Neighborhood Restrictions in Geographic IR

Steven Schockaert[*] [†]
Ghent University
Krijgslaan 281 - S9
9000 Gent, Belgium

Martine De Cock
Ghent University
Krijgslaan 281 - S9
9000 Gent, Belgium

ABSTRACT

Geographic information retrieval (GIR) systems allow users to specify a geographic context, in addition to a more traditional query, enabling the system to pinpoint interesting search results whose relevancy is location-dependent. In particular local search services have become a widely used mechanism to find businesses, such as hotels, restaurants, and shops, which satisfy a geographical restriction. Unfortunately, many useful types of geographic restrictions are currently not supported in these systems, including restrictions that specify the neighborhood in which the business should be located. As the boundaries of city neighborhoods are not readily available, automated techniques to construct representations of the spatial extent of neighborhoods are required to support this kind of restrictions. In this paper, we propose such a technique, using fuzzy footprints to cope with the inherent vagueness of most neighborhood boundaries, and we provide experimental results that demonstrate the potential of our technique in a local search setting.

Categories and Subject Descriptors

I.2 [**Artificial Intelligence**]: Knowledge Representation Formalisms and Methods; H.3 [**Information Storage and Retrieval**]: Information Search and Retrieval

General Terms

Algorithms, Experimentation

Keywords

Geographic Information Retrieval, Fuzzy Footprints

1. INTRODUCTION

An increasing number of research efforts in information retrieval are concerned with providing users with more focused information. For example, question answering systems [4, 16, 22] try to obtain the exact answer to a question of the user — posed in natural language — rather than returning a list of documents, or even paragraphs, which might contain this answer. Another witness of this trend are geographic information retrieval systems [5, 12, 17, 19] and in particular local search services, such as Google Maps[1], Yahoo! local[2] and Microsoft's Live Search[3]. The purpose of these services is to find lists of businesses of a given kind that satisfy some geographical constraint, e.g., *hotels near the Dam square, Amsterdam*. Usually, the kind of business the user is interested in is specified as a list of keywords and phrases, while the geographic constraint is specified by providing an address (or landmark) near which the business should be located.

Despite their overwhelming popularity, the existing local search services suffer from two important limitations. First, the content of the knowledge base that can be queried is to a large extent based on structured information that is a priori available to the system. The static nature of this knowledge base stands in stark contrast to the way traditional search engines work, using a crawler to dynamically update their indices. This makes the creation and updating of the knowledge base an expensive and time-consuming process, and, moreover, it limits the information in the knowledge base to the kind that is typically found in the well-known yellow pages. Current research in GIR systems mainly aims at automating the creation of such a knowledge base, which involves, among others, identifying the geographical scope of web resources based on the occurrence of place names, full addresses, telephone numbers, etc.

Second, only very simple geographic constraints can be specified. While current systems allow for geographic constraints of the form *NEAR* <*address*> or, to some extent, *NEAR* <*landmark*>, other kinds of constraints may be desirable in practice. One particularly useful example are constraints specifying the neighborhood in which the business should be located, e.g., *restaurants in Amsterdam's Museum Quarter*. Local search services and GIR systems in general use gazetteers as their primary source of geographical background knowledge. Unfortunately, most gazetteers contain no information at all about neighborhoods, districts and other types of (non-political) regions, while others provide only a centroid, i.e., the geographical coordinates of a location that is considered to be the center of the region.

[*] Research Assistant of the Research Foundation - Flanders.
[†] Steven.Schockaert@UGent.be (corresponding author)

[1] `http://local.google.com`
[2] `http://local.yahoo.com`
[3] `http://local.live.com`

SIGIR'07, July 23–27, 2007, Amsterdam, The Netherlands.

The main reason for this is that the boundaries of most regions are ill-defined. For example, the absence of region boundaries in the well-known GNIS gazetteer is motivated as follows[4]:

> Regions are application driven and highly susceptible to perception. Sometimes, people might agree on the core of a region, but agreement deteriorates rapidly outward from that core.

As a consequence, current local search services provide almost no support for geographical restrictions involving neighborhood names.

In this paper, we propose a technique to automatically construct a representation of the spatial extent of neighborhoods. We are particularly interested in the impact of using such a representation in a local search context. Because of the subjective and vague nature of many neighborhoods, we do not want to commit ourselves to one single boundary for each neighborhood. Rather, we will represent the extent of a neighborhood as a fuzzy set of locations, called a fuzzy footprint, i.e., a mapping F from locations to the unit interval $[0, 1]$. For a location x, $F(x)$ expresses the degree to which x belongs to the neighborhood.

