

Calculation of Target Locations for Web Resources

Saeid Asadi¹, Jiajie Xu², Yuan Shi³, Joachim Diederich¹, and Xiaofang Zhou¹

¹ School of ITEE, the University of Queensland, Brisbane, QLD 4072, Australia
{asadi, zxf, joachimd}@itee.uq.edu.au

²S4112087@student.uq.edu.au

³State Key Lab of Software Engineering, Wuhan University, China 430072
yuanshisklse@gmail.com

Abstract. A location-based search engine must be able to find and assign proper locations to Web resources. Host, content and metadata location information are not sufficient to describe the location of resources as they are ambiguous or unavailable for many documents. We introduce target location as the location of users of Web resources. Target location is content-independent and can be applied to all types of Web resources. A novel method is introduced which uses log files and IPs to track the visitors of websites. The experiments show that target location can be calculated for almost all documents on the Web at country level and to the majority of them in state and city levels. It can be assigned to Web resources as a new definition and dimension of location. It can be used separately or with other relevant locations to define the geography of Web resources. This compensates insufficient geographical information on Web resources and would facilitate the design and development of location-based search engines.

Keywords: location-based search, Web search, geographical search engine, target location

1 Introduction

The World Wide Web has caused significant changes in traditional models of publishing, communicating, shopping and so on. Search engines have emerged as the main tool to capture resources on the Web and sophisticated search tools handle hundreds of millions of queries every day. Over 150 million queries are sent to Google and Yahoo each single day from the United States alone [1]. Among different services offered by search engines, location-based search has received significant attention. Location-based or geographic Web search intends to associate Web resources with real locations and offer the results based on the geographical area a query refers to. The Web has made it possible to have world wide access to information which was a desire for many centuries. However, the fast growth of Web-based services emerged in the recent years and demands for location-based

information, has challenged the existing engines. Many queries (e.g. for driving, shopping or accommodation) have geospatial dimensions. Users search for services and facilities in a suburb, city or any other geographic area.

Studies such as [2,3] reveal that a significant portion of queries on existing general search engines have geospatial. So far, general search engines have not been able to answer location-based queries properly and their results are poor for this kind of queries [4]. Lack of geographic information and poor definition and allocation of locations to Web resources is a basic challenge that frustrates any attempts on developing location-based search engines.

This paper sets out to illustrate the challenges of defining the location of webpages and other resources on the Web. We think information extraction techniques can be applied to only a small portion of Web resources and complementary techniques must be employed to include more documents in the scope of location-based search engines. We introduce target location as a reliable way to calculate at least one location for every document on the Web. Target location deals with geographic areas that the users of a webpage belong to. The contribution of this paper is describing how to obtain and calculate target locations for all Web resources including texts, photos and other types of media.

2 Related Studies

Search engines have emerged since 1993 in parallel with the development of the World Wide Web. Major search engines have employed sophisticated techniques to collect as many documents as possible and to improve results effectively by using clustering or ranking models. Unlike general search, location-based search tools have not been used widely. Google Local¹ can search only for local business information in USA, UK and Canada [7]. Other location-based search engines such as SPIRITS, Northern Light and GeoSearch have remained incomplete or they have stopped their limited services. Local search homepages intend to search among webpages with limited domains or languages. For example, Yahoo7² is a search interface for Australia and New Zealand. These services are close to location-based tools but they are still too general and limited to particular domains and regions.

A geographical search engine must present the result in an effective way which is often different from general search. Map-based presentation, geographical ranking and location-based clustering are related techniques. Watters & Amoudi [4] report on an algorithm for re-ranking search results in a geographical approach. GeoSearcher tries to identify the geographic coordinates (latitudes and longitudes) of webpages by assistance of online gazetteers. Having the coordinates, it calculates the distance of selected webpages from a reference point. The results are then ranked in ascending order of distance. The distance-based ranking is a useful model for location-based search engines but it often results in wrongly ordered results as many geographical areas on earth do not have regular rectangular or circular shapes.

