A Topic-specific Crawler with Dynamic Concept Context Graph Based on FCA

Xiaolei LI¹, Yajun DU¹*, Yanyan WEI¹, Yong ZHENG²

¹School of Mathematics and Computer Engineering, Xihua University, Sichuan 610039, China
²Institute of Electronics, Taiyuan University of Science and Technology, Taiyuan 030024, China

Abstract

An excellent topic-specific crawler should retrieve as many high related web pages as possible in the limited time. The historical clicked web pages can express the users’ interest in a certain extent. By using the knowledge of Formal Concept Analysis (FCA), some information is extracted out to construct the concept lattice which is used to build the Concept Context Graph (CCG). In this paper, we construct a knowledge background for the topic-specific crawler by using the relevant web pages. And a Dynamic CCG (DCCG) is proposed in this paper which updates the concepts in a way of elimination mechanism all the crawling process. At last, several different CCGs are taken into the experiment for the comparison in their performance.

Keywords: CCG; Topic-specific Crawler; FCA; DCCG

1 Introduction

With the rapid development of the Internet, people are increasingly dependent on computer network to find the required information. The network has become an important source of accessing information in people’s daily life. The emergence of search engine can make people query related web pages by keywords quickly. As we all know, the defect of general search engine is not retrieving enough web pages but obtaining a high precision rate in returned lists. Since the recall rate is no longer a problem, all the search engines try to improve their precision rate. With the increase of diversity information on the network, traditional search engine cannot meet people’s demand for personalized information search. According to the latest report of the Netcraft statistics, the global site has more than 160 million. Google indexes 1000 billion web pages with more than 80% web pages are not been retrieved [1]. Especially, when use the general search engine to search information about a particular field, the results may deviate from the topic. In such a huge amount of data in the network to locate target information quickly has

*Project supported by the National Nature Science Foundation of China (No. 61271413). The Graduate Student Innovation Foundation of Xihua University under Grant(No. ycjj201309).
*Corresponding author.
Email address: duyajun@mail.xhu.edu.cn (Yajun DU).

1553–9105 / Copyright © 2014 Binary Information Press
DOI: 10.12733/jcis9506
February 15, 2014
become an urgent problem. All of these issues have contributed to the birth of the topic-specific crawler. A topic-specific crawler will compute the relevance between the retrieved web page and the specific topic to determine whether the new web page should be discarded or retained [2]. The core work of topic-specific crawler is the relevance algorithm. In 1994, Bra proposed Fish Search algorithm [3]. The crawling behavior of web pages is seen as the migration of fish, and each URL is seen as a fish. In 1998, Hersovici expanded the Fish Search which called Shark Search [4]. VSM was used to calculate the relevance value in the Shark Search algorithm. In 2002, Bergmark increased the tunnel crossing ability which based on Best First Search [5]. In 2005, Li proposed decision tree algorithm by utilizing the anchor text [6]. The crawling priority was decided by the anchor text. In 2007, Dong [7] has firstly introduced the FCA into focused crawling and improved the matching from keywords to concepts. They proposed a method to estimate the similarity between concepts by calculating the distance of concepts. In 2008, Yang [8] proposed a concept similarity context graph which was based on FCA. The core idea of Yang is calculating the similarity value between concepts and core concept to estimate the priority score of unvisited URLs. In 2009, with the thinking of concepts can be layered according to the attribute, Peng [9] proposed a topic-specific crawler which was based on CCG. Gao [10] [11] improved the CCG by updating the concepts in CCG to guide the topic-specific crawler. In the crawling process, those relevant concepts will be added into the CCG and those irrelevant concepts will be deleted. Gao introduced the idea of updating the concepts in the CCG, but this also bring into some new problems. For example, with the updating of the concepts in the CCG, the number of concepts will reach the expansion state. Du [12] used the knowledge of FCA to measure the concepts similarity and proposed a method of web page’s rank by using the information content approach based on users’ web logs. In 2013, Li [13] improved the original CCG by using the theory of six degrees of separation. In the experiment, he got a result that a CCG can achieve the best performance with an average layer of 2.6. By above researches, we find that the research of topic-specific crawler undergoes from link-based methods to semantic-based methods. The semantic-based methods tried to construct a formal context and mine information to guide the topic-specific crawler. In this paper, we proposed a dynamic CCG which can not only update the concepts but also keep the concept in an acceptable number all through the crawling process. And with the elimination mechanism of updating the CCG, the concepts which retained in the DCCG will have a high similarity with the topic. We organize this paper in the following way: Section 2 is the related work of different CCGs. Section 3 is the method of how to form a dynamic CCG. In Section 4, the experimental results under different CCG are presented. Section 5 is the conclusions and future work.

2 Related Works

In this section, we will first review the knowledge of formal concept analysis and then introduce some CCGs. All these CCGs can act an appreciable performance, but some imperfections need to be made up to make the CCGs have a better performance.