Particularly for an information retrieval task, fuzzy footprints are more suitable than regions with crisp boundaries, since the membership degrees allow to rank the results based on the extent to which they satisfy the geographical constraint. First, businesses in the core of the neighborhood are returned as everybody would agree that these businesses satisfy the geographical constraint. Next, businesses with a decreasing degree of membership are returned, i.e., businesses for which there might be an increasing amount of disagreement.

This paper is structured as follows. In the next section, we review some existing work on the representation of vague regions. In Section 3, we introduce our algorithm for constructing fuzzy footprints of neighborhoods. Next, in Section 4, we evaluate the performance of our fuzzy footprints in the context of geographic information retrieval. Finally, Section 5 presents some concluding remarks and directions for future work.

2. RELATED WORK

More often than not, the boundaries of a geographic region are only vaguely defined (e.g., [2, 3, 10, 11, 26]). Various formalisms have already been adopted to represent the spatial extent of such a vague region, including supervaluation semantics [2, 18, 26], pairs of crisp (i.e., non-fuzzy) sets [3, 7, 9] and fuzzy footprints [13, 14, 15, 25].

In [20], a user study was conducted which indicates that neighborhoods like *downtown* are indeed perceived as vague by most people. Moreover, when comparing the interpretations of the same neighborhood by different people, a fair amount of agreement was witnessed, although two people seldom agree on the exact (vague) boundaries. The results of this study are important as they indicate that constructing a fuzzy set to represent the spatial extent of a neighborhood is indeed meaningful. In [14], a statistical model is described that predicts the probability that a user would use a particular neighborhood name R to describe the location of a particular shop s. This model provides some evidence that this probability depends on the distance between s and the center c of R, and on the density of shops on the path between c and s.

A few automatic procedures to construct representations of vague regions already exist. In some approaches, a single crisp boundary is constructed to represent a vague region, assuming that the vagueness of the boundary is not important for the intended application. In [23], for example, an algorithm is discussed to find a reasonable polygon for a vague region R, based on a set of points that are assumed to lie in the region, and a set of points that are assumed to lie outside the region. These sets are extracted automatically from web pages containing phrases like *x is located in R*. Another way of obtaining such a polygon is proposed in [1], pursuing a similar strategy.

A graded approach is introduced in [21], using an interpolation technique to obtain a representation similar to fuzzy footprints from a weighted set of points that are assumed to lie in the region. This set of points is obtained by querying Google[5] for pages about the region and assuming that every place on these pages is located in the region. In this way, many false positives are obtained, i.e., places that are incorrectly assumed to lie in the region. To make the approach more robust to such errors in the input, the points are weighted based on their frequency of occurrence [8]. These weights, however, reflect the importance of a particular place, rather than a degree of membership in the corresponding region. Finally, [25] presents a technique to refine the definition of a fuzzy footprint, using natural language constraints found in web documents such as *x is located in the north of R*.

All of the aforementioned approaches deal only with large-scale regions such as the Alpes, Western Europe, etc. To our knowledge, the automatic construction of (fuzzy) footprints for city neighborhoods has not yet been considered. As for large-scale regions, official boundaries for city neighborhoods are usually nonexistent. Moreover, various users studies (e.g., [24, 6]) have shown that in cases where official definitions do exist, these definitions rarely correspond to residents' perception of the neighborhood boundaries.

3. FUZZY FOOTPRINTS

3.1 Finding places in a neighborhood

To construct a fuzzy footprint for a neighborhood L, we use the web to extract names of places that are assumed to lie in L. For large-scale regions, simple patterns are often used to find out, for example, that Amsterdam and Brussels are located in Western Europe, from a sentence like *Many cities in Western Europe, including Amsterdam and Brussels, ...* For most city neighborhoods, too few places can be obtained in this way.

One possible solution would be to query Google for pages about the neighborhood and extract all addresses on these pages, in a similar way as was done in [8, 21] for large-scale regions. However, this requires disambiguation techniques to decide which pages are focusing on the intended neighborhood, and which pages are focusing on neighborhoods in other cities, or even non-geographical entities with the same

[4] `http://geonames.usgs.gov/faqs.html#25`, accessed January 18, 2007.

[5] `http://www.google.com`

name. As an alternative, we use the Yahoo! local API[6] to find appropriate places. Queries to Yahoo! local consist of two parts: a geographic restriction and the actual keyword-based query specifying which businesses we are interested in. For example, to find places that lie in Seattle's Belltown neighborhood, we would send a query with *Seattle* as the geographic restriction and *Belltown* as the actual query. What is returned is a list of businesses in Seattle that contain the word *Belltown* in the name of the business, in the accompanying natural language description of the business, in a user review, or in one of the other fields describing the business.

