Web-a-Where: Geotagging Web Content

Einat Amitay Nadav Har'El Ron Sivan Aya Soffer
IBM Haifa Research Lab
Haifa 31905, Israel
{*einat,nyh,rsivan,ayas*}*@il.ibm.com*

ABSTRACT

We describe Web-a-Where, a system for associating geography with Web pages. Web-a-Where locates mentions of places and determines the place each name refers to. In addition, it assigns to each page a geographic *focus* — a locality that the page discusses as a whole. The tagging process is simple and fast, aimed to be applied to large collections of Web pages and to facilitate a variety of location-based applications and data analyses.

Geotagging involves arbitrating two types of ambiguities: geo/non-geo and geo/geo. A geo/non-geo ambiguity occurs when a place name also has a non-geographic meaning, such as a person name (e.g., Berlin) or a common word (Turkey). Geo/geo ambiguity arises when distinct places have the same name, as in London, England vs. London, Ontario.

An implementation of the tagger within the framework of the WebFountain data mining system is described, and evaluated on several corpora of real Web pages. Precision of up to 82% on individual geotags is achieved. We also evaluate the relative contribution of various heuristics the tagger employs, and evaluate the focus-finding algorithm using a corpus pretagged with localities, showing that as many as 91% of the foci reported are correct up to the country level.

Categories and Subject Descriptors

H.3.3 [**Information Search and Retrieval**]

General Terms

Design, Algorithms, Experimentation, Verification

Keywords

Information retrieval, Text mining, Natural language processing, Geographic tagging, Gazetteer, Disambiguation.

1. INTRODUCTION

Understanding place names mentioned in texts such as Web pages, news articles, emails, etc. can greatly benefit data mining systems. Users could add geographic criteria to their queries that search engines could process intelligently. The geographic distribution of the matching pages could be displayed, showing how a certain product is popular in a certain geographic region and unpopular in others, for example. Mining could be narrowed to a certain geographic region (e.g., only process pages that talk about England). Correlation between mentions of place names, or place names and other terms, could be analyzed, for example, to find which places are most associated by bloggers with fashion, parties, vacations, or good food. A variety of location-based services for mobile devices can also be developed.

To accomplish these and other goals, an understanding of the geographic orientation of a page is needed. This is usually extracted from the places the page refers to, which in turn depend on effective disambiguation of the place names listed in the page and its associated structures. This paper deals with the challenges posed by both the disambiguation of individual place mentions and their integration into a coherent page focus.

A page may have two types of geography associated with it: a *source* and a *target*. Source geography has to do with the origin of the page, the physical location of the server it is stored at, the address of its author or owner, etc. Target geography is determined by the contents of the page and relates to the topic the page is discussing. For example, a news article about Northern Ireland appearing on the CNN site would have a target geography of United Kingdom and perhaps Ireland as well, but a source geography of the USA.

In principle, given a list of all possible place names, all that needs to be done is to search the text for names in the list, and for each name found, look up its meaning in the list. The problem is that most place names, and the vast majority of names found on the Web, are ambiguous.

There are two types of ambiguities: geo/non-geo and geo/-geo. Geo/non-geo ambiguity is the case of a place name having another, non geographic meaning, as in Gary (Indiana), Mobile (Alabama) or Reading (England). Some of the most common English words are also place names: As (Belgium), Of (Turkey) and To (Myanmar). Geo/geo ambiguity arises when two distinct places have the same name. Almost every major city in the Old World has a sister city of the same name in the New: London, Paris, Vienna, Berlin, Moscow, Cairo, Rome, Athens and Jerusalem, to name a few. In the USA there are 18 cities named Jerusalem, 24 named Paris and 63 Springfields in 34 states (Alabama alone has four). Smith *et al* [26] report that 92% of all names occurring in their corpus are ambiguous. On Web-pages, our statistics

SIGIR'04, July 25–29, 2004, Sheffield, South Yorkshire, UK.

show that 37% of the potential geographic-name mentions have several possible geographic meanings, and the average number of possible meanings per mention is roughly 2.

The problem of geographic name disambiguation has been tackled almost independently in circles of linguistics, machine understanding and AI, as well as by information retrieval and knowledge management researchers, resulting in characteristically differing solutions. While NLP employs machine learning to recognize names from their structure and context, the data mining approach resorts to glossaries and gazetteers to make the determination, and say which place is being referred to. The latter approach cannot find names that are not in the list, as the former can, but normally results in a simpler algorithm. It also does not require training data which is sometimes hard to come by. Moreover, since Web data mining entails processing tremendous quantities of data, the algorithm used to find place names must be fast enough to handle hundreds of pages a second on an ordinary work station. NLP algorithms tend to be too complex to work this fast.

The system we present, Web-a-Where, employs the gazetteer approach. Its goal is to identify all geographic mentions in Web pages, assign a geographic location and confidence level to each, and derive a focus (or foci) for the entire page. While our approach is inspired by several previously published works that employed the gazetteer approach, we devised several enhancements designed to improve the precision of individual place name tagging. For example, we employ a new method to correctly identify place names that also have a very commonly used non-geographic meaning. Our system automatically compiles a list of such locations, and requires significantly more evidence before declaring them a place name.

