7-21].
Table 8: Comparisons of our model with the latest sentiment classification models (or the latest sentiment classification methods) in [22-27].
Table 9: Comparisons of our model's positives and negatives with the latest sentiment classification models (or the latest sentiment classification methods) in [22-27].

TABLE 1 The results of the 1,000,000 English documents in the testing data set

	Testing Dataset	Correct Classification	Incorrect Classification
Negative	500,000	428,714	71,286
Positive	500,000	430,786	69,214
Summary	1,000,000	859,500	140,500

TABLE 2 The accuracy of our new model for the 1,000,000 English documents in the testing data set

Proposed Model	Class	Accuracy
Our new model	Negative Positive	85.95%

TABLE 3 Average time of the classification of our new model for the 1,000,000 English documents in testing data set

	Average time of the classification /1,000,000 English documents
The K-Medoids Algorithm (PAM)in the sequential environment	5,267,195 seconds
The K-Medoids Algorithm (PAM) in the Cloudera distributed system – 3 nodes	1,688,939 seconds
The K-Medoids Algorithm (PAM) in the Cloudera distributed system – 6 nodes	868,384 seconds
The K-Medoids Algorithm (PAM) in the Cloudera distributed system – 9 nodes	584,647 seconds

TABLE 4 Comparisons of our model's results with the works in [1, 2, 3]

Studies	PAM	CT	Sentiment Classification	PNS	SD	DT	Language	VSM
[1]	No	No	No	No	Yes	No	EL	Yes
[2]	No	No	Yes	No	Yes	No	EL	Yes
[3]	No	No	Yes	No	Yes	Yes	EL	Yes
Our work	Yes	Yes	Yes	Yes	Yes	Yes	EL	Yes

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

TABLE 5 Comparisons of our model's advantages and disadvantages with the works in [1, 2, 3]

Researches	Approach	Advantages	Disadvantages
[1]	Examining the vector space model, an information retrieval technique and its variation	In this work, the authors have given an insider to the working of vector space model techniques used for efficient retrieval techniques. It is the bare fact that each system has its own strengths and weaknesses. What we have sorted out in the authors' work for vector space modelling is that the model is easy to understand and cheaper to implement, considering the fact that the system should be cost effective (i.e., should follow the space/time constraint. It is also very popular. Although the system has all these properties, it is facing some major drawbacks.	The drawbacks are that the system yields no theoretical findings. Weights associated with the vectors are very arbitrary, and this system is an independent system, thus requiring separate attention. Though it is a promising technique, the current level of success of the vector space model techniques used for information retrieval are not able to satisfy user needs and need extensive attention.
[2]	+Latent Dirichlet allocation (LDA). +Multi-label text classification tasks and apply various feature sets. +Several combinations of features, like bi-grams and uni-grams.	In this work, the authors consider multi-label text classification tasks and apply various feature sets. The authors consider a subset of multi-labelled files of the Reuters-21578 corpus. The authors use traditional TF-IDF values of the features and tried both considering and ignoring stop words. The authors also tried several combinations of features, like bi-grams and uni-grams. The authors also experimented with adding LDA results into vector space models as new features. These last experiments obtained the best results.	No mention
[3]	The K-Nearest Neighbours algorithm for English sentiment classification in the Cloudera distributed system.	In this study, the authors introduce a new weighting method based on statistical estimation of the importance of a word for a specific categorization problem. One benefit of this method is that it can make feature selection implicit, since useless features of the categorization problem considered get a very small weight. Extensive experiments reported in the work show that this new weighting method improves significantly the classification accuracy as measured on many categorization tasks.	Despite positive results in some settings, GainRatio failed to show that supervised weighting methods are generally higher than unsupervised ones. The authors believe that ConfWeight is a promising supervised weighting technique that behaves gracefully both with and without feature selection. Therefore, the authors advocate its use in further experiments.
Our work	The K-Medoids Algorithm (PAM) for English sentiment classification in the Cloudera distributed system. The advantages and disadvantages of the proposed model are shown in the Conclusion section.		