¹ <http://local.google.com/>

² <http://au.yahoo.com/>

A significant challenge for location-based search engines is defining and assigning different locations to Web resources. Basically, a location-aware engine must be able to analyze webpages for geographical information. Information extraction rules and techniques have been used to extract addresses and location names from webpages [8,9,12,13]. Olga [10] describes an algorithm that assigns a geographical class to unknown names by adding patterns to a candidate name and sending the results as new queries to same search engines. Based on the number of returned documents, the algorithm determines which category is suitable for a name. Gravano et al. [11] divided Web queries into global and local based on search results. If the best results of a query refer to local webpages, the query is considered a local query; otherwise it will be global. Many webpages do not have addresses, geographic metadata or geographic features and IE techniques are unable to find related locations for these. Ding & Gravano [6] used the link structure of Web to assign locations to Web resources. The *geographic scope* of a Web resource is calculated as a city, state etc. where most of the back links refer to. This method can assign locations to webpages even if they are not mentioned in the text. It can also tag non-textual resources e.g. photos and films with location names. However, as many resources don't have enough back links, the model is not always applicable. Other studies [14,15] have focused on geographical indexing, address and location extraction, and tagging webpages with location names

3 Target Locations of Web Resources

In this section, we first describe different definitions of location on the Web and then introduce target location in detail.

3.1 Web Resource Locations

The location of a Web resource can be defined in many ways. We divide them in three groups: host, content and target locations.

Host's Location. Every website is stored on one or more servers which are located somewhere in the world. Therefore, the physical location of a resource or $L_{(w)}$ is defined as the location of the server which is hosting the resource w :

$L_{(w)}$: location of w 's hosting servers

Host's location is not the best way to define webpage location. Websites usually are hosted on servers which are not necessarily related to a topic of them.

Content Location. By content location we mean location(s) mentioned in a webpage's content including geographic features mentioned in text, footnotes, contact addresses, metatags, headings and so on. In this case, the location of a webpage is the geographical location of its objects or entities:

$L_{(w)}$: location of geographic entities in w

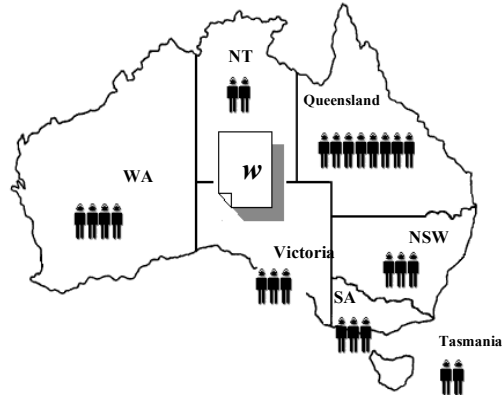


Fig. 1. An example of distribution of a Web resource users in different geographical locations (Here within the states of Australia)

Different algorithms have been developed to extract addresses, postcodes and geographic features. Geo-tagging techniques extract all locations and disambiguate them by using gazetteers or pattern learning approaches. Web-a-Where [5] finds one or more focuses for a page. For example, if Brisbane, Gold Coast and Cairns are mentioned on a page, the focus of the page will be Queensland.

Target Location. Websites are usually designed to serve a specific group of people. This population is often considered in the design and development of the page. Target location is an area a webpage is dedicated to or the location of its users. It can be said that webpages have at least one target which could be a small area like a suburb or town or a very large region. For example, the target location of a page entitled ‘Pizza delivery in Toowong’ can be Toowong suburb in Brisbane, Australia; while the target of Google search engine is the entire world. Target location is defined as:

$$L_{(w)}: \text{location of people that use } w$$

The geographical distribution of the users could be used as the target location of that page. If a website is designed for people of Toowong suburb for example, it is likely that most of the people who use this site be from Toowong or areas around it. As the target locations of many resources are not determined explicitly, a reverse approach can be used to find this information. Suppose that 1000 people have visited webpage w and most of them are from Queensland. Queensland is then a possible target location of w (Fig. 1).

