The Agent of extracting Internet Information with Lead Order

Zan Mo [1], Chuliang Huang[2],and Aijun Liu[3]
1 Associate professor at School of Economic and Management,
Guangdong University of Technology, 510090 Guangzhou, China
Email: mozan@126.com
2 Graduate student at School of Economic and Management,
Guangdong University of Technology, 510090 Guangzhou, China
Email: chuliang750@163.com
3 Graduate student at School of Economic and Management,
Guangdong University of Technology, 510090 Guangzhou, China
Email: laijnking@gmail.com

Abstract. In order to carry out e-commerce better, advanced technologies to access business information are in need urgently. An agent is described to deal with the problems of extracting internet information that caused by the non-standard and skimble-scamble structure of Chinese websites. The agent designed includes three modules which respond to the process of extracting information separately. A method of HTTP tree and a kind of Lead algorithm is proposed to generate a lead order, with which the required web can be retrieved easily. How to transform the extracted information structuralized with natural language is also discussed.

1 Introduction

In the era of e-commerce, only the one who can quickly access and distinguish information can gain business opportunities. Currently, Internet has developed into the world's largest information base and the main channel of global e-commerce, of which the WWW (World Wide Web) develops most rapidly. WWW offers users the information they need in the form of hypertext, including technical material, business information, news, entertainment, and other information of different types and forms. The information constitutes an unusually huge heterogeneous, open distributed database. Searching information which we are interested in such a large ocean of data is very difficult. It needs some smart technology of information extraction then, which is also the urgent bottleneck of the process of carrying out e-commerce.

The agent, developed in recent years, has been proved as the right professional assistant in this field. It can access specific information from the WWW, and arrange the information into the forms we need, such as: collect and collate information, manage

financial affairs, health consult, tour guide, etc. This paper describes a knowledge-based information extraction agent, which can extract meaningful information from the Internet website on the support of domain knowledge [1-4].

In the section 2 of this paper, we describe some common questions for extracting information from websites, which were caused by the non-standard and skimble-scamble structure of Chinese websites. For dealing with the above questions, we divide the process of extracting meaningful information from internet into three steps in the section 3, and try to design an agent which includes three modules responding to these steps. In the section 4, firstly, we research on the method to generate the Lead based on HTTP Tree in detail, through which users can retrieve the required website easily, then we extract information with the method of DOM from retrieved web page, and finally we discuss how to structure the extracted information.

2 Question Description

Freitag [5] and Kushmerick[6] pointed out separately in their papers that: Information extraction is a complex problem because many of the electronic sources connected in the Web do not provide their information in a standard way. The representational manner of website information, which is semi-structure, is more complex. The inherent heterogeneous and dynamic properties of www make it difficult to access the information we need. These can mainly be depicted as following:

1) The specialty of information source elements. Website documents, in which there exist hyperlinks, are the basic elements that composing information source. For each document, from the perspective of object model, is a kind of tree structure. This is different from RDS (Relationship Data Base) or OODB (Oriented Object Database), for the basic information source elements of RDS and OODB are consisted of records. For this reason, manipulation and query of information source are different from RDS and OODB.

2) The independency and dynamic properties of information source. With the change of time, the content and mode of information source change as well. Generally speaking, the models of RDS and OODB change little when designs are finished; the main work is adding, deleting and maintaining the records. However, the change of the mode of information source of www is more frequent. In addition to the mentioned above, the hyperlink relationship of documents often changes quite a lot.

3) HTML is a tool to format information. Browser can explain HTML clearly and express it exactly. However, browser itself knows nothing about what the information is. That is to say, the machine itself can not understand the content of what HTML displays. However, we can understand the content of information very well with its help.

4) Information is expressed in manner of words or tables, but it is not convenient to be extracted by other programs or structure method.

For the first problem, it needs a suitable way to describe the information source which takes the document as a basic element. No. 1 and No. 2 issues are interrelated. For the dynamic varying information source, it needs to generate an information describing method quickly, which can adapt to the changed structure. This paper uses the HTTP tree to describe the relationship between web pages and the method to retrieve the web page. Even if the relationship between websites has changed, we are still able to use the HTTP tree to generate a new method to find out the websites we need. For the third problem, we can describe the required information from the websites through the document object model. Document object models are mainly used to describe the structure and content of each element inner HTML. The content or structure in each element is not limit. As for the content, it can be any usable data model, such as text, graph, sound and cartoon, etc. While for the structure, elements may be composed by simple structure and complex structure. Simple structure means atom element, while complex structure compound

elements. Each compound element consists of many sub-elements, and sub-element may also be atom element or compound element. Thereby, it is feasible to use dot '.' to describe the websites which have hierarchical structures. For the fourth problem, we can structure the text information by using the concept node method with the support of domain knowledge.

