Available online at http://rjas.rsu.ac.th

Rangsit Journal of Arts and Sciences, January-June 2017 RJAS Vol. 7 No. 1, pp. 11-22
Copyright © 2011, Rangsit University
ISSN 2229-063X (Print)/ISSN 2392-554X (Online)

A comparative study of clustering techniques for non-segmented language documents

Todsanai Chumwatana

College of Information and Communication Technology, Rangsit University, Patumthani 12000, Thailand
E-mail: todsanai.c@rsu.ac.th

Submitted 28 January 2016; accepted in final form 20 February 2017
Available online 29 June 2017

Abstract

Document clustering has become an important area of study due to the rapid increase in the number of electronic documents. It can be employed to group and categorize documents, as well as provide a useful summary of the categories for browsing purposes. Until now, many clustering techniques have been developed for grouping and clustering documents both in segmented and non-segmented languages, like English and some Asian languages, respectively. However, document clustering can be a complicated task for many Asian languages such as Chinese, Japanese, Korean and Thai, because these languages are written without explicit word boundary delimiters such as white space. The aim of this paper is to provide a comprehensive and comparative study of non-segmented document clustering techniques using self-organizing map (SOM) and *k*-means, as they are two classic and well known methods in the area of text clustering. To illustrate these two methods, experimental and comparative studies on clustering non-segmented documents by using SOM and *k*-means are revealed in this paper. The keyword extraction is first applied to search for the member of occurrences. These members are then used as an input for the next clustering process. The experimental results show that *k*-means technique is simple and has low computation cost. Meanwhile, SOM is relatively complex, but the clustering performance is more visual and easy to comprehend. Consequently, *k*-means technique has become a well-known text clustering method and is used by many fields due to its straightforwardness, while SOM performs well for detection of noisy documents, thus making it more suitable for some applications such as navigation of document collection and multi-document summarization.

Keywords**:** *document clustering, k-means, non-segmented languages, self-organizing map*

1. Introduction

While the vast number of documents in non-segmented languages can be searched from the internet, developing a tool which can help users to effectively search and easily organize the information they are looking for has become a crucial and urgent matter. Document clustering has been considered as a practical approach to achieve the goal because of its capability of organizing large volumes of information into a smaller number of meaningful groups (Baeza-Yates & Ribeiro-Neto, 1999; Jain, Murty, & Flynn, 1999; Willett, 1988). This technique, also known as text clustering, is used to identify the similarity of a document and summarize large number of documents using key attributes of the clusters (Kalpana, & Vigneshwari, 2016). It is widely accepted that document clustering uses unsupervised learning techniques and assists fast information retrieval or filtering (Cutting, Pedersen, & Tukey, 1992), because it enables document categorization by arranging the documents into groups based on the similarities of their member occurrences. Regarding the document clustering in information retrieval, a document is considered as a bag of words. Although a document normally consists of a sequence of sentences and each sentence is composed of grammatically ordered words, when performing document clustering, the positions of words are disregarded. Instead, the frequencies of the words appearing in documents are used as key parameters to analyze the similarity of documents (Matveeva, 2006). Those documents containing the similarity of words and frequencies will be grouped under the same cluster. This clustering process is directly applied to European languages where words are clearly defined by the word delimiters such as space or other punctuation marks. Because of this characteristic of European languages, the European texts are explicitly segmented into word tokens. Many algorithms have been developed to calculate the similarity of documents and to build clusters for fast information retrieval. On the contrary, document clustering turns to be a challenging task for many Asian languages such as Chinese, Japanese, Korean and Thai, because a sentence in these

languages is written continuously as a sequence of characters without any explicit word boundary delimiters. So, they are recognized as non-segmented languages. Due to this characteristic, similarities texts of non-segmented language document are unable to be calculated directly. A preprocessing step needs to be performed to discover keywords from non-segmented language documents prior clustering. As a result, most approaches for clustering non-segmented language documents consist of two stages including keyword extraction and a document clustering process as shown in Figure 1

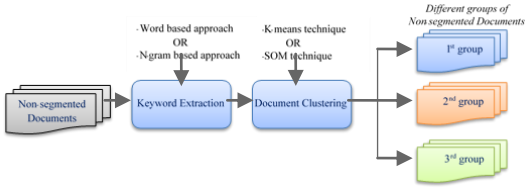

Figure 1 The process of non-segmented language document clustering

Figure 1 depicts the main approaches utilized to assess similarities. The word based and n-gram based approaches will be described in the next section and the document clustering section will provide the details of *k*-means and SOM techniques, followed by the comparison of clustering techniques in the final section.

Before keyword extraction and clustering techniques are described in more detail, the background of non-segmented languages is first provided to assist with understanding of the problem..

2. Background and related works

Unlike European languages where sentences are explicitly segmented by spaces or symbols, many Asian languages such as Chinese, Japanese, Korea or Thai are non-segmented texts. These languages share similar characteristics with each other in terms of the structure of writing. They are written in a string of symbols without explicit word boundary delimiters, such as white spaces, semicolons, and commas. The spaces in these languages are sometimes used to interrupt an idea or to help the reader pay attention to the texts, but they do not signify a split between words, phrases or sentences (Jaruskulchai & Kruengkrai, 2003). In this section, the Chinese language is selected from among non-segmented languages for the illustration because it is one of most widely used non-segmented languages. The Chinese language is similar to the other non-segmented languages in many ways. For example, each character has its own meaning, so it can be regarded as a word. On the other hand, several Chinese characters can be linked together to make a phrase. A phrase may consist of two, three, or more characters, but there are no spaces between Chinese characters except punctuation marks such as '，' or '．' (full stop) (Kwok, 1997). Chinese writings consist of mainly Han characters (hanzi), which are also used in the Japanese language (known as kanji) and in Korean (known as hanja). In modern Chinese, the pictograph words have been simplified by using characters made up of seven strokes (horizontal and vertical strokes, left-falling and right-falling strokes, a point stroke, and a hook stroke) as shown in Figure 2 (see Soapberry at http://www.4c.com.tw/photo/index/4c[1828-1]/[re]14649/200810141742.jpg)

Figure 2 Example of Chinese language

Words can be composed of one or more characters in texts and a word boundary is not necessarily used between two characters. In addition, Chinese does not have variations of words: no changes of tenses, gender, and no plural forms. The number of commonly used Chinese characters is 8,000 to 13,000 characters (Wieger, 1965).