3.2 Sources of Geographical Information

There are different sources for obtaining geographical information from Web resources. HTML tags are a basic source for extraction of location names and

addresses. However, this information is often limited to a portion of textual resources. Link structure of the Web is another source. It can be assumed that a webpage receives more citations from its geographical target area. Ding & Gravano [6] introduced the concept of *geographic scope* which refers to one or more locations that most of the back links to a Web resource come from. The problem is that the majority of resources do not have enough back links to calculate their geographic scope.

We propose *visitor location* analysis instead of link and content analysis or as a complementary solution whenever other methods are not available. A Web resource w is visited or used by people in different geographic areas. Our assumption is that people in the target location of w visit it more than people in other areas. For example, the website *www.qldevents.com.au* that provides information about current events and programs in Queensland might be visited by people in Queensland State more than the people in Victoria. As a result, the target location of this webpage would be Queensland. Visitor information could be a useful way to find a Web resource's target location and it can be more practical than the link-based approach. Similar to geographic scope, visitor information can be applied to all resources and it is not limited to textual documents. Visitor information can be tracked and extracted by analyzing IP addresses which often refer to geographic areas. However, extracting geographic information from IPs is time consuming and requires complex information extraction and normalization steps. Surfing the Web anonymously can also affect the quality of visitor-based approach.

A typical connection to the Internet occurs through an ISP which registers a range of IPs. Geographically, an ISP is located in a city and serves that city or region. When a user visits a website, his/her information is stored in a log file. Log files are stored on hosting servers and contain valuable information about the history of visits to a website. Logs contain the user's IP, date, time and length of the visit and the list of pages and files visited by a user. There are some online databases and which provide information about the IPs. The structured data on these databases tagged as telephone, address, country name etc. provides a useful source of geographic information.

3.3 Measures for Calculation of Target Location

Different measures can be used to aim at the selection of the best locations as targets. We measure *popularity*, *distribution* and *stability* of candidate locations to calculate the temporary and constant target locations of a Web resource.

Popularity. If most of the users of Web resource w are from the location ℓ , it can be said that w is a popular page or document in ℓ rather than other locations. Popularity of Web resources has been measured by [6] using a *power* formula. In a particular location ℓ , *power* is the number pages with link to w compared to the total number of webpages existing in ℓ . The main problem with this formula is that there is no reliable way to find out how many Web resources exist in ℓ . Instead, our *power* formula compares the popularity of w in ℓ to the total number of visitors of w :

$$Power(w, \ell) = \frac{Visitors(w, \ell)}{Visitors(w)} \quad (1)$$

Distribution. Geographical areas often consist of several smaller divisions. For example, a country usually has several states or provinces. Calculations on large locations ignoring their sub-locations can lead to poor accuracy. For example, if many visitors of w are from Australia but all of them are from Queensland State, then it seems that Queensland is a better target than Australia. If they are distributed smoothly over different states, then Australia can be a target location. A vector-space definition of distribution or *spread*, described in [6], computes the similarity between vectors of webpages and vectors of links. Instead of links, we use the number of visitors in a sub-location ℓ_i :

$$Spread(w, \ell) = \frac{\sum_{i=1}^{i=n} \ell_i}{\sqrt{\sum_{i=1}^{i=n} \ell_i^2}} \quad (2)$$

We use spread as a threshold to select few candidates and then use power as a threshold to select temporal targets (TTLs).

