A Graph-Based Text Similarity Algorithm

Zuoguo Liu  
College of Computer Science & Information  
GuiZhou University  
GuiYang city, GuiZhou province, China, 13385142056  
412769371@qq.com  

Xiaorong Chen  
College of Computer Science & Information  
GuiZhou University  
GuiYang city, GuiZhou province, China, 13809443285  
xrchengz@163.com  

Abstract—This paper is trying to research a text similarity algorithm which based on graph theory. A text is mapped into a graph which consists of terms as its nodes and term sequences as its undirected edges. The Maximum Common Subgraph (MCS) of two graphs is useful for analyzing their similarity and the similarity of two texts is divided into two parts: nodes similarity and edges similarity. Each part is calculated respectively and text similarity is the sum of two parts.

Keywords—graph theory; mapped graph; maximum common subgraph

I. INTRODUCTION

As an important domain of data mining and text processing, text clustering technology has appeared for a long time. Many models such as VSM (Vector Space Model), DBScan (Density-Based Scan) and SOM (Self-Organizing Map) have been researched and improved repeatedly. It is a key step to calculate similarities, or distances, amount texts. This paper will elaborate a text similarity algorithm which maps Chinese texts into graphs then calculates the similarity of two texts by comparing their graphs. The Graph-Based Text Similarity algorithm will be abbreviated to GBTS in this paper.

Like traditional ones, GBTS algorithm includes three stages: pretreatment, mining and analyzing, and result showing [1]. Since it’s the main problem that how to process high dimension data in text clustering, GBTS is oriented to long texts but not short ones or abstracts. It means that time efficiency should give way to space efficiency in order that stability and accuracy can get guarantee.

II. GRAPH-BASED TEXT SIMILARITY ALGORITHM

A text consists of large numbers of words which appear in some relatively fixed sequences and those sequences form a mapped graph which takes words as its nodes and sequences as its edges. A mapped graph contains most of the information of a text so that the similarity of two texts can be calculated by comparing their mapped graphs. What’s more, the similarity of two texts should be a value in the interval $[0, 1]$.

Article [2] has elaborated a Graph Structure Presentation Method Based Chinese Clustering (GSPM) and illustrated that the Maximum Common Subgraph (MCS) of two graphs represents their similarity and the more nodes and edges a MCS contains, the higher similarity two graphs have. It is the key that how the MCS of two graphs is obtained and calculated into a mathematical value to represent the similarity.

III. THE PROCESS OF GBTS ALGORITHM

A. Pretreatment

Fig. 1 shows the whole process of pretreatment.

![Figure 1. Process of pretreatment](image)

First of all, a text must be reorganized to the uniform format which computer can process.

Then, text segmentation is necessary in Chinese text processing in order to segment each text into words. Since it’s not the core of this research, A System—Institute of Computing Technology, Chinese Lexical Analysis System (ICTCLAS) from Institute of computing technology of Chinese Academy of Sciences, is used for segmentation and tagging at this step.

The third step is word properties selecting, or stopped words filtering. A word properties list has been established in a document before ICTCLAS tags each word by word property. All of notional words such as nouns and verbs will be remained while function words will be abandoned.

At last Term Frequency (TF) and Inverse Document Frequency (IDF) will be calculated and the value of TF*IDF will be a standard [3]. A term will be abandoned if its value of TF*IDF little than threshold.

B. Text expressing model

Text expressing is the first step of text analyzing. A text must be mapped into a graph in this paper.
Differing from Article [2], the mapped graph is defined as a quadruple as \( \{ N, E, W_n, W_e \} \) in this paper. N is the set of nodes, E is the set of undirected edges, \( W_n \) is the set of node weights and \( W_e \) is the set of edge weights.

Fig. 2 shows the structure of text expressing model.

Assume that term A and term B appear in a text and they adjoin each other, it’s certainly that A and B are two nodes of the graph and there is an undirected edge (A, B) links node A and node B.

The next problem is how to calculate the weight of nodes and edges in graphs. The values of TF will be selected as the weights of nodes.

According to Article [4][5], definitions are given out:

\[
W_n(d,i) = TF(d,i) 
\]

(1)

\[
W_e(d,i,j) = \frac{TF(d,i,j)}{TF(d,i) + TF(d,j) - TF(d,i,j)} 
\]

(2)

\( TF(d,i) \) is the frequency of term \( i \) in text \( d \), and \( TF(d,i,j) \) is the frequency of term \( i \) and \( j \) which \( j \) adjoins \( i \).