In the higher levels, the Chinese language can be classified as a non-segmented language. Due to the reason that it does not have word delimiters, readers have to use their own knowledge to analyze context and separate words from the sentences like other non-segmented languages.

Due to the nature of the non-segmented language, keyword extraction has become one of the essential preprocessing methods in the area of document clustering. Usually, keywords are considered as a key factor to determine the main content of the whole documents. Nouns appeared in the documents are the most part containing the

semantics, although there are many compositions in a natural language text such as nouns, pronouns, articles, verbs, adjectives, adverbs, and conjunctions.

In text mining, keyword extraction is one of the major applications (Hearst, 1999; Tan, 1999). Information Extraction (IE) is an indispensable task in text mining that explains the process of discovering interesting keywords based on unstructured natural-language texts. Most of keyword extraction methods mentioned in the literature were attained by getting the set of words from given texts constructed. Then, the keywords were selected from the set of words while the preprocessing step was being executed.

The methods of extracting keywords in non-segmented languages proposed in the previous works can be classified into two main categories: word based and n-gram based approaches.

2.1 Word based approach

In the word based approach, there are several techniques proposed to split Asian texts such as Chinese (Cai & Zhao, 2016; Kwok, 1997), Japanese (Croft, 1993), Korea (Lee & Ahn, 1996) and Thai (Sukhahuta & Smith, 2000) into term tokens.

Usually, a word segmentation technique is applied to extract the keywords prior organizing the keywords into clustering techniques. Mostly, the word segmentation techniques are language-dependent and rely on language analysis or a dictionary. Due to the characteristic of being non-segmented, many Asian languages must be taken into the preparation of word segmentation which is very time-consuming. Therefore, word segmentation has become a challenging task in Natural Language Processing (NLP) for Asian languages.

A segmentation algorithm is usually used to segment text documents into words or terms before the next processing step can be performed. Although words can be manually identified by human experts for non-segmented languages, the process is time consuming and labor intensive. In non-segmented languages, several researchers have attempted to develop more efficient techniques of text segmentation to divide text documents into words or terms (Hasan & Matsumoto, 2000). For instance, the majority of the methods proposed for extracting words or terms in the Chinese language falls into one of two main categories: character based (CB) and word based (WB) (Kwok, 1997).

For Thai language and some other Asian languages, a plethora of algorithms is available for text segmentation. Approaches can be sorted into dictionary-based, rule-based and machine learning based categories. Dictionary-based methods match each word of the dictionary against the text (Brent & Tao, 2001; Sornlertlamvanich, 1993) and their performance depends on the size and the quality of the dictionary. The morphology of languages enables rule based techniques (Theeramunkong, Sornlertlamvanich, Tanhermhong, & Chinnan, 2000), but the accuracy then depends again on hand-crafted rules. Machine learning techniques (Haruechaiyasak, Kongyoung, & Damrongrat, 2000) use tagged training corpora to build a statistical model able to identify boundaries between words in text. Although this approach does not require the use of dictionary or language analysis, it still needs corpus and its performance depends critically on the characteristics of the document domain and the size of the training corpus. Also, the preparation of this approach is time consuming.

2.2 N-gram based approach

The n-gram technique was first introduced and tested as index-terms by Adams in 1991 (Adams, 1991). This technique is a language-independent approach, which does not require the use of language analysis, dictionary, or corpus. This makes the n-gram technique more popular and widely used to segment many Asian languages due to its being language-independent (Cavnar & Trenkle, 1994; Majumder, Mitra, & Chaudhuri, 2002; Ogawa & Matsuda, 1998). It can also be applied to other non-segmented texts such as genome or protein sequences (Jaruskulchai & Kruengkrai, 2003; Williams & Zobel, 2002).

However, selection of the dimension of the gram term is important for these non-segmented languages so that they are appropriate for each language. For instance, it has been shown that the bi-gram term is effective for clustering Chinese documents (Chien, 1995; Jiao, Liu, & Jia, 2007; Liang, Lee, & Yang, 1996; Lin, 2007). Furthermore, most Chinese bi-gram terms do not lose the semantics of words. In Japanese, the dimension of the gram term had also been found to be equal to two (Chien, 1995). In bioinformatics, CAFÉ (Williams & Zobel, 2002) is a well-known method which uses the n-gram base approach. It uses 9-gram terms for the genome sequence and 3-gram terms for the protein sequence.

Meanwhile the n-gram based approach with n equal to three and four characters seems to have the best parameters to achieve retrieval effectively for the Thai language (Chuleerat, 1998; Jaruskulchai, 1996). This is because the highest frequency of the top 20 3-gram and 4-gram terms are complete words in the Thai language, where Thai words have varying lengths. As a result, three and four are both used as the best parameters in the n-gram terms extraction for the Thai language. It has been shown (Jaruskulchai, 1996) that the n should be greater than 2 for the Thai language. By selecting the n greater than two, the possibility of achieving the effective clustering is increased, since the minimum number of characters for Thai word appearance is two, with at least one of them being a consonant. Furthermore, each Thai character cannot represent a word or a meaning like Chinese or Japanese. In Thai, the smallest unit which can represent a word or a meaning is a syllable.

Due to these reasons, there is no single parameter for n-gram that is best for all non-segmented texts and applications. The following paragraphs will describe the process of n-gram term extraction.