To the best of our knowledge, Web-a-Where's algorithm for finding the geographical focus of a page has not been published elsewhere. Our scheme is capable of assigning as focus a location the page does not mention, inferring it from places it does. An additional contribution of this work is the evaluation platform that we have developed. Most of the work published on this topic does not provide experimental proof of effectiveness. Measuring the precision of a tagger depends either on pretagged corpora or manual labor, both in short supply. We created several test sets from readily available Web pages, and analyzed the contribution of various heuristics used for geographic name tagging. We additionally tested our focus-finding algorithm by comparing its results with the classifications provided by the editors of the Open Directory Project. We show that on 91% of the pages for which a focus is reported, the focus is correct up to the country level.

The rest of this paper is organized as follows. Section 2 surveys existing methods for finding place names in text. Section 3 explains in detail how Web-a-Where tags individual place names, and the geographic knowledge it requires. Section 4 describes our novel algorithm for determining the focus of a page. Section 5 provides some implementation details, as well as experimental results obtained by running the tagger over several data sets. Section 6 concludes and discusses planned future work.

2. SURVEY OF PREVIOUS WORK

Substantial effort has been applied to the more general task of Named Entity Recognition (NER) involved in identifying proper names in text. Both statistical-learning [13, 20, 22, 28, 29] and natural language processing [14, 25, 26] provide good results on some texts, at times even approaching human performance (precision of 96% according to [27]). Hybrid approaches involving aspects of both have also been successfully applied [23]. Most of the work published in this area, however, has been tested primarily on news articles or on the standard NER tasks provided by various conferences [10, 11], all corpora of well-edited text.

More recently, the specific task of determining which place is meant by a particular occurrence of a place name, known as *grounding* (also referred to as *localization*), has been gaining attention. Clearly, grounding requires general-world knowledge and cannot rely completely on information found in the text or even in a whole corpus. This general knowledge is provided in a *gazetteer*, which traditionally lists the names of all places in an atlas. Grounding need not be confined to geographic applications, however: it has been creatively applied to anatomical term understanding [17] (where an anatomical atlas serves as a gazetteer) or to finding names of proteins [16] (where a repository of proteins names serves this purpose). Only a minority of the published works include any experimental results: most describe algorithm details without measurements.

Most published algorithms do not employ machine learning, but are rather based on various NLP heuristics. This may be explained by the fact that machine-learning algorithms are more expensive and require training data that is not readily available. In the mean time, the presence of gazetteers simplifies the task of deciding which text tokens to consider.

Many researchers (the present work included) adopt plausible principles to help distill the correct sense of a name. Examples include:

- Single sense per discourse — an ambiguous term is likely to mean only one of its senses when used multiple times within one page unless specifically qualified (e.g., "He drove from Portland, ME to Portland, OR").
- Place names appearing in one context tend to indicate nearby locations. Vienna and Alexandria mentioned in the same paragraph are more likely to indicate the two communities in Northern Virginia than the larger and better known cities in Austria and Egypt, respectively.

Li *et al.* present in [18, 19] a rigorous 5-step algorithm that is typical of many other publications. First, only words in the text which are also listed in the gazetteer are considered. Next, NLP techniques are called upon to help weed out non-geo terms (avoiding prefixes like "Mr." but embracing "city of"). Then, the "single sense per discourse" principle is applied. Next, one of the senses of each ambiguous place name is selected such that overall distances between all senses is minimized. Names that remain unresolved are assigned a default sense, the most important one associated with the given name. The authors report 93.8% precision on news and travel guide data.

Leidner *et al.* present a similar algorithm [17], showing its application to non-geographic domains.

In Bilhaut *et al.* [12], the language understanding step is expanded to interpret phrases such as "south of Genève-Bordeaux line", which refers to a region which includes neither city. Their examples draw from documents in French produced by the French government.

In Smith *et al* [26] the first two steps of the algorithm are reversed: a general NER is done first, and only words that are suspected of being place names are looked up in the gazetteer. Selecting a sense for an ambiguous place name is also different: the scope over which minimization is sought is limited to the unambiguous names appearing in a textual window about that name. Smith *et al* report precision of 75% – 93% over a digital library of historical texts.

The suggestion to apply such techniques to Web pages (rather than to well-edited corpora of news and digital libraries) was first made by McCurley in [21]. He analyzes the various aspects of a Web page that could have a geographic association, from its URL and the language it is written in to the phone numbers and ZIP codes it may list. Names found in the text may be looked up in gazetteers if they are places, or in the White Pages directory (to extract their address) in case they are persons. Hyperlinks may be followed to see if the linked pages have a strong geographic association that could be reflected back. His approach, however, depends heavily on information such as place names, postal tracts and phone directories that are free and available online where the USA is concerned, but are much harder to come by in other parts of the world and for many jurisdictions may not exist at all.

Google introduced a new service, Search by Location [1], where a search can be limited to pages referencing a given address. Again, only US addresses are supported.

Rauch *et al* [24] describe a complete geography-based search system, being developed commercially by MetaCarta [3]. Their algorithm enhances the general scheme described above by maintaining a history of disambiguations over the corpus. They claim that over time one can get closer to assessing $P(name, place)$, the probability that a given name means a given place. $P(name, place)$ can then be used to improve disambiguation in the future. They also include a step that attempts to understand relative positioning, as in "15 miles north of Washington". The system is able to digest Internet pages and assign a geographical focus to each, which is then used both to limit a search to a particular geographical region and to display the spatial distribution of the result.

