Applying Table based AHC Algorithm to Semantic Word Clustering

Taeho Jo
President
Alpha AI Publication
Cheongju, South Korea
tjo018@naver.com

***Abstract*—This article proposes the modified AHC (Agglomerative Hierarchical Clustering) algorithm which clusters tables, instead of numerical vectors, as the approach to the word clustering. The motivations of this research are the successful results from applying the table based algorithms to the text clustering tasks in previous works and the expectation of synergy effect between the text clustering and the word clustering. In this research, we define the similarity metric between tables representing words, and modify the AHC algorithm by adopting the proposed similarity metric as the approach to the word clustering. The proposed AHC algorithm is empirically validated as the better approach in clustering words in news articles and opinions. In using the table based AHC algorithm, it is easier to trace results from clustering words.**

***Keywords*-Word Clustering, Table Similarity, Table based AHC**

I. Introduction

Word clustering refers to the process of segmenting a group of words into subgroups of similar words. Words are clustered based on their lexical similarities based on their spellings or their semantic ones based on their meanings. The scope of this research is restricted to the latter where words are clustered based on their meanings. Texts in the given corpus are features of representing words, and a similarity between words is computed based on their collocations within each text. Even if various types of clustering are available, this research focus on only hard clustering where each item is allowed to only one cluster.

Let us consider the issues with which this research tries to tackle in encoding words into numerical vectors. When we use texts as features of numerical vectors representing words, we need many features for implementing the robust word clustering systems[1]. Since almost numerical vectors which represent words have their zero values dominantly, what they called the sparse distribution, they have very little discrimination for computing their distances[22]. When we use the grammatical properties as features, implementation of encoding process becomes very difficult and complicated. Therefore, this research attempts to solve the problems by encoding words into tables, instead of numerical vectors.

Let us mention some agenda which are proposed by this research, in order to solve the above problems. Each word is encoded into a table which consists of entries of texts including it and its weights. We define the similarity measure between tables which is always given a normalized value, as the operation on tables. Using the similarity measure, we modify the AHC (Agglomerate Hierarchical Clustering) algorithm into the table based version where each object is given as a table. Therefore, in this research, we apply the modified AHC algorithm to the text clustering as its approach.

We will mention some benefits which are expected from this research. The tables which represent texts may be regarded as the more compact representations than numerical vectors, since the system robustness can be maintained with much less than elements. We expect more discrimination among tables than among numerical vectors in computing the similarity between them, because the sparse distribution is not available in tables. Because of the normalized similarity measure, from the proposed AHC version, we obtain both its better performance and more stability. However, note that in the proposed system, the table size is given as the external parameter, and it should be optimized between the reliability and the computation speed.

Let us mention the organization of this research. In Section II, we explore the previous works which are relevant to this research. In Section III, we describe in detail what we propose in this research. In Section IV, we validate empirically the proposed approach by comparing it with the traditional one. In Section V, we mention the significance of this research and the remaining tasks as the conclusion.

II. Previous Works

This section is concerned with the previous works which are relevant to this research. In Section II-A, we explore the previous cases of applying the AHC algorithm to text mining tasks. In Section II-B, we survey the schemes of encoding texts or words into structured data. In Section II-C, we describe the previous machine learning algorithms which receive alternative structured data such as tables and string vectors to numerical vectors. Therefore, in this section, we provide the history about this research, by surveying the relevant previous works.

A. Application to Word Clustering Tasks

This section is concerned with the previous cases of applying the modern version of AHC algorithm to clustering tasks. We will present the previous cases of applying the modernized AHC algorithm to semantic word clustering. We will also explore ones on using the AHC version for clustering texts based on their contents. We will mention the clustering evaluation measure called clustering index and cases of using it for evaluating clustering results. This section is intended to explore previous cases of applying the modern version to the word clustering and the text clustering.

Let us survey on the previous works where the modernized AHC algorithms were applied to the semantic word clustering. The similarities among features were considered for modernizing the AHC algorithm as the approach to the word clustering [5]. The words were encoded into string vectors, instead of numerical vectors, in using the AHC algorithm for the word clustering, in order to avoid the problems in encoding them into numerical vectors [7]. It was proposed that the AHC algorithm should be modernized where a graph is received as its input data [8]. In the above literatures, the cases of using the modernized versions of AHC algorithm for the word clustering are presented.

A word may expanded into a text which consists of more than one paragraph as clustering targets. The AHC algorithm which adopts the similarity metric which considers the similarities among features was used for clustering texts [17]. The AHC algorithm which clusters string vectors, directly, was proposed as the approach to the text clustering, as well as the word clustering [18]. Another modernized AHC algorithm which processes graphs directly was applied to the text clustering [19]. We will consider the association of two types of clustering in future research.

In this research, we use the clustering index as the evaluation metric of clustering results. The clustering index was initially proposed for evaluating the dynamic document organization system by Jo in 2006 [2]. The clustering index was described in detail in [24]. The clustering index was also mentioned for tuning the parameters in using a clustering algorithm in [20]. The clustering index is the metric which integrate the intra-cluster similarity and the inter-cluster discrimination, following the style of the F1 measure.

Let us mention some distinguished points of this research from the works which were mentioned, above. We explored the cases of applying the modernized versions of AHC algorithm to the word clustering and the text clustering. We presented the historical notes about the clustering index which is used as the evaluation metric in this study. In this research, we propose as the approach to the word clustering, the AHC algorithm which was modernized into the version which clusters table directly based on their similarities. The words are encoded into tables and the proposed version is applied to the word clustering tasks, for validate its performance, empirically.

B. Word and Text Encoding

This section is concerned with the previous works on encoding words or texts into non-numerical vectors. Some problems in encoding them into numerical vectors were discovered in previous works. So the works challenged against the problems by encoding them into other types of structured data. We mention the tables, the string vectors, and the graphs as alternatives to numerical vectors. This section is intended to survey previous cases of encoding texts or words into one of the three types.