Assume that document *d* consists of a string of characters a_1, a_2, ..., a_N. An n-gram term is a substring of *n* overlap or non-overlap successive characters extracted from the string. Extracting a set of n-gram terms from the documents *d* can be done by using the 1-sliding technique (Kim, Whang, Lee, & Lee, 2005) by sliding a window of length *n* from a_1 to a_N and storing the characters located in the window. Therefore, the *i*th n-gram term extracted from document *d* is the substring a_i, a_{i+1}, ..., a_{i+n}. Figure 3 shows 1-gram, 2-gram, ..., n-gram overlap sequence of Chinese text.

Let *Chinese text*: 假的作真的时真的亦为假的

1-gram terms: 假, 的, 作, 真, 的, 时, 真, 的, 亦, 为, 假, 的
2-gram terms: 假的, 的作, 作真, 真的, 的时, 时真, 真的, 的亦, 亦为, 为假, 假的
:
:
n-gram terms: 假的作真的时真的亦为假的

Figure 3 Substrings of 1-gram, 2-gram, ..., n-gram overlap sequence of Chinese text

After keyword extraction is performed by using word based or n-gram based approaches, keywords are then transformed into the feature vector of the words that appear in the documents. The term-weights (usually term-frequencies) of the words are also contained in each feature vector. The vector space model (VSM) has been a standard model of representing documents by containing the set of words with their frequencies (Liu, 2007). In the VSM, each document is replaced by the vector of the words. The vector size is dependent on the number of keywords that appear in the documents. For instance, let w_{ik} be the weight of keyword *k* that appears in the document *i*, and $D_i = (w_{i1}, w_{i2}, ..., w_{it})$ is the feature vector for document *i*, where *t* is the number of unique words of all documents. Therefore, the size of the feature vector is equal to *t* dimension as shown in Figure 4.

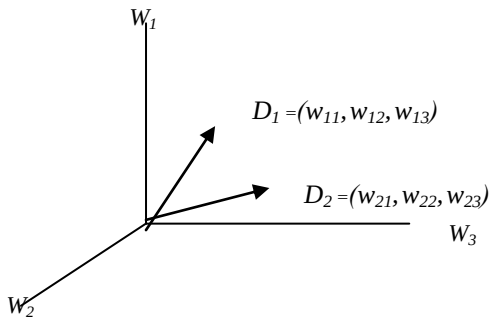

Figure 4 An example of the document vectors in 3-dimension

From Figure 4, the similarity between two documents can be computed with one of several similarity measures based on two corresponding feature vectors, e.g., cosine measure, Jaccard measure, and Euclidean distance measure (Feldman & Sanger, 2006). In document clustering, there are two main approaches: hierarchical and partitional approaches (Jain, 1988; Kaufman & Rousseeuw, 1990; Steinbach & Kumar, 2000). The hierarchical approach produces document clusters by using a nested sequence of partitions that can be represented in the form of a tree structure called a dendrogram. The root of the tree contains one cluster covering all data points, and a singleton cluster of individual data points is shown on the leaves of the tree. There are two basic methods when performing hierarchical clustering: agglomerative (bottom up) and divisive (top down) clustering (Steinbach & Kumar, 2000). The advantages of the hierarchical approach are that it can take any form of similarity function, and also the hierarchy of clusters allows users to discover clusters at any level of detail. However, this

technique may suffer from the chain effect, and its space requirement is at least quadratic or $O(n^2)$ compared to the *k*-means algorithm that provide $O(Iknm)$ where I is the number of necessary iterations, k is the number of clusters, n is the number of documents and m is the dimensionality of the vectors. The partitional approach (Zhao, 2002), on the other hand, can be divided into several techniques, e.g., *k*-means (Huang, 1998), Fuzzy c-means (Dembele & Kastner, 2003), and QT (quality threshold) (Heyer, Kruglyak, & Yooseph, 1999) algorithms. The *k*-means algorithm is more widely used among all clustering algorithms because of its efficiency and simplicity. The basic idea of *k*-means algorithm is that it divides a given data set into *k* clusters defined by users where each cluster has a center point, also called centroid that can be used to represent the cluster. However, its weaknesses are that it is only applicable to data sets where the notion of the mean is defined, the number of clusters can be identified by users, and sensitivity to data points that are very far from outliers (Liu, 2007). Furthermore, a self-organizing map (SOM) (Chumwatana, Wong, & Xie, 2010; Fung, Wong, Eren, Charlebois, & Crocker, 1997) can be used as one of the clustering algorithms in the family of an artificial neural network. The self-organizing map is an unsupervised neural network architecture, capable of ordering high dimensional data in such a way that similar inputs are grouped spatially close to each other. To use SOM in document clustering, text documents are described by features with high dimensionality, and SOM based techniques have been applied to document clustering (Yang, Lee, & Hsiao, 2015). In the following section, *k*-means and Self-organizing map techniques will be described in detail as these two techniques are well known in the area of document clustering (Olszewski, 2016).

3. Document clustering

In document clustering, there are two well-known techniques: *k*-means and self-organizing map which will be described in this study. Later in the subsequent sections, comparison and discussion on these techniques will also be revealed.

3.1 *K*-means algorithm

K-means algorithm is partition-based clustering method (Arora & Varshney, 2016). When *k*-means is used for document clustering, all the documents will be put into *k* clusters randomly, and then the clustering partition will be adjusted according to some principles until the clustering results are stable. *K*-means algorithm separates a given data set into *k* clusters where each cluster has the center point, also called centroid, that can be used to represent the cluster. *k*-data points are randomly selected as the centroids by the algorithm. All data points are then assigned to the closest centroid by computing the distance between every data point and each centroid. Therefore, each centroid and its members can form a cluster. The algorithm also re-computes the centroid of each cluster using the data in the current cluster, and this step is repeated until the centroids stabilize. To aid in understanding, the process of *k*-means algorithm for document clustering can be described step by step as follows:

Input: *n* documents to be clustered, the cluster number *k* defined by user

Output: *k* clusters, and each document will be assigned to one cluster

1) Choose *k* documents randomly as the initial clustering document seeds;

2) Calculate the centroid of each cluster;

3) According to the mean vector of all documents in each cluster, assign each document into most similar cluster by computing the distance between input document and the centroid of each cluster;

4) Update the mean vector of each cluster according to the document vector in it;

5) Repeat steps 2, 3 and 4 until the partition is stable;

6) Output the generated clusters and the partition.