Ding *et al* [15] find what they call the geographical *scope* for a web site — the location of the its intended audience. They implemented two methods: the first analyzes the geographic location of the hosts linking into the site in question (this method is unrelated to the present work). The second method applies one simple disambiguation heuristic after a NER stage, followed by a statistical algorithm to determine the *scope* of all places mentioned in the site. Their scope-finding algorithm requires more samples than our focus algorithm, and cannot meaningfully work on individual web-pages that contain very few places mentioned.

3. TAGGING INDIVIDUAL PLACE NAMES

The main component of our system is the geotagger. It finds and disambiguates geographic names, currently those of cities, states and countries. Disambiguation means assigning a canonical *taxonomy* node to each phrase in the text that is deemed to refer to a place. Like an address, a taxonomy node indicates a single, unambiguous place on the globe by hierarchically specifying its name and the names of all the regions encompassing it. For example, `Paris/France/Europe` or `Haifa/Israel/Asia`.

These taxonomy nodes provide the users with powerful search options. For example, searching for `France/Europe` could return a page that never mentions France explicitly, only names of cities determined to be French.

The list of geographic names, their canonical taxonomies and other pertinent information is kept in a database known as a *gazetteer*, described in Section 3.1. The gazetteer is one of the most significant components of the system.

The processing of a page is done in three phases:

1. **Spotting:** the page text is scanned for occurrences of names appearing in the gazetteer. See Section 3.2.

2. **Disambiguation:** each spot is examined and is assigned a meaning. The confidence of the assignment is also calculated. See Section 3.3.

3. **Focus determination:** knowledge from the individual spots is aggregated to yield up to four geographies which encompass most of the spots and hence represent the geographic focus (or foci) of the page as a whole. See Section 4.

3.1 The Gazetteer

The gazetteer contains a hierarchical view of the world, divided (in our current implementation) into continents, countries, states (for some countries), and cities. As explained above, this hierarchy associates each geographic entity (i.e., place) with a canonical taxonomy node. Each place is also associated with a number of names and/or abbreviations — for example "Alabama", "AL" and "Ala." are all names of the same state. World coordinates and a population estimate are also assigned to each place, as these are useful in the disambiguation algorithm.

3.1.1 Contents and sources

The gazetteer used by our system contains all of the world's countries and many of its cities (those having 5,000 inhabitants or more). It also contains states and provinces for the USA, Canada, Australia, China and the UK, as well as standard abbreviations of country and state names. Overall, nearly 40,000 places around the world are listed, together with alternative spellings and abbreviations for a total of about 75,000 names — most in English and some in the vernacular. The gazetteer is kept as an XML file, and is automatically created from a number of freely available data sources using a set of Perl scripts. This allows for flexibility in generating a gazetteer for different purposes, controlling its size or focus of attention.

In preparing the gazetteer, we used the following data source: **GNIS** [7] for U.S. locations, **World–gazetteer.com** [9] for non-U.S. locations, **UNSD** [6] for countries and continents, **ISO 3166-1** [2] for country abbreviations, and other sources for state abbreviations.

3.1.2 Non-Geo Terms

Geo/non-geo ambiguity can be particularly bothersome when the non-geo sense is very commonly used. This can be the case when the senses are in different languages, so the ambiguity is not ordinarily visible. Places named "To" (Myanmar), or "Of" (Turkey) create mounds of false spots for the geotagger to sift through, the vast majority of which are non-geographic.

The gazetteer contains a special section listing place names that are also very commonly used words. The geotagger

then requires much more evidence before it tags such an ambiguous term as geographic. Thus "Mobile" (even capitalized) is considered non-geo unless followed by "Alabama" (or an abbreviation thereof).

To identify the place names that also have a very common non-geo sense we counted the number of times each name in the gazetteer occurred in a large corpus of well-edited pages[1]. Two tests were applied to the results:

- Names that appeared more than 100 times, but in most cases were not capitalized as a name should be, were included in the gazetteer's non-geo section. "Asbestos" (Quebec) and "Humble" (Texas) were thus discovered as having a strong non-geo meaning.

- Names mentioned much more frequently than their population would warrant were also included. Although "Grove" (Spain) and "Atlantic" (Iowa) were encountered capitalized most of the time, their high frequency did not match their small populations (10,976 and 7,474, respectively).

This was the only part of gazetteer generation we were unsuccessful in completely automating: the list required a manual pass to weed out obvious errors and add words that were subsequently found to have been missed. For example, "Aspen" had to be removed from the list because although the population of Aspen, Colorado is only 5,465, its high frequency of mention is not due to a competing non-geo meaning but rather to its renown as a ski resort. On the other hand, "Metro", a city of 100,000 in Indonesia, had to be manually added to the list: although "Metro" normally occurs properly capitalized, and at a frequency commensurate with this city's population, the reference in most cases is to a "Metro area", the Paris underground or to similar non-geographic entities.