3 Design of the Agent Model for Extracting Information

Base on the above analyses, the process of extracting meaningful information from internet can be divided into three steps. Firstly, we retrieve the website we need, then extract the needed information and finally structure it. Through these three steps, information on the fixed website can be packaged as a fine source with the fine structure. So, we design an agent, which includes three independent modules. These modules are responding to these three steps to deal with the information. And they can be modified properly to reuse, even if their application domain has been changed.

Fig.1 has shown the details of the theoretic model of information agent, which can be described as following:

Phase 1. The agent module 1 connects the website sources via protocol such as HTTP (Hypertext Transport Protocol), and gets the web URL, which represents a pointer to a resource on the WWW. Then it retrieves the web page with the support of the HTTP Tree model with Lead Order.

Phase 2. With the method of document object model (DOM), the module 2 extracts information from the resulting website which resides on a temp base, and then resides on the temp base again.

Phase 3. The transform module constructs the resulting information with the natural language, and finally gets the information that we are interested in.

4 Hypertext Digraph and Lead Order

Define 1. Digraph D is defined as a couple *(V, U)*, where V is a nonempty set, and its elements are called **peak**; ***U*** is a subset of the order-set $V \times V$, its elements are called **arc**. The two ends associated the arc have a certain order, **arc** *(u, v)* is different from **arc** *(v, u)*.As for **arc** a= *(u, v)*, we can call that ***u*** is the jumping-off point of the arc while ***v*** is the end-point.

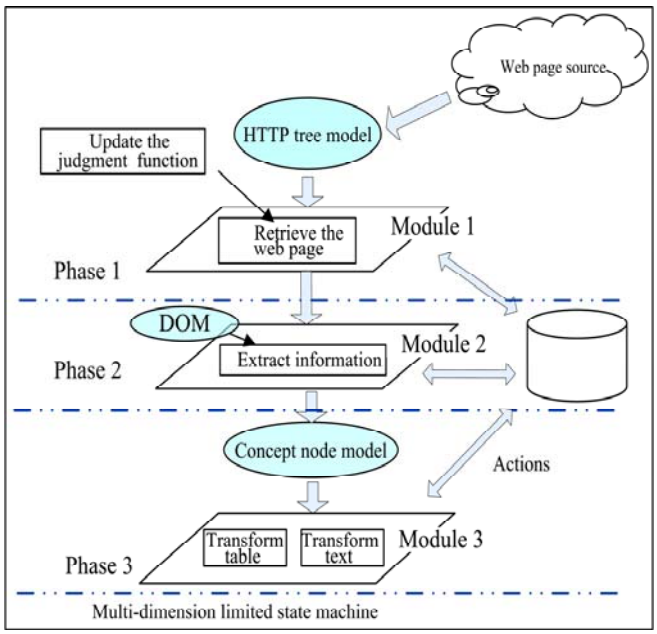

Fig. 1. The theoretic model of information

As shown in the chart A of Fig. 2, document which obeys to HTML syntax is called a ***peak***, and the link relationship between two documents is called an ***arc***. This link relationship has two forms to show, one is "**Get**" method of HTTP, and the other is "**Post**" method of HTTP.

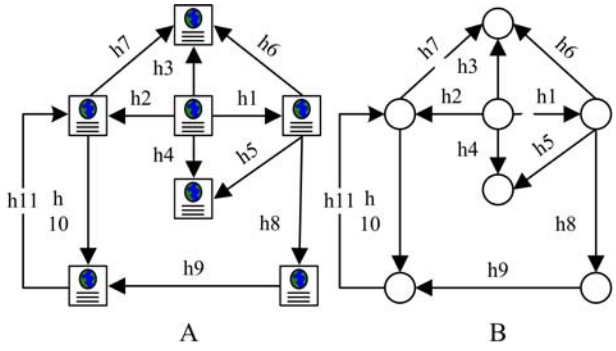

Fig. 2. Te Hypertext Digraph

Define 2. If each arc of Digraph D has been labeled with link method, then the Digraph D is called **Hypertext Digraph**. Generally speaking, the method to link two websites can always be either **Get** or **Post**. For example, if web V_0 and V_1 are linked with the Post method, then we can express this link as ***Post*** (V_0, V_1), otherwise, ***Get*** (V_0, V_1).