According to above process, Figure 5 shows the steps of *k*-means algorithm and *k* is equal to two.

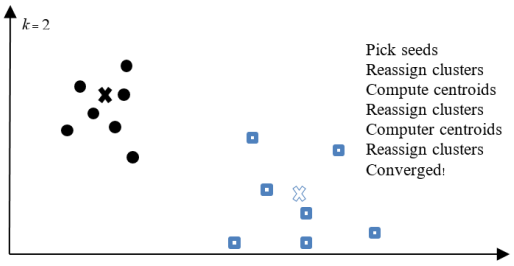

Figure 5 Process of *k*-means algorithm

In practical applications, the efficiency of *k*-means algorithm is dependent on the selection of *k* value. For instance, if *k* is set to *n*, it means that the number of clusters is equal to the number of documents, which is meaningless. In the case of $k = 1$,

it means that there is no partition in clustering. As a result, the selection of appropriate k value is regarded as the most important part in k-means algorithm.

3.2 Self-organizing map algorithm

In 1984, Teuvo Kohonen developed Self-organizing map or SOM which is one of the unsupervised learning methods in the family of artificial neural networks (Chumwatana, Wong, & Xie, 2009). This technique is well known and has been used in many areas (Merkevičius, Garšva, & Simutis, 2015; Rajchl et al., 2016; Berrada et al., 2016). The SOM can be visualized as a regular two-dimensional array of cells or nodes called neurons. This algorithm defines a mapping from the input vector onto a two-dimensional array of nodes. When the input vector $x(t) \in R^n$ is given, it is connected to all neurons in the SOM array denoted as vector $m_i(t) \in R^n$, which are associated by each neuron and is gradually modified in the learning process. The input vector $x(t) \in R^n$ is the input data set where t represents the keywords of the input documents. These input data sets have to be mapped with all neurons in the map which is denoted as a two-dimensional network of cells or the model vector $m_i(t) \in R^n$.

In mapping, the node where vector m_i is most similar to the input vector x will be activated. This node is often called a best-matching node or a winner. The winner and a number of its neighboring nodes in the SOM array are then turned towards the input vector x according to the learning principle. To aid in understanding, the process of clustering using SOM will be described step by step as follows:

Input: n documents to be clustered

Output: m neurons, and each document will be assigned to one neuron

1) Generate a set of keywords together with their frequency from the input training document;

2) Calculate the weight of each keyword occurring in each input document represented by document vector;

3) Label these documents into neurons according to the similarity of their document vectors;

4) Train SOM algorithm using neurons in the document cluster map;

5) Match new input document vectors with all neurons in the cluster map by considering the similarity of keywords;

6) Output the different generated neurons.

According to above processes, the following is an example of document clustering using SOM.

Let D be a document collection consisting of n documents, d_1, d_2, ..., d_n. Firstly, the keyword extraction technique is used to generate a set of keywords together with their frequency f from the document collection.

Assuming the above process produces m keywords from the document collection, denoted $KW = (kw_1, kw_2, ..., kw_m)$, where kw_i is the ith keyword generated from the document collection. The weight w_{ij} which represents the frequency of keyword kw_i occurring in document d_j for each keyword and each document will then be calculated. Finally, an $m*n$ matrix of such weights is constructed. In this matrix, row i represents the frequencies of occurrence of the ith keyword kw_i in the n documents, while jth column represents the document vector for document j, as depicted in Figure 6.

fm_i

$$V = \begin{pmatrix} 4 & 3 & 7 & 2 & 0 & 0 & \dots & 0 & 0 \\ 3 & 3 & 0 & 0 & 2 & 0 & & 2 & 0 \\ 6 & 4 & 5 & 2 & 0 & 0 & & 0 & 0 \\ 2 & 2 & 2 & 1 & 0 & 0 & & 0 & 0 \\ 3 & 3 & 4 & 3 & 0 & 0 & & 0 & 0 \\ 0 & 0 & 0 & 8 & 4 & 7 & & 4 & 9 \\ \vdots & & & & & & & & \\ 0 & 0 & 0 & 3 & 0 & 4 & \dots & 0 & 0 \\ 0 & 0 & 0 & 2 & 0 & 2 & & 0 & 0 \\ 0 & 0 & 0 & 4 & 3 & 4 & & 3 & 3 \\ 0 & 0 & 2 & 3 & 4 & 3 & & 4 & 0 \end{pmatrix}$$

ผลการแข่งขัน (Competition result)
เป็นอันดับที่ (Rank)
ตารางการแข่งขัน (Competition timetable)
กรรมการตัดสิน (Umpire)
รอบรองชนะเลิศ (Semi-final round)
การเมือง (Politic)
โรงแรม (Hotel)
โปรโมชั่น (Promotion)
ร้านอาหาร (Restaurant)
ส่วนลด (Discount)

d_1 d_2 d_3 d_4 d_5 d_l ... d_{n-1} d_n

Figure 6 The example of the document matrix

Figure 6 shows an example of a document matrix, where each element w_{ij} is more than 0 if kw_i occurs in the document d_j or 0 if kw_i does not appear in the document d_j, i.e.,

$$w_{ij} = \begin{cases} > 0 \text{ if } kw_i \text{ occurs in } d_j \\ 0 \text{ otherwise} \end{cases}$$

After all document vectors are generated from non-segmented document corpus, SOM is then used to represent all vectors as its own members for further clustering process. The document will be assigned into a neuron by considering the keywords in each document vector. Consequently, the set of same documents will be placed into the same neuron if they share the same keywords and the similar documents will be placed into neighboring neurons

in the case that they have similar keywords. On the other hand, the documents which contain totally different keywords will be mapped to far neurons, called distant neurons. Finally, this expresses that the neurons can build a document cluster map which contains all documents in different position on the document cluster map according to the extracted keywords in each document. The organization of the document cluster map that groups same documents, similar documents and different documents into the same neuron, neighboring neuron, and distant neuron respectively is shown in Figure 7. Keywords in the boxes in Figure 7 are extracted from non-segmented documents and they can represent the type of documents in the collection.