3.2 Spotting place name candidates

The process of finding geographic names starts by finding (or *spotting*) all the possible geographic names in each page. Any case-insensitive appearance of "London" will be spotted — only later a disambiguation pass will try to decide if this is a reference to a place (not a person, as in "Jack London") and if so, which of the cities named London is meant.

The list of words to spot is the list of all names in the gazetteer. Short abbreviations are not spotted since often they are too ambiguous, as IN (Indiana or India, but also a common English preposition) or AT (for Austria). However, such abbreviations are used later by the geotagger to help disambiguate other spots, as in the phrase "Gary, IN".

3.3 Disambiguating spots

The disambiguation algorithm sequentially applies several heuristics to each spot. The steps are as follows:

1. If the tokens in the vicinity of a spot can uniquely qualify it, as in "IL" immediately following a spot of "Chicago", the geotagger assigns this unique meaning to that spot with a confidence in the range of 0.95 – 1 to reflect its high level of certainty. Combinations that look like qualified names but are not unique (e.g., "Springfield, USA" of which there are many, or "London, Germany" of which there are none) are left unassigned for the moment.

2. Each unresolved spot is assigned its default meaning: the geographical entity with the largest population among those having that name. The confidence of this assignment is set to the low level of 0.5.

3. In case the page has multiple spots of the same name where only one is qualified, the meaning of the qualified spot is delegated to the others ("single sense per discourse"). The assignment is given a confidence in the range of 0.8 – 0.9, depending on whether the delegated meaning matches the spot's default meaning.

4. A *disambiguating context* for the spots that are still unresolved (those whose confidence is below 0.7) is now sought. A context is a region in whose confines most unresolved spots become unique. For example, assume "London" and "Hamilton" appear in the same page without further qualifications. Possible interpretations of London include "England, UK" and "Ontario, Canada", while Hamilton exists in "Ohio, USA", "Ontario, Canada" as well as in "New Zealand". The smallest disambiguating context is therefore "Ontario, Canada", which is the only one common to both lists. The meanings induced by the context are assigned with a confidence in the range of 0.65 – 0.75, again depending on whether the assigned meaning matches the spot's default meaning.

   The hypothesis behind this approach is that page authors imply a context, and do not qualify names whose meaning is clear from that context. It stands to reason, then, that all unqualified names in the page share a context. The names that are qualified, in fact, may have been qualified precisely because they do not share this common context and therefore could be misinterpreted if appeared unadorned.

4. DETERMINING PAGE FOCUS

Once we determined the correct meaning of every geographical name mentioned in the page, we would also like a way to separate the wheat from the chaff — to decide which geographic mentions are incidental, and which constitute the actual *focus* of the page. Knowing this focus might be useful, for example, if the user wants to search for pages about California, rather than finding the multitude of pages that mention in passing some city in California or pages that list all the states of the union.

4.1 Rationale of focus algorithm

The basic idea is that if several cities from the same region are mentioned, this might mean that this region is the focus. For example, a page mentioning San Francisco (Calif.), Los Angeles (Calif.) and San Diego (Calif.) can be said to be about California. A page mentioning San Jose (Calif.), Chicago (Ill.) and Louisiana can be said to be about the United States. A page that is predominantly about the United States with a single mention of Paris France can still be said to be only about the US. Repeated mentions of the same place count: A page mentioning the state of California five times is probably just as likely to be about California as a page mentioning five different cities in California.

[1]The corpus consisted of about 1,200,000 pages taken at some point from the *.gov* domain.

Sometimes we cannot say that a page has only one focus. For example, two different countries might be repeatedly mentioned in some news story. In such cases we will want to list several geographic regions as foci. However, we must still try to coalesce many places into one region before declaring foci, so that a page that lists the 50 states of the United States will not be said to have 50 separate foci, but rather one focus — the United States.

The other extreme should be avoided as well: if a small region is the real focus of a page, we should not unnecessarily report a larger region. It is all too easy, but not very productive, to report several continents as being the "focus".

The focus-finding algorithm assumes that all geographic names have already been disambiguated correctly. When the disambiguation algorithm makes a bad guess, it should give it a low confidence score. In finding the focus, we should take these confidence scores into account, giving higher weight to information coming from locations with higher confidence.

4.2 Outline of focus algorithm

The basic idea is as follows. Each geographic mention, disambiguated into a taxonomy node (e.g., `Paris/France/Europe`), adds a certain score to the importance of this place in the page, while adding lower scores to the enclosing hierarchies — `France/Europe` and `Europe`. We sum up the scores contributed by all places in the page, and then sort the hierarchies by importance. We ignore places that are already part of or enclose a more important place, as well as places whose importance score is not high enough.

The reason that places contribute less score to their enclosing regions is that this allows the more specific place to "win" if it is the only place mentioned in this region, while permitting the region to be chosen as focus if several different places in it are mentioned with no emphasis on any.

Note that the region that the focus-finding algorithm reports is a taxonomy node from the gazetteer. Our gazetteer does not list regions such as "The Tri-state area", "Southern California" or "The Middle East", and therefore such regions cannot be reported as foci — the focus is limited to being a city, state, country or continent.