3.2 Weighting the input data

Using Yahoo! local, we usually obtain a relatively high number of places for the neighborhood of interest. However, not all of these places are actually located in the neighborhood. To increase the robustness of the algorithm, we attach weights to each of the places, expressing our confidence that they are actually located in the neighborhood. Let $p_1, p_2, \ldots, p_k$ be the list of places that was returned for some neighborhood of interest L. Our confidence in each of these places is based on two different assumptions:

1. The position of a place in the list that was returned by Yahoo! local is a good indication of the probability that the place is actually located in the neighborhood.

2. The further a place is from the center of a neighborhood, the less likely it is located in this neighborhood.

The first assumption is inspired by the fact that the ordering of the businesses in the list returned by Yahoo! local, is based on the importance of the query terms (i.e., the neighborhood name) in their descriptions. For example, the places with the highest ranks are places whose name contains the name of the neighborhood. We can be quite confident that these places are indeed located in the neighborhood; e.g., *Belltown Pizza* is probably located in the Belltown neighborhood. On the other hand, places that are further down the list often contain the neighborhood name, for example, only in some user review. Our confidence a_i in the fact that p_i is indeed located in L, based on the first assumption, is defined by:

$$a_i = \begin{cases} 1 & \text{if } L \text{ occurs in the name of } p_i \\ \max(0.3, 1 - \frac{i}{k}) & \text{otherwise} \end{cases}$$

Note that our confidence in the correctness of a place is at least 0.3. This ensures that even the places towards the end of the list will have some — albeit limited — impact on the final result.

The idea behind the second assumption is that, although quite a few of the places returned by Yahoo! local may not be located in the corresponding neighborhood, we can still identify the center of the neighborhood in a very accurate way. To model this notion of center, we use the medoid m of the set $p_1, p_2, \ldots, p_k$, defined by:

$$m = \operatorname*{argmin}_{p \in \{p_1, \ldots, p_k\}} \sum_{i=1}^{k} d(p, p_i) \tag{1}$$

where d is the straight-line distance (circle distance). In other words, the medoid is the place for which the sum of the distances to all the other places is minimal. Our confidence b_i in the fact that p_i is indeed located in L, based on the second assumption, is defined as a decreasing function of the distance between p_i and m:

$$b_i = \begin{cases} 1 & \text{if } d(p_i, m) \leq \alpha \\ \frac{\alpha + \beta - d(p_i, m)}{\beta} & \text{if } \alpha < d(p_i, m) < \alpha + \beta \\ 0 & \text{if } d(p_i, m) \geq \alpha + \beta \end{cases}$$

Note how the values of α and β reflect how tolerant we are w.r.t. our second assumption. We can, for example, define these values based on how close to the medoid *most* of the places are located. In particular, let $\pi_1, \pi_2, \ldots, \pi_k$ be a permutation of $1, 2, \ldots, k$ such that $d(p_{\pi_1}, m) \leq d(p_{\pi_2}, m) \leq \cdots \leq d(p_{\pi_k}, m)$. We assume that at least 60% of the places are correct, i.e., located in the neighborhood L. This is reflected in the following definition of α (assuming, for simplicity, that k is a multiple of 5):

$$\alpha = d(p_{\pi_{0.6k}}, m) \tag{2}$$

The value of β will determine how tolerant we are for the remaining places. The idea is that the difference $d(p_{\pi_{0.6k}}, m) - d(p_{\pi_{0.4k}}, m)$ gives a good indication of how tightly the neighborhood is clustered around the center m. This leads to the following definition of β:

$$\beta = 4(d(p_{\pi_{0.6k}}, m) - d(p_{\pi_{0.4k}}, m))$$

Finally, our overall confidence c_i in the correctness of a place p_i is defined as the product of a_i and b_i:

$$c_i = a_i b_i$$

3.3 Defining neighborhoods

Neighborhood boundaries are generally considered to be inherently fuzzy [24]. However, apart from their gradual nature, neighborhood boundaries are also ill-defined because of a lack of agreement between different people. Several studies have shown, for example, that the perception of the boundaries of a neighborhood is influenced by factors such as age, gender, length of residence, socio-econonomic class, etc. (e.g., [6]). Hence, the degree of membership of a place in a fuzzy footprint should reflect how much people agree that this place is part of the neighborhood.

A related problem is that the definition of neighborhood boundaries is context-dependent. For example, in some contexts, Seattle's Belltown neighborhood is considered to be a part of Downtown Seattle, while in other contexts it is assumed that the two neighborhoods are bordering on each other. We cope with this by defining a neighborhood relative to some list of neighborhoods $\mathcal{L} = \{L_1, L_2, \ldots, L_n\}$. In the first context, $\mathcal{L}$ will contain both Downtown and Belltown, while in the second context, Belltown will be excluded from $\mathcal{L}$. Intuitively, the list $\mathcal{L}$ defines a partitioning of a city into a set of neighborhoods, i.e., such that the spatial extent of the city is equal to the union of the spatial extents of the neighborhoods in $\mathcal{L}$, and such that the spatial extents of the neighborhoods are pairwise disjoint. However, we also allow *default regions* like Central Seattle, whose spatial extent, in this context, would cover all places in Central Seattle that are not contained in any of the other neighborhoods. In this way, our approach can also be used when a complete enumeration of every neighborhood is not available.