Let us mention the previous cases of encoding texts or words into tables in modernizing other machine learning algorithms. Words were encoded into tables in applying the KNN algorithm for categorizing semantically words [11]. They were encoded so in using it for the keyword extraction [12]. Texts were encoded into tables in using it for the text categorization [13]. In the above literatures we presented the cases of encoding texts or words into tables in modernizing it.

Let us consider encoding words or texts into string vectors for modernizing the machine learning algorithms. Words were encoded into string vectors for applying the KNN algorithm to the word categorization [14]. Words were encoded into string vectors for applying the KNN algorithm for extracting keywords from a text [15]. In order to apply the KNN algorithm to the text categorization, texts were encoded into string vectors [16]. The previous cases of encoding words or texts into string vectors are presented in the above literatures.

Let us mention the cases on encoding words or texts into graphs. It was proposed that words should be encoded into graphs, in applying the KNN algorithm to the topic based word classification [9]. It was proposed that words should be also encoded so in applying it to the keyword extraction [10]. It was proposed that texts should be encoded into graphs, in applying it to the text categorization [21]. In the above literatures, we present the previous cases of encoding raw data into graphs.

We mentioned the three schemes of encoding words or texts in other tasks. We adopt the first scheme where words are encoded into tables, in this study. We define the similarity metric between tables, and modify the AHC algorithm into the version which clusters tables directly. The modified version of AHC algorithm is used for implementing the semantic word clustering system. We validate empirically the modified version by comparing it with the traditional version, in clustering words.

C. Non-Numerical Vector based Clustering Algorithms

This section is concerned with the previous works on non-numerical vector based clustering algorithms. In the previous section, we explored the previous cases of encoding words

or texts into alternative structured form to numerical vectors. In this section, we mention the string kernel clustering algorithm, the table matching algorithm, and the Neural Text Self Organizer, as the clustering algorithms which process non-numerical vectors directly. Because the text clustering is relevant to the word clustering, in this section, we focus on the text clustering in surveying the previous works. This section is intended to survey the previous works which are involved in one of the three clustering algorithms which are mentioned above.

Let us consider using the string kernel for clustering texts. It was initially proposed as a kernel function of SVM (Support Vector Machine) by Lodhi et al in 2002 [26]. Subsequently, it was used for modifying the k means algorithm as the approach to the text clustering by Karatzoglou and Feinerer in 2006 [25]. The spectral clustering algorithm was modified using the string kernel by Shi et al. in 2010 [27]. In the above literatures, we presented the cases of using the string kernel for the text clustering as well as the text classification.

Let us mention the table based matching algorithm as another type of approach to the text categorization. It was initially proposed as a method of categorizing texts by Jo and Cho, in 2008 [22]. It was applied to fuzzy classification of texts which allows to assign more than one category to each text by Jo in 2008 [3]. It was upgraded into the more robust and stable approach to the text categorization by Jo in 2015 [6]. In using the table based matching algorithm which is mentioned in the above literatures, texts should be encoded into tables.

Let us mention the neural network model, Neural Text Self Organizer, which was specialized for the text clustering. It was initially proposed as the approach to the text clustering by Jo and Japkowicz, in 2005 [23]. It was mentioned by surveying text clustering methods by Zheng et al. in 2006 [28]. Its clustering performance was confirmed by comparing it with the Kohonen Networks in the text clustering by Jo in 2010 [4]. Texts should encoded into string vectors in using the Neural Text Self Organizer.

We mentioned the two clustering algorithms and one classification algorithm as non-numerical vector based ones. The string kernel based clustering algorithm clusters raw texts directly, the table based matching algorithm classifies tables, and the Neural Text Self Organizer clusters string vectors. In this research, words are encoded into tables like the second non-numerical vector based one. The AHC algorithm is modified into the version which clusters tables directly as the approach to the semantic word clustering. The modified version is empirically validated in the semantic word clustering tasks, compared with the traditional version.

III. Proposed Approach

This section is concerned with the table AHC (Agglomerative Hierarchical Clustering) algorithm as the approach to the word clustering tasks. In Section III-A, we describe the process of encoding words into tables, which is called text preprocessing. In Section III-B, we cover the scheme of computing a similarity between two tables into a normalized value between zero and one. In Section III-C, we mention the proposed version of AHC algorithm as the word clustering tool, and in Section III-D, present the architecture of the system which we try to implement by adopting the proposed AHC algorithm.. Therefore, this article is intended to describe the encoding scheme, the similarity computation method, and the proposed version of AHC algorithm for implementing the word clustering systems.

A. Word Encoding

This section is concerned with the transformation of words into tables. We surveyed the previous cases of converting texts into tables in text mining tasks in Section II-B and II-C. In this section, we will mention the three steps which are presented in Figure 1-3, as the process of encoding words into tables. In the table representing a word, each entry is given as a pair of text identifier and weight. This section is intended to describe the three steps which are involved in encoding so.

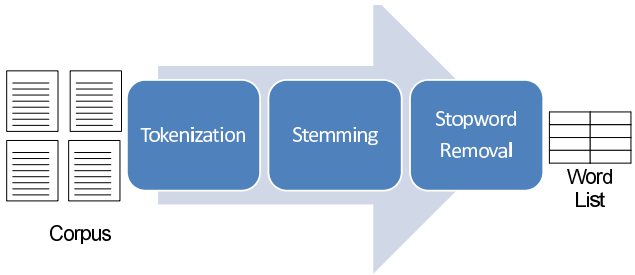

Figure 1. Overall Process of Word Indexing

The process of indexing a corpus into a list of words as the first step of encoding words into tables is illustrated in Figure 1. The tokenization is the segmentation of a text into tokens by white spaces and punctuation marks. The stemming is the transformation of each token into its root form; change of plural form of a noun into its singular form, for example. The stop-word removal is the process of excluding grammatical words such as conjunctions and prepositions from the token list. Nouns, verbs, and adjectives are usually remaining words from the process.