An example will make the algorithm clearer: A certain page contained four mentions of `Orlando/Florida` (confidence 0.5), three `Texas` (0.75), eight `Fort Worth/Texas` (0.75), three `Dallas/Texas` (0.75), one `Garland/Texas` (0.75), and one `Iraq` (0.5). A human that was asked to judge what is the geographical focus of this page responded with "It's about Texas and perhaps also Orlando". Indeed, that page comes from the "Orlando Weekly" site, in a forum titled "Just a look at The Texas Local Music Scene...". Our scoring algorithm (given in detail below) gave the following scores:

```
6.41 Texas/United States/North America
4.97 United States/North America
4.50 Fort Worth/Texas/United States/North America
3.48 North America
1.68 Dallas/Texas/United States/North America
1.00 Orlando/Florida/United States/North America
0.70 Florida/United States/North America
0.56 Garland/Texas/United States/North America
0.25 Iraq/Asia
0.17 Asia
```

The algorithm proceeds to go over this sorted list from the top. Texas got the top score (because several separate cities — Fort Worth, Dallas and Garland contributed to it, even though each city contributed more to its own score) and is chosen as a focus. The next highest scorer, the United States, already covers Texas so it is dropped: it doesn't make sense to say that both Texas and something that covers Texas are in focus. The next scorer, Fort Worth, is covered by Texas and is dropped for the same reason, as are North America and Dallas which follow it in the list. We then get to Orlando/Florida, which does not cover the existing focus of Texas nor is it covered by it, and is taken as a second focus. The remaining scores (e.g., for Iraq/Asia) are below the importance threshold (0.9) and are ignored. This page therefore ends up with two foci: Texas and Orlando, with Texas being the first (stronger) focus.

4.3 The focus scoring algorithm

We shall now describe the focus finding algorithm more formally. The algorithm loops over the disambiguated geographical places in the page aggregating the *importance* of the various levels of the taxonomy nodes. For a place with a taxonomy node of the form `A/B/C` whose disambiguation confidence is $p \in [0, 1]$, we add to the importance of `A/B/C` the score p^2. A quadratic scoring function was chosen in order to increase the relative weight of very certain disambiguations, but is rather arbitrary since it depends on how p is calculated in the first place. After adding p^2 to `A/B/C`'s score, we add $p^2 d$ to the enclosing region `B/C`, and $p^2 d^2$ to `C`, where $0 < d < 1$ (0.7 in our implementation), causing the importance to decay as was explained above.

After sorting the resulting taxonomy levels by score, we loop over them from highest to lowest, stopping at the low threshold (0.9 in our implementation), or if sufficiently many foci were already found (in our case, 4). We skip taxonomy levels that cover or are covered by one already selected as focus. Otherwise we add this level to the list of foci.

The aforementioned weights and thresholds are based on some experimentation, but they are by no means optimal. Additional research is needed to discover the optimal values.

5. IMPLEMENTATION AND RESULTS

Web-a-Where was implemented using the WebFountain [8] data-mining framework developed at IBM research. WebFountain provides facilities for crawling the Web, storing the resulting pages and indexing their contents. In conjunction with each page, the page store can keep any amount of additional information, organized in *keys*. Pages can be processed by so-called *miners*, which can read the page and its associated keys and add new keys as well as modify or delete existing ones. Miners may be chained so that the output of one is efficiently fed as input to another. Together with a set of APIs for miner development, WebFountain encourages the construction of specialized miners, each dedicated to a single task, that can be used as building blocks for complex data mining undertakings. For example, an existing Spotter miner, employing a string match algorithm and capable of efficiently handling even lists of a million names to spot, was used to implement the spotting phase (see section 3.2).

The Web-a-Where geotagger, implemented as a WebFountain miner, outputs the meaning (a taxonomy node) for each place name in the text, together with a confidence figure for that meaning. It also produces a set of up to four foci per page in a separate key, describing the geographical orientation of the page as a whole (see section 4).

Altogether, the resulting WebFountain setup processed roughly 14 Web-pages per second on a desktop PC.

The evaluation of Web-a-Where was conducted in two stages. In the first stage we evaluated the precision and accuracy of the assigned geotags by manually going over a set of results and marking them for correctness. In the second stage we extracted a random sample from the Open Directory Project's (ODP) "Regional" sub-directory and used their RDF assigned tags to test our focus algorithm.

5.1 Evaluating the geotagging process

Due to the difficulty in defining a "representative" collection of Web pages, Web-a-Where was tested on three different Web-page collections, representing different themes, quality, and coherence of writing. All told, 600 pages containing over 7,000 geotags were analyzed. The criteria for their selection are described below.

The first collection, the "Arbitrary Collection", represents well-authored rich-content pages that are intended for daily use by a large and general audience. The collection was generated in the following way: we queried Google with three different queries: `+the`, `+and`, `+in`, and collected the top 1000 results for each query. We then sorted the URLs alphabetically and removed duplicates. From that list we removed pages smaller than 3K and then arbitrarily chose 200 pages. This method of collecting pages is biased by the search engine ranking assigned to pages. Assuming the queries we chose appear in the majority of English-language pages on the Web, the pages with the highest page-rank are mostly intended for large audiences, and in English.

The ".GOV Collection" represents 200 pages taken at random from among nearly 1,200,000 pages harvested from the *.gov* domain for the standard test of TREC'03 [5]. These pages are almost exclusively in well-edited English, dealing with various US government issues.