[6] `http://developer.yahoo.com/search/local/V2/localSearch.html`

For convenience, we use L_i both to refer to the name of a neighborhood, and to the fuzzy footprint describing its spatial extent. To construct this fuzzy footprint, we use the places $p_1^i, p_2^i, \ldots, p_{k_i}^i$ extracted from Yahoo! local, and their confidence scores $c_1^i, c_2^i, \ldots, c_{k_i}^i$, where all the places p_j^i are assumed to be located in L_i. Such a set of places has been obtained for every neighborhood in $\mathcal{L}$. Let $\mathcal{P}$ be the set of all these places, and let $\mathcal{P}_i = \{p_1^i, p_2^i, \ldots, p_{k_i}^i\}$ be the set of places corresponding to neighborhood L_i; note that $\mathcal{P} = \bigcup_{i=1}^n \mathcal{P}_i$.

The main idea to define the membership degree $L_i(x)$ of an arbitrary location x (i.e., not necessarily corresponding to a place in $\mathcal{P}$) in the neighborhood L_i, is to use the fraction of nearby places that are assumed to lie in L_i, i.e., included in the set $\mathcal{P}_i$. This idea is closely related to a voting model for fuzzy sets, where the degree of membership of an object in a fuzzy set modelling a certain vague property, reflects the percentage of people that would answer positive when asked whether or not this object satisfies the property. However, rather than treating all nearby places (votes) in the same way, the impact of each place is weighted based on its confidence score and its distance to x:

$$L_i(x) = \frac{\sum_{x_0 \in N_x} f_i(x_0) g(x, x_0)}{\sum_{x_0 \in N_x} \max_{j=1}^n f_j(x_0) g(x, x_0)} \qquad (3)$$

where f_j is defined for a place x_0 as ($j \in \{1, 2, \ldots, n\}$)

$$f_j(x_0) = \begin{cases} c_s^j & \text{if } x_0 = p_s^j \text{ for some } s \in \{1, 2, \ldots, k_j\} \\ 0 & \text{otherwise} \end{cases}$$

The value $f_j(x_0)$ is equal to our confidence that x_0 is located in L_j, provided $x_0 \in \mathcal{P}_j$, i.e., provided x_0 is contained in the list of businesses returned by Yahoo! local for the neighborhood L_j; otherwise, $f_j(x_0) = 0$. The function g should be a decreasing function of the distance between x and x_0. We used the function g defined for two locations x and x_0 as

$$g(x, x_0) = \frac{1}{1 + d(x, x_0)}$$

Finally, the set N_x is the set of places that are considered to be nearby x ($N_x \subseteq \mathcal{P}$).

3.4 Analyzing the fuzzy footprints

The impact of using different definitions of N_x is illustrated in Figures 1 and 2. In Figure 1 (resp. 2), the crosses correspond to the places from $\mathcal{P}$ that are assumed to lie in the Capitol Hill (resp. Downtown) neighborhood of Seattle, while the dots correspond to the places that are assumed to lie in one of the other neighborhoods. The empty area in the lower left corner corresponds to the sea. Both figures display the same places, although the crosses from one figure will correspond to dots in the other figure.

When N_x only contains the 5 places of $\mathcal{P}$ that are closest to x, the resulting fuzzy footprint is very sensitive to the actual input, i.e., to the businesses that were returned by Yahoo! local. When increasing the number of places that are considered, the resulting fuzzy footprint becomes smoother, and less sensitive to individual places in the input. In the remainder of this paper, we will assume that N_x contains the 100 places that are closest to x. Note that when other sources would be available which would allow to find large sets of places for each neighborhood in a more accurate way, optimal performance might be achieved by looking at a lower number of nearby places.

Note that, as can be seen from Figure 2, a large part of the area that is covered by the fuzzy footprint of Downtown is actually located in the sea. Although this is clearly incorrect, it is of no importance in the context of local search, since there are no businesses located in the sea (otherwise no parts of the sea would have been covered by the fuzzy footprints in the first place). In contexts where this would be a problem, this can easily be solved by intersecting the fuzzy footprints with a detailed footprint of Seattle, based on the official boundaries, which can be found in the Tiger gazetteer[7].

Dark regions in the representation of Capitol Hill in Figure 1 correspond to light regions in the representation of Downtown in Figure 2, and vice versa. This is particularly noticeable in Figures 1(a) and 2(a). More generally, we can show that for any location x, it holds that

$$\sum_{i=1}^n L_i(x) = 1$$

In other words, the fuzzy footprints $L_1, L_2, \ldots, L_n$ define a fuzzy partition of the city under consideration. This is important because it ensures that our intended intuitive meaning of $\mathcal{L}$ as a an exhaustive and mutual exclusive set of neighborhoods is reflected in the definition of the fuzzy footprints.