The inverted index where each word is linked to its related texts is illustrated in Figure 2. Each text is indexed into a list of words in the corpus in the previous step. In this step, the structure where each text is linked into its associated words is converted into one which is illustrated in Figure 2. The fact that each word is given as axis, instead of each text, is the reason of calling what is presented in Figure 2 inverted index. A list of texts becomes the information about the word for encoding it into a table.

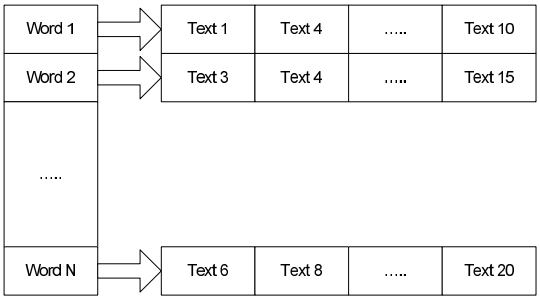

Figure 2. Inverted Index

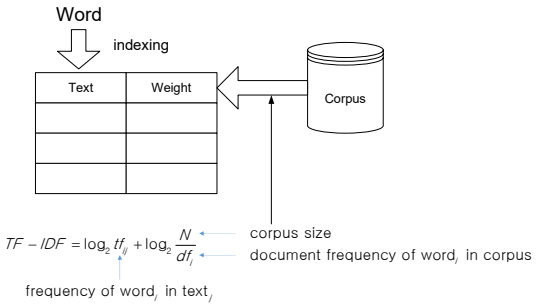

Figure 3. Text Weighting

The process of weighting text identifiers in the table which represents a word is illustrated in Figure 3. In the previous step, the inverted index where each word is linked to its relevant texts was constructed. For each text identifier, the TF-IDF (Term Frequency and Inverse Document Frequency)is computed by the equation which is presented in Figure 3. The TF-IDF weight is proportional to the frequency of the word in the given text, but reversely proportional to the number of texts which include itself in the corpus, called document frequency. The corpus is required for computing the TF-IDF weight.

The three steps for encoding a word into a table are presented in Figure 1-3. If constant weights are assigned to all text identifiers, the table may be viewed as a set of text identifiers. The TF-IDF weight is adopted for discriminating texts which are related with the word in this research. Each entry in the table is expanded into one with a text identifier and its multiple weights by adopting the multiple weighting schemes. We need to define operations on tables for modifying the machine learning algorithms into the versions which process them directly..

B. Table Similarity

This section is concerned with the similarity metric between tables. In the previous section, we described the process of encoding words into tables. We need to define the similarity metric between tables as an operation for modifying the AHC algorithm into the version which clusters tables directly. A table is expressed as set of entries each of which consists of a text identifier and a weight and the similarity between tables is computed based on their shared text identifiers. This section is intended to describe the process of computing a similarity between tables.

Let us mention the function of a table for mapping a table into an item set. A table is expressed as a set of entries as shown in equation (1),

$$T = \{(text\_id_1, weight_1), (text\_id_2, weight_2), \ldots, (text\_id_{|T|}, weight_{|T|})\} \quad (1)$$

where $text\_id_i$ is a text identifier which include the word and $weight_i$ is its weight in the text identified by $text\_id_i$. The table function is defined for generating a list of text identifiers as expressed in equation (2),

$$F(T) = \{text\_id_1, text\_id_2, \ldots, text\_id_{|T|}\} \quad (2)$$

The elements in the set, $F(T)$, is given text identifiers which include the word which is represented by the table, T. The function will be used for computing the similarity between two tables.

Let us mention the process of computing the similarity between two tables which represent words. The two tables are expressed as two sets of entries in equation (3) and (4),

$$T_1 = \{(text\_id_{11}, weight_{11}), (text\_id_{12}, weight_{12}), \ldots, (text\_id_{1|T|}, weight_{1|T|})\} \quad (3)$$

$$T_2 = \{(text\_id_{21}, weight_{21}), (text\_id_{22}, weight_{22}), \ldots, (text\_id_{2|T|}, weight_{2|T|})\} \quad (4)$$

The two tables are mapped into the sets of text identifiers which are shown in equation (5) and (6), by applying the table function to equation (3) and (4),

$$F(T_1) = \{text\_id_{11}, text\_id_{12}, \ldots, text\_id_{1|T|}\} \quad (5)$$

$$F(T_2) = \{text\_id_{21}, text\_id_{22}, \ldots, text\_id_{2|T|}\} \quad (6)$$

The set of shared text identifiers which is shown in equation (7) is obtained by applying the intersection to equation (5) and (6),

$$F(T_1) \cap F(T_2) = \{stext\_id_1, stext\_id_2, \ldots, stext\_id_k\} \tag{7}$$

The shared table is constructed by taking their weights from the two tables, T_1 and T_2 as shown in equation (8),

$$\begin{aligned} ST = \{&(stext\_id_1, weight_{11}, weight_{21}), \ldots, \\ &(stext\_id_k, weight_{1k}, weight_{2k})\} \end{aligned} \tag{8}$$

In equation (8), $weight_{1i}$ indicates the weight from the table, T_1, and $weight_{2i}$ indicates the weight from the table, T_2 to the text identifier, $stext\_id_1$.

Let us mention the process of computing the similarity between two tables after extracting the shared entries. The weights of the two tables are given as sums of entry weights, as expressed in equation (9) and (10),

$$W(T_1) = \sum_{i=1}^{|T_1|} weight_{1i} \tag{9}$$

$$W(T_2) = \sum_{i=1}^{|T_2|} weight_{2i} \tag{10}$$

The dual weight sums in the shared table, ST, are defined as equation (11) and (12),

$$W_1(ST) = \sum_{i=1}^{k} sweight_{1i} \tag{11}$$

$$W_2(ST) = \sum_{i=1}^{k} weight_{2i} \tag{12}$$

The similarity between the tables, T_1 and T_2 is computed by equation (13),

$$sim(T_1, T_2) = \frac{W_1(ST) + W_2(ST)}{W(T_1) + W(T_2)} \tag{13}$$

The similarity between tables is always given as normalized value between zero and one.

