Available online at www.sciencedirect.com

SciVerse ScienceDirect

Procedia Engineering 15 (2011) 3058 – 3062

Procedia Engineering

www.elsevier.com/locate/procedia

Advanced in Control Engineering and Information Science

The Design of Intelligence Collection System Based on Internet

Xiaojun Liu*

Basic Department,the chinese people's armed police force academy, Langfang 065000,China

Abstract

Internet has become an important way of collecting intelligence for its richness of resources and timeliness. Based on the features of collection and retrieval of public intelligence, an intelligence collection system using knowledge base and user interest model is developed. The system can automatically collect intelligence according to authoritative websites or designated websites, create index after information processing including word segmentation, information filter and conversion, and realize full-text retrieval. The system large scale enhances the timeliness and accuracy of intelligence collection.

Selection and/or peer-review under responsibility of [CEIS 2011]

Keywords: Intelligence; Search engine; Knowledge base; User interest model

1. Introduction

The intelligence collection is not only the material base of intelligence practice but the premise and condition of intelligence research. With the development of computer network, internet has become one important way of public intelligence collection [1]. However, as a source of massive information, although the internet is an open and widespread information space, some of its inherent characteristics

Corresponding author.Tel.:+86-316-2068423
E-mail:wjxy_lxj@hotmail.com

doi:10.1016/j.proeng.2011.08.573

already have hindered the on-line people from using information resources fully. Directing at the characteristics of open information collection and searching on the internet, the paper developed a subject-based search engine on the knowledge base and users' interest model.

2. The Principle of Search Engine

The search engine locates at the bottom hierarchy of web information. It regards the web information as processing object, and provides users and information retrieval agent with retrieval service. It mainly contains five basic parts: spider, analyzer, indexer, retriever and user interface (UI).

The function of spider is roaming and collecting information in the internet in accordance with hyperlinked queue or the stack maintained by the system. The spider downloads the web pages from the initial URL included in the stack, extracts new hyperlinks and adds them to the queue. The above process is repeated until queue (or stack) is empty. In the realization of the spiders, the distributed parallel computing technique is used to speed information finding and update.

In order to establish index, analyzer firstly analysis the document downloaded by spiders. Generally document analyzing techniques include words segmentation, filter and conversion, etc, which are often related closely to the specific language and system-indexing model. In words segmentation, most systems extract entries from the whole text, but others also from some part of one document (for example, title and header). Entries also have a variety of types excepting word or phrase. Usually after words segmentation, banned word table is used to remove high frequency entries. Some systems make changes about entries: the plural or single conversion, affix removal, synonyms conversion, etc.

Indexer extracts index entries which are used to indicate the document and generate the index table of document libraries. High quality indexer is one of the key factors to successful retrieval system of web information. A good index model, with speedy retrieval and small storage space, is easy to realize and maintain. Search engine generally draws lessons from such retrieval models in the traditional information search as the index of inverted file, vector space model, probability model and so on.

According to the user's query, retriever rapidly detects the documents in the index base, makes the relevance evaluation between documents and inquires, sorts the output results and achieves the relevance feedback to a user.

The user interface provides users with visual inquires input, displays inquire results and offers users the relevance feedback. Its design and implementation make use of the theories of human-computer interaction in order to fully adapt to human habits of thinking. User input interface can be divided into simple interface and complex interface. Simple interface only provides text box in which users input query strings. Complex interface allows users to make such limitation on inquiry as logic operations, close relations, the scope of domain name, appearance position, information time, length, etc.

3. The Realization of Subject-Based Search Engine Technology

At present, the comprehensive search engine is not very satisfactory in both recall and precision, and the timeliness of inquire results even hardly meets the demand of information users. The "information overload" problem caused by search engine limits the application of search engine [3]. The subject-based search engine applied to the intelligence collection, shown in Fig1, can make a pertinence collection in designated websites or the authoritative websites. The subject-based search engine is characterized as simple use, high efficiency, high precision and timeliness, etc.