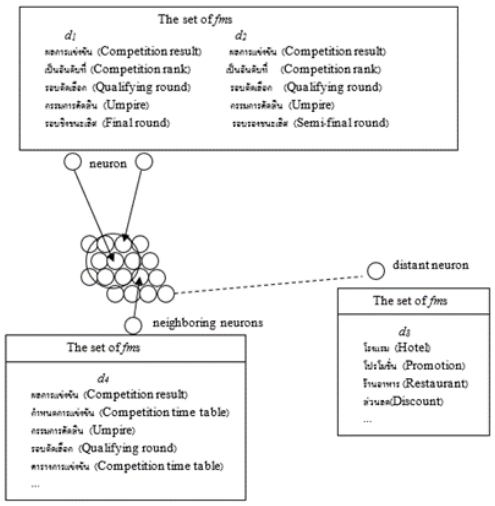

Figure 7 An example of document cluster map

After the SOM has been trained, the document clusters are formed by labeling each neuron that contains certain documents of similar type. The documents in the same neuron may not contain exactly the same set of keywords, but they contain mostly overlapping keywords.

As a result, the trained SOM will then be used to match the input data x (input documents) with the neurons in the document cluster map as will be described in the following section.

Consider the input vector $x = /x_1, x_2, \ldots, x_n/^t \in R^n$ as the input data sets where t is the KW of the input documents. These input data sets have to be matched with all neurons in the map that is denoted as a two-dimensional network of cells or the model vector $m_i = /m_{i1}, m_{i2}, \ldots, m_{in}/^t \in R^n$ depicted in Figure 8.

Each neuron i in the network contains the model vector m_i, which has the same number of keywords as the input vector x.

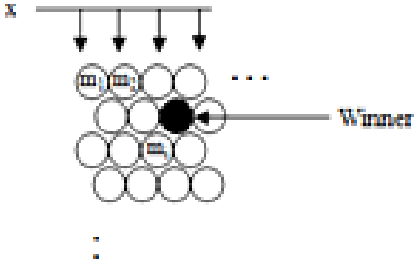

Figure 8 Self-organizing map

From Figure 8, the input vector x is compared with all neurons in the model vector m_i to find the best matching node called the winner. The winner unit is the neuron on the map where the set of the keywords of the input vector x is the same or similar to the set of the keywords of the model vector m_i by using some matching criterion e.g. the Euclidean distances between x and m_i. As a result, this method can be used to cluster documents into different groups, and it also suggests that this can be used to reduce the search time for the relevant document.

4. Experimental studies, comparison results and discussion

The experiment of clustering non-segmented documents is presented in this section. By using the SOM and k-means techniques, the dataset of 60 Thai language documents was used as an input of this study. By randomly collecting documents from Thai news websites, the dataset consisted of 15 sports, 15 travel, 15 political, and 15 education documents as show in the Table 1.

Table 1 The input dataset of 60 Thai language documents

Categories	Document ID
Sport	Spo1, Spo2, Spo3, Spo4, Spo5, Spo6, Spo7, Spo8, Spo9, Spo10, Spo11, Spo12, Spo13, Spo14, Spo15
Travel	Tra1, Tra2, Tra3, Tra4, Tra5, Tra6, Tra7, Tra8, Tra9, Tra10, Tra11, Tra12, Tra13, Tra14, Tra15
Political	Pol1, Pol2, Pol3, Pol4, Pol5, Pol6, Pol7, Pol8, Pol9, Pol10, Pol11, Pol12, Pol13, Pol14, Pol15
Education	Edu1, Edu2, Edu3, Edu4, Edu5, Edu6, Edu7, Edu8, Edu9, Edu10, Edu11, Edu12, Edu13, Edu14, Edu15

From Table 1, all Thai language documents were used as an input for the first keyword extraction process. After that, the 2 clustering techniques, SOM and *k*-means, were applied to the set of extracted keywords. Firstly, keywords were extracted by using Thai word segmentation techniques from the input dataset to get the words segmented. Then, keywords were transformed into feature vectors of the individual words which appeared in the documents. Not only the keywords but the term frequencies of the words were also included in each feature vector. The feature vectors were used to compute the similarity of the documents. By applying SOM technique, the results showed that it was capable of automatically clustering 60 documents into 5 neurons on the map, and it also could categorize the similar documents into a group within the same neuron as presented in Table 2.

Table 2 Clustering results of using SOM technique

Neuron ID	Document ID
Neuron 1	Tra12
Neuron 2	Pol2, Pol3, Pol4, Pol5, Pol6, Pol7, Pol8, Pol9, Pol10, Pol11, Pol12, Pol13, Pol14, Pol15
Neuron 3	Edu5, Edu9, Edu14, Edu15, Tra1, Tra3, Tra11, Tra14
Neuron 4	Tra2, Tra4, Tra5, Tra6, Tra7, Tra8, Tra9, Tra13, Tra15
Neuron 5	Pol1, Edu1, Edu2, Edu3, Edu4, Edu6, Edu7, Edu8, Edu10, Edu11, Edu12, Edu13, Spo1, Spo2, Spo3, Spo4, Spo5, Spo6, Spo7, Spo8, Spo9, Spo10, Spo11, Spo12, Spo13, Spo14, Spo15, Tra10

In *k*-means technique, the number of clusters was set to five, to equal the number of neurons. The experimental result showed that this technique can cluster 50 documents into five clusters as shown in Table 3 below.

