"十一五"国家科技支撑计划重点项目

"十一五"文化遗产保护领域国家科技支撑计划重点项目论文集

古代建筑保护技术及传统工艺科学化研究（Ⅰ）

科技部社会发展科技司
国家文物局博物馆与社会文物司（科技司） 编

文物出版社

“十一五”文化遗产保护领域国家科技支撑计划重点项目论文集

古代建筑保护技术及传统工艺科学化研究（Ⅰ）

科技部社会发展科技司
国家文物局博物馆与社会文物司（科技司） 编

文物出版社

封面设计　周小玮
责任印制　陆　联
责任编辑　吴　湘

图书在版编目(CIP)数据

古代建筑保护技术及传统工艺科学化研究／科技部社会发展科技司,国家文物局博物馆与社会文物司(科技司)编.—北京:文物出版社,2010.11
(“十一五”文化遗产保护领域国家科技支撑计划重点项目论文集)
ISBN 978-7-5010-3080-4

Ⅰ.①古…　Ⅱ.①科…②国…　Ⅲ.①古建筑-文物保护-中国-文集②古建筑-建筑艺术-中国-文集
Ⅳ.①TU-87②TU-092.2

中国版本图书馆CIP数据核字(2010)第215617号

“十一五”文化遗产保护领域国家科技支撑计划重点项目论文集
古代建筑保护技术及传统工艺科学化研究

科技部社会发展科技司
国家文物局博物馆与社会文物司(科技司)　编

*

文物出版社出版发行
(北京东直门内北小街2号楼)
http://www.wenwu.com
E-mail:web@wenwu.com
北京君升印刷有限公司印刷
新华书店经销
787×1092　1/16　印张:50
2010年11月第1版　2010年11月第1次印刷
ISBN 978-7-5010-3080-4　定价:200.00元(全2册)

"十一五"文化遗产保护领域
国家科技支撑计划课题成果论文集
编辑委员会

序　　言

历史悠久、弥足珍贵的中华民族文化遗产，既是不可再生、不可替代的深厚物质资源，更是博大精深、绵延不断的文化资源和精神资源，有着重要的历史、艺术和科学价值，对国家的统一、民族的团结、社会的和谐、人民的幸福具有重要的意义。

文化遗产保护科技是一个开放的复杂巨系统，包括人文社会科学、自然科学、工程技术科学等一切与文化遗产保护相关的科学和技术。作为多学科高度交叉综合的集成体，文化遗产保护科技已经在文化遗产价值的调查、认定、研究、展示、利用和传承，文化遗产本体的保存、保全和修复，以及对文化遗产相关环境的控制与治理中发挥着越来越重要的作用。文化遗产保护科技的进步对文化遗产事业的发展具有决定性影响，已成为推动着我国从文化遗产大国向文化遗产保护强国转变的核心要素，同时也将对国家科学和技术的整体发展做出贡献。

“十一五”期间，在科技部的大力支持下，文化遗产领域有4个项目15项课题列入国家科技支撑计划第一批启动项目。包括“文化遗产保护关键技术研究”、“中华文明探源工程”、“大遗址保护关键技术研究与开发”、“古代建筑保护技术及传统工艺科学化研究”。随后，“石质文物保护关键技术及南京报恩寺地宫出土文物保护关键技术研究”、“中华文明探源工程及相关文物保护关键技术研究”又相继获得批复实施。国家文物局以组织实施国家科技支撑计划等重大科研项目为契机，努力推动体制机制创新，积极寻找建立跨学科、跨领域、跨行业、跨部门的合作机制与模式。通过重大科技计划的实施，统筹考虑行业的技术研发、装备升级、人才培养、基地建设和体制机制创新，实现了文化遗产保护科技的跨越式发展。

截止2010年初，第一批启动的4个项目15项课题已全部通过了结项验收，据不完全统计，共研发新技术（工艺）21项，新产品、新材料、新装置36项，获得自主知识产权和专利179项，制定技术标准40项，培养博