Stability. A location is eligible as target if it remains in the scope of a webpage over time. To consider the freshness of webpages, the extracted target locations can receive different time-dependent weights. The more recent a location has been selected as a target, the higher weight it receives. Stability is measured as:

$$Stability(\ell, w, T) = \frac{\sum_{n=t_0}^{n=t_n} \frac{1}{2^n} \ell}{\sum_{n=t_0}^{n=t_n} \frac{1}{2^n} L} \quad (3)$$

where t_i is a subdivision of time T , ℓ is a location selected as a temporal target (TTL) of w in t_i , and L refers to all TTLs selected in t_i . We use stability score as a threshold to select CTLs from TTLs.

Pruning Models. Three pruning strategies are used to choose constant targets (CTLs) from candidate TTLs after measuring their stability:

Top-k Threshold. Given an integer k , constant target locations are selected from the top of the list while candidates are ordered based on the decrease in *stability*.

Fixed Threshold. A fixed threshold of *stability* is defined and candidates above this threshold are selected.

Relative Threshold. A percentage threshold of *stability* is defined and candidates above this threshold are selected.

We use same pruning models to choose temporal targets (TTLs) from the extracted locations. In this case, power is used instead of stability as a threshold.

3.4 The Algorithm

A summary of the algorithm for the proposed visitor-based model of estimating and calculation of target locations is mentioned below. In a sub-period t_i , we calculate power and spread of the extracted locations and then select a few of them as temporal targets (TTLs) using three pruning models. In the end of T , we calculate the stability of all locations selected as TTLs in $t_0, t_1, t_2, \dots, t_n$. Finally, we select constant target locations (CTLs) from TTLs using stability as a threshold for different pruning strategies.

```
For  $t_i$  =a time subsequence of  $T$ :
  For  $\ell \in N$  ( $N$  =extracted locations from IPs):
    A = list of candidate targets
    Calculate power and spread of  $\ell$ 
    if spread  $\ell \geq$  threshold  $\tau_s$ 
      Add  $\ell$  to A
    Select TTLs from A:
      Case 1. Select  $k \ell \in A$  with the highest power
      Case 2. Select all  $\ell \in A$  with power score  $\geq \tau_p$ 
      Case 3. Relative threshold pruning:
        Select all  $\ell \in A$  with  $power \geq \frac{n}{m}$  top power in A

Then (in the end of  $T$ )
For  $\ell \in B$  ( $B = \sum TTLs$ ):
  Calculate stability of  $\ell$ 
  Select CTLs from B:
    Case 1. Select  $k \ell \in B$  with the highest stability
    Case 2. Select all  $\ell \in B$  with stability  $> \tau_b$ 
    Case 3. Pruning with relative threshold of stability:
      Select all  $\ell \in B$  with  $Stability \geq \frac{n}{m}$  top stability in B
```

4 Experiments and Evaluation

4.1 Dataset and Experimental Setup

A set of 90 resources was chosen for this study and each document was tagged with at least one CITY, STATE/PROVINCE or COUNTRY name which indicates the most relevant locations to that page. Textual pages, photos and sound files were included in the dataset. We used addresses and other geographical information to judge the most relevant locations for a resource. For example, if a webpage is talking about a

restaurant in Paris, it would be tagged with Paris as its location. A picture of this restaurant will be tagged with Paris too. The next step was finding the location of visitors. The log files of all selected resources were collected in 12 weeks. We pick an IP from a log and search it in online databases like Whois and RIPE³ and automatically extract country codes, telephone codes and addresses. The output of this procedure is a list of locations accompanied with IPs. Locations were disambiguated manually as this wasn't the aim of this paper.

Privacy is a big issue in a visitor-based approach. Only authorized people have access to log files. It is notable that for search engines, the privacy problem could be solved with a substitute method. Instead of analyzing real server log files, search engines can track and analyze URLs. A URL represents a Web resource and therefore, clicking on a URL link can be interpreted as using the corresponding resource. We have used server log files whenever they were available; otherwise, websites with traffic and statistic tools were selected for our dataset. Most of the resources selected for this research use either Webstats4u⁴ or Statcounter⁵ services with a setting that allows all people to see the history of visits. The total number of analyzed visitor entries is 5530 and for unique visitors it was 3563.