If an arrow diagram can be drawn from the start point to the end for each **arc** ***(u, v)*** in the Hypertext Digraph with ***h*** labeling the corresponding arc, the Hypertext Digraph D can be expressed by a geometrical figure as chart B of Fig. 2.

Define 3 The number of arcs which begin with the peak v is called **out-degree**, marked as $d_{p^+}(v)$.In Hypertext digraph, the number in the set which includes all non-local website pages linked by the initial web page with post method is out-degree, marked as $d_{p^+}(v)$.

Define 4 The number of arcs which end with the peaks ***v*** is called **in-degree**, marked

as $d_p^-(v)$ .In Hypertext digraph, the number in the set which includes all non-local website pages linked by the initial web page with Get method is in-degree, marked as $d_p^-(v)$;

Define 5 In Hypertext digraph, the number of all local website pages linked by the initial knowledge website with Get method is called **sub-degree**, marked as $d_c(v)$.

As Fig.3 shows, web ***V*** is the initial knowledge web, it links three non-local websites V_{g1}, V_{g2}, and V_{g3}, with the Get method, and links two non-local websites V_{p1}, V_{p2} with the Post method, and meanwhile, it links three local websites V_{c1}, V_{c2}, V_{c3}, which have relation with the initial knowledge web. So we can say, for the initial web ***V***, $d_p^+(v)$ is 3; $d_p^-(v)$ is 2 and $d_c(v)$ is 3.

In the Digraph, a triple ***h*** is used to label each corresponding arc, and regarded as a link method of HTTP. The ***h*** can be expressed as following:

{HTTP Method，URL，Parameter}

There are two HTTP Methods: ***Post*** and ***Get***. URL is Uniform Resource Locator pointing to documents. ***Parameter***, standing for ***h***'s parameter, is a quadruple :{[$d_p^+(v)$], [$d_p^-(v)$], [$d_c(v)$], [*child.h*]}, whereas, $d_p^+(v)$, $d_p^-(v)$ denotes the aimed web pages' out-degree and in-degree separately , $d_c(v)$ denotes the aimed web pages' sub-degree, and [*child.h*] denotes the HTTP method to link the aimed website and its sub-page.

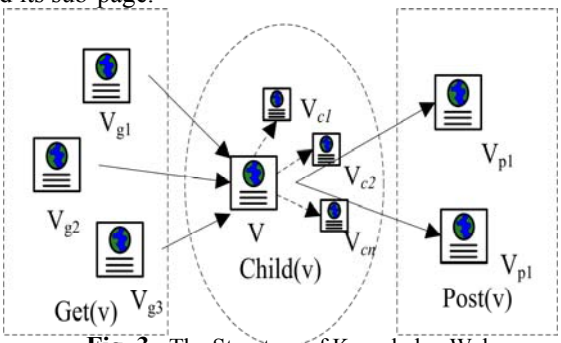

Fig. 3. The Structure of Knowledge Web

Define 6 The limited non-empty sequence $w = v_0 h_1 v_1 \ldots v_{k-1} h_k v_k$, where the items are alternate with ***peak*** and ***arc***, the start point of h_i is v_{i-1}, and end point is v_i （$i = 1, 2, 3 \ldots k$）. if there is no same arcs in ***w***, the ***w*** can be called a direct chain, where v_0 is the start point, v_{ki} is the end point, and ***k*** is its strength, namely, ***q*** （v_0，v_k） is a direct chain. The direct chains with non-similar peak are called direct paths, and the direct chains which have superposition of start point and end point are called direct loops. If there is a direct ***path (u, v)*** in Digraph **D**, we can regard that the peak ***u*** can reach to ***v***. So, $h_1 h_2 \ldots h_k$ are called the ***Lead*** of direct ***path (u, v)***.

Suppose **D** ***(u, v)*** was a Hypertext Digraph, where ***u*** and ***v*** are two random peaks, then we can get the following natures based on Graph Theory:

Nature 1 *If* ***u*** can reach ***v*** and vice versa, then ***D*** was strongly connected or bi-directionally connected.

Nature 2 *If* ***u*** can reach ***v*** or vice versa, then we say that ***D*** was weakly connected or single-directionally connected.