士、硕士研究生301名，发表文章513篇，出版专著15本。一些科研成果已广泛应用于第三次文物普查、长城资源调查、重点文物保护工程、大遗址保护工程、灾后文化遗产抢救性保护、馆藏文物保存环境改善、博物馆展示提升等重大工程和重点工作，文化遗产保护科技含量大幅提升，行业自主创新能力得到显著提高。

为进一步做好科技成果的推广工作，我们就文化遗产领域科技支撑计划的部分成果汇编成册，这既是文化遗产保护科技成果的展示，也是向所有关心文化遗产保护的社会各界的回报。

值此，向勇于实践、不断创新的科技工作者，向文化遗产的保护者和守卫者致以崇高的敬意。

编　者

目　录

第一部分　古代建筑保护技术信息系统研发

第二部分　古代建筑虚拟修复及 Web 表现技术研究

第三部分 古代建筑琉璃构件保护技术及传统工艺科学化研究

第四部分　古代建筑油饰彩画保护技术及传统工艺科学化研究

Ontology Based Query Expansionin Vertical Search Engine

Ma Liangjun[1], Chen Lin[1], Gao Yibo[1], Yang Yiping[1]

(1. Institute of Automation, China Academy of Sciences, Beijing, 100190, China)

Abstract: Queries to search engine on the Internet are usually short, and cannot provide enough information for effective retrievals. Researchers have developed query expansion to cope with the problem and proved its usefulness. But previous researches have mainly on the general search engine and do not give the semantic enough attention. In this paper, we introduced a novel algorithm especially for the vertical search engine, which makes full use of the character that knowledge in special domain can be described more availably and powerfully than that in the open domain. In the algorithm, we utilize the knowledge, formalized by ontology, to generate semantic diagraph for combinations of words in one query. And then according to the semantic distance between the vertexes in the diagraph, we selected the candidates to be added. Terms added into the initial query are obviously related in semantic with the initial one. And we experimentally show that it can improve the search result clearly.

1 Introduction

The proliferation of the World Wide Web prompts the wide application of search engine. Especially, theemergence of the WEB2.0 is booming the vertical search engine [1]. However, short queries [2] and the incompatibility between the terms in queries and the index have infected the performance of the two kinds of searching engine. Many researchers have invented various algorithms of query expansion to solve the short query problem. And most of them focus on the automatic query expansion [3]-[7][13]. Generally, the automatic query expansion can be categorized into local strategy and global strategy.

A query method based on local strategy usually extracts expansion terms from part of the initial query results by applying the relevance feedback technology, with the vector processing method and the probabilistic feedback method as basic ones. Early experiment [5] has shown improvement in precision for small test in practice it. Recently [3] [13] have improved the strategy by involving user logs and applying the minimal feedback separately. But the strategy consumes much time and is affected by the first query result badly in some case; and also the users may be reluctant to select the relevant files.

As in the global strategy, researchers like to make use of the thesauri, which is co-occurrence-based, automatically built or handed-crafted, to expand the initial query words. In [4] [6] [7], researchers have examine the method and developed it. This strategy is easy to apply and more timeless. Comparing with applying the co-occurrence-based thesauri, people like to study and apply the co-occurrence-based thesauri, but in the lasted result [4], Hui Fang showed using the manual crafted thesauri, for example *WordNet*, will have a better result than just using the co-occurrence-based thesauri. And the limitation of co-occurrence-based query expansion was mentioned even earlier in [9].

But the algorithmsabove are mostly for the general search engine, of course, they do not pay enough attention to the knowledge in domain. Terms added into the initial queries are always lack of clear semantic relationship. But the abundant, more available and describable domain knowledge is just the specialty of the vertical search engine versus the general one. What's more, mining the relationship between terms based on the domain knowledge becomes more possible.

And then taking account of the result of [3]-[9] and [13], the more related terms added with the initial query, the much better the result shows.

In this paper, we developed a novel method of query expansion, which is based on the ontology-described knowledge and focus on the semantic relationship of terms. The method can make sure terms added are obviously related with the initial query in semantics. Furthermore, it is developed especially for the vertical search engine. In this method, we firstly establish the domain knowledge database by adopting ontology as the describing tool; secondly parse the query string into terms, and then construct semantic diagraph with each term as the first vertex based on the domain knowledge; thirdly calculate the semantic distance between the first vertex and each vertex in the semantic diagraph, and then according to the threshold, select the expanded terms of each semantic diagraph; at last, we combine all the terms gotten from the semantic diagraph with logic operator and then obtain the result of query expansion.