Table 3 Clustering results of using *k*-means approach

Cluster ID	Document ID
Cluster 1	Edu1, Edu3, Edu4, Edu6, Edu7, Edu8, Edu10, Edu11, Edu12, Edu13, Spo9, Spo10, Spo11, Spo13, Tra1, Tra2, Tra3, Tra4, Tra5, Tra7, Tra8, Tra9, Tra10, Tra11, Tra12, Tra13, Tra14, Tra15, Pol15
Cluster 2	Edu2, Edu5, Edu9, Edu14, Edu15
Cluster 3	Spo1, Spo12, Spo14, Spo16
Cluster 4	Spo2, Spo3, Spo4, Spo5, Spo6, Spo7, Spo8, Spo15
Cluster 5	Pol1, Pol2, Pol3, Pol4, Pol5, Pol6, Pol7, Pol8, Pol9, Pol10, Pol11, Pol12, Pol13, Pol14

From the experimental results above, both techniques of document clustering obviously provide good results in the group of political documents. In SOM technique, the political documents were clustered into neuron 2. Whereas, in *k*-means technique, they were put together in cluster 5. Interestingly, the group of education documents also performed well in clustering experiments by both techniques. It was found that most of the education documents were grouped into neuron 5 and cluster 1 in the SOM technique and *k*-means techniques, respectively. However, in SOM technique, the group of education and sports were mapped into the same neuron which is neuron 5. This is because these two groups contained the same keywords which overlapped each other. Likewise, in *k*-means techniques, the group of travel and education documents was included together in cluster 1 as they had many overlapping words. Furthermore, it also shows that some of the education and sports documents are allocated into different clusters as displayed in Table 3. Meanwhile, in Table 2, there are some errors which occurred within the group of travel documents as they were scattered into different neurons because of their overlapping words with other groups. To compare and discuss these two clustering techniques more clearly, the advantages and disadvantages of *k*-means and self-organizing map techniques are also provided in Table 4 and Table 5.

Table 4 Advantages and disadvantages of *k*-means techniques

***k*-means technique**
Advantages: • *k*-means technique is simple and easy to implement. • With a large number of documents and when *k* is small, *k*-means technique is computationally faster than hierarchical clustering and self-organizing map techniques. • *k*-means technique produces tighter clusters than hierarchical clustering and self-organizing map techniques. • *k*-means technique gives best results when data set are distinct or explicitly separated from each other.
Disadvantages: • Fixed number of clusters can make it difficult to predict which *k* should be appropriate. • *k*-means technique does not work well with non-globular clusters. • Different initial partitions can result in different final clusters. • The learning algorithm requires a priori specification of the number of centroids. • The learning algorithm is not invariant to non-linear transformations. • Euclidean distance measures can unequally weight underlying factors. • The learning algorithm provides the local optima of the squared error function. • Unable to handle noisy data and outliers.

Table 5 Advantages and disadvantages of self-organizing map techniques

Self-organizing map technique (SOM)
Advantages: • SOM is easily interpreted and understood. • The reduction of dimensionality and grid clustering makes it easy to observe similarities in the data. • SOM is capable of handling several types of classification problems, and also providing a useful, interactive and intelligible summary of the data. • SOM is fully capable of clustering large and complex data sets. • SOM can be trained in a short amount of time, after that it can be used to cluster data set efficiently
Disadvantages: • SOM requires necessary and sufficient data in order to develop meaningful clusters. • SOM is often difficult to obtain a perfect mapping where groupings are unique within the map. • SOM requires that nearby data points behave similarly.

In summary, *k*-means technique is simple and easy to implement, and also has low computation cost. As a result, *k*-means technique has become a well-known text clustering method and is used by many fields. Meanwhile, SOM is relatively complex, but the clustering performance is more visual and easy to comprehend. The SOM also performs well for detection of noisy documents and topology preservation, thus making it more suitable for applications such as navigation of document collection, multi-document summarization, et cetera.

5. Conclusions

This paper provides the comparative studies of document clustering techniques for non-segmented languages such as Chinese, Japanese, Korean or Thai. The techniques revealed in this paper are *k*-means and self-organizing map as they are the two most popular methods and are well known in the area of text clustering. In order to cluster non-segmented languages, the process can be divided into two main phases: preprocessing phase and clustering phase. In the preprocessing phase, the keyword extraction: word based and n-gram based approaches are first applied to extract the keywords, together with their number of occurrences, from the non-segmented documents. In the clustering phase, clustering techniques: *k*-means and self-organizing map techniques are then applied to group similar document by using the bag of keywords extracted from the first phase. The experimental studies and comparison results on clustering 60 Thai text documents are presented in this paper. From the experimental results, the SOM technique can be used to cluster Thai documents into different clusters with more accuracy than *k*-means technique. This is because the SOM technique first performed the training process before clustering, meanwhile *k*-means technique does not have the training process. The advantages and disadvantages of the two clustering techniques are also assessed in this paper. Tables 4 and 5 showed that *k*-means is easy to realize and it usually has low computation cost, so it has become a well-known text clustering method used by many fields. Meanwhile, SOM is more complex, but the clustering performance is more efficient. Although these two techniques are different in some ways, they are both applicable for clustering non-segmented languages in order to enhance the performance of information retrieval, summarization and the other areas in natural language processing.