Nature 3 If there is a peak ***w*** to reach ***u*** and ***v*** for each couple peak ***u*** and ***v***, then ***D***

was drafted strongly connected.

If the HTML documents are organized according to strongly directed or weakly directed digraph, then it is much easier to access the HTML documents. So the required web page can easily be found in a website by constructing a series of lead. The ***Lead*** is ***h*** :{ GET, URL, NULL}. If the required website can't be reached directly, according to Nature 3, then we should consider using peak ***w*** to reach the required web, thus a Lead should be generated to guide the user to reach the web.

In view of the character of web, Lead order can be generated by using the improved *Kruskal* algorithm [7]. The central thought of the Lead Order algorithm is that: firstly, take the pointed peak V_0 as the initial peak, then choose the point which has the maximal out-degree and associated with V_0 without loop with the Lead order—if there are two or more points having similar maximum of out-degree or in-degree, we can choose the point which has maximal sub-degree, then add it into Lead order. Through several iterations, the algorithm stops until the degree is less than the pointed variable. Finally the Lead order is generated as this form:V_0{*Get, URL,*{ $d_p^+(v_0)$ *=5}}*V_1{*Post, URL,* { $d_p^-(v_1)=3$ *}}*V_2{*Post, URL,* { $d_p^-(v_1)=3$ *,child1.*{*Get, URL,* $d_p(v_{2.1})=3$ *} }}……$V_{k-1}h_kV_k$.

Fig. 4 has described the pseudo code of Lead algorithms.

The lead mentioned above could only find one website that we are interested in. However, in some cases, the interesting webs are a correlated group set. We need not construct all leads for every interesting website, but just construct a common lead order with ***wild-card***. By using the common lead order, the HTML documents can be leaded from one to many. According to the data matched with wild-card, the common lead is divided into two kinds: one matches the document object model element, and the other matches all value range of the node parameters.

We can construct a lead ***h*** as following:

h = { Get , document.div[0].table[0].tr[0].td[0].ol[0].li[i:].p[0].a[0].href , parameter }*

Where,***** is a *wild-card*, which expresses the sub element 'li' of ***document.div[0].table[0].tr[0].td[0].ol[0]*** in document. For example: ***li[i:1-5]*** expresses sub element of the first to fifth li. Specifically, the wild-card ***** expresses all sub element of ***ol[0]***.

```
Procedure: Lead Order
Lead: = the empty set of Lead order
V0:= the pointed initial peak
While d_p(v_i)<=φ (φ>=0) //****φ denotes the
                          threshold variable, and is a
                          integer***//
BEGIN
  V_i: = the point has maximum of d_p(v_i) and
       associated with V_{i-1} without circle with Lead,
  IF (Maximum d_p(v_i) is not unique)
  Then: Choose the peak has maximum d_c(v_i) of V_i
End
  Add V_i into Lead
  Lead:= Lead has been added V_i
End
```

Fig. 4. The pseudo code of Lead Algorithm

If the names of required documents are fixed, the agent can directly get the name and remember it, achieve the HTTP method of those documents. However, the names are not fixed in many cases, so the URL should be surely dynamical. Thus, we can use some key words as the searching form of leading URL, and combine the lead and the keywords to get the method to actually express a serial of interested webs. This method can resolve the problem that caused by the changeable name of the document.

5 Structuring the Text Information

With the help of the lead order and document object model, the agent can transform the interested information into a form of pure-text, and analyze these texts by method---- concept node, which combines mechanical matching method (MMM) and characteristic dictionary method (CDM), and can be understood by natural language[8].

5.1 Mechanical Matching Method (MMM)

The basic thought of *MMM* is: building a dictionary includes all words in advance, segmenting the sub-string of ***S*** according to a certain confirmed principle for the designated non-segmentation characters strings ***S***. and then if the sub-string matches with some lemma in the dictionary, then the sub-string is a word. Continue to segment the remained parts, until the rest is vacant. Otherwise the sub-string is not a word, then return to cutting the sub-string of ***S*** to match.

The data structure of *MMM* is always simple. Generally speaking, the dictionary can be divided into basic dictionary and professional dictionary. In order to improve efficiency, they can be subdivided into the single-character dictionaries, dual-word

dictionaries, ternate-word dictionaries, four-word dictionaries and multi-word dictionaries etc. As to MMM, the lemma in each dictionary is very simple. It only need to record its inner expresses, and don't have to attach other information.

This paper makes use of *MMM* to analyze some phrases, and the practice shows that it is more effective to analyze the little character string ***S*** which has known the meaning of this method.