TABLE 6 Comparisons of our model's results with the works related to the K-Medoids Algorithm (PAM) in [7-21]

Studies	PAM	CT	Sentiment Classification	PNS	SD	DT	Language	VSM
[7]	Yes	Yes	NM	NM	NM	NM	NM	NM
[8]	Yes	Yes	NM	NM	NM	NM	NM	NM
[9]	Yes	Yes	NM	NM	NM	NM	NM	NM
[10]	Yes	Yes	NM	NM	NM	NM	NM	NM
[11]	Yes	Yes	NM	NM	NM	NM	NM	NM
[12]	Yes	Yes	NM	NM	NM	NM	NM	NM
[13]	Yes	Yes	NM	NM	NM	NM	NM	NM
[14]	Yes	Yes	NM	NM	NM	NM	NM	NM
[15]	Yes	Yes	NM	NM	NM	NM	NM	NM
[16]	Yes	Yes	NM	NM	NM	NM	NM	NM
[17]	Yes	Yes	NM	NM	NM	NM	NM	NM
[18]	Yes	Yes	NM	NM	NM	NM	NM	NM
[19]	Yes	Yes	NM	NM	NM	NM	NM	NM
[20]	Yes	Yes	NM	NM	NM	NM	NM	NM
[21]	Yes	Yes	NM	NM	NM	NM	NM	NM
Our work	Yes	Yes	Yes	Yes	Yes	Yes	EL	Yes

TABLE 7 Comparisons of our model's merits and demerits with the works related to the K-Medoids Algorithm (PAM) in [7-21]

Works	Approach	Merits	Demerits
[7]	A simple and fast algorithm for K-medoids clustering	The proposed algorithm calculates the distance matrix once and uses it for finding new medoids at every iterative step. To evaluate the proposed algorithm, the authors use some real and artificial data sets and compare with the results of other algorithms in terms of the adjusted Rand index. Experimental results show that the proposed algorithm takes a significantly reduced time in computation with comparable performance against the partitioning around medoids.	No mention
[8]	A fuzzy relative of the k-medoids algorithm with application to web document and snippet clustering	The objective functions are based on selecting c representative objects (medoids) from the data set in such a way that the total dissimilarity within each cluster is minimized. A comparison of FCMdd with the relational fuzzy c-means algorithm shows that FCMdd is much faster. The authors present examples of applications of these algorithms to web document and snippet clustering.	No mention
[9]	Computational Complexity between K-Means and K-Medoids Clustering Algorithms for Normal and Uniform Distributions of Data Points	In this research, the most representative algorithms K-Means and K-Medoids were examined and analysed based on their basic approach. The best algorithm in each category was found out based on their performance. The input data points are generated by two ways, one by using normal distribution and another by applying uniform distribution. Results: The randomly distributed data points were taken as input to these algorithms and clusters are found out for each algorithm. The algorithms were implemented using JAVA language and the performance was analysed based on their clustering quality. The execution time for the algorithms in each category was compared for different runs. The accuracy of the algorithm was investigated during different execution of the program on the input data points. Conclusion: The average time taken by K-Means algorithm is greater than the time taken by K-Medoids algorithm for both the case of normal and uniform distributions. The results proved to be satisfactory.	No mention