2.1 Formal concept analysis

FCA is a powerful tool for data analysis and rule extraction from the formal context which is proposed by Wille. Concept Lattice (CL) is the core data structure. Each node of the CL is a
concept that is composed of extension and intension. The extension of the concept is understood as a collection of objects belonging to this concept, while the intension is considered the characteristics common to all of these objects. All concepts, together with the generalization/instantiated relationship between them form a concept lattice. For instance, consider a context named smart-

<table>
<thead>
<tr>
<th></th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>d</th>
<th>e</th>
<th>f</th>
</tr>
</thead>
<tbody>
<tr>
<td>P1</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>P2</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
</tr>
<tr>
<td>P3</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>P4</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>P5</td>
<td></td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>P6</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 1: The binary relation of the formal context

phone, suppose that the set O is defined by the following six objects which represent six different phones: O = {P1, P2, P3, P4, P5, P6}. And the set A is defined by six possible attributes of these objects: A = {3G, TI, touchscreen, GPS, Navigation, Android, IOS}. For the sake of convenience, we make “a” → “3G”, “b” → “TI”, “c” → “touchscreen”, “d” → “Navigation”, “e” → “Android”, “f” → “IOS” where “→” means “stands for”. R = (O, A, I) is a formal context for the topic of “smartphone”, and its binary relation is described in Table 1. And Fig. 1 shows the Hasse graph of the concept lattice corresponding to the formal context in Table 1.

2.2 The different CCGs

At present, there are several different CCGs. All the CCGs are constructed by the information which are extracted from users history clicked web pages. But the construction methods are slightly different. The following describes two of the optimizing CCGs.
2.2.1 Incrementally updating concept context graph

In the process of crawling, some new found web pages can express the topic well. So, Gao proposed a new improved CCG which can update the background knowledge for the topic-specific crawler. The method of constructing the CCG is the same with Peng’s which is according to the attribute number of a concept. The innovation of the incrementally updating CCG is that an unvisited page is seen as an Incremental Concept (IC). Some definitions are as follows: D-topic is attributes set of the topic in CCG. D-visit is attributes set of unvisited web page. D-user is attributes set which users provide. D-new is an attribute set.

**Definition 1** (new concept) The intersection of D-topic and D-visit generates the D-new. If D-new is $\emptyset$, we call D-visit a new concept. That means the concept is off-topic and should be discarded.

**Definition 2** (updated concept) If the D-new is not $\emptyset$, D-visit $\subseteq$ D-topic and D-visit not belong to the CCG, then D-visit is called a updated concept.

Taking the Table 1 for example again, when user’s query is “acdf”, the concepts contain the user’s query are core concept and the CCG corresponding to the Table 1 is shown in Fig. 2. When crawler visits a new web page which can be expressed by the term of “ad”, we take the page ($\{P9\}, \{ad\}$) as an updated concept. The intersection of the updated concept and the core concepts contains two attributes. The updated concept should locate in the 1-layer. Because all the core concepts contain the attributes $\{ad\}$, the similarity between each core concept and the updated concept should be calculated. The details of the calculation process are described as follows:

$$
\text{SimCC}(\{P2, acdf\}, \{P2P7, ad\}) = \frac{1}{2} * \frac{1}{2} + \frac{2}{4} * \frac{1}{2} = 0.5
$$

$$
\text{SimCC}(\emptyset, abcdef), \{P7\emptyset, ad\}) = \frac{1}{2} * \frac{1}{2} + \frac{2}{6} * \frac{1}{2} = 0.42
$$

![Fig. 2: The CCG corresponding to the Table 1](image)

By the calculating, we can see that the core concept ($\{P2\}, \{acdf\}$) and the updated concept have the higher similarity. So the new concept is the inserted into the 1-layer and related to the core concept ($\{P2\}, \{acdf\}$). In the Fig. 2, the green node represents the new updated concept.
2.2.2 Optimizing concept context graph

Yang proposed a concept similarity context graph in 2008. A definition between two concepts was defined by Yang as follow:

\[ SimCC((O_1, A_1), (O_2, A_2)) = \frac{|O_1 \cap O_2|}{r} \cdot w + \frac{|A_1 \cap A_2|}{m} \cdot (1 - w) \]  
(1)

where \( r \) is the greater cardinality between set \( O_1 \) and \( O_2 \), \( m \) is the greater cardinality between attribute set \( A_1 \) and \( A_2 \). The range of weight \( w \) is set at \( 0 < w < 1 \). The score of a concept in the lattice is defined by Yang as follows:

\[ f(x) = \begin{cases} 1 & \text{when } c \text{ is core concept,} \\ Score(d) \cdot SimCC(c, d) & \text{when } c \text{ is not core concept.} \end{cases} \]  
(2)

where \( d \) is the concept \( c \)'s parent concept or child concept, and \( SimCC(c, d) \) is the similarity between \( c \) and \( d \). \( Score(d) \) represent the score of concept \( d \). The concept score is the foreshadowing for the URL prediction. In 2013, Li found the CCG can be assigned a specific level to make the CCG performing more efficiency. In his results, a CCG with 3 layers can reach an acceptable performance. In the crawling process, a new web page is seen as a virtual concept. If the new virtual concept can map into a layer of the OCCG, the virtual concept will be expanded by the method which Gao has proposed. Once the new expanded concept is assigned into the corresponding layer, the similarity value between the concept and the core concept is calculated with the method of formula 1. And the similarity value is seen as the priority value of the new web page.