Above, we mentioned the similarity between two tables as a normalized value between zero and one. If the two tables are identical to each other as shown in equation (14),

$$T_1 = T_2 \tag{14}$$

the similarity between them is 1.0, as shown in equation (15),

$$\begin{aligned} sim(T_1, T_2) &= \frac{W_1(ST) + W_2(ST)}{W(T_1) + W(T_2)} \\ &= \frac{2W_1(ST))}{2W(T_1)} = \frac{2W_1(T_1))}{2W(T_1)} = 1.0 \end{aligned} \tag{15}$$

If the two tables are completely different from each other as shown in equation (16),

$$F(T_1) \cap F(T_2) = \oslash, |ST| = 0 \tag{16}$$

the similarity between them is zero, as shown in equation (17),

$$\begin{aligned} sim(T_1, T_2) &= \frac{W_1(ST) + W_2(ST)}{W(T_1) + W(T_2)} \\ &= \frac{0}{W(T_1) + W(T_2)} = 0.0 \end{aligned} \tag{17}$$

The similarity between two tables is given as a normalized value between zero and one, as shown in equation (18),

$$\begin{aligned} &ST \subseteq T_1, ST \subseteq T_2 \\ &W_1(ST) + W_2(ST) \leq W(T_1) + W(T_2) \end{aligned} \tag{18}$$

The similarity threshold is set between zero and one in modifying machine learning algorithms using the operation.

C. Proposed Version of AHC Algorithm

This section is concerned with the proposed version of AHC algorithm as the approach to the semantic word clustering, which is illustrated in Figure 4. The process of encoding words into tables was described in Section III-A, and words in the group are assumed to be given as tables. The similarity metric between tables which is described in Section III-B is used for computing the similarity between clusters, in proceeding the AHC algorithm. Some variants may be derived from the AHC algorithm by considering various schemes of computing the cluster similarities and merging clusters. This section is intended to describe the proposed version of AHC algorithm which clusters tables directly and its variants.

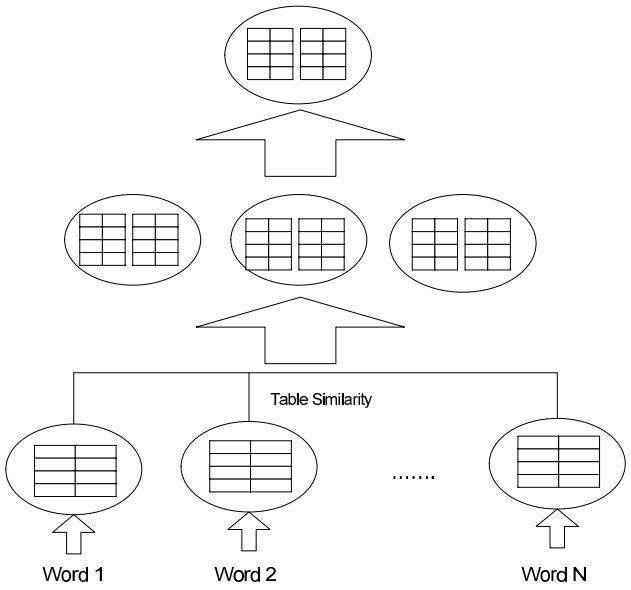

Figure 4. Proposed Version of AHC Algorithm

Let us mention the computation of the similarity between two clusters. The two clusters are notated by sets of tables: $C_1 = \{T_{11}, T_{12}, \ldots, T_{1|C_1|}\}$ and $C_2 = \{T_{21}, T_{22}, \ldots, T_{2|C_2|}\}$. All possible pairs of tables are generated from the two clusters, and for each pair, its similarity is computed by the equation which was defined in Section III-B. The similarity between the two clusters is computed by equation (19),

$$sim(C_1, C2) = \frac{1}{|C_1||C_2|} \sum_{i=1}^{|C_1|} \sum_{j=1}^{|C_2|} sim(T_{1i}, T_{2j}) \tag{19}$$

The similarity between two tables is always given as a normalized value between zero and one, so the similarity between two clusters which is computed by equation (19) is also given as a normalized value.

Let us mention the process of clustering data items by the AHC algorithm. The tables which are mapped from words in the group are notated by the set, $\{T_1, T_2, \ldots, T_N\}$, and the set of initial clusters is expressed as $\{C_1^1, C_2^1, \ldots, C_{N_1}^1\}$, where $C_i = \{T_i\}$, the super script 1 means the initial iteration, and $N_1 = N$ which is the number of clusters in the first iteration. All possible pairs of clusters, $Pair(C_i^k, C_j^k), i < j$, are generated, and the similarity between two clusters $sim(C_i^k, C_j^k)$ is computed for each pair by equation (19). Clusters in the pair with the maximal similarity are merged into a cluster as shown in equation (20),

$$\begin{aligned} Pair_{\max}(C_i^k, C_j^k) &= \underset{i<j}{\overset{N_k}{\operatorname{argmax}}}\, sim(C_i^k, C_j^k) \\ C_{merge}^{k+1} &= merge(Pair_{\max}(C_i^k, C_j^k)) \end{aligned} \tag{20}$$

and the number of clusters in the $k+1$ th iteration is $N_{k+1} = N_k - 1$ by decrementing the number of clusters by merging it. The AHC algorithm proceeds clustering by iterating the computation of similarities between clusters in all possible pairs and merge of pair with the maximal similarity into one cluster.