Works	Approach	Merits	Demerits
[10]	A New and Efficient K-Medoid Algorithm for Spatial Clustering	The new algorithm utilizes the TIN of medoids to facilitate local computation when searching for the optimal medoids. It is more efficient than most existing k-medoids methods while retaining the exact the same clustering quality of the basic k-medoids algorithm. The application of the new algorithm to road network extraction from classified imagery is also discussed and the preliminary results are encouraging.	No mention
[11]	A genetic k-medoids clustering algorithm	As a result the proposed algorithm can efficiently evolve appropriate partitionings while making no a priori assumption about the number of clusters present in the datasets. In the experiments, the authors show the effectiveness of the proposed algorithm and compare it with other related clustering methods.	No mention
[12]	An improved k-medoids clustering algorithm	The authors can get k clusters from the root of the CF-Tree. This algorithm improves obviously the drawbacks of the k-medoids algorithm, such as the time complexity, scalability on large dataset, and can't find the clusters of sizes different very much and the convex shapes. Experiments show that this algorithm enhances the quality and scalability of clustering	No mention
[13]	Clustering Rules: A Comparison of Partitioning and Hierarchical Clustering Algorithms	This work describes the application of a number of different clustering algorithms to these rules, in order to identify similar rules and to better understand the data.	No mention
[14]	Leaders–Subleaders: An efficient hierarchical clustering algorithm for large data sets	As an example, a two level clustering algorithm –'Leaders–Subleaders', an extension of the leader algorithm is presented. Classification accuracy (CA) obtained using the representatives generated by the Leaders–Subleaders method is found to be better than that of using leaders as representatives. Even if more number of prototypes are generated, classification time is less as only a part of the hierarchical structure is searched.	No mention
[15]	Ranked k-medoids: A fast and accurate rank-based partitioning algorithm for clustering large datasets	Comparison between our algorithm and two other partitioning algorithms is performed by using four well-known external validation measures over seven standard datasets. The results for the larger datasets show the superiority of the proposed algorithm over two other algorithms in terms of speed and accuracy.	No mention
[16]	Using of clustering algorithm CWSP-PAM for rural network planning	In this research the partitioning around medoids (PAM) original algorithm have been modified. Results demonstrate the effectiveness and flexibility of the modifying algorithm in tackling the important problem of rural network planning. Comparisons with related work are presented showing the advantages of the CWSP-PAM (Clustering with Shortest Path-PAM) algorithm introduced in this work.	No mention
[17]	Using of Clustering and Ant-Colony Algorithms CWSP-PAM-ANT in Network Planning	In the present survey, the CWSP-PAM algorithm is modified by introducing the ant-colony-based algorithm in the second step of the network planning process to find the optimal path between any node and the corresponding switch node. Results demonstrate the effectiveness and flexibility of the modifying algorithm in tackling the important problem of network planning	No mention
[18]	Using Pivots to Speed-Up k-Medoids Clustering	The authors target the class of k-medoids algorithms in general, and propose a technique that selects a well-positioned subset of central elements to serve as the initial set of medoids for the clustering process. The authors' technique leads to a substantially smaller amount of distance calculations, thus improving the algorithm's efficiency when compared to existing methods, without sacrificing effectiveness. A salient feature of our proposed technique is that it is not a new k-medoid clustering algorithm per se, rather, it can be used in conjunction with any existing clustering algorithm that is based on the k-medoid paradigm. Experimental results, using both synthetic and real datasets, confirm the efficiency, effectiveness and scalability of the proposed technique.	No mention

Works	Approach	Merits	Demerits
[19]	An Incremental Multi-Centroid, Multi-Run Sampling Scheme For K-medoids-based Algorithms	Experimental results demonstrate the proposed scheme can not only reduce by more than 80% computation time but also reduce the average distance per object compared with CLARA and CLARANS. IMCMRS is also superior to MCMRS. 1 Introduction Clustering is a useful practice of classification imposed over a finite set of objects. The goal of clustering is to group sets of objects into classes such that single groups have similar characteristics, while dissimilar objects are in separate groups. Various existing clustering algorithms have been proposed and designed to fit various formats and constraints of application including k-means, k-medoids, BIRCH, CURE, CHAMELEON, DBSCAN.	No mention
[20]	K-medoids clustering based on MapReduce and optimal search of medoids	This work proposed an improved algorithm based on MapReduce and optimal search of medoids to cluster big data. Firstly, according to the basic properties of triangular geometry, this paper reduced calculation of distances among data elements to help search medoids quickly and reduce the calculation complexity of k-medoids. Secondly, according to the working principle of MapReduce, Map function is responsible for calculating the distances between each data element and medoids, and assigns data elements to their clusters; Reduce function will check for the results from Map function, search new medoids by the optimal search strategy of medoids again, and return new results to Map function in the next MapReduce process. The experiment results showed that our algorithm in this study has high efficiency and good effectiveness.	No mention
[21]	New Approach for K-mean and K-medoids Algorithm	The new approach for the k mean algorithm eliminates the deficiency of exiting k mean. It first calculates the initial centro ids k as per requirements of users and then gives better, effective and stable cluster. It also takes less execution time because it eliminates unnecessary distance computation by using previous iteration. The new approach for k - medoids selects initial k medoids systematically based on initial centroids. It generates stable clusters to improve accuracy.	No mention
Our work	The K-Medoids Algorithm (PAM) for English sentiment classification in the Cloudera distributed system. Our research's merits and demerits are shown in the Conclusion section.		

