# Yellow Page driven Methods of Collecting and Scoring Spatial Web Documents

Takeshi Sagara
Center of Information Fusion, The University of Tokyo
4-6-1 Komaba, Meguro-ku
Tokyo 153-8505 JAPAN
+81-3-5452-6256
sagara@iis.u-tokyo.ac.jp

Masaru Kitsuregawa
Center of Information Fusion, The University of Tokyo
4-6-1 Komaba, Meguro-ku
Tokyo 153-8505 JAPAN
+81-3-5452-6254
kitsure@tkl.iis.u-tokyo.ac.jp

## Categories and Subject Descriptors
H.3.3 [Information Search and Retrieval]: Information Filtering

## General Terms
Algorithms.

## Keywords
Web, Information Retrieval, Geographic Information.

## 1. INTRODUCTION

The web is thought as a promising source of spatial contents so that some research projects had been reported which try to collect geo-referenced web documents and add spatial index to them. There are some types of explicitly geo-referenced web documents, for example, XML map data, raster map images, HTML documents with embedded coordinate values (ex. ICBM tags), however we are focusing on implicit geo-referenced documents; i.e. HTML documents including location names. We call them "spatial documents".

In this paper, we discuss an efficient algorithm for retrieving spatial documents from the Web, and propose scoring method for sorting the documents. Finally, a prototype system of spatial search engine is developed.

## 2. Crawling spatial documents

The basic algorithm for adding spatial index to spatial documents is quite simple; (a) Parse the document and extract location names. (b) Convert those names into coordinate values such as longitude and latitude. The process (a) is called 'geoparse' and (b) is called 'geocode'[1]. These 2 processes are the most important technologies to exploit spatial documents.

We had already developed practical geoparser and geocoder for Japanese spatial documents. The system was reported as a "spatial document management system (SDMS)" in 2001[2]. The system can add street-block-level spatial index in almost all part of Japan to the documents written in Japanese.

Using SDMS, user can retrieve web documents by combination of spatial query such as range retrieval and full text search. For example, you can ask the system like "Show me all documents which are geo-referenced within 1km distance from Shinjuku station and contain 'restaurant'". However, there are 2 big issues remained to answer this question.

1st issue is how to retrieve web pages which will be geo-referenced in a certain area. Since it is not practical to collect and add spatial index to all web pages in the world, efficient strategy is required to find spatial documents from the web.

2nd issue is how to select documents which satisfy both spatial and keyword conditions. Even though a document is geo-referenced in the queried area and contains queried keyword such as 'restaurant', it does not ensure that the document describes "restaurant in the area", because the document may mention about an office in the area and a restaurant outside of the area. Confirming relevancy between the location name and the keyword essentially requires semantic text understanding.

Furthermore, we should consider that the keyword often represents rather a concept or a category in geographical information retrieval. In the restaurant query, the user would like to get the list of restaurants and their documents, but not the list of documents including "restaurant". In our experience, some web documents introduce restaurants as bar or pub, and they don't contain "restaurant" in their text, thus these documents can't be retrieved by the query.

Therefore, we utilize the yellow page for collecting web pages. The yellow page contains almost all shops with their name, postal address, phone number and category of business. The crawling algorithm is shown below.

**Algorithm 1: Crawling spatial documents.**

```
Step 1.  Pick up a record $\boldsymbol{y}$ from the yellow page.
Step 2.  Make search keyword from name, address, phone number in $\boldsymbol{y}$.
Step 3.  Calculate coordinate values $\boldsymbol{g}$ of $\boldsymbol{y}$ by geocoding the address.
Step 4.  Collect web documents $\boldsymbol{d}_{1..\boldsymbol{m}}$ ($\boldsymbol{m}$: maximum pages) using keyword index.
Step 5.  Check $\boldsymbol{d}_i$, $\boldsymbol{i}$ = 1 .. $\boldsymbol{m}$ and store ($\boldsymbol{y}$, $\boldsymbol{g}$, $\boldsymbol{d}_i$) to the relational database if
    a.  $\boldsymbol{d}_i$ contains correct name and address, or
    b.  $\boldsymbol{d}_i$ contains correct phone number
Step 6.  Go back to step 1 while more record remains.
```

The reason why the document contains correct name will not be stored in step 5 is, it may describes another shop in another place with same name.