Let us mention the clustering index which is used for evaluating the traditional version and the proposed one of the AHC algorithm. The intra-cluster similarity of the cluster, C_i, and the inter-cluster similarity of the two clusters, C_i and C_j are notated respectively by $intra\_sim(C_i)$ and $inter\_sim(C_i, C_j)$ and the clustering results are expressed as a set of clusters, $C = \{C_1, C_2, \ldots, C_{|C|}\}$. The intra-cluster similarity over the clustering results,C, is computed by equation (21),

$$intra\_sim(C) = \frac{1}{|C|} \sum_{i=1}^{|C|} intra\_sim(C_i) \tag{21}$$

and the inter-cluster similarity over entire cluster, C is computed by equation (22),

$$inter\_sim(C) = \frac{2}{|C|(|C|-1)} \sum_{i<j}^{|C|} inter\_sim(C_i, C_j) \tag{22}$$

The clustering index is computed by equation (23),

$$CI(C) = \frac{2 \cdot intra\_sim(C) \cdot (1 - inter\_sim(C))}{intra\_sim(C) + (1 - inter\_sim(C))} \tag{23}$$

The desired goal of clustering data items is to maximize the intra cluster similarity and minimize the inter cluster similarity.

We described the proposed version of the AHC algorithm as the approach to the data clustering. Raw data is encoded into tables for using the proposed version for clustering data items. We use the similarity metric between tables for computing similarities among items. The similarity between clusters is the average over all possible similarities of data items. The desired number of clusters is set as the termination condition in proceeding clustering by the AHC algorithm.

D. Word Clustering System

This section is concerned with the semantic word clustering system which adopts the table based AHC algorithm. In Section III-C, we described the AHC algorithm which clusters tables directly. The main functions of this system are to encode words into tables and to cluster them semantically. Data items are clustered by iterating computing similarities among clusters and merge two clusters into one. This section is intended to describe the semantic word clustering system with respect to its functions and architecture.

The words are gathered as clustering targets. Because unsupervised learning algorithms are used for clustering data, the words are assumed to be unlabeled. The words are encoded into tables by the process which was mentioned in Section III-A. The similarity metric which is described in Section III-B is defined and the AHC algorithm which is described in Section III-C is adopted as the clustering method. The number of clusters should be set as the termination condition in the system.

The entire architecture of the proposed word clustering system is illustrated in Figure 5. All words which are given as the input are encoded into tables. They are clustered by the AHC algorithm which was described in Section III-C in the similarity computation module and the clustering module. The table clusters are restored into the word clusters by the decoder. There are the four modules in the system: the encoding module, the similarity computation module, the clustering module, and the decoding module.

The execution process of the proposed system is illustrated as a block diagram in Figure 6. The words which are clustered are encoded into tables by the encoding module.

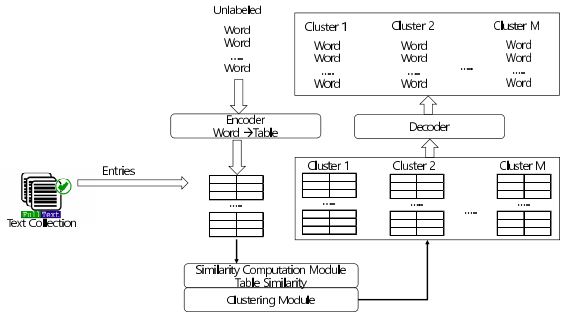

Figure 5. Proposed System Architecture

The tables are clustered by the AHC algorithm by iterating computing the similarity among table clusters and merging clusters. Clusters each of which contain semantic similar words are given as the final output in the system. In advance we need to decide the number of clusters as an external parameter.

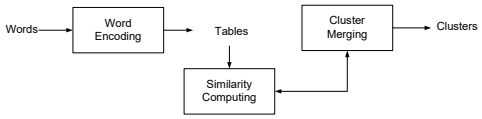

Figure 6. Execution Process of Proposed System

Let us make some remarks on the proposed system which is illustrated in Figure 5 as the architecture. Words are encoded into tables, instead of numerical vectors. Tables which represent words are clustered by the proposed AHC algorithm, directly. The clustering performance is improve by what is proposed in this research as shown in Section IV. In the next research, we present the graphical user interface and the source codes which are necessary for implementing the system as a complete one.

IV. Experiments

This section is concerned with the empirical experiments for validating the proposed version of AHC algorithm, and consists of the five sections. In Section IV-A, we present the results from applying the proposed version of AHC to the word clustering on the collection, NewsPage.com. In Section IV-B, we show the results from applying it for clustering words from the collection, Opinosis. In Section IV-C and IV-D, we mention the results from comparing the two versions of AHC algorithm with each other in clustering words from 20NewsGroups.

A. NewsPage.com

This section is concerned with the experiments for validating the better performance of the proposed version on the collection: NewsPage.com. We set the number of clusters as four, following the number of categories for evaluating the performance, and gather words from the collection, category by category, as the labeled ones. In the clustering process, each word is arranged into one of the four clusters, exclusively, in this set of experiments. We use the clustering index which was proposed in [2] for evaluating the clustering performances. Therefore, this section is intended to observe the performance of the traditional and proposed versions of AHC algorithm with different input sizes.

In Table I, we specify NewsPage.com as the text collection which is used as the source for extracting classified words, in this set of experiments. The text collection, NewsPage.com, was also used for evaluating approaches to text categorization, in previous works [5]. We extract the 300 important words from each topic for building the collection of classified words for evaluating the approaches to word clustering. We segment the entire collection which consists totally of 1200 words into the four subgroups, depending on their semantic similarities. In each category, words are selected by their frequencies concentrated on the given topic combined with subjectivity, from the text collection.

Table I
The Number of Texts and Words in NewsPage.com

Category	#Texts	#Words
Business	500	300
Health	500	300
Internet	500	300
Sports	500	300
Total	2000	1200

Let us mention the experimental process for validating empirically the proposed approach to the task of word clustering. We extract the important words from each category in the above text collection, and encode them into numerical vectors and tables. The 1200 examples are clustered into the four clusters by the both versions of AHC algorithm. We use the clustering index which combines the two measures, the intra-cluster similarity and the inter-cluster similarity,

for evaluating the both versions. The clustering index is described in detail in [24], and used previously for evaluating the clustering algorithms [2].