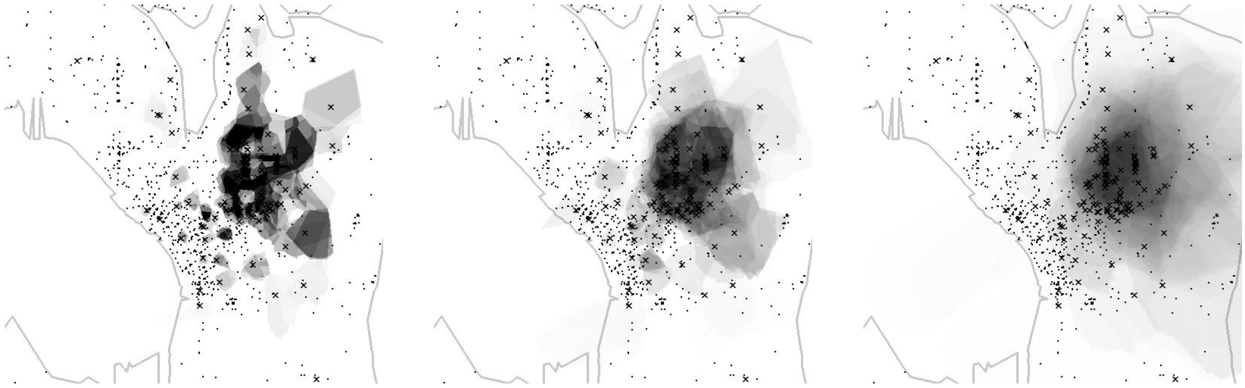
Figure 3 illustrates our ability to deal with the context-dependency of neighborhood boundaries. In particular, the definition of Downtown Seattle is shown in two different contexts. In the first context, Belltown is assumed to be a neighborhood next to Downtown, i.e., Belltown, as well as Downtown, is included in the list of neighborhoods $\mathcal{L}$. The resulting fuzzy footprints for Belltown and Downtown are shown in Figures 3(a) and 3(b). In the second context, Belltown is not included in the list of neighborhoods $\mathcal{L}$, and is thus implicitly assumed to be a part of Downtown. The fuzzy footprint for Downtown in this second context is shown in Figure 3(c). Note that the places and fuzzy footprints in Figure 3 are shown at a smaller scale than those in Figures 1 and 2.

4. EXPERIMENTAL RESULTS

As there are no official boundaries for most city neighborhoods, it is difficult to evaluate the quality of our fuzzy footprints directly. Instead, we will analyse the impact of using the fuzzy footprints in a local search context. To this end, we will compare the results of a query of the form *restaurants in <neighborhood>*, obtained using our fuzzy footprints, against a manual classification of restaurants by neighborhood. We extracted such a manual classification from *restaurants.com*[8] for 15 US cities: Atlanta, Austin, Baltimore, Boston, Cambridge, Chicago, Las Vegas, Los Angeles, Miami, Minneapolis, New York, Philadelphia, San Diego, San Francisco, and Seattle. For these 15 cities, *restaurants.com* contains information about 13681 restaurants in 410 different neighborhoods. To allow for meaningful precision and recall scores, we limited our experiments to queries about the 149 neighborhoods containing at least 25 restaurants.

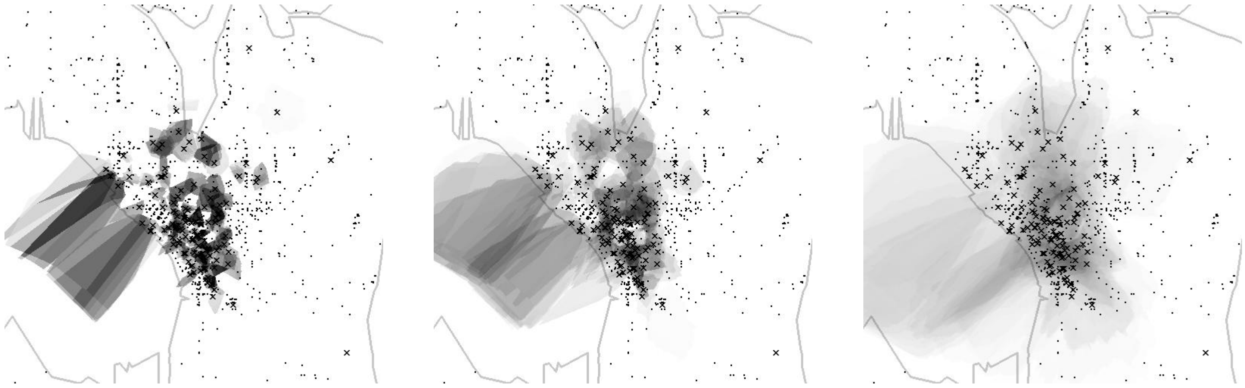
[7] http://www.census.gov/geo/tiger99/tl_1999.html