The "ODP Collection" comprises of 200 randomly chosen pages from the Regional sub-directory of the Open Directory Project [4]. All pages chosen were larger than 3k. This collection represents pages that were chosen by a human editor to be of either "belonging" to a geographical place, or describing a geographical place. The chosen pages are not necessarily well authored, they may refer to very small communities, and may not be intended for or composed by native English speakers.

We geotagged all three collections with Web-a-Where and then manually checked the geotags for correctness. Each geotag was labeled to be either "correct", error of type "Geo/Non-Geo" (the entity was wrongly identified as a geographic place), error of type "Geo/Geo" (an incorrect geography was selected from the gazetteer, e.g. `Paris/France` instead of `Paris/Texas`), or error of type "Not in Gazetteer" (the correct location does not appear in our gazetteer, e.g. Hollywood in Los Angeles, California is not an independent municipality and is absent from our gazetteer, and thus any occurrence of Hollywood is tagged as `Hollywood/Florida`, the only Hollywood in the gazetteer).

Web-a-Where assigned 2307 geotags within the Arbitrary collection, 2558 geotags in the .GOV collection, and 2217 geotags in the ODP collection. Results are provided for each store separately and are summarized in Figure 1.

It is apparent in all collections that the most acute problem is the "Geo/Non-Geo" error type. The differences between the collections are probably caused by the generality of the page at hand. The Arbitrary Collection (81.7% accuracy) is more coherent, almost always in English, intended for the general public, and will thus tend to attach more disambiguation cues to place names and concepts; On the other hand, the ODP Collection (63.1% accuracy) often refers to smaller places, will have local readership, and hence contains little or no reference to the geographical ambiguity that might be created. The .GOV Collection is found somewhere in between the two (73.3% accuracy). It probably has little disambiguating cues for the larger US locations, but provides more elaboration for the smaller locations that are not as familiar to the general population.

Heuristic / Collection	Arbitrary	.GOV	ODP
Full algorithm	81.7%	73.3%	63.1%
No population data	72.7%	68.1%	58.5%
No implied context	82.7%	74.4%	62.0%
No seen-qualified	80.5%	72.3%	61.7%

Table 1: Effect of removing different heuristics on the geotagging precision

Analyzing the contribution of each component in our geotagging algorithm is shown in in table 1. We first examine the impact of the population heuristic by randomly selecting a default location for a place name from the candidates in the gazetteer, rather than selecting the one with the largest population. On all three collections, our performance drops significantly. We hurt the Arbitrary Collection most (decreasing its accuracy by 9%). The .GOV Collection's accuracy drops by 5.2% and the accuracy of the ODP Collection is down by 4.6%. Given our previous observation about the different nature of these three collections, these results show that the population heuristic is most relevant in pages that by default mention well known, large places as in the case of the Arbitrary collection.

We next tested omitting the implied context heuristic. This resulted in a slight improvement in the accuracy of the Arbitrary and the .GOV collections but lowered the accuracy for the ODP collection. Finally we omitted the seen-qualified component of the geotagging algorithm. This degraded the performance on all three collections but not as significantly as the population heuristic.

5.2 Testing Page Focus

We evaluated the focus-finding algorithm (section 4) by comparing its decisions to those of human editors.

Fortunately, a comprehensive source of Web-pages already tagged by their geographic focus (as judged by human editors) is freely available. The Open Directory Project (ODP) [4] is the largest (3.8 million pages) human-edited directory of the Web, and is maintained by a vast community of volunteer editors. One section of this index, the "Regional" section, is devoted to English-language pages with a coherent geographic focus (e.g., sites about a place, sites about a company located in a certain city). Each page is located in the "Regional" hierarchy according to its geographic focus.

This availability of almost one million real Web-pages pretagged with their geographic focus allowed us to automatically test Web-a-Where on a very large sample of Web-pages, much bigger than we could afford to manually tag ourselves. We chose a random sample of over 20,000 Web-pages from the ODP's Regional section that were larger than 3k. We

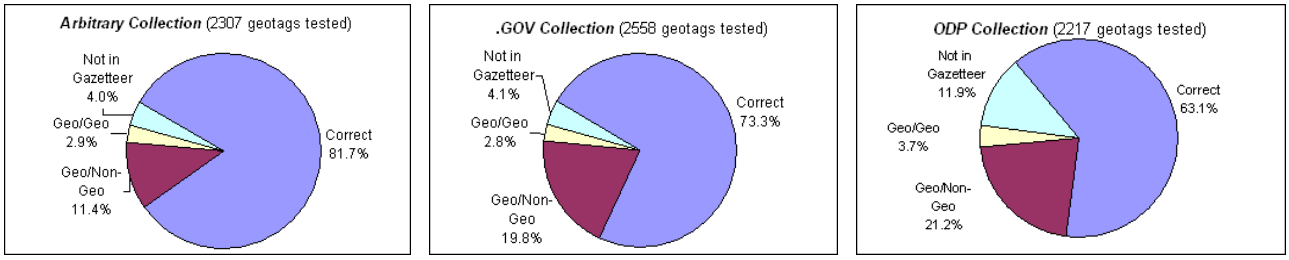

Figure 1: Precision of individual Web-a-Where geotags, in different collections, and types of errors

92% correct up to country level			*8% incorrect country*	
38% Precise match	30% Correct state or city	24% Correct country	4% Correct continent	4% Continent wrong

Table 2: Comparison of Web-a-Where-determined focus to human-determined one (ODP)

ran Web-a-Where on this sample and compared the foci it reported to those listed in the ODP index.