2 Domain Knowledge Structure

In this section, we'd like to introduce the structure of the domain knowledge database. And the domain knowledge database is the basis of this method introduced in this paper.

Firstly, taking into account of the ontology's powerful ability to describe knowledge, we chose it as the core of the domain knowledge sources.

Besides that, we suggest the domain knowledge source includes, not limited, the ontology, the fact database and, for each language supported in the search engine, a lexicon that includes an onomasticon, a lexicon of names, a thesaurus that only includes the synonym, and some supporting category tree, especially a shallow semantic analyzer.

And Figure 1 shows the sketch structure of the domain knowledge:

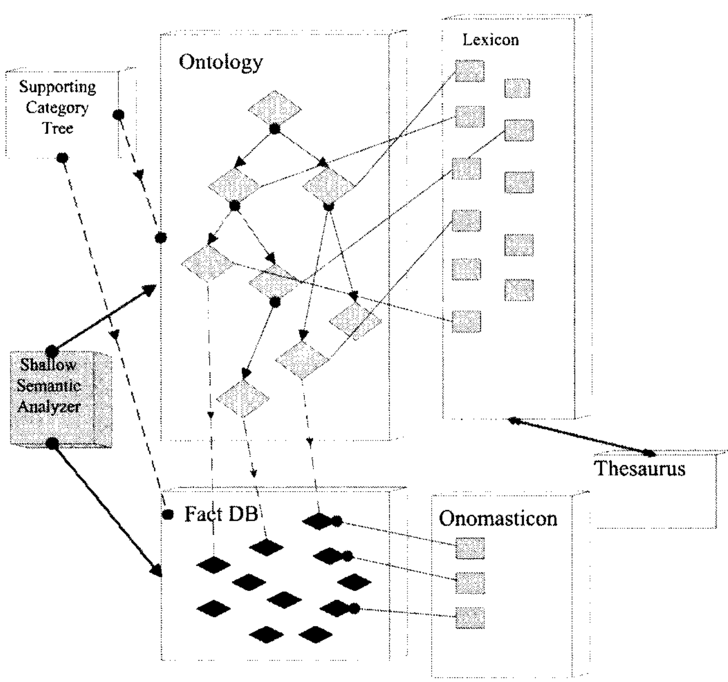

Fig. 1 the sketch structure of the domain knowledge

The ontology can be divided into two classes, according to the fact they describe, one is themain ontology which is the basic standard that most material classified according to; the other is the supporting ontology in which the items are candidates for values of the instances' of the main ontology or the ontology's slots.

For the values in the instances' or the ontology's slots which cannot be described in ontology but can be expressed in hierarchy, we store the values in hierarchy which are named after the supporting category tree.

And the shallow semantic analyzeris used to gets some special attributes from user's query. For example, a query "1940 dollars, computer", the analyzer will obtain a pair of attribute "*price*: 1940 *dollars*" . In our method, it simply utilizes the regular expression to obtain the attribute pairs. e. g. the regular expression: [一二三四五六七八九]? [百千万]? [元] | \d\d* [百千万]? [元] to match the pattern of "三千元" (which means three thousand *yuans*) in Chinese, and get the attribute pair of *price*: 3000*yuan*.

3 Semantic Diagraph

In this section, we will post a structure to express the semantic relationship among words or phrase based on the domain knowledge. Referring other researchers' result, we also give an equation to calculate the semantic distance in a diagraph, which quantifies the relationship between vertexes in the diagraph.

3.1 The definition of the Semantic Diagraph

Semantic diagraph is created to describe the semantic relationship among words, phrases and strings, partly referred to semantic net mentioned in [6], with its vertexes expressing the words, phrases or strings, whose edges labeled with relationship of the Tail and Head and each edge is assigned with a weight. In the semantic diagraph, we use the distance among the vertexes, the relationship between any of the two vertexes, and the weight of the edge to mark the semantic distance of the two vertexes.