This algorithm is relatively fast hence it uses normal keyword index search. We implemented and applied the algorithm actually for about 100 thousand restaurants in Tokyo area and collected about 450 thousand spatial documents (which satisfy one of the conditions in step 5.) in 4 days. The documents can be considered that they really describe ***y*** and their locations can be referenced by ***g***, therefore the relevancy between the location name and the keyword doesn't become a problem. Additionally, the documents can be retrieved by categories of business stored in y.

## 3. Scoring spatial documents

Scoring documents is one of the most important technologies of IR, since user would not like to read all documents. In geographical information retrieval, we divided scoring method into 2 different levels. The 1st level scoring method evaluates "popularity" of each geographic object, and 2nd one evaluates "quality" of each document.

For example, when a user asks about "restaurants near to the station", the answer should be a list of restaurants sorted by some measure, such as taste or reputation. Hence such subjective information is too difficult to extract from the documents, popularity is used alternatively. Conceptually, if there is more number of web documents describing restaurant A than restaurant B, the restaurant A can be thought as more popular on the web than B. More precise definition will be shown at the end of this section.

When the user gets interest in restaurant A, he/she would like to examine by reading spatial documents related to the restaurant. In this case, reliable and informative documents should be presented with high priority. To determine the reliability of web pages, some techniques, which had been already developed based on link relationships such as pagerank, can be applied. In our implementation, reliability can take binary value, i.e. 'reliable' and 'not reliable'. When the pagerank of a document is higher than the threshold, it is considered as reliable. Additionally, we confirmed by an evaluation experiment that documents containing both correct phone number and address are reliable with high possibility. Therefore, when the document contains phone number and address, it considered reliable regardless of its pagerank.

To evaluate the document is informative or not, the number of characters is simply used. However, since some web documents are describing multiple geographic objects sequentially in one page, such as a list of the author's favorite restaurants, sentences which really mentioned about the object must be extracted before counting characters. Avoiding text understanding, we use a heuristic approach to implement the extraction process. First, "spatial block" is defined as a smallest part of web document which contains more than one spatial content, and the block is segmented by any HTML tag. The extraction is done by the block. Second, two assumptions are made.

1. A spatial block contains more than 1 location name.
2. Spatial blocks in a web page are placed in parallel under certain tag-block.

The spatial block can be extracted by the algorithm shown below.

**Algorithm 2: Spatial block extraction.**

```
Step 1.  Create HTML tag tree from the web document.
Step 2.  Mark every node which text contains location names.
Step 3.  Mark every parent node of marked node repeatedly.
Step 4.  Find most upper level nodes which are marked, and it's
         sibling node is also marked. Every marked node in the
         level is spatial block.
Step 5.  Extract the spatial block which contains location name
         focused on.
```

Although the algorithm can work only when the blocks have location names and segmented by HTML tags, it is so flexible that it can extract either a line surrounded by '`<tr>`' and '`</tr>`' tags from a table, or a block surrounded by '`<p>`' and '`</p>`' (Fig. 1).

The quality of a document ***q*** is defined by combination of reliability ***r*** and number of characters in the spatial block ***n*** as;

1. if ***r*** = 'reliable' and ***n*** >= ***th*** then ***q*** := 1.0
2. if ***r*** = 'reliable' and ***n*** < th then ***q*** := ***n*** / ***th***
3. if ***r*** = 'not reliable' and ***n*** >= ***th*** then ***q*** := ***w***
4. if ***r*** = 'not reliable' and ***n*** < ***th*** then ***q*** := ***w*** (***n*** / ***th***)

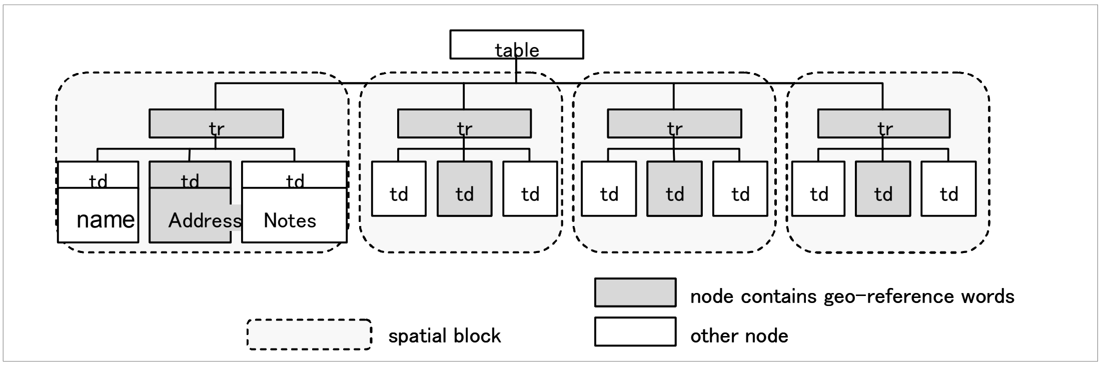

**Figure 1. Extracting spatial blocks from spatial document**

,where ***th*** is a threshold value, ***w*** is a weight (0 < ***w*** < 1.0).