After extracting different locations from the logs, *power* and *spread* formulas were used respectively to find the weight of each location. The spread formula requires the number and weight of sub-locations. Any unrecognized sub-location was counted as a sub-location called OTHER to facilitate calculation of geographic distribution. We have limited the granularity of our model to city level as it is hard to disambiguate suburbs and streets inside a city. As a result, we have treated cities different from states and countries and the spread weight is considered 1 for all cities.

The procedure continues with using a threshold of spread τ_s to select candidates and then select temporal target locations (TTLs) with *top-k* model, fixed and relative pruning models described in 3.3. Power was used as a threshold τ_p to select TTLs in a 12-week period. At the end of this period, the *stability* of TTLs were calculated and used as a threshold τ_b to select constant target locations (CTLs) with three pruning models.

Recall, precision and f-measure have been used to measure the accuracy of visitor-based model. As we have already tagged each document in our dataset with a corresponding geographic name, we define recall and precision as following (here for TTLs):

$$Recall = \frac{TTLs_estimated}{(TTLs_estimated) + (TTLs_not_estimated)}$$

$$Precision = \frac{TTLs_estimated}{(TTLs_estimated) + (Others_estimated)}$$

³ <http://www.ripe.net/>

⁴ <http://www.webstats4u.com/>

⁵ <http://www.statcounter.com/>

As our approach might find more than one target location for each document, we use the average precision formula to measure how accurately it ranks the calculated locations. Average precision is the sum of the precision at each relevant hit divided by the total number of relevant documents in the collection:

$$Average\ Precision = \sum_{j=1}^{j=n} \frac{Precision(j) * Relevance(j)}{R}$$

where j is a hit in the hit list and R is the total number of relevant documents in the entire dataset. $Relevance(j)$ is 0 if it is not a relevant hit and 1 if it is relevant.

4.2 Experimental Results

We have evaluated our proposed model in different ways. Because of the limitation on the length of this paper, a selection of our experiments and results are presented here. Table 1 shows how successfully our model returns location names for an IP. The total number of IPs is 5530. The last column indicates that using online databases such as Whois, our model can return at least one country for a single IP in 98.77% of all cases. The successful tagging rate is around 78% and 74.5% for STATE/PROVINCE and CITY levels consequently. This indicates that target location can be applied to almost all Web resources at country level and to a majority of them at state and city levels.

We have used the geographic data to calculate the *target location* of documents in the dataset. As an example, we show how target locations are calculated for the webpage <http://www.boroujerd.info/english.htm> which is related to Boroujerd City in the Lorestan Province of Iran. Table 2 summarizes the results for calculation of power and spread in the 1st week as well as calculation of stability and CTLs in the end of the 12th week. The spread threshold τ_s is 0.6. In the Top- k model, $k=3$. The fixed threshold of $\tau_p=0.150$ and in the relative model, $\tau_p=35\%$. At the end of the 12th week, we calculate the *stability* of each TTL and then estimate CTLs. Again for the Top- k model, $k=3$. For the fixed threshold pruning, $\tau_b=0.150$. And finally, for the relative threshold model, $\tau_b=15\%$ of the highest stability in the TTL set.

Table 3 shows recall and precision of our model for calculation of constant target locations (CTLs). The last row shows that the recall of our model is at least 0.80 using any pruning approach and the precision is 0.64 or more.

Table 1. Successful tagging of IPs with locations

	No. of IPs with zero match	No. of IPs with 1 match	No. of IPs with 2 or more matches	% of success in the total set
COUNTRY	68	5343	119	98.77
STATE/PROVINCE	1216	3924	390	78.01
CITY	1414	3717	399	74.43

The basic precision formula indicates how many correct target locations our system can find. However, it cannot determine how accurately the results are ranked. Ideally, our model must be able not only to find correct locations but also to rank them based on their relevance to documents. For example, if a Web resource w has been judged to be geographically related to Australia, New Zealand or Fiji respectively, our model must be able to select these locations and rank them correctly. Average precision has been used to evaluate how well our model ranks target locations. Fig. 2 shows the average precision of different strategies used to select constant target locations. The Top- k pruning model shows a higher average precision than the other techniques; it can rank the most relevant target locations higher than other pruning models. It is also indicated that visitor-based algorithm performs better for calculation of target location at country level.

