Modification of AHC algorithm for Clustering Words into Feature Similarity based Version

Taeho Jo  
School of Game  
Hongik University  
Sejong, South Korea  
tjo018@hongik.ac.kr

Abstract—In this research, we propose the AHC version which considers the feature similarity as well as the feature value similarity, as the approach to the word clustering. Actually, features are dependent on each other in the text mining tasks, including the word clustering, and we need to reinforce both the word clustering and the text clustering by connecting them with each other. In this research, words are encoded into numerical vectors whose features are text identifiers, the similarity metric between two numerical vectors, considering the semantic similarities among features is defined, and the AHC algorithm is modified by adopting the proposed similarity metric. We adopt the clustering index as the evaluation metric, and validate empirically that the proposed AHC version is better than the traditional version. In future, we connect mutually the word clustering with the text clustering for reinforcing them at same time.

I. INTRODUCTION

The word clustering is referred to the process of segmenting a group of words into subgroups of similar words. Even if we consider the lexical clustering where words are clustered based on their spellings and the grammatical clustering where words are clustered based on their grammatical functions as word clustering, the two tasks are excluded from the research scope. The scope is restructured to the semantic clustering where words are clustered based on their topics or meaning. We mention the AHC algorithm as approach and modify it into the version which considers the feature similarity as well as the feature value similarity for computing the similarity between numerical vectors. In this section, we describe briefly the motivation, the idea, and the validation of this research.

Let us consider the motivations for doing this research. The fact that we may expect the synergy effect between the word clustering and the text clustering motivates for setting the former as the task of this research. The AHC algorithm is a simple approach to the data clustering for starting to modify machine learning algorithms. Texts which are features for encoding words into numerical vectors have their own semantic relations with other. In this research, by introducing the feature similarity as well as the feature value similarity, we expect the discriminations among numerical vectors to be improved.

In this research, we propose the modified AHC version as the approach to the word clustering. The words are encoded into numerical vectors whose features are texts, and the similarity metric between them which consider the similarities among features is defined. By adopting the proposed similarity metric, for computing the similarity between the test example and each training example, the AHC algorithm is modified. The modified version is applied to the word clustering which is covered in this research. The discriminations among numerical vectors are improved by considering the similarities among features as well as ones among feature values.

In this research, we will validate empirically the proposed approach to the word clustering as the better version than the traditional AHC version. We extract words which are classified their own topics from the news collection: 20NewsGroups. The traditional AHC version and the proposed version are compared with each other. We observe the better results of the proposed AHC version in clustering words. It potentially possible to reduce the dimension by considering the feature similarity.

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 significances of this research and the remaining tasks as the conclusion.

II. PREVIOUS WORKS

Let us survey the previous cases of encoding texts into structured forms for using the machine learning algorithms to text mining tasks. The three main problems, huge dimensionality, sparse distribution, and poor transparency, have existed inherently in encoding them into numerical vectors. In previous works, various schemes of preprocessing texts have been proposed, in order to solve the problems. In this survey, we focus on the process of encoding texts into alternative structured forms to numerical vectors. In other words, this section is intended to explore previous works on solutions to the problems.
Let us mention the popularity of encoding texts into numerical vectors, and the proposal and the application of string kernels as the solution to the above problems. In 2002, Sebastiani presented the numerical vectors as the standard representations of texts in applying the machine learning algorithms to the text classifications [4]. In 2002, Lodhi et al. proposed the string kernel as a kernel function of raw texts in using the SVM (Support Vector Machine) to the text classification [5]. In 2004, Lesile et al. used the version of SVM which proposed by Lodhi et al. to the protein classification [6]. In 2004, Kate and Mooney used also the SVM version for classifying sentences by their meanings [7].

It was proposed that texts are encoded into tables instead of numerical vectors, as the solutions to the above problems. In 2008, Jo and Cho proposed the table matching algorithm as the approach to text classification [8]. In 2008, Jo applied also his proposed approach to the text clustering, as well as the text categorization [12]. In 2011, Jo described as the technique of automatic text classification in his patent document [10]. In 2015, Jo improved the table matching algorithm into its more stable version [11].

Previously, it was proposed that texts should be encoded into string vectors as other structured forms. In 2008, Jo modified the k means algorithm into the version which processes string vectors as the approach to the text clustering [12]. In 2010, Jo modified the two supervised learning algorithms, the KNN and the SVM, into the version as the improved approaches to the text classification [13]. In 2010, Jo proposed the unsupervised neural networks, called Neural Text Self Organizer, which receives the string vector as its input data [14]. In 2010, Jo applied the supervised neural networks, called Neural Text Categorizer, which gets a string vector as its input, as the approach to the text classification [15].

The above previous works proposed the string kernel as the kernel function of raw texts in the SVM, and tables and string vectors as representations of texts, in order to solve the problems. Because the string kernel takes very much computation time for computing their values, it was used for processing short strings or sentences rather than texts. In the previous works on encoding texts into tables, only table matching algorithm was proposed; there is no attempt to modify the machine algorithms into their table based version. In the previous works on encoding texts into string vectors, only frequency was considered for defining features of string vectors. Texts which are used as features of numerical vectors which represent words have their semantic similarities among them, so the similarities will be used for processing sparse numerical vectors, in this research.