The results of this comparison are given in table 2. Several points are due noting when interpreting these results:

- While the ODP indeed lists a geographic focus for every page in the sample, the geography of many of the pages was determined by looking at images, sub-pages, and at other non-textual information not available to our tagger. In fact, as much as 27% of the pages did not have a single geographic name in them. We therefore cannot expect our tagger to find a focus in many of the pages in the sample.
- The determination of a page's focus is more subjective than the meaning of a single geographic name. It's not always clear if a page is about "England" or about the "United Kingdom", for example. Also, the ODP's hierarchy is less rigid than ours and pages may be marked as belonging to a metro area (e.g., "San Francisco Bay Area"), region ("Northeast Tennessee"), etc. The ODP is also known for its over-representation of small categories — in this case many tiny towns that are too small to appear in our gazetteer. Accordingly, we cannot always expect an exact match between the focus reported by our tagger and the one listed in the ODP. Instead, we need to define what qualifies a "good enough" match. Table 2, for example, divides these matches into several quality types.
- The focus algorithm uses the information the tagger gives it on the meaning of individual places mentioned on the page. In section 5.1 we saw that as much as 37% of these were wrong. It must be emphasized, therefore, that this test evaluates not just the focus algorithm, but Web-a-Where as a whole.

Given these difficulties, Web-a-Where did quite well. It found a page focus in 75% of the pages with one or more geotags. Of the pages with focus, in 91% the focus had the correct country, in 65% the focus matched up to the 3rd hierarchy level (state or city), and in 38% the focus matched the ODP listing precisely.

These numbers are averages over all pages in the sample. We should note that Web-a-Where seemed to do better on pages classified by the ODP as focused on the USA (a little over half of the pages in the sample). American-focused pages more often were given a focus (80% vs. 68%, of the pages that had any geotag), and more often given a correct focus (for all the above correctness measures). Nearly all (97%) American pages with focus were given the correct country (the United States), compared to 81% of non-American pages with focus whose country was identified correctly. As much as 8% of non-American pages with focus were labeled with the wrong continent — many of them mislabeled as USA.

When research on Web-a-Where started, an interesting question was raised: will enlarging the gazetteer by adding smaller and smaller towns improve the page-focus determination, or hurt it? On one hand, by adding smaller towns to the gazetteer, more references to small towns are recognized and this should improve focus determination. But on the other hand, most small towns are never mentioned in actual Web-pages and add a lot of ambiguity. This question can be answered by re-running the focus text with different gazetteers generated with different town-size cutoffs. Figure 2 shows that all relevant quality measures (number of pages with focus, number of pages for which we got the correct country, and so on) increase as we lower the town-size cutoff, from 500,000 to 5,000 people. This is a good sign — Web-a-Where is helped, not hurt, by an improved gazetteer.

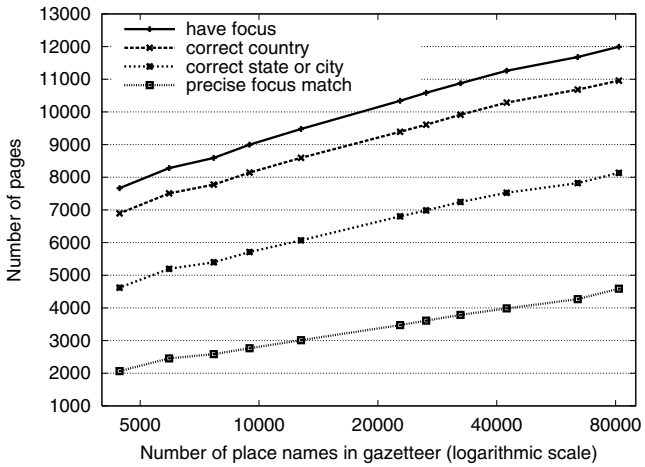

Figure 2: More pages get correct focus as the gazetteer is enlarged (lowering population cutoff).

6. CONCLUSIONS AND FUTURE WORK

Web-a-Where, a system for geotagging Web content has been presented. Our experiments show that we are able to correctly tag individual name place occurrences 80% of the time and that we are able to recognize the correct focus of a page 91% of the time. While the accuracy can be further improved, it is currently sufficient for its applications.

The main source of errors is geo/non-geo ambiguity. There are several additional directions we plan to explore in order to address this issue. For example, we could recognize person names as non-geo, or rule out uncapitalized words in an otherwise properly-capitalized text. Use of a part-of-speech tagger can also be explored, though its impact on performance would have to be considered.

Geo/geo accuracy can be improved by improving the "disambiguating context" heuristics, and by devising additional ones. Heuristics based on the *coordinates* of places should be compared with the taxonomy-based methods we now use. Finally, we believe that analyzing linkage among Web pages can be used to help disambiguation and we plan to explore that possibility as well.