In Figure 7, we illustrate experimental results from clustering words using the both versions of AHC algorithm. The y-axis indicate the clustering index and is the measure for evaluating the clustering results. In the x-axis, each group indicates the input size as the dimension of numerical vectors which represent words. In each group, the gray bar and the black bar indicate the results of the traditional version and the proposed version of AHC algorithm, respectively. The most right group in Figure 7 indicates the average aver the results of the left four groups.

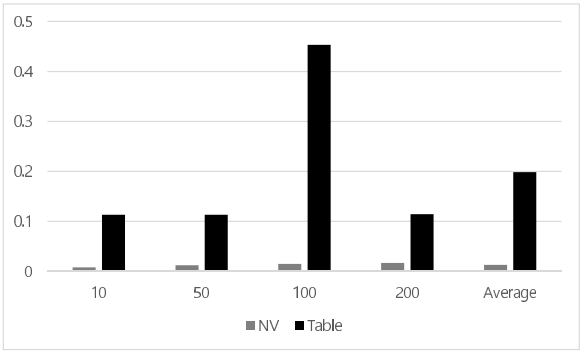

Figure 7. Results from Clustering Words in Text Collection: News-Page.com

Let us make the discussions on the results from doing the word clustering, using the both versions of AHC algorithm, as shown in Figure 7. In the proposed version of AHC algorithm, the clustering index which is the performance measure of these clustering tasks is in the range between 0.1 and 0.45. The proposed version of the AHC Algorithm works much better in the all input sizes, as shown in Figure 7. The reason of the better performance is the improved discriminations among representations of words, by encoding words into tables as alternative structured forms to numerical vectors. From this set of experiments, we conclude that the proposed version works much better than the traditional one, in averaging over the four cases.

B. Opinopsis

This section is concerned with the set of experiments for validating the better performance of the proposed version: Opniopsis. In this set of experiments, the three categories are predefined in the collection, and we collect words category by category as the classified ones. A group of words is exclusively segmented into the three clusters. In this set of experiments, we also use the clustering index. Therefore, in this section, we observe the performances of the both versions of AHC algorithm with the different input sizes on another collection.

In Table II, we illustrate the text collection, Opinosis, which is used as the source for extracting the classified words, in this set of experiments. The collection, Opinosis, was used in previous works for evaluating approaches to text categorization. We extract the 300 important words from each topic as the collection of classified words, for evaluating the approaches to word clustering. The group of totally 900 words is segmented into the three subgroups by the clustering algorithms, according to the number of the predefined categories. The words are extracted by both their frequencies which are concentrated in their own categories, in this set of experiments.

Table II
THE NUMBER OF TEXTS AND WORDS IN OPINIOPSIS

Category	#Texts	#Words
Car	23	300
Electronic	16	300
Hotel	12	300
Total	51	900

We perform this set of experiments by the process which is described in section IV-A. We extract the 300 important words by scanning individual texts in each category, and encode them into numerical vectors and tables, with the input sizes: 10, 50, 100, and 200. The group of total 900 examples is clustered by the both versions of AHC algorithm into the three clusters, using the cosine similarity and the proposed one. In this set of experiments, we use also the clustering index which combines the intra-cluster similarity and the inverse inter-cluster similarity with each other, for evaluating the both versions. We adopted the external evaluation where the labeled examples are used for evaluating clustering algorithms which is mentioned in [4].

In Figure 8, we illustrate the experimental results from clustering words using the both versions of AHC algorithm. Like Figure 7, the y-axis indicates the value of clustering index, and x-axis indicates the group of the two versions of AHC algorithm by an input size. In each group, the grey bar and the black bar indicate the achievements of the traditional version and the proposed on of AHC algorithm. In Figure 8, the most right group indicates the averages over the achievements of both versions of the left four groups. Therefore, Figure 8 shows the results from clustering words into the three subgroups by both versions, on the collection: Opinosis.

We discuss the results from doing the word clustering, using the both versions of AHC algorithm, on Opinosis, shown in Figure 8. The values of clustering index of both versions range between less than 0.1 and 0.75. The proposed version of AHC algorithm works better than the traditional ones in all input sizes. The reason of its better performance is the improved discriminations among tables as alternative representations of words to numerical vectors. From this set of experiments, we conclude that the proposed one works

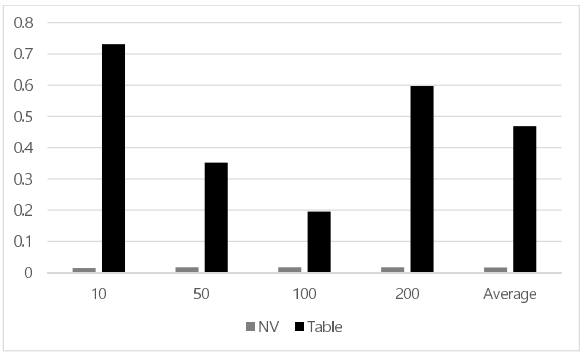

Figure 8. Results from Clustering Words in Text Collection: Opiniopsis

outstandingly better in averaging over the four cases.

C. 20NewsGroups I: General Version

This section is concerned with one more set of experiments for validating empirically the better performance of the proposed version on the text collection: 20NewsGroups I. In this set of experiments, we predefine the four general categories and gather words from the collection category by category as the classified ones. The task of in this set of experiments is to cluster the gathered words into the four clusters based on their semantic similarities, exclusively. The both versions of AHC algorithm are evaluated by the clustering index, like the previous set of experiments. Therefore, in this section, we observe the performances of the both versions with the different input sizes.