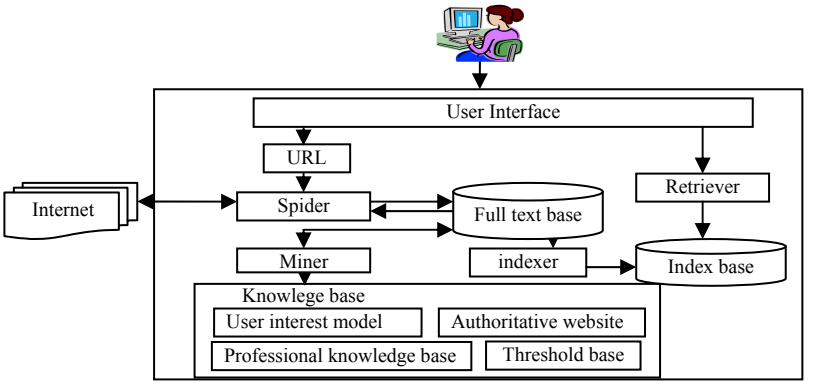

Fig.1. The diagram of working principle of system

3.1. Spider

Spider, receiving web site table (website*1*, website*2*, … webisite*n*) from the authority websites of the repository, analyzes and processes all the relevant link pages from the beginning of URL. The working process includes getting a WebSite, downloading the page to memory, and making the following analysis:

- judge the linking pages: according to the threshold *k* from threshold base, to judge whether the searched web is the content page or the linking page. If the linking page, extract hyperlinks; if the content page, judge whether it is relative to the user's interests. Dim *D* as average link length of HTML webpage. The methods of judging linking pages are shown as follows:

$$Am(D) = \frac{Lengh(D)}{NumberofHyperlink(D)} \tag{1}$$

NumberofHyperlink is the number of hyperlinks; *Lengh (D)* is the length of HTML document *D*. If *Am(D)* less then the threshold *k* the web page is a link page.

- extract hyperlinks form the link page: according to the tag <href> of the link page in the HTML document, the hyperlinks contained in the document are extracted. The phase of process also includes the conversion from relative URL to absolute URL in the page. URL and its content are include in a pair of tag: <a> and in HTML document, such as:

[Case] ARUND THE NATION; MAN GETS 10 YEARS FOR SMUGGLING COCAINE.

The HTML code is ARUND THE NATION; MAN GETS 10 YEARS FOR SMUGGLING COCAINE. The HTML code shows that the title information in hyperlink is the words between the tag and , that is, ARUND THE NATION; MAN GETS 10 YEARS FOR SMUGGLING COCAINE. The hyperlink is "http://www.nytimes.com/1986/12/06/us/arund-the-nation-man-gets-10-years-for-smuggling-cocaine.html?ref=smuggling", the content of the two parts in which is just the paper intends to extract.

The browser provides a HtmlDocument2 interface that can make developers easily get the URL and content of link, so this paper puts forward a web-linking extraction algorithm based on HTML tags and keywords. Set the output URL and the title as *T*, the page content of downloaded as *W*, the extracted document as *LHTML*, scanning variable as *i*, the lines of *LHTML* as *L*, the intermediate variable for HTML document as *t_text* and *tL* in length, the searching keywords as *KeyWord*. Hyperlink extraction algorithm is described as below:

Download *W*

LHTML←IHtmlDocument2(*W*).getElementsByTagName('a')
i=0
While *i*<*L*
if *KeyWord*□*LHTML.Line.String*(*i*)
t_text←*LHTML.Line.String(i)*
delete the tags of HTML
i=*i*+1
End
i=0
While *i*<*tL*
T<=acquire the hyperlink
i=*i*+1
End
T<=*IHtmlDocument2(t_text).outText*

- extract the text of content page and judge the relevance between the text and user's interests: according to the keywords vector provided by users, document content is judged and classified. Different types of materials are classified. Documents and Web pages are also classified and identified according to the different user's interests. For image, video and other media forms, the source and subject are identified in the database system according to media types in different locations.