5.2 Characteristic Dictionary Method (CDM)

The basic thought of the *CDM* is: building a characteristic dictionary in advance, which includes various words with segmentation character; segmenting ***S*** into several sub-strings according to characteristic dictionary for the designated non-segmentation characters strings ***S***; segmenting every sub-string by *MMM* separately. Since each sub-string is shorter than ***S***, the last problem in *MMM* mentioned above can be resolved.

The theoretical foundation of the *CDM* is that: Though the modal symbol of Chinese is not as abundant as western languages such as English, there are still some symbols in Chinese. These symbols offer the important basis for segmenting Chinese, and they can be used to segment automatically. Generally speaking, various affixes (including prefix and suffix), functional words and overlap-words, etc. can be regarded as the segmentation character. Though their quantity is limited, it is feasible and effective to separate and dispose them at first because frequency of their utilization is commonly high.

Since different kinds of characteristic word often require different treatments, the lemmas of characteristic dictionary have to record not only their inner expressions, but also their types. Generally, the scale of characteristic dictionary is not large, so the dictionary can often be folded into memory once, and can be order by its utilization frequency. So the words are segmented according to descending order of frequency.

The basis of choosing characteristic word is word formation of Chinese grammar or sentence formation in concrete language environment, etc. However, there is also some exceptive phenomenon in Chinese. As to this, it should consider as comprehensive as possible while building the characteristic dictionary, that is to say, it should estimate various exceptive situations for special-purpose processing

Since each lemma of dictionary is often an abstract of several words, segmenting these words has no unitary disposal. As a result, there is no need to include these words in the dictionary of MMM, as which can not only economize the space but also accelerate the speed of searching.

A common character of the two kinds of segmenting methods provided above is considering the word's form alone. However, each word in Chinese also has morphological features and the meaning besides form. In addition, the morphological feature and the meaning of the adjoined vocabulary must be consistent; otherwise, it will not conform to the grammar or illogic. In other words, morphological features and the meaning of adjoined vocabulary must satisfy a certain restraint relationship. These restrained relationships are important basis to judge whether the automatic segmentation result is right or not, so it should be present in the segmenting method possibly.

6 Conclusion

In this paper, we have proposed some common questions for extracting information from websites. During dealing with these questions, we divided the process of extracting information from internet into three steps and design an agent. Then we research on the method to generate the Lead order based on HTTP Tree in detail, through which users can

retrieve the required websites easily. We extract information with the method of DOM from retrieved websites, and finally we discuss how to structure the extracted information.

Acknowledgment

This work was supported in part by the Guangdong science fund under Grant No06300278 and Doctoral Subject point special fund for Guangdong University of Technology under grand 053019.

References

1. Maes P., Moukas A., Amalthaea: An Evolving Multi-Agent Information Filtering and Discovery System for the WWW, Autonomous Agents and Multi-Agent Systems, 1(1)1998.
2. Hamdi, M.S, Information extraction using multi-agents, International Conference on Internet Computing - IC'03, pt. 1, pp.77-82,Vol.1(2003).
3. Arpteg, Anders, Multi-page list extraction: An agent-oriented approach to user-driven information extraction, , 2005 International Conference on Integration of Knowledge Intensive Multi-Agent Systems, KIMAS'05: Modeling, Exploration, and Engineering, v 2005, 2005 International Conference on Integration of Knowledge Intensive Multi-Agent Systems, KIMAS'05: Modeling, Exploration, and Engineering, 2005, pp.431-437.
4. Vlahovic, N. , Application of information extraction using information management agent for Croatian financial markets, WSEAS Transactions on Business and Economics, v 3, n 5, May 2006, pp.434-441.
5. Freitag, D. (1998). Information extraction from HTML: Application of a general learning approach. In Proceedings of the 15th National Conference on Artificial Intelligence (AAAI-98). Menlo Park, CA: AAAI Press.
6. Kushmerick, N., Weld, D. S., & Doorenbos, R. B. (1997). Wrapper induction for information extraction. In Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI'97),pp.729–737.
7. Kenneth H.Rosen.Discrete Mathematics and Its applications (4^{th} edition), 2002.11 McGraw-Hill College,pp.473-479.
8. Grosz, Barbara. The Contexts of Collaboration. In K. Korta, E Sosa, eds. Cognition, Agency and Rationality, Dordrecht: Kluwer Press, 1999, pp.175-188.