In fact, it can be found that the frequencies of terms could be replaced by the count of their appearances. Therefore, Formula (2) will be replaced by:

\[
W_e(d,i,j) = \frac{C(d,i,j)}{C(d,i) + C(d,j) - C(d,i,j)} 
\]

(3)

\( C(d,i) \) is the count of term \( i \) which appears in text \( d \).

C. Similarity of texts

It has been noted that the MCS of two graphs represents their similarity. Because the complete definition of MCS is too strict to be processed, a relatively extended definition of MCS is given out as this:

For two graphs \( G_1 \) and \( G_2 \), if there is another graph \( g \) conforms to the relationship that \( g \subseteq G_1, g \subseteq G_2 \), and there is not any graph \( g' \) conforms to this relationship so that \( |g'| > |g| \), then \( g \) is the MCS of \( G_1 \) and \( G_2 \), and it is marked as \( MCS(G_1,G_2) \). \( g \subseteq G \) means that \( g \) is a subgraph of \( G \), namely, all the nodes and edges of \( g \) appear in \( G \).

Assume that Text A and B have been mapped into two graphs. Nodes and edges of \( MCS(A,B) \) will be processed respectively.

For creating nodes, it is needed to get all the common nodes. And the weight of a common node will be set as the smaller weight between the two texts.

For creating edges, all the common edges which link common nodes must be checked and the weight of a common edge will be set as the smaller weight the same with creating nodes.

Since the nodes and edges are processed respectively, the similarity value should be separated into two parts: Nodes Similarity (NS) and Edges Similarity (ES). GSPM defines them as this:

\[
NS(A,B) = \alpha \times \frac{|N(MCS(A,B))|}{\text{MAX}[|N(A)|,|N(B)|]} 
\]

(4)

\[
ES(A,B) = (1-\alpha) \times \frac{\text{MIN}[W_{e,ij}(A), W_{e,ij}(B)]}{\text{MAX}[E(A),E(B)]} 
\]

(5)

\[
\text{Similarity}(A,B) = NS(A,B) + ES(A,B) 
\]

(6)

Because the formulas defined in GSPM are limited, some new similarity formulas are created in this paper:

\[
NS(A,B) = \alpha \times \left( \frac{|N(MCS(A,B))|}{|N(A)|} + \frac{|N(MCS(A,B))|}{|N(B)|} \right) 
\]

(7)

\[
ES(A,B) = \frac{1-\alpha}{2} \times \left( \frac{\sum W_e(MCS(A,B))}{\sum W_e(A)} + \frac{\sum W_e(MCS(A,B))}{\sum W_e(B)} \right) 
\]

(8)

\( W_n \) is node weights set, \( W_e \) is edge weights set, \( |N(G)| \) and \( |E(G)| \) are the sizes of nodes set and edges set of graph \( G \) while \( \alpha \) is a parameter ranges from 0 to 1. It is obvious that nodes similarity is ignored when \( \alpha = 0 \) and edges similarity is ignored when \( \alpha = 1 \). What’s more, the value of Similarity is in the interval \([0,1]\).
D. Example

Assume that two Chinese texts have been gotten as this, Text A: “文本 聚类 知识 挖掘 领域 重要 技术 手段，文本 信息 挖掘 知识 检索 具有 重要作用”; Text B: “文本 聚类 文本 挖掘 重要 技术，应用 文本 挖掘 信息 检索 方面”。The mapped graphs and MCS should be gotten as this:

IV. Experiments and Analyses

The Chinese text corpus from Fudan University [6] is selected as experiment data. Ten of the categories are chosen. They are: C3-Art, C5-Education, C19-Computer, C29-Transport, C31-Environment, C32-Agriculture, C34-Economy, C35-Law, C37-Military and C38-Politics. 100 texts are chosen randomly from each category then 1000 texts are processed in experiment. As comparison, the GSPM illustrated in Article [2] will be compared with GBTS. TABLE I shows the comparison.