So, a semantic diagraph can now be defined as follows:

Definition 1. A Semantic Diagraph structure is a couple

$$SD: = (V, E),$$

Where

- SD: a semantic diagraph;
- V: the vertexes of a semantic diagraph. It can bea word, phrase or string.
- E: the edges of a semantic diagraph. And it is defined as the definition 2.

Definition 2: An edge of a semantic diagraph between vertex v_i and vertex v_j:

$$E_{ij}: = \{v_i, v_j, r_{ij}, w_r\}$$

Where:

- v_i and v_j are the two vertexes that the edge E_{ij} connected; and i is the distance of the vertex v_i to the initial vertex v_0. And the distance of the v_i and v_0 is always regarded as the jumps in nearest path of the two Vertexes.

– r_{ij} is the relationship between the two vertexes, generally it can be categorized into the following categories: *hyperonymy* (Mount Tai is-a Mountain, donated with @), *hyponymy* (hyperonymy's reverse, donated with anti@), *meronymy* (computer has-a hardware, donated with %), *holonymy* (meronymy's reverse, donated with anti%), *similarity* (love similar-to like, donated with $). Including the relationship above, there are still some a little more complicated ones. (1) *Attribute*: Mount Tai is a node of ontology WORLD-HERITAGE, with an attribute named LOCATION whose value is Tai'an, a city of China. So Tai'an is *attribute-of* Mount Tai. And Attribute is donated with #; (2) *HasAttributValue*: In opposition to the Attribute, Mount Tai *HasAttributeValue* Tai'an, which is denoted with anti#. (3) *Instance*: to describe the relationship from class in ontology to its instances; (4) *Class*: to describe the relationship from an instance to its class in the ontology.

– w_r is the weight assigned to the relationship r.

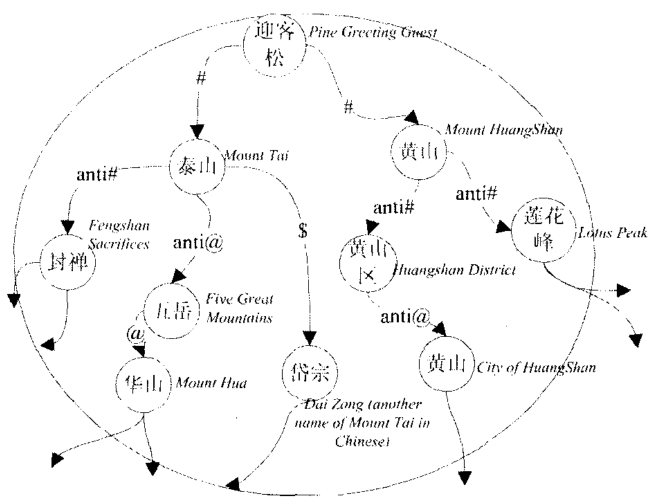

Fig. 2 part of a semantic diagraph with the initial vertex "迎客松" (means the Pine Greeting Guest)

3.2 Semantic Distance

Semantic Diagraph is established directly on the connection in the lexicon, named entity database and the ontology. The difference between two vertexes in semantic is mainly related with the two variables: the distance of the two vertexes in the nearest path, the relationshipbetween every two connected vertexes in the nearest path.

We use the semantic distance to express the semantic difference and relation between vertexes. Consulting the study in [10] on the similarity computing based on *HowNet* that the further the distance of the two vertexes is, the more dissimilar the two vertexes are. So we define semantic distance between the initial vertex and any other vertex v_i in the semantic diagraph as follows:

Definition 3: The semantic distance between the initial vertex and any other vertex v_i:

$$S_i = (-1) \cdot \log_2\left\{\prod_{n=1}^{i} T_n \cdot \left(\frac{k}{i+\lambda}\right)^m\right\} + 1$$
$$= -\sum_{n=1}^{i} \log_2 T_n - m \cdot \log_2\left(\frac{k}{i+\lambda}\right) + 1 \qquad (1)$$

Where T_n is the weight of the relationship between vertex v_{n-1} and v_n in the nearest path from v_0 to v_i, which is various and confined within $[0, 1]$, especially when $i=0$, $T_n = 1$. k and λ are integer and various. m is an attenuation, which is integer, various, and $m > 2$.