7. ACKNOWLEDGMENTS

We we would like to thank Ronny Lempel for his insightful remarks and many suggestions. We would also like to thank Wayne Niblack, Todd Bender and Zengyan Zhang of Almaden Research Laboratory for their input and support.

8. REFERENCES

[1] Google Search by Location http://labs.google.com/location.

[2] ISO 3166 code lists. http://www.iso.ch/iso/en/prods-services/iso3166ma/02iso-3166-code-lists/index.html.

[3] $M\epsilon\tau\alpha$CARTA, Inc. 875 Massachusetts Avenue, Cambridge, MA 02139. http://www.metacarta.com.

[4] ODP: Regional. http://dmoz.org/regional.

[5] Text REtrieval Conference 2003: .gov test collection. http://es.cmis.csiro.au/trecweb/access_to_data.html.

[6] United Nations department of economic and social affairs. http://unstats.un.org/unsd.

[7] USGS Geographic Names Information System (GNIS). http://geonames.usgs.gov.

[8] WebFountain framework for data mining. http://www.almaden.ibm.com/webfountain.

[9] World Gazetteer. http://www.world-gazetteer.com.

[10] The 6th message understanding conference task definition, March 1995. http://www.cs.nyu.edu/cs/-faculty/grishman/COtask21.book_1.html.

[11] Language-independent named entity recognition: shared task, 2002. http://cnts.uia.ac.be/conll2002/ner.

[12] F. Bilhaut, T. Charnois, P. Enjalbert, and Y. Mathet. Geographic reference analysis for geographic document querying. In *Workshop on the Analysis of Geographic References*, Edmonton, Alberta, Canada, May 2003. NAACL-HLT.

[13] J. D. Burger, J. C. Henderson, and W. T. Morgan. Statistical named entity recognizer adaptation. In *Proceedings of CoNLL-2002*, pages 163–166, 2002.

[14] S. Cucerzan and D. Yarowsky. Language independent NER using a unified model of internal and contextual evidence. In *Proceedings of CoNLL-2002*, pages 171–175. Taipei, Taiwan, 2002.

[15] J. Ding, L. Gravano, and N. Shivakumar. Computing geographical scopes of web resources. In *Proceedings of the 26th VLDB Conference*, Cairo, Egypt, 2000.

[16] G. Eriksson, K. Franzén, F. Olsson, L. Asker, and P. Lidén. Exploiting syntax when detecting protein names in text. In *Proceedings of Workshop on Natural Language Processing in Biomedical Applications*, 2002.

[17] J. Leidner, G. Sinclair, and B. Webber. Grounding spatial named entities for information extraction and question answering. In *Workshop on the Analysis of Geographic References*, Edmonton, Alberta, Canada, May 2003. NAACL-HLT.

[18] H. Li, R. K. Srihari, C. Niu, and W. Li. Location normalization for information extraction. In *Proc. of the 19th Conference on Computational Linguistics (COLING-02)*, Taipei, Taiwan, August 2002. ACL.

[19] H. Li, R. K. Srihari, C. Niu, and W. Li. infoXtract location normalization: a hybrid approach to geographical references in information extraction. In *Workshop on the Analysis of Geographic References*, Edmonton, Canada, May 2003. NAACL-HLT.

[20] R. Malouf. Markov models for language-independent named entity recognition. In *Proceedings of CoNLL-2002*, pages 187–190, Taipei, Taiwan, 2002.

[21] K. S. McCurley. Geospatial mapping and navigation of the web. In *Proc. of the 10th int. conference on World Wide Web*, pages 221–229. ACM Press, 2001.

[22] P. McNamee and J. Mayfield. Entity extraction without language-specific resources. In *Proceedings of CoNLL-2002*, pages 183–186. Taipei, Taiwan, 2002.

[23] J. Patrick, C. Whitelaw, and R. Munro. Slinerc: The sydney language-independent named entity recogniser and classifier. In *Proceedings of CoNLL-2002*, pages 199–202. Taipei, Taiwan, 2002.

[24] E. Rauch, M. Bukatin, and K. Baker. A confidence-based framework for disambiguating geographic terms. In *Workshop on the Analysis of Geographic References*, Edmonton, Alberta, Canada, May 2003. NAACL-HLT.

[25] Y. Ravin and N. Wacholder. Extracting names from natural-language text. Technical Report RC-20338, IBM Research Division, T.J.Watson, Yorktown Heights, NY, October 1997.

[26] D. A. Smith and G. Crane. Disambiguating geographic names in a historical digital library. In *Proceedings of the 5th European Conference on Research and Advanced Technology for Digital Libraries (ECDL'01)*, Lecture Notes in Computer Science, pages 127–136, Darmstadt, September 2001. Springer.

[27] B. Sundheim. Overview of results of the MUC-6 evaluation. In *Proc. of the 6th message understanding conference*, pages 13–32, Columbia, MD, Nov. 1995.

[28] D. Wu, G. Ngai, M. Carpuat, J. Larsen, and Y. Yang. Boosting for named entity recognition. In *Proceedings of CoNLL-2002*, pages 195–198. Taipei, Taiwan, 2002.

[29] G. Zhou and J. Su. Named entity tagging using an HMM-based chunk tagger. In *Proceedings of the 40th Annual meeting of the ACL*, pages 209–219, Philadelphia, PA, July 2002.