[8] http://www.restaurants.com



(a) Capitol Hill — 5 places (b) Capitol Hill — 20 places (c) Capitol Hill — 100 places

Figure 1: Definitions of the fuzzy footprints for Seattle's Capitol Hill neighborhood for varying definitions of the set N_x of places nearby x. In (a) the set N_x consists of the 5 places closest to x, while in (b) and (c), 20 places and 100 places are used respectively. Darker regions correspond to a higher degree of membership.



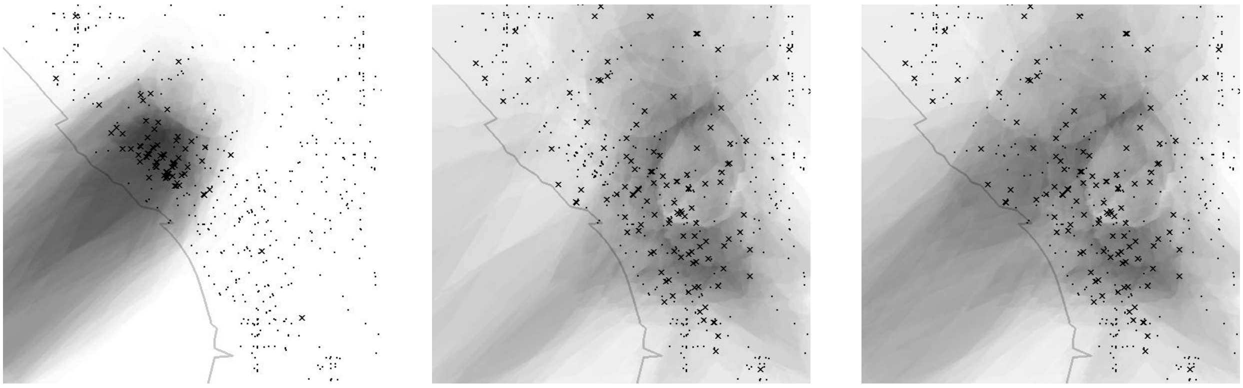
(a) Downtown — 5 places (b) Downtown — 20 places (c) Downtown — 100 places

Figure 2: Definitions of the fuzzy footprints for Downtown Seattle when the closest (a) 5 places, (b) 20 places, and (c) 100 places are contained in N_x. Darker regions correspond to a higher degree of membership.

In the first experiment, we investigate the precision and recall (w.r.t. the manual classification from *restaurants.com*) of the complete set of restaurants that are returned by the system, ignoring any ranking of the restaurants. When using the fuzzy footprints, a threshold value λ in $]0,1]$ has to be chosen to decide which restaurants to return. The set of restaurants that is returned is then equal to the set of restaurants whose degree of membership in the fuzzy footprint for the neighborhood imposed by the query is at least λ. Clearly this threshold parameter can be used to tune the performance of the system towards better precision or better recall. Figure 4 shows the resulting precision/recall trade-off, and compares our approach with two baseline techniques.

Both baseline systems return all restaurants that are located within a certain radius of the medoid of the neighborhood, as defined in (1). For the first baseline, this radius is a constant r. By increasing or decreasing the value of r, the precision/recall trade-off can be adjusted. The second baseline system returns all restaurants within a radius of $r_0\alpha$, where α is defined in (2), and r_0 is a constant. The idea here is that the value of α gives a good indication of the size of the neighborhood, and should thus be useful to determine an appropriate radius. As can be seen from Figure 4, neither of the two baselines is better than the other. When high precision is needed, the first baseline performs better than the second, while for higher recall values, the second baseline outperforms the first.

Clearly, the system using our fuzzy footprints constitutes a significant improvement over both baseline systems. For low recall values, the precision is almost 1, from which we can conclude that the fuzzy footprints correctly identify the core of the neighborhoods. Towards the higher recall values, the gain in performance over the baseline systems somewhat decreases. This can be explained by the fact that to obtain a high recall, some restaurants may have to be included for which there might be disagreement about which neighborhood they belong to. As the assignment of a single neighborhood to such restaurants is, to some extent, arbitrary, it becomes harder to make a more intelligent decision than simply returning every restaurant within a certain radius. In



(a) Belltown — Context 1 (b) Downtown — Context 1 (c) Downtown — Context 2

Figure 3: In (a), a fuzzy footprint of the Belltown neighborhood in Seattle is shown, where darker regions correspond to a higher degree of membership. Figures (b) and (c) show a fuzzy footprint of Downtown Seattle in two different contexts, viz. when Belltown is included in the list of neighborhoods $\mathcal{L}$ (Context 1), and when it is not (Context 2).