For any of the two vertexes in the diagraph, assuming v_i and v_j, the semantic distance is defined as follows:

Definition 4: Assuming $i > j$, the semantic distance of the vertex v_i and v_j in the semantic diagraph:

$$S_{ij} = S_i - S_j = \sum_{n=1}^{j} \log_2 T_n + m \cdot \log_2 \frac{k}{i+\lambda} - \sum_{m=1}^{j} \log_2 T_m + m \cdot \log_2 \frac{k}{j+\lambda} \qquad (2)$$

Where T_n is the weight of the relationship between vertex v_{n-1} and v_n in the nearest path from v_0 to v_j.

4 Ontology based Query Expansion in Vertical Search Engine

In this section we'd like to introduce a novel algorithm that applying the domain knowledge and the semantic diagraph to select the expanded terms. Shortly the algorithm is as follows:

Assuming Q is the initial query,

(1) Shallow Semantic Analysis: utilizing the shallow semantic analyzer, we obtain the attribute pairs from Q and then delete the matched items from Q. Let the attribute pairs are A.

(2) Based on*onomasticon* in the domain knowledge source, we obtain the named entities N from the remaining of Q, and delete the matched items.

(3) Let Q' be the remaining Q' $= \{w_1, w_2, w_3, \cdots, w_n\}$ is the initial query, and w_i is the ith of all possible word combination in Q' (stop words are pruned as usual)

(4) For each attribute pair in$A$, and each item in Q', we construct the semantic diagraphs according to the definition of the semantic diagraph in Section 3.

And then we obtain semantic diagraph collection:

$$SD = \{sd \mid invertex(sd) = i, i \in A \text{ or } i \in Q'\} \qquad (3)$$

Where*inivertex* (*sd*) $= i$ means that the semantic diagraph *sd*'s initial vertex is *i*.

When building the semantic diagraph, rule below should be followed:

Rule 1: to generate the sub-nodes of v_i in a semantic diagraph based on the relationship named *Attribute* mentioned in Section 3, if v_i is a node in the supporting ontology or in the supporting category tree, all the nodes in the ontology that have the attribute's value equal to v_i or its sub-nodes should be added into the semantic diagraph as the sub-vertexes of v_i;

As showed in Figure 2, it is part of a semantic diagraph created in our test environment with an initial vertex "迎客松" in Chinese (means the Pine Greeting Guests).

(5) Calculate the semantic distance between any vertex and the initial vertex in each semantic diagraph, according to the equation (1) and (2), and compare it with the threshold set before, select the vertexes, whose semantic distance to the initial vertex is smaller than the threshold, as the items to be added.

For each semantic diagraphsd_i, we can obtain the candidates:

$$EP_i = \{v_j \mid S_j < S', v_j \in sd_i\} \tag{4}$$

WhereEP_i means candidates gotten from the semantic diagraph sd_i; S_j means the semantic distance between v_j and the initial vertex of sd_i. S' means that the threshold set ahead; and the v_j means the vertex.

(6) Combine all temporal result to get the final result EPQ:

$$EPQ = \cap EP_i \tag{5}$$

When applying the combining, somespecial case should be pay attention to:

Rule 2: when $v_i \cap v_j$and $v_i$and $v_j$are both nodes in the main ontology or in the supporting ontology:

If v_i is sub-node of v_j, $v_i \cap v_j = v_i$, else $v_i \cap v_j = v_j$

Rule 3: when $v_i \cup v_j$and $v_i$and $v_j$are both nodes in the main ontology or in the supporting ontology:

If v_i is sub-node of v_j, $v_i \cup v_j = v_j$, else $v_i \cap v_j = v_i$

Rule 4: when $v_i \cap v_j$and $v_i$and $v_j$are not both nodes in the main ontology or in the supporting ontology:

If $v_i$is the same to v_j. $v_i \cap v_j = v_i = v_j$; else null.

(7) Submit the last result of the query expansion, EPQ, to the search engine. And get the result.

5 Experiments

In this section, we introduced the environment of our experiments, and evaluate the effec-