6. Notification

Portions of this research work were presented at the Knowledge Management International Conference (KMICe) 2014, 12-15 August 2014, Malaysia and published in as Chumwatana (2014) Using Clustering Techniques for non-segmented Language Document

Management: A Comparison of K-mean and Self Organizing Map Techniques. Proceedings in Knowledge Management International Conference (KMICe) 2014, 12-15 August 2014, Malaysia, PID214, 600-605 (Chumwatana, 2014). This paper is extended and provide more detail of clustering non-segmented language documents as well as discussion on advantages and disadvantages of techniques

7. References

Adams, E. (1991). *A study of trigrams and their feasibility as index terms in a full text information retrieval system* (PhD's thesis, George Washington University, USA).

Arora, P., & Varshney, S. (2016). Analysis of k-means and k-medoids algorithm for big data. *Procedia Computer Science*, *78*, 507-512. DOI: 10.1016/j.procs.2016.02.095

Baeza-Yates, R., & Ribeiro-Neto, B. (1999). *Modern information retrieval.* New York, USA: ACM Press.

Berrada, M., Hmaidi, A. E., Monyr, N., Abrid, D., Abdallaoui, A., Essahlaoui, A., & Ouali, A. E. (2016). Self-organizing map for the detection of seasonal variations in Sidi Chahed Dam sediments (Northern Morocco). *Hydrological Sciences Journal*, *61*(3), 628-635. http://dx.doi.org/10.1080/02626667.2014.964717

Brent, M. R., & Tao, X. (2001). Chinese text segmentation with MBDP-1: making the most of training corpora. *In Proceedings of the 39th Annual Meeting on Association for Computational Linguistics (ACL 2001)*. Toulouse, France, pp. 90-97. DOI: 10.3115/1073012.1073025

Cai, D., & Zhao, H. (2016). Neural word segmentation learning for Chinese. *arXiv preprint arXiv:1606.04300 [cs.CL]*. https://arxiv.org/abs/1606.04300

Cavnar, W., & Trenkle, J. (1994). N-gram based text categorization. *In Proceedings of the 3rd Annual Symposium on Document Analysis and Information Retrieval.* Las Vegas, USA, pp. 161-175.

Chien, L. F. (1995). Fast and quasi-natural language search for gigabytes of Chinese texts. *In Proceedings of 18th ACM SIGIR Conference on Research and Development in Information Retrieval*. New York, USA, pp. 112-120. DOI: 10.1145/215206.215345

Chuleerat, J. (1998). *An automatic indexing for Thai text retrieval* (PhD's thesis, George Washington University, USA).

Chumwatana, T. (2014). Using clustering techniques for non-segmented language document management: A comparison of k-mean and self organizing map techniques. In *Proceedings in Knowledge Management International Conference (KMICe) 2014*, 12-15 August 2014, Malaysia, PID214, 600-605.

Chumwatana, T., Wong, W. K., & Xie, H. (2009). Non-segmented document clustering using self-organizing map and frequent max substring technique. In *16th International Conference on Neural Information Processing (ICONIP 2009)*, Bangkok, Thailand. pp. 691-698.

Chumwatana, T., Wong, K.W. & Xie, H. (2010) A SOM-based document clustering using frequent max substrings for non-segmented texts. *Journal of Intelligent Learning Systems and Applications, 2* (03), 117-125. http://dx.doi.org/10.4236/jilsa.2010.23015

Croft, W. B. (1993). A comparison of indexing techniques for Japanese text retrieval. In *Proceedings of ACM SIGIR International Conference on Research and Development in Information Retrieval*, pp. 237-246.

Cutting, R. D., Pedersen, J. O., & Tukey, J. W. (1992). Scatter/gather*:* A cluster-based approach to browsing large document collections. In *The 15th Annual ACM-SIGIR '92*, pp. 318-329.

Dembele, P., & Kastner, P. (2003). Fuzzy C-means method for clustering microarray data. *Bioinformatics*, *19*(8), 973-980. DOI: 10.1093/bioinformatics/btg119

Feldman, R., & Sanger, J. (2006). *The text mining handbook: Advanced approaches in analyzing unstructured data.* UK: Cambridge University Press.

Fung, C. C., Wong, W. K., Eren, H., Charlebois, R., & Crocker, H. (1997). Modular artificial neural network for prediction of petrophysical properties from well Log data. *IEEE Transactions on*

Instrumentation & Measurement, 46(6), 1295-1299.

Haruechaiyasak, C., Kongyoung, S., & Damrongrat, C. (2000). LearnLexTo:a machine-learning based word segmentation for indexing Thai texts. In *Proceedings of iNEWS'08: Proceedings of the 2nd ACM Workshop on Improving non English web searching.* Napa Valley, CA, USA, pp. 85-88.

Hasan, M. M., & Matsumoto, Y. (2000). Chinese-Japanese cross language information retrieval: A Han character based approach. In *Proceedings of the SIGLEX Workshop on Word Senses and Multi-linguality (ACL-2000)*. Hong Kong, pp. 19-26.

Hearst, M. A. (1999). Untangling text data mining. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics (ACL)* University of Maryland, June 20-26, 1999, pp. 3-10.

Heyer, L. J., Kruglyak, S. & Yooseph, S. (1999). Exploring expression data: identification and analysis of coexpressed genes. *Genome Research, 9*(11), 1106-1115.

Huang, Z. (1998). Extensions to the k-means algorithm for clustering large datasets with categorical values. *Data Mining and Knowledge Discovery*, 3, 283-304. DOI: 10.1023/A:1009769707641

Jain, A. K. (1988). *Algorithms for clustering data*. Englewood Cliffs, New Jersey, USA: Prentice Hall.

Jain, A. K., Murty, M. N., & Flynn, P. J. (1999) Data clustering: a review. *ACM Computing Surveys (CSUR), 31*(3), 264-323.

Jaruskulchai, C. (1996). Thai text segmentation: problems and potential solutions. In *the Sixth Annual Workshop on Science and Technology Exchange between Thai Professionals in North America and Thailand.* Edmonton, Alberta, Canada.