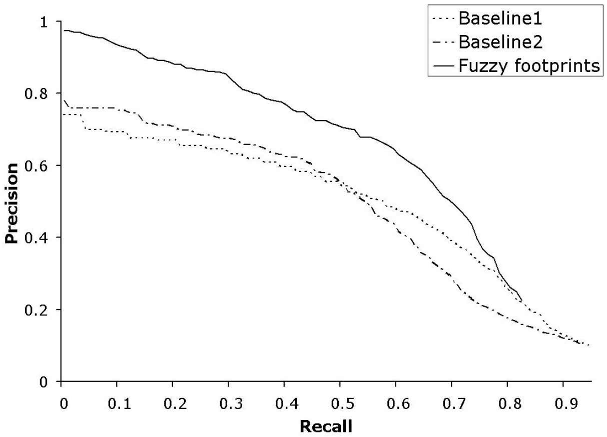


Figure 4: Precision/recall trade-off for experiment 1, using a system based on fuzzy footprints and two baseline systems. Each of the three systems involves a parameter that can be adjusted in favor of precision or recall.

fact, a similar behaviour could be expected when comparing different human assignments.

Using the fuzzy footprints, it is not possible to obtain a recall value that is, on average, more than 0.83 for the 149 queries considered. This means that on average, 17% of the restaurants that are located in a neighborhood have membership degree 0 in the corresponding fuzzy footprint. The main reason for this is that for some lesser known neighborhoods, too few businesses are returned by Yahoo! local. As a consequence, some of the fuzzy footprints are completely incorrect, and cover (almost) none of the restaurants that are actually located in the neighborhood. In other words, the problem lies mainly in the data acquisition phase, rather than in the construction of the fuzzy footprints. Note that the precision values corresponding to the lower recall values are not affected by this, because the membership degrees of restaurants in the fuzzy footprints of these problematic neighborhoods are all very low. If the threshold λ is set to be sufficiently high, then no restaurants at all will be returned for these neighborhoods.

The evaluation task in the first experiment can be seen as a two–stage process. First, the restaurants have to be ranked, based on how likely it is that they should be included in the result set, and then a choice has to be made about how many restaurants to return, i.e., at which point to cut off the ranked list of restaurants. Note that the two baseline systems in our first experiment only differ in the second stage of this process; both use the distance of the restaurants to the medoid of the neighborhood as the ranking criterium. In a second experiment, we only looked at the ranking of the restaurants. In the system based on fuzzy footprints, the degree of membership of a restaurant in the fuzzy footprint of the neighborhood is used to rank the restaurants. The distance between each of the restaurants and the medoid is used to break ties, and, in particular, to rank the restaurants that have membership degree 0. In the baseline system, the restaurants are ranked according to their distance to the medoid. For each of the 149 queries, the precision at different recall levels, and at different list cutoffs, was calculated. The resulting precision/recall graph is shown in Figure 5.

This figure shows that using fuzzy footprints results in a better ranking of the restaurants. However, the improvement over the baseline system is less apparent than in the first experiment. At very low recall points, the precision of the fuzzy footprint approach and the baseline is even (almost) identical. This is because in both systems the top restaurants in the list are usually the same, i.e., the restaurants in the immediate vicinity of the medoid of the neighborhood. Similar conclusions can be drawn from the results in Table 1, which compares the precision at different (fixed) list cutoffs for both rankings.

For the task of ranking the restaurants, only the relative order of the membership degrees is important. The second stage of the process from experiment 1, on the other hand, is based on the actual values of the membership degrees. Although the second experiment demonstrates that

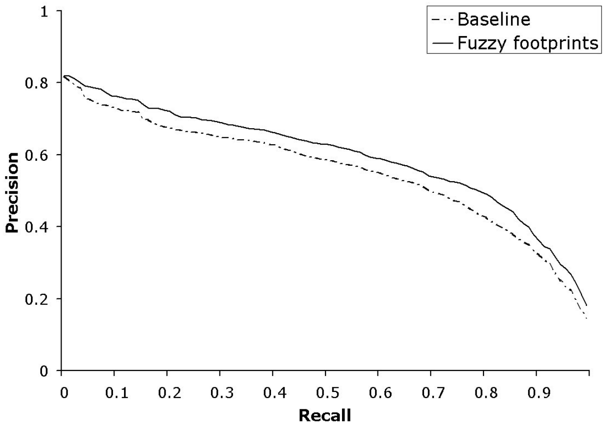


Figure 5: Averaged precision/recall graph for experiment 2, using a system based on fuzzy footprints and a baseline system.