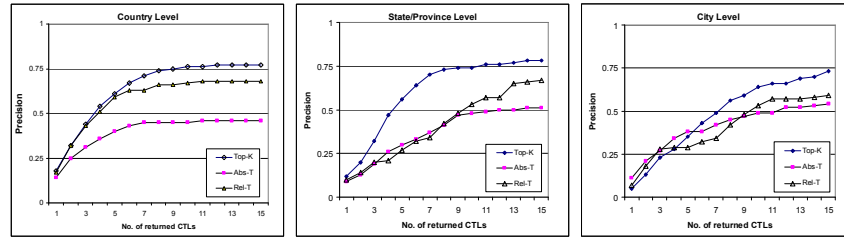


Fig. 2. Average precision of different pruning models for selection of constant target locations.

Table 2. Calculation of power and spread in the 1st week; and calculation of stability and CTLs in the end of the 12th week for the webpage <http://www.boroujerd.info/english.htm>.

Measure	Level	Matched	1 st Top	2 nd Top	3 rd Top	
Power in week 1	COUNTRY	35	Iran	USA	UAE	
	STATE/PROVINCE	31	Lorestan (Iran)	Tehran	California	
	CITY	31	Boroujerd	Tehran	Dubai (UAE)	
Spread in week 1	COUNTRY	35	Iran	USA	Germany	
	STATE/PROVINCE	31	California	Isfahan (Iran)	New York	
	CITY	31	-	-	-	
Stability	COUNTRY	85	Iran	USA	UAE	
	STATE/PROVINCE	61	Tehran (Iran)	Khorasan (Iran)	Lorestan	
	CITY	50	Tehran	Borujerd	Dubai	
Constant Target Locations (CTL)	Top-3	COUNTRY	3	Iran	USA	Germany
		STATE/PROVINCE	3	Tehran	Lorestan	California
		CITY	3	Boroujerd	Tehran	Dubai
	Absolute $T=0.15$	COUNTRY	2	Iran	USA	-
		STATE/PROVINCE	2	Tehran	Lorestan	-
		CITY	4	Boroujerd	Tehran	Dubai
	Relative $T=15\%$	COUNTRY	5	Iran	USA	UAE
		STATE/PROVINCE	2	Tehran	Lorestan	-
		CITY	2	Boroujerd	Tehran	-

Table 3. Recall and Precision of three models for calculation of constant target locations (CTLs) at different geographical levels

	Top-k		Absolute T		Relative T	
	Recall	Precision	Recall	Precision	Recall	Precision
Country	0.92	0.68	0.93	0.51	0.88	0.62
State/Province	0.88	0.75	0.80	0.69	0.81	0.67
City	0.87	0.76	0.67	0.71	0.81	0.67
All locations	0.89	0.73	0.80	0.64	0.83	0.65

We have compared our visitor-based model with content-based and link-based approaches. Table 4 indicates the rate of successful extraction or assignment of locations to Web resources using different models. The visitor based approach receives the highest score in country and state levels. This is because of the vast information which is stored with registered IPs.

Table 5 summarizes the results of comparing f-measures of different pruning techniques in visitor, link and content-based approaches. The visitor-based approach works better at country level. The link-based approach is better at city level.

Table 4. A comparison among visitor, link and content based approaches for accuracy of extraction of correct location names

	Visitor	Link	Content
Country	98%	76%	64%
State/Province	78%	69%	58%
City	78%	76%	65%
All locations	85%	74%	62%

Table 5. A comparison of the f-measures of visitor, link and content based approaches for calculating the target location of Web Resources.

