Computational Linguistics and Intelligent Text Processing

5th International Conference, CICLing 2004
Seoul, Korea, February 2004
Proceedings
Table of Contents

The Impact of Enriched Linguistic Annotation on the Performance of Extracting Relation Triples .................................................. 647
Sanheu Kim, Paul Lewis, Kirk Martin

Text Categorization

An iNN Model-Based Approach and Its Application in Text Categorization ................................................................. 569
Gonghe Guo, Hui Wang, David Belp, Yuxin Bh, Kieran Omer

Automatic Learning Features Using Bootstrapping for Text Categorization ................................................................. 571
Weibing Chen, Jingke Zhu, Hongwei Wu, Tianhe Xu

Reconstruction of Class Relevance Scores for Improving Text Classification ................................................................. 581
Song-Hun Kim, Hae-Chang Rim

Raising High-Degree Overlapped Character bigrams into Trigrams for Dimensionality Reduction in Chinese Text Categorization ................................................................. 584
Dezhi Xu, Mousong Sun

Information Retrieval and Text Categorization with Semantic Indexing ................................................................. 596
Paolo Russo, Antonio Sessa, Ferran Plo, Daniel Jiménez, Vicente Vidal

Document Clustering

Sampling and Feature Selection in a Genetic Algorithm for Document Clustering ................................................................. 601
Asenzo Castilla, Magno T. González de Leão, Raquel Martínez

A New Efficient Clustering Algorithm for Organizing Dynamic Data Collection ................................................................. 613
Kwon-sung Kim, Sungyung Han

Domain-Informed Topic Detection ................................................................. 617
Conrad Flynn, John Dunning

Summarization

Assessing the Impact of Lexical Chain Scoring Methods and Sentence Extraction Schemes on Summarization .... 627
William Daven, Nicola Storey, Joe Carlyle, John Dunning

A Term Weighting Method Based on Lexical Chain for Automatic Summarization ................................................................. 636
Young-In Song, Kyoung-Soo Han, Hae-Chang Rim
A New Efficient Clustering Algorithm for Organizing Dynamic Data Collection

Kwangchul Shin and Sanggyung Han*

School of Computer Science and Engineering, Chung-Ang Univ.
221 Heukseok Dong Dongjak-Ku, Seoul, Korea
kwshin@cs.cuee.com.ac.kr, hankyung@cau.ac.kr

Abstract. We deal with dynamic information organization for more efficient Internet browsing. As the appropriate algorithm for this purpose, we propose modified ART (artificial resonance theory) algorithm, which functions similarly with the dynamic Star-clustering algorithm but performs a more efficient time complexity of O(nk) instead of O(kn) found in the dynamic Star-clustering algorithm. In order to see how fast the proposed algorithm is in producing clusters for organizing information, the algorithm is tested on CLASTIC in comparison with the dynamic Star-clustering algorithm.

1 Introduction

From the very beginning of the information-oriented society era, gathering information has been very important issue. However, in current Internet environment there are too many documents, this makes users to waste time. Information organization techniques capable of automatically grouping related documents make it easy for users to recognize the contents of documents and to find what they want [1].

The most recent study on information organization deals with Star-clustering [2]. The Star-clustering algorithm presents the information system by applying the undirected, weighted similarity graph G(V, E, μ) and forms a dense subgraph O'(V, E, μ) based on G in order to organize the information. The Star-clustering algorithm also can be executed dynamically, which means each document clustered one by one. Compared to the formerly used average link or single link algorithm, the Star-clustering algorithm scored higher in the multi-precision measurement. However, in order to execute dynamic Star-clustering algorithm, the required time complexity is O(kn^2). And O(kn^2) is too much time wasting when one has a massive amount of document groups to process in real time.

In this study, we suggest a new algorithm, which retains the benefit of the Star-clustering algorithm but has complexity only O(nk), where k is the number of produced clusters. Our algorithm combines ART (artificial resonance theory) [5], a real time clustering algorithm, and concept vector [4]. By controlling the vigilance parameter in ART, we can form clusters that have certain number of documents and certain coherence.

* Work supported by the FRI of the Chung-Ang University.
** Corresponding author.

© Springer-Verlag Berlin Heidelberg 2004
2 Suggested Method

Our algorithm uses the vector space model based on the cosine similarity, which applies the concept vector $\mathbf{c}_i$ and assigns a different vigilance parameter to each cluster in order to use the dynamic vigilance parameter changes for each cluster. So in order to have newly produced clusters to have coherent vigilance parameter values, the global vigilance parameter $\rho_0$ is applied, and vigilance parameter $\rho_1$ is applied to cluster $x_i$ to control the vigilance parameter of each cluster.

**Initialization**: The number of clusters is initialized as $i$, and the input patterns (document vectors) are normalized to be formed in units of the L2 norm. Then a cluster is made with the first input pattern and the global vigilance parameter $\rho_0$ and the vigilance parameter of the first cluster $\rho_1$ is set at a value over 93.

$$w_0 = x_i, \rho_0 = 0.33, \rho_1 = \rho_0$$

Since the amount of matches between the input pattern $x_i$ and cluster $x_j$ is perceived by the cosine similarity, the number of clusters to be produced can be increased by seeing the initial vigilance parameter at a higher level.