TABLE 8 Comparisons of our model with the latest sentiment classification models (or the latest sentiment classification methods) in [22, 23, 24, 25, 26, 27]

Studies	PAM	CT	Sentiment Classification	PNS	SD	DT	Language	VSM
[22]	No	No	Yes	NM	Yes	Yes	Yes	vector
[23]	No	No	Yes	NM	Yes	Yes	NM	NM
[24]	No	No	Yes	NM	Yes	Yes	EL	NM
[25]	No	No	Yes	NM	Yes	Yes	NM	NM
[26]	No	No	Yes	No	No	No	EL	No
[27]	No	No	Yes	No	No	No	EL	No
Our work	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes

TABLE 9 Comparisons of our model's positives and negatives the latest sentiment classification models (or the latest sentiment classification methods) in [22, 23, 24, 25, 26, 27]

Works	Approach	Merits	Demerits
[22]	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
[23]	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
[24]	Exploiting New Sentiment-Based Meta-Level Features for Effective Sentiment Analysis	Experiments performed with a substantial number of datasets (nineteen) demonstrate that the effectiveness of the proposed sentiment-based meta-level features is not only superior to the traditional bag-of-words representation (by up to 16%) but also is also superior in most cases to state-of-art meta-level features previously proposed in the literature for text classification tasks that do not take into account any idiosyncrasies of sentiment analysis. The authors' proposal is also largely superior to the best lexicon-based methods as well as to supervised combinations of them. In fact, the proposed approach is the only one to produce the best results in all tested datasets in all scenarios.	A line of future research would be to explore the authors' meta features with other classification algorithms and feature selection techniques in different sentiment analysis tasks such as scoring movies or products according to their related reviews.
[25]	Rule-Based Machine Learning Algorithms	The proposed approach is tested by experimenting with online books and political reviews and demonstrates the efficacy through Kappa measures, which have a higher accuracy of 97.4% and a lower error rate. The weighted average of different accuracy measures like Precision, Recall, and TP-Rate depicts higher efficiency rate and lower FP-Rate. Comparative experiments on various rule-based machine learning algorithms have been performed through a ten-fold cross validation training model for sentiment classification.	No mention
[26]	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 that were not in five dictionaries before. The study shows that 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.	No mention
[27]	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	The K-Medoids Algorithm (PAM) for English sentiment classification in the Cloudera distributed system. The positives and negatives of the proposed model are given in the Conclusion section.		

Appendix of Codes

Algorithm 1: Transferring one English document into one multi-dimensional vector.
Algorithm 2: Creating the positive vector group.
Algorithm 3: Creating the negative vector group.
Algorithm 4: Clustering one multi-dimensional vector (corresponding to one English document of the English testing data set) into the positive vector group PositiveVectorGroup, the negative vector group NegativeVectorGroup, or not.