And the popularity ***p*** of a geographic object is defined as total of ***q*** which related to the object.

## 4. A prototype system of spatial search engine: restaurant search

Using proposed methods, a prototype system of spatial search engine was developed. The system can retrieve restaurants by combination of its location, category of business and keywords. The search results will be shown by both a street map and a list of restaurants sorted by their popularity. By clicking a rectangle on the map or name of a restaurant, list of all web documents related to the restaurant will be shown sorted by their quality. The list contains links to their original pages, and text from the spatial blocks.

## 5. Conclusion

In this paper, algorithms for collecting and scoring spatial documents from the web are explained, and a prototype system is developed based on the algorithms. Quantitative evaluation of the proposed algorithms is planned.

(1,486 words)

## 6. REFERENCES

[1] Kevin S. McCurley, *Geospatial Mapping and Navigation of the Web*, Hong Kong, ACM WWW10, 2001,

[2] Sagara, T., Arikawa, M., Sakauchi, M., *Spatial Document Management System Using Spatial Data Fusion*, IIWAS2001, Linz, 2001, 399-409.

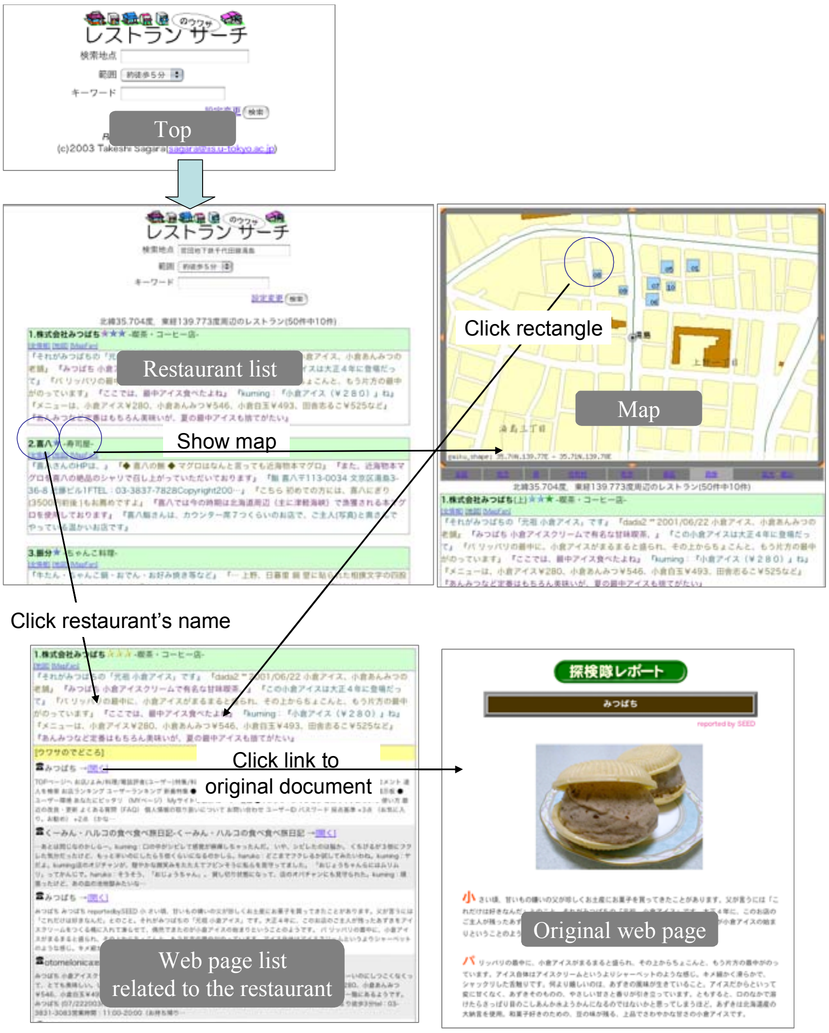

**Figure 2. A prototype system of spatial search engine**