**Activation Function (AF)**: The activation function used for measuring the compatibility between the input pattern and the weighting vector is calculated as the cosine similarity between the two vectors as shown below.

$$AF(x_j, x_i) = \cos(\theta_{x_j, x_i}) = \frac{x_j \cdot x_i}{\|x_j\| \cdot \|x_i\|}$$

Here, the weighting vector is the sum of cluster $x_j$'s input patterns:

$$w_j = \sum_{x_i} x_i$$

Unlike Kocy ART (7), the suggested algorithm does not have the Matching Function calculated. The Activation Function substitutes it. In other words, the activation function (2) also serves as the matching function. So no additional calculation is necessary for the matching function.

**Selecting the Resonance Cluster and Modulating the Vigilance Parameter**: Because the activation function takes over the place of the matching function, the resonance cluster is obtained by applying the following formulas.

$$AF(x_j, x_i) = \rho_{0 - \rho_1} \arg\max_{x_j} \{AF(x_j, x_i)\}$$

It is checked whether the weighing vector of the cluster that is the most similar to the input pattern meets the condition of the cluster's vigilance parameter.

When the cluster is not applicable to formula (3), then the closest cluster is selected and tested on the formula. If so cluster meets the vigilance parameter conditions, then a new cluster is formed and the input pattern in concern is allocated. Here, the newly produced cluster $x_i$'s vigilance parameter $\rho_1$ must be set as $\rho_0$. This is to have the new cluster's vigilance parameter to accord to the global vigilance parameter. If the newly produced cluster's vigilance parameter is too high, the input pattern cannot be allocated and the cluster becomes isolated.
During the production of new clusters, the vigilance parameter of all clusters and the global vigilance parameter are lowered the same amount according to the following formula. This is to reduce the probability of continuous cluster production and have the input patterns accord to the current cluster.

\[ \rho_{\pi_j}^{(0)} = \rho_{\pi_j}^{(1)} - \zeta \cdot \delta, \quad j = 1, \ldots, k, \]

\[ \rho_\psi^{(0)} = \rho_\psi^{(1)} - \zeta', \quad \text{where} \quad \zeta' \text{ is a control parameter} \]

**Removing the weighting and controlling the vigilance parameter:** If for the selected cluster, \( \pi_j \) satisfies (3), the input pattern is allocated to \( \pi_j \) and the weighting vector and concept vector of \( \pi_j \) is modulated by the following formula.

\[ w^{(i+1)} = w^{(i)} + x_i, \quad c^{(i+1)} = \frac{w^{(i+1)}}{\|w^{(i+1)}\|} \]

Applying the formula below changes the vigilance parameter for the cluster in concern.

\[ \rho_{\pi_j}^{(i+1)} = \rho_{\pi_j}^{(i)} + \delta \beta, \quad \delta, \beta \in [0, 0.002] \]

where \( \delta \) is a control parameter.

Cluster \( \pi_j \) ’s vigilance parameter \( \rho_{\pi_j} \) is modulated in order to prevent the input pattern being allocated entirely to a single cluster. If the number of input patterns in a certain cluster continues to increase, the non-vectors among the cluster’s weighting vector elements also increase. In this case, the cosine similarity measured by using the vector’s inner product is relatively higher than that of other clusters. This results in having the input patterns all gather in a single cluster and ultimately lowers the clustering coherences. Therefore, heightening the vigilance parameter of the clusters where the input patterns are to be allocated must prevent such cases.

### 3 Experimental Results

To verify its speed, the suggested algorithm has been tested on data set clas8c3 (ftp.cs.cornell.edu/pub/trusam), composed of 3893 documents extracted from the well-known data sets MEDLINE, CUL, and CRANFIELD. First, the MC program (www.cs.tamu.edu/uturn/miln/MC) vectorizes the 3893 data. In this process, stopwords and words of the frequency below 0.5% and above 15% are deleted. From the remaining 4262 terms, the t5 scheme makes 3893 document vectors.

For the experiment, besides dynamic Star-clustering algorithm, we also implemented ART, which clusters data dynamically, and \( k \)-means, Fast static-clustering algorithm.

At first, we make experiments with suggested algorithm, namely Modified ART, and as a result, we extracted five clusters, which has at least 20 data, in order of coherence. In five clusters, 195 data has been extracted and average coherence of clusters is 0.997. And run time is 14.04 seconds. Parameters of ART and dynamic Star are modified to have similar number of data with modified ART, and in case of \( k \)-means, \( k \) is set to 57 to get some number of produced clusters as modified ART for
Table 1: The Results over CLASSICS. ART stands for the run time. FNC for the total number of problem clusters, and ND for the number of data.

<table>
<thead>
<tr>
<th>Methodology</th>
<th>ART (μ=0.1, ω=0.63)</th>
<th>Dynamic (μ=0.3)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>NO</td>
<td>FNC</td>
</tr>
<tr>
<td>Cluster 1</td>
<td>1</td>
<td>9</td>
</tr>
<tr>
<td>Cluster 2</td>
<td>2</td>
<td>15.1</td>
</tr>
<tr>
<td>Cluster 3</td>
<td>3</td>
<td>14.8</td>
</tr>
<tr>
<td>Cluster 4</td>
<td>4</td>
<td>17.1</td>
</tr>
<tr>
<td>Cluster 5</td>
<td>5</td>
<td>17.1</td>
</tr>
</tbody>
</table>

The table shows that suggested modified ART can produce quart clusters with 4 or 6 times lower than k-means and dynamic ART, and in case of ART, within almost same time, modified ART can produce clusters with 25% better coherence.

References