<table>
<thead>
<tr>
<th>Category</th>
<th>GSPM Precision rate</th>
<th>GSPM Recall rate</th>
<th>GBTS Precision rate</th>
<th>GBTS Recall rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Art</td>
<td>78.45%</td>
<td>55.00%</td>
<td>72.31%</td>
<td>60.18%</td>
</tr>
<tr>
<td>Education</td>
<td>95.24%</td>
<td>100.00%</td>
<td>92.00%</td>
<td>88.42%</td>
</tr>
<tr>
<td>Computer</td>
<td>81.82%</td>
<td>90.00%</td>
<td>88.52%</td>
<td>91.70%</td>
</tr>
<tr>
<td>Transport</td>
<td>68.42%</td>
<td>65.00%</td>
<td>74.60%</td>
<td>72.34%</td>
</tr>
<tr>
<td>Environment</td>
<td>58.82%</td>
<td>50.00%</td>
<td>80.31%</td>
<td>75.43%</td>
</tr>
<tr>
<td>Agriculture</td>
<td>100.00%</td>
<td>100.00%</td>
<td>80.88%</td>
<td>89.19%</td>
</tr>
<tr>
<td>Economy</td>
<td>76.19%</td>
<td>80.00%</td>
<td>37.83%</td>
<td>43.43%</td>
</tr>
<tr>
<td>Law</td>
<td>72.73%</td>
<td>80.00%</td>
<td>76.26%</td>
<td>77.72%</td>
</tr>
<tr>
<td>Military</td>
<td>47.37%</td>
<td>45.00%</td>
<td>81.04%</td>
<td>87.90%</td>
</tr>
<tr>
<td>Politics</td>
<td>77.78%</td>
<td>70.00%</td>
<td>83.55%</td>
<td>69.58%</td>
</tr>
<tr>
<td>Average</td>
<td>75.70%</td>
<td>73.50%</td>
<td>76.73%</td>
<td>75.59%</td>
</tr>
</tbody>
</table>

Comparison shows that GBTS is more efficient than GSPM on average and GBTS has higher stability in most categories. The reason comes from the differences between the two algorithms.

Firstly, GSPM maps a text into a directed graph while GBTS maps it into an undirected graph. In the example above which texts are very short, comparison will be given out.
Fig. 6 and 7 show that undirected graph which used in GBTS will remain more edges in MCS than directed graph while the number of nodes is the same. It means that GBTS is more favorable for edges similarity while it has same effects on nodes similarity with GSPM. The advantage will become more obvious if texts are longer.

Secondly, Formula (4) and (5) are limited. Assume that Text B is contained completely or nearly by Text A. Their mapped graphs should be like Fig. 8.

![Figure 8. MCS of GBTS](image)

It is obvious that the two texts are similar because Graph A contains Graph B, but the value of NS and ES may not be large enough because Graph A is $\text{MAX}(A,B)$ and it may be much bigger than $\text{MCS}(A,B)$. However, Formula (7) and (8) will calculate average value so that final similarity will not be less than 50%. Notice in this case:

\[
\text{NS}(A,B) = \frac{\alpha}{2} \times \left( \frac{|\text{MCS}(A,B)|}{N(A)} + 1 \right) > \frac{\alpha}{2}
\]

\[
\text{ES}(A,B) = \frac{1 - \alpha}{2} \times \left( \frac{\sum W_e(\text{MCS}(A,B))}{\sum W_e(A)} + 1 \right) > \frac{1 - \alpha}{2}
\]

\[
\text{Similarity}(A,B) = \text{NS}(A,B) + \text{ES}(A,B) > 0.5
\]

What’s more, each of GSPM and GBTS has its advantages and disadvantages. GBTS process more efficiently in categories of art, computer, transport, environment, military and politics while GSPM shows more efficiency in education, agriculture and economy.

V. CONCLUSION

In this paper, a Graph-Based Text Similarity algorithm has been worked out and compared with GSPM algorithm. Both of the two algorithms concentrate on using graph theory to analyze texts and calculate similarities in long texts. Comparison shows that GBTS has higher stability while GSPM has higher upper limit of success in some categories.

ACKNOWLEDGMENT

The research is supported by GuiZhou provincial industry and informatization development special fund.

This paper is accomplished under the guidance of Professor Xiaorong Chen from College of Computer Science & Information, GuiZhou University.

Thanks are given to Institute of computing technology of Chinese Academy of Sciences for offering ICTCLAS.

Last but not least, Thanks are also given to Fudan University and Institute of computing technology of Chinese Academy of Sciences for collecting Chinese text corpus.

REFERENCES