ALGORITHM 1: Transferring one English document into one multi-dimensional vector

---

```
Input: one English document
Output: one multi-dimensional vector
Begin
Step 0: Set LengthOfMultiDimensionalVector := 0;
Step 1: Set MultiDimensionalVector := {}{} with n_max rows and m_max columns.
Step 2: Set arraySentences := Split the English document into many separate sentences based on
“.” Or “!” or “?”;
Step 3: For i = 0; i < arraySentences.length; i++, do:
Step 4:   Set OneDimensionalVector := Transfer arraySentences[i] into one vector (one dimensiona)
based on VSM [1, 2, 3];
Step 5:   If OneDimensionalVector.length is less than m_max Then
Step 6:             For j = OneDimensionalVector.length; j< m_max; j++; do:
Step 7:                       OneDimensionalVector[j] := 0;
Step 8:             End For;
Step 9:   End If;
Step 10: MultiDimensionalVector.AddOneDimensionalVector (OneDimensionalVector);
Step 11: LengthOfMultiDimensionalVector = LengthOfMultiDimensionalVector + 1;
Step 12: End For;
Step 13: If LengthOfMultiDimensionalVector is less than n_max Then
Step 14: For k = LengthOfMultiDimensionalVector; k < n_max; k++; do:
Step 15:             MultiDimensionalVector.AddOneDimensionalVector (zero vector – one
dimension);
Step 16: End For;
Step 17: Return MultiDimensionalVector;
End;
```

ALGORITHM 2: Creating the positive vector group

```
Input: the 1,000,000 positive English documents of the English training data set.
Output: the positive vector group PositiveVectorGroup
Begin
Step 0: Set PositiveVectorGroup := {};
Step 1: For i = 0; i < 1,000,000; i++; do:
Step 2:   Set OneMultiDimensionalVector := Call Algorithm 1 with the positive English document
i in the English training data set;
Step 3:   PositiveVectorGroup.AddMultiDimensionalVector(OneMultiDimensionalVector);
Step 4: End For;
Step 5: Return PositiveVectorGroup;
End;
```

ALGORITHM 3: Creating the negative vector group

Input: the 1,000,000 negative English documents of the English training data set.
Output: the negative vector group NegativeVectorGroup
Begin
Step 0: Set NegativeVectorGroup := {};
Step 1: For i = 0; i < 1,000,000; i++; do:
Step 2: Set OneMultiDimensionalVector := Call Algorithm 1 with the negative English document i in the English training data set;
Step 3: NegativeVectorGroup.AddMultiDimensionalVector(OneMultiDimensionalVector);
Step 4: End For;
Step 5: Return NegativeVectorGroup;
End;

ALGORITHM 4: Clustering one multi-dimensional vector (corresponding to one English document of the English testing data set) into the positive vector group PositiveVectorGroup, the negative vector group NegativeVectorGroup, or not

Input: one multi-dimensional vector A (corresponding to one English document of the English testing data set), the positive vector group PositiveVectorGroup, the negative vector group NegativeVectorGroup;
Output: positive, negative, neutral;
Begin
Step 1: Implement the K-Medoids Algorithm (PAM) based on the K-Medoids Algorithm in [7-21] with input is one multi-dimensional vector (corresponding to one English document of the English testing data set), the positive vector group PositiveVectorGroup, the negative vector group NegativeVectorGroup;
Step 2: With the results of Step 1, If the vector is clustered into the positive vector group Then
Step 3: Return positive;
Step 4: Else If the vector is clustered into the negative vector group Then
Step 5: Return negative;
Step 6: Else
Step 7: Return neutral;
Step 8: End If;
Step 9: Return neutral;
End;

AUTHORS

Dr. Vo Ngoc Phu

Current position, grades: researcher; lecturer; Dr of computer science
University studies: Duy Tan University
Scientific interest: artificial intelligence; intelligent systems; expert systems; data mining; machine learning; distributed network environments; natural language processing;
Publications: 19 ISI manuscripts and 2 conferences
Experience: 10 years of researching; teaching

Dr. Vo Thi Ngoc Tran

Current position, grades: researcher; lecturer; Dr of computer science
University studies: Ho Chi Minh City University of Technology
Scientific interest: artificial intelligence; intelligent systems; expert systems; data mining; machine learning; distributed network environments; natural language processing;
Publications: 19 ISI manuscripts and 1 conferences
Experience: 10 years of researching; teaching