3 Dynamic CCG

Once the formal context is constructed, a specific topic can be expressed by this knowledge background. By the method of CCG’s construction, all concepts are allocated in the corresponding layer of the CCG. For the efficiency of crawling, we set a threshold of concept number. When a new web page is visited, it will be seen as a virtual concept at first. The representative words of this web page are extracted out firstly. According to the number of attribute intersection between core concept and the virtual concept, the proper level of this virtual concept can be finalized.

3.1 The updating of concepts

The DCCG updates all the way during the crawling process. In this paper, we still use the concept of update concept which is proposed by Gao. If the virtual concept of a new web page is an update concept, the DCCG will check its threshold to determine whether the virtual concept should be inserted or not. If the concept number of the DCCG is less than the threshold, the new virtual concept should be expand and insert into the corresponding level of the DCCG. The inserting a new concept process is as follows:

**Step 1:** If the virtual concept contains all the attributes of the core concept and the entire attributes of the virtual concept are include in the FC, the object set will expand by adding the core concept’s object elements. It will allocate in the 1-layer of the DCCG, else go to Step 2.
**Step 2:** If the intersection of attribute between the virtual concept and the core concept is not null and the number of the insertion is i, then the similarity value will be calculated between the virtual concept and the concepts in the N-i layer where N is the number of core concept attribute. At last, the concept which gets the highest similarity value with the virtual concept is assigned as the father concept of the new virtual concept. The new concept expands its object set by inserting all the object elements of the assigned father concept. Then go to step 3.

**Step 3:** If the intersection is null, we think this web page is deviation from the topic. This web page should be ignored.

This progress is the dynamic updating of the concepts. As the new concepts are inserted into the CCG, the total number of concepts will reach the threshold. So this method will cause a layer updating of the CCG. The subsection of the 3.2 is mainly about CCG’s layer updating.

### 3.2 The updating of layer

When the concept number reaches the threshold, the concepts in the CCG will execute the elimination mechanism which excludes the minimal correlated concept in the CCG. When a new web page is met, the similarity value will be calculated between the virtual concept and the core concept. If the virtual concept should locate in the most out layer, then each concept in the layer is compared to the virtual concept. If the similarity value is smaller than the concept which has the smallest similarity value in the most out layer, the virtual concept will be discarded. When a virtual concept is assigned in the layer except the most out layer, the concept with the smallest similarity value will be deleted in the most out layer. In this way, the DCCG can keep the threshold unchanged. The entire operation process is shown in the Fig. 3. In the Fig. 3, the expansion module is used to expand the virtual concept which represents a web page. The detail expansion we have discussed in the former. Here, we focus on the Logic Module (LM) which keeps the dynamic activity of the CCG. For convenience, the concepts are just represented by the similarity value in the CCG. In Fig. 4 and Fig. 5, the updating of CCG is shown. The Fig. 4 is the original CCG which is constructed by the selected URLs from result list of Google for a specific topic. The Fig. 5 displays the evolution process of the original CCG. The concepts with blue color are new updated concepts. Assumed that the threshold is 10 in the example, the subgraph (a) of Fig. 5 displays a new concept with the similarity value of 0.30 is inserted into the 3-layer. Then the concept number reaches the threshold, and the situation of subgraph (b) will happen. In the subgraph (b) of Fig. 5, an old concept which owns the similarity value of 0.18 is
replaced by a new concept with a higher similarity value. This elimination mechanism executes like the law of survival of the fittest which eliminated the least relevant concept in the CCG. The subgraph (c) indicates those concepts in the 3-layer are all replaces by the new concepts which are inserted into the in inner layer. Under this mechanism, the layers of CCG will become fewer and fewer. And all the concepts have a high similarity value to the core concept in the 0-layer.

This will result a high correlation CCG to the specific topic. In the section 4, the experiment will be done on several different CCGs to compare their efficiency.

4 Experiments

For this experiment, we choose 3 different topics which include AutoRacing, Porsche and Samsung. And the threshold is set to 20 for the DCCG in our experiment. In the Fig. 6, we display the different P, R and F values. As we can see in the Fig. 6, the line volatility of DCCG are tiny compared to the other two CCGs. This means the DCCG can get satisfied values both in recall and precision. The OCCG and IUCCG get the result with either high recall and low precision or low recall and high precision.
5 Conclusions and Future Work

For a search engine, the goal is to retrieve as many relevant web pages as possible. But P and R are two factors which restrict each other. In order to achieve a state of balance, the DCCG can guarantee both the P and R satisfied. In the future, we will research a suitable value of the threshold to find what threshold value can cause the DCCG’s performance better.

References