### III. Proposed Approach

This section is concerned with what we propose in this research. Words are encoded into numerical vectors and the feature similarity is considered for computing similarities among them. The AHC algorithm is modified into the version where the feature similarity is computed as well as the feature value one. The modified version is applied to the topic based word clustering. In this section, we describe what is proposed in this research.

Let us explain the process of encoding a word into a numerical vector. The texts in the corpus are given as feature candidates and among them, some are selected by their coverage to given words. The word is given as the input and its TF-IDF (Term Frequency-Inverse Document Frequency) weights or its frequencies to texts which are given as features are computed as the feature values which indicate relationship of the word with the texts. In other words, the word is represented into a numerical vector which consists of weights and frequencies. Numerical vectors which represent words or texts, tend to have their sparse distribution where zero values are dominant.

Figure 1 illustrates the outline of computing the proposed similarity metric between two numerical vectors. $d_1, d_2, \ldots, d_n$ are the text identifiers which are selected from the corpus as features and the two words, $t_1$ and $t_2$ are represented into the two numerical vectors: $t_1 = [w_{11}, w_{12}, \ldots, w_{1n}]$ and $t_2 = [w_{21}, w_{22}, \ldots, w_{2n}]$. The similarity between the two features, $d_i$ and $d_j$, is computed by Equation (1)

$$s_{ij} = sim(d_i, d_j) = \frac{2 \times tf(d_i, d_j)}{tf(d_i) + tf(d_j)}$$ (1)

where $tf(d_i, d_j)$ is the number of words which are shared by the two texts, $d_1$ and $d_2$, and $tf(d_i)$ is the number of words which are included in the text, $d_i$. The similarity between the two numerical vectors which considers the feature similarity, $s_{ij}$, is computed by Equation (2)

$$sim(t_1, t_2) = \frac{\sum_{i=1}^{n} \sum_{j=1}^{n} s_{ij} \cdot w_{1i} \cdot w_{2j}}{||t_1|| \cdot ||t_2||}$$ (2)

where $||t_1|| = \sqrt{\sum_{i=1}^{n} w_{1i}^2}$ and $||t_2|| = \sqrt{\sum_{i=1}^{n} w_{2i}^2}$. The complexity in computing the proposed similarity metric is quadratic to the dimension, $n, O(n^2)$. $f_1, f_2, \ldots, f_d$ $X = \begin{bmatrix} X_1, X_2, \ldots, X_d \end{bmatrix}$ $Y = \begin{bmatrix} Y_1, Y_2, \ldots, Y_d \end{bmatrix}$

Figure 1. The Combination of Feature and Feature Value Similarity

The proposed AHC algorithm is presented in Figure 2. The proposed system encodes the words which are given as the clustering targets into numerical vectors and starts with
singletons as many as items. It computes the similarities of all possible cluster pairs by Equation(2) and merges the most similar clusters into a single cluster. It iterates the above steps until the number of clusters is reduced to the desired one. By discriminating the similarities and the attributes, we derive AHC variants from this version.

![Figure 2. The Proposed Version of AHC Algorithm](image)

Let us make some remarks on what is proposed in this research. Even if the AHC algorithm is a very simple machine learning algorithm, it is useful for implementing a light version of clustering system. Even if it takes much time for computing the proposed similarity metric, it tackled against the poor discriminations from the sparse distribution of numerical vectors. We may use words which are called contexts as features for representing words into numerical vectors, as well as text identifiers. The proposed AHC algorithm is described in more detail in [16].

### IV. Experiments

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 20NewsGroups 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>
<thead>
<tr>
<th>Category</th>
<th>#Texts</th>
<th>#Words</th>
</tr>
</thead>
<tbody>
<tr>
<td>Electro</td>
<td>1000</td>
<td>300</td>
</tr>
<tr>
<td>Medicine</td>
<td>1000</td>
<td>300</td>
</tr>
<tr>
<td>Script</td>
<td>1000</td>
<td>300</td>
</tr>
<tr>
<td>Space</td>
<td>1000</td>
<td>300</td>
</tr>
<tr>
<td>Total</td>
<td>4000</td>
<td>1200</td>
</tr>
</tbody>
</table>

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 3, 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.

![Figure 3. Results from Clustering Words in Text Collection: 20NewsGroup II](image)
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 4. The reason of the better performances is the discriminations among feature vectors which is improved by considering the feature similarities as well as feature value ones. From this set of experiments, it is concluded that the proposed version of AHC algorithm is much feasible to the task of word clustering.

V. Conclusion
Let us mention the remaining tasks for doing the further research. We need to validate the proposed approach in specific domains such as medicine, engineering, and economics, as well as in generic domains such as ones of news articles. We may consider the computation of similarities among some main features rather than among all features for reducing the computation time. We try to modify other machine learning algorithms such as Naive Bayes, Perceptrons, and SVM (Support Vector Machine) based on both kinds of similarities. By adopting the proposed approach, we may implement the word clustering system as a real program.

REFERENCES