Jaruskulchai, C., & Kruengkrai, C. (2003). A practical text summarizer by paragraph extraction for Thai. In *Proceedings of the sixth international workshop on Information retrieval with Asian languages, Japan, 11*, 9-16. DOI: 10.3115/1118935.1118937

Jiao, H., Liu, Q., & Jia, H. (2007). Chinese keyword extraction based on n-gram and word co-occurrence. *In 2007 International Conference on Computational Intelligence and Security Workshops (CISW 2007).* 15-19 December 2007. China, pp. 152-155. DOI: 10.1109/CISW.2007.4425468

Kalpana, S., & Vigneshwari, S. (2016). Selecting multiview point similarity from different methods of similarity measure to perform document comparison. *Indian Journal of Science and Technology*, *9*(10). DOI: 10.17485/ijst/2016/v9i10/88903

Kaufman, L., & Rousseeuw, P. J. (1990). *Finding groups in data*: *An introduction to cluster analysis*. *John Wiley and Sons*.

Kim, M. S., Whang, K. Y., Lee, J. G., & Lee, M. J. (2005). n-Gram/2L: A space and time efficient two-level n-Gram inverted index structure. In *Proceedings of the 31st VLDB Conference*, Trondheim, Norway, pp. 325-336.

Kohonen, T. (1984). *Self-organization and associative memory*. vol. 8: *More about biological memory*. Springer-Verlag Berlin Heidelberg.

Kwok, K. L. (1997). Comparing representations in Chinese information retrieval. In *Proceedings of the 20th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. Philadelphia, USA, pp. 34-41.

Lee, J. H., & Ahn, J. S. (1996). Using n-grams for Korean text retrieval. *In SIGIR '96 Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval*. Zurich, Switzerland, August 18 - 22, 1996, pp. 216-224. DOI: 10.1145/243199.243269

Liang, T., Lee, S. Y., & Yang, W. P. (1996). Optimal weight assignment for a Chinese signature file. *In Journal of Information Processing and Management*, *32*(2), 227-237. https://doi.org/10.1016/S0306-4573(96)85008-4

Lin, Y. T. (2007). *Chinese-English dictionary of modern usage*. Hong Kong: Chinese University of Hong Kong Press.

Liu, B. (2007). *Web data mining*: *Exploring hyperlinks, contents, and usage data*. New York, USA: Springer-Verlag Berlin Heidelberg.

Majumder, P., Mitra, M., & Chaudhuri, B. B. (2002). N-gram: a language independent approach to IR and NLP. *In International Conference on Universal Knowledge*.

Matveeva, I. (2006). Document representation and multilevel measures of document similarity. In *Proceedings of the 2006 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology: companion volume: doctoral consortium*. New York, June 04-09, pp. 235-238. DOI: 10.3115/1225797.1225804

Merkevičius, E., Garšva, G., & Simutis, R. (2015). Forecasting of credit classes with the self-organizing maps. *Information technology and control*, *33*(4), 61-66.

Ogawa, Y., & Matsudua, T. (1998). Optimizing query evaluation in n-gram indexing. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval.* Melbourne, Australia, August 24-28, 1998, pp. 367-368. DOI: 10.1145/290941.291057

Olszewski, D. (2016). Asymmetric k-means clustering of the asymmetric self-organizing map. *Neural Processing Letters*, *43*(1), 231-253. DOI: 10.1007/s11063-015-9415-8

Rajchl, M., Baxter, J. S., McLeod, A. J., Yuan, J., Qiu, W., Peters, T. M., & Khan, A. R. (2016). Hierarchical max-flow segmentation framework for multi-atlas segmentation with Kohonen self-organizing map based Gaussian mixture modeling. *Medical image analysis*, *27*, 45-56. DOI: 10.1016/j.media. 2015.05.005

Sornlertlamvanich, V. (1993). Word segmentation for Thai in machine translation system. *Machine Translation, National Electronics and Computer Technology Center*, 50-56. Bangkok.

Steinbach, G. K. M., & Kumar, V. (2000). A comparison of document clustering techniques. In *KDD Workshop on Text Mining*.

Sukhahuta, R., & Smith, D. (2000). Information extraction for Thai documents. In *IRAL '00 Proceedings of the fifth international workshop on on Information retrieval with Asian languages*, Hong Kong, China. September 30 - October 01, 2000, pp. 103-110. DOI: 10.1145/355214.355229

Tan, A. (1999). Text mining: The state of the art and the challenges. In *Proceedings of the Pacific Asia Conf on Knowledge Discovery and Data Mining PAKDD'99 workshop on Knowledge Discovery from Advanced Databases (KDAD'99)*, page 65-70.

Theeramunkong, T., Sornlertlamvanich, V., Tanhermhong, T., & Chinnan, W. (2000). Character-cluster based Thai information retrieval. In *Proc. of the 5th Int. Workshop on Information Retrieval with Asian Languages*. Hong Kong, pp.75-80.

Wieger, L. (1965). Chinese characters: Their origin, etymology, history, classification and signification: A thorough study from Chinese documents. *The Catholic Mission Press*.

Willett, P. (1988). Recent trends in hierarchic document clustering: a critical review. *Information Processing and Management*, *24*(5), 577-597.

Williams, H. E., & Zobel, J. (2002). Indexing and retrieval for genomic databases. *In IEEE Transaction on Knowledge and Data Engineering*, *14*(1), 63-78. DOI: 10.1109/69.979973

Yang, H. C., Lee, C. H., & Hsiao, H. W. (2015). Incorporating self-organizing map with text mining techniques for text hierarchy generation. *Applied Soft Computing*, *34*(C), 251-259. DOI: 10.1016/j.asoc.2015.05.005

Zhao, Y., & Karypis, G. (2002). Comparison of agglomerative and partitional document clustering algorithms. (No. TR-02-014). Minnesota University, Minneapolis, Department of Computer Science

.