In Table III, we specify the general version of 20NewsGroups which is used for evaluating the two versions of AHC algorithm. In 20NewsGroup, the hierarchical classification system is defined with the two levels; in the first level, the six categories, alt, comp, rec, sci, talk, misc, and soc, are defined, and among them, the four categories are selected, as shown in Table III. In each category, we select 1000 texts at random, and extract 300 important words from them as the labeled words. In the process of gathering the classified words, they are selected by their frequencies which are concentrated in their corresponding categories. Therefore, following the external evaluation, we use the classified words for evaluating clustering results.

Table III
THE NUMBER OF TEXTS AND WORDS IN 20NEWSGROUPS I

Category	#Texts	#Words
Comp	1000	300
Rec	1000	300
Sci	1000	300
Talk	1000	300
Total	4000	1200

The experimental process is identical is that in the previous sets of experiments. In each category, we extract the 300 important words and encode them into numerical vectors and tables with the input sizes, 10, 50, 100, and 200. The totally 1200 words are clustered by the two versions of AHC algorithm, based on their similarities. We use the clustering index which combines the intra-cluster similarity and the inverse inter-cluster similarity with each other, for evaluating the both versions, identically to the previous sets of experiments. We use the labeled words and their target labels are hidden during clustering process.

In Figure 9, we illustrate the experimental results from clustering the words using the both versions of AHC algorithm on the broad version of 20NewsGroups. Figure 9 has the identical frame of presenting the results to those of Figure 7 and 8. In each group, the gray bar and the black bar indicates the achievements of the traditional version and the proposed version of AHC algorithm, respectively. This figure presents the results from clustering words into the four clusters by changing their input sizes. We adopt the external evaluation as the paradigm of evaluating the clustering results, in this set of experiments.

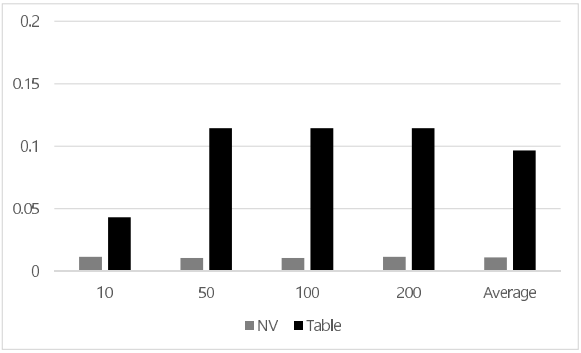

Figure 9. Results from Clustering Words in Text Collection: 20NewsGroup I

Let us discuss the results from doing the word clustering using the both versions of AHC algorithm on the broad version of 20NewsGroups, as shown in Figure 9. The clustering indices of the both versions range between less than 0.1 and 0.12. The proposed version shows the much better results in all of the input sizes. The reason of the better results is the improved discrimination among word representations. From this set of experiments, we conclude the proposed version win completely over the traditional one, in averaging their four achievements.

D. 20NewsGroups II: Specific Version

This section is concerned with one more set of experiments where the better performance of the proposed version is validated on another different version of 20NewsGroups. In this set of experiments, the four specific categories are predefined and words are gathered from each topic as the classified ones. The task of this set of experiments is to cluster exclusively words into four clusters. We use the clustering index like the previous sets of experiments as the

evaluation metric. Therefore, in this section, we observe the performances of the both versions of AHC algorithm, with the different input sizes.

In Table 4, we specify the second version of 20News-Groups which is used in this set of experiments. Within the general category, sci, the four categories, electro, medicine, script, and space, are predefined. We build the collection of labeled words by extracting the 300 important words from approximately 1000 texts in each specific category. In this set of experiments, the group of 1,200 words is clustered into the four groups. We use the classified words for evaluating the results from clustering them, like the case in the previous set of experiments.

Table IV
THE NUMBER OF TEXTS AND WORDS IN 20NEWSGROUPS II

Category	#Texts	#Words
Electro	1000	300
Medicine	1000	300
Script	1000	300
Space	1000	300
Total	4000	1200

The process of doing this set of experiments is same to that in the previous sets of experiments. We extract the identical number of words from all texts in each category, and encode them into numerical vectors. We cluster 1200 words by the two versions of AHC algorithm into the four clusters. We use the clustering index based on the intra-cluster similarity and inverse inter-cluster similarity, for evaluating the both versions. We evaluate the results from clustering items, using the labeled examples, following the external validity.

We present the experimental results from clustering the words using the both versions of AHC algorithm on the specific version of 20NewsGroups. The frame of illustrating the classification results is identical to the previous ones. In each group, the gray bar and the black bar stand for the achievements of the traditional version and the proposed version, respectively. The y-axis in Figure 10, indicates the clustering index which is used as the performance metric. In clustering words, each of them is allowed to belong to only one cluster like the cases in the previous sets of experiments.

Let us discuss the results from clustering the words using the both versions of AHC algorithm on the specific version of 20NewsGroups, as shown in Figure 10. The clustering indices of both versions range between less than 0.1 and 0.12. The proposed version shows its strongly better performances in the all input sized, as shown in Figure 10. The reason of the better performances is the discriminations among feature vectors which is improved by encoding words into tables, instead of numerical vectors. From this set of experiments, it is concluded that the proposed version of AHC algorithm is much feasible to the task of word clustering.

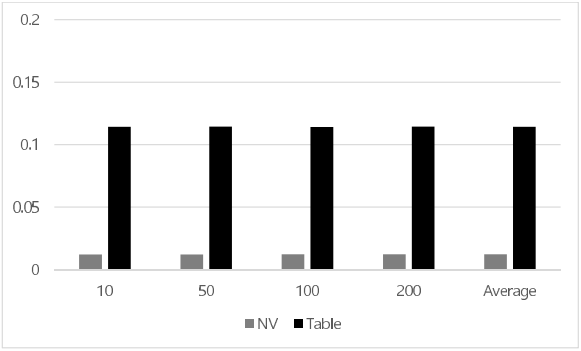

Figure 10. Results from Clustering Words in Text Collection: 20News-Group II

V. CONCLUSION

Let us discuss the entire results from clustering word using the two versions of AHC algorithm. In these sets of experiments, the traditional and proposed version are compared with each other in the tasks of word clustering. The proposed version shows the better results in all of the four collections. The clustering indices of the traditional version is always less than 0.1, while those of the proposed version range between 0.1 and 0.72. Through the four sets of experiments, we conclude that the proposed version improve the word clustering performance very strongly as the contribution of this research.