According to the HTML language designing code, the title of web page is tagged as <TITLE></TITLE>; the content basically as several < P > < / P >. Therefore, in the process of extracting the document content, the characteristics of HTML document can be used to extract the content tagged as <TITLE>and<P>. The text relevant to the subject can be ultimately obtained after filtering the extracted results and removing the HTML tags. Besides, the document content is not only tagged as <P> but as . So the situation should be taken into consideration in designing the extraction algorithm of document content.

The relevance between the extracted URL and the subject has been judged before the implementation of web information extraction. Nonetheless, the extracted content from a web page may vary greatly with the set subject. This phenomenon will affect the extraction accuracy of subject page information. So the relevance between page and subject needs to be judged in order to filter out the irrelevant pages.

In this system, the vector space model is used because of its strongly handling capacity and easy flexibility. Dim *D1* as subject vector and *D2* as distinguishing page vector, it will be

$$Sim(D1, D2) = \frac{\sum_{k=1}^{N} W_{1k} * W_{2k}}{\sqrt{\sum_{k=1}^{N} W_{1k}{}^{2} \sum_{k=1}^{N} W_{2k}{}^{2}}} \tag{2}$$

Compare the value size of *Sim (D1, D2)* with threshold *d*: if *Sim (D1, D2)* greater than or equal to *d*, it means that pages are related to the subject and saved in the database. Otherwise the page should be discarded.

3.2. Indexer

The anchor of HTML documents can describe the linking document. Each standard HTML document also has title logo. Generally speaking, those tags are very good to instruct and summarize the link

documents. Therefore, after the web page is downloaded by subject search engine, the title and the anchor are extracted from each HTML document and taken as the indexes. The indexes and documents are stored together in the corresponding field of the data bank.

3.3. Retriever and user interface

Searcher includes retriever and user interface. The system realized two modes of users' retrieval and browsing:

- The collected intelligence data are stored in the corresponding list of storage by concentration and in classified files. Index server provided by windows system automatically establishes index for the specified files and realizes the retrieval and browse with the man-machine interface designed by this paper.

Advantage: Expansibility is great, for the original material can be easily retrieved and browsed rather than undertaking the work on index.

Disadvantage: Flexibility is not enough, for reclassification/recluster cannot realize according to the new classification structure.

- The collected data are stored in a database whose corresponding classification information is saved with one field. It is friendly for users to make full use of the full text index provided by SQL server and the index generated from title and anchor in processing.

Advantage: In view of the different approaches and structures of classification, the classification category of documents can be flexibly changed.

Disadvantage: Scalability isn't very good, for the existing data and the intelligence material cannot be indexed and browsed unless there is a special design for an artificially input interface or an artificial entry.

4. Conclusion

Based on analysis of the structure of searching engine, this paper develops the subject-based searching engine which is specially applied in the intelligence collection. The system collects the open intelligence in network, with uninterrupted operation and regularly searching, ensures the accuracy, comprehension and speed of collection. The experimental results show that the system searches the HTML document with high precision but video and PDF document with low efficiency. The major reason is that anchor was not accurate enough: the annotation of the documents does not correctly describe the content but uses unmeaning or wrong numbers and characters.

References

[1] Hussey, David. Sources of information for competitor analysis, Strategic Change, 1998

[2] Wang jicheng, Pan jingui, Zhang fuyan. Research on Web Text Mining. Journal of Computer Research and Development 2000; 37(3):513-520

[3] Luan xidao. Internet Information Collection and Processing Technology. National University of Defense Technology 2003

[4] Zhang xunmo. Analysis of Search Engine Design, Computer Engineering & Science 2002;24(2)

[5] Chakrabarti. the Web's Link Structure. Compute. IEEE Computer 1999; 32(8):60-67.

[6] Kleinberg J M,Tomkins. Application of Linear Algebra in Information Retrieval and Hypertext Analysis. the 18th ACM Symp.on Principles of Database Systems 1999-05:185-193.

[7]Wang K, Zhou S,Liew S C. Building Hierarchical Classifiers Using Class Proximity. proc of VLDB' 97, 1999:363-374