	Top-k			Absolute Threshold			Average Threshold		
	Visitor	Link	Content	Visitor	Link	Content	Visitor	Link	Content
Country	0.85	0.77	0.65	0.68	0.52	0.66	0.76	0.74	0.68
State	0.74	0.72	0.58	0.58	0.5	0.61	0.66	0.74	0.66
City	0.73	0.73	0.65	0.57	0.5	0.66	0.65	0.74	0.69
Average	0.77	0.74	0.63	0.61	0.51	0.64	0.69	0.74	0.68

7 Conclusion

A location-based search engine must be able to relate Web resources with geographical locations. In this paper we showed that content and metadata information are not sufficient to judge the geography of a webpage. Information

extraction techniques in isolation cannot address the location of non-textual resources on the Web. Target location was introduced in this paper as the locality of the majority of users or visitors of a webpage. Target location is content-independent and can be applied to different media types on the Web. We introduced a novel model for extraction and calculation of target locations. Target location can be calculated and for almost all types of media on the Web, best at country level. It can be assigned to Web resources as a new definition and dimension of location. It can also be used separately or in conjunction other relevant locations to define the geography or location of a Web resource.

References

1. Sullivan, D.: Searches per day. *Search Engine Watch* (April 20, 2006). Available at: <http://searchenginewatch.com/reports/article.php/2156461> [Last visit April 22, 2006].
2. Sanderson, M., Kohler, J.: Analyzing geographic queries. Proceedings of Workshop on Geographic Information Retrieval *GIR in SIGIR'04*, Sheffield, UK (2004).
3. Asadi, S., Chang, C., Zhou, X Diederich, J.: Searching the World Wide Web for local services and facilities: A review on the patterns of location-based queries. *WAIM'05*, Hong Zhou, China (2005).
4. Watters, C., Amoudi, G.: GeoSearcher: Location-based ranking of search engine results. *JASIST*, 54(2) (2003) 140-151.
5. Amitay, E., Har'El, N., Sivan, R., Soffer, A.: Web-a-Where: Geotagging Web Content. *SIGIR'04*, Sheffield, UK (2004) 273-280.
6. Ding, J., Gravano, L., Shivakumar, N.: Computing geographical scopes of Web resources. Proceedings of the 26th *VLDB* Conference, Cairo, Egypt (2000).
7. Newcomb, K.: Google Gets Local in Canada. *ClickZ News*, (Sep. 23, 2004). Available at: <http://www.clickz.com/news/article.php/3411681> [Last visit M 12, 2006].
8. Buyukkokten, O., et al.: Exploiting geographical location information of Webpages. *SIGMOD WebDB'99*, Philadelphia, USA (1999).
9. Pouliquen, B., et al.: Geographical Information Recognition and Visualization in Texts Written in Various Languages. *SAC'04*, Nicosia, Cyprus (2004).
10. Olga, O.: Extracting Geographical Knowledge from the Internet. Proceedings of ICDM-AM International Workshop on Active Mining, Maebashi, Japan (2002).
11. Gravano, L., et al.: Categorizing Web Queries According to Geographical Locality. *CIKM'03*, New Orleans, USA (2003).
12. Li, H., et al.: Location normalization for information extraction. Proceedings of 19th Int'l Conference on Computational Linguistics, Taipei (2002) 549-555.
13. Tu, H.: Pattern recognitions and geographical data standardization. Proceedings of Geoinformatics'99 Conference, Ann Arbor (1999) 1-7.
14. Markowetz, A. et al.: Design and implementation of a geographic search engine. *WebDB'05*, Baltimore, Maryland, USA (2005).
15. Woodruff, A. G., Plaunt, C.: Gipsy: Automated geographic indexing of text documents. *JASIST*, 45(9) (1994) 645-655.