Let us consider the remaining tasks for doing the further research. We need to validate and customize the proposed research in the word clustering in one of specific domains: engineering, science, and medicine. Because various schemes of weighting words are available, more than one weight may be assigned to each word, so it need to be considered in computing the similarity between tables. Other unsupervised machine learning algorithms may be modified as well as AHC into their table based versions. By adopting the proposed approach, we implement the word clustering system as a module of other programs or an independent program.

REFERENCES

[1] L.D. Baker and A. K. McCallum , "Distributional clustering of words for text classification", pp96-103 Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval, 1998.

[2] T. Jo, "The Implementation of Dynamic Document Organization using Text Categorization and Text Clustering", PhD Dissertation of University of Ottawa, 2006.

[3] T. Jo, "Table based Matching Algorithm for Soft Categorization of News Articles in Reuter 21578", 875-882, Journal of Korea Multimedia Society, Vol 11, No 6, 2008.

[4] T. Jo, "NTSO (Neural Text Self Organizer): A New Neural Network for Text Clustering", 31-43, Journal of Network Technology, Vol 1, No 1, 2010.

[5] T. Jo, “AHC based Clustering considering Feature Similarities”, 67-70, The Proceedings of 11th International Conference on Data Mining, 2015.

[6] T. Jo, “Normalized Table Matching Algorithm as Approach to Text Categorization”, 839-849, Soft Computing, Vol 19, No 4, 2015.

[7] T. Jo, “String Vector based AHC as Approach to Word Clustering”, 133-138, The Proceedings of 12th International Conference on Data Mining, 2016.

[8] T, Jo, “Encoding Words into Graphs for Clustering Word by AHC Algorithm”, 90-95, The Proceedings of 12th International Conference on Multimedia Information Technology and Applications, 2016.

[9] T. Jo, “Graph based KNN for Content based Word Classification”, 24-29, The Proceedings of 12th International Conference on Multimedia Information Technology and Applications, 2016.

[10] T. Jo, “Extracting Keywords by Graph based KNN”, 96-101, The Proceedings of 12th International Conference on Multimedia Information Technology and Applications, 2016.

[11] T. Jo, “Table based K Nearest Neighbor for Word Categorization in News Articles”, 1214-1217, The Proceedings of 25th International Conference on Computational Science & Computational Intelligence, 2018.

[12] T. Jo, “Keyword Extraction in News Articles using Table based K Nearest Neighbors”, 1230-1233, The Proceedings of 25th International Conference on Computational Science & Computational Intelligence, 2018.

[13] T. Jo, “Modification into Table based K Nearest Neighbor for News Article Classification”, 49-50, The Proceedings of 1st International Conference on Advanced Engineering and ICT-Convergence, 2018.

[14] T. Jo, “Modification of K Nearest Neighbor into String Vector based Version for Classifying Words in Current Affairs”, 72-75, The Proceedings of International Conference on Information and Knowledge Engineering, 2018.

[15] T. Jo, “Modifying K Nearest Neighbor into String Vector based Version for Extracting Keywords from News Articles”, 43-46, The Proceedings of International Conference on Applied Cognitive Computing, 2018.

[16] Taeho Jo, ”Improving K Nearest Neighbor into String Vector Version for Text Categorization”, pp 1091-1097, ICACT Transaction on Communication Technology, Vol 7, No 1, 2018.

[17] T. Jo, “Feature Similarity AHC Algorithm for Clustering News Articles”, 49-54, The Proceedings of 2nd International Conference on Advanced Engineering and ICT-Convergence, 2019.

[18] T. Jo, “Introduction of String Vectors to AHC Algorithm for Clustering News Articles”, 150-153, The Proceedings of 21st International Conference on Artificial Intelligence, 2019.

[19] T. Jo, “Graph based Version for Clustering Texts in Current Affair Domain”, 171-174, The Proceedings of 15st International Conference on Data Science, 2019.

[20] T. Jo, “Text Mining: Concepts and Big Data Challenge”, Springer, 2019.

[21] T. Jo, “Graph based Version of K Nearest Neighbor for classifying News Articles”, 4-7, The Proceedings of International Conference on Green and Human Information Technology Part I, 2019.

[22] T. Jo and D. Cho, “Index Based Approach for Text Categorization”, 127-132, International Journal of Mathematics and Computers in Simulation, Vol 2, No 1, 2007.

[23] T. Jo and N. Japkowicz, “Text Clustering using NTSO”, 558-563, The Proceedings of Internaitonal Joint Conference on Neural Networks, 2005.

[24] T. Jo and M. Lee, “The Evaluation Measure of Text Clustering for the Variable Number of Clusters”, 871-879, Lecture Notes in Computer Science, Vol 4492, 2007.

[25] A. Karatzoglou and I. Feinerer, “Text Clustering with String Kernels in R”, 91-98, Advances in Data Analysis, 2006.

[26] H. Lodhi, C. Saunders, J. Shawe-Taylor, N. Cristianini, and C. Watkins, “Text Classification with String Kernels”, 419-444, Journal of Machine Learning Research, Vol 2, No 2, 2002.

[27] Q. Shi, X. Qiao, and X. Guangquan, ”Using String Kernel for Document Clustering”, pp40-46, I.J. Information Technology and Computer Science, Vol 2, 2010.

[28] Y. Zheng, X. Cheng, R. Huang, and Y. Man, “A Comparative Study on Text Clustering Methods”, 644-651, Advanced Data Mining and Applications, 2006.