Table 1: Precision at fixed list cutoffs for the fuzzy footprint approach and the baseline system in experiment 2. $P@n$ denotes the precision of the first n restaurants in the list.

	Baseline	Fuzzy footprints
P@1	0.660	0.654
P@2	0.629	0.673
P@3	0.648	0.667
P@4	0.657	0.684
P@5	0.657	0.679
P@10	0.648	0.677
P@20	0.618	0.653
P@30	0.578	0.623
P@40	0.535	0.585
P@50	0.499	0.546

the ordering of the restaurants imposed by the membership degrees of the fuzzy footprints is useful, it also shows that a significant gain in precision is achieved by choosing the right cutoff position, based on the absolute membership degrees of the fuzzy footprints.

5. CONCLUDING REMARKS

In this paper, we introduced a technique to implement neighborhood restrictions in GIR systems. We discussed how places that lie in a given neighborhood can be found using an existing local search service, how confidence scores can be attached to these places to increase the robustness of the approach, and how these places can be used to obtain fuzzy footprints. The difficulties in finding large sets of places that lie in each neighborhood necessitate strategies that introduce a lot of noise, i.e., many places will actually be located in a different neighborhood than they are assumed to be. Therefore, any technique to construct representations of city neighborhoods, based on data gathered from the web, has to be extremely robust to such falsely classified places. In our approach, this robustness is achieved by considering a sufficiently high number of nearby places in the set N_x in (3). The higher this value, the less the definition of the fuzzy footprints is influenced by one, or a few, places.

It is important to realize that neighborhood boundaries are inherently ill-defined, and, to some extent, subjective, and that a perfect definition of the spatial extent of neighborhoods can therefore not exist. However, as previous studies have shown, people tend to agree on the core of a neighborhood. The experimental results in this paper demonstrate that our technique can successfully identify this core, and that, moreover, a reasonable choice is made about the degree of membership of the borderline cases, suggesting that supporting neighborhood restrictions based on automatically generated representations is indeed feasible.

Our fuzzy footprints are always defined relative to some list of neighborhoods. In this way, different fuzzy footprints can be obtained for the same neighborhood in different contexts. This is important, since the interpretation of the boundaries of a neighborhood may be context-dependent. However, we have not discussed how such a list can be obtained. One possibility is to use manually defined lists of neighborhoods for every city of interest. Another promising solution would be to extract such a list automatically from web documents. In this way, a more complete list of neighborhoods may be obtained for some cities. Also, it may be useful to try to extract information about spatial relationships between neighborhoods, districts, and other types of regions in cities (e.g., A is a part of B, A is bordering on B) . These spatial relationships could be used to weaken queries that are too restrictive, to improve the quality of the fuzzy footprints, and to try to find an appropriate context (i.e., list of neighborhoods) automatically, depending on the query or the user's profile.

6. REFERENCES

[1] A. Arampatzis, M. van Kreveld, I. Reinbacher, C. Jones, S. Vaid, P. Clough, H. Joho, and M. Sanderson. Web-based delineation of imprecise regions. *Computers, Environment and Urban Systems*, 30(4):436–459, 2006.

[2] B. Bennett. What is a forest? on the vagueness of certain geographic concepts. *Topoi*, 20(2):189–201, 2001.

[3] T. Bittner and J. Stell. Vagueness and rough location. *Geoinformatica*, 6(2):99–121, 2002.

[4] E. Brill, J. Lin, M. Banko, S. Dumais, and A. Ng. Data-intensive question answering. In *Proceedings of the 10th Text REtrieveal Conference (TREC 2001)*, 2001.

[5] B. Bucher, P. Clough, H. Joho, R. Purves, and A. Syed. Geographic IR systems: requirements and evaluation. In *Proceedings of the 22nd International Cartographic Conference*, 2005.

[6] C. Cho. *Study on the effects of resident-perceived neighborhood boundaries on public services accessibility & its relation to utilization: using geographic information system, focusing on the case of public parks in Austin, Texas.* PhD thesis, Texas A&M University, 2003.

[7] E. Clementini and P. Di Felice. Approximate topological relations. *International Journal of Approximate Reasoning*, 16(2):173–204, 1997.

[8] P. Clough, H. Joho, C. Jones, and R. Purves. Modelling vague places with knowledge from the web. *Unpublished, available at* `http://ext.dcs.shef.ac.`

uk/~u0015/darwinProjectProposals/SandersonMark/clough.pdf, 2005.

[9] A. Cohn and N. Gotts. The 'egg-yolk' representation of regions with indeterminate boundaries. In *Geographic Objects with Indeterminate Boundaries (P.A. Burrough and A.U. Frank, eds.)*, pages 171–187, 1996.

[10] M. Erwig and M. Schneider. Vague regions. In *Proceedings of the 5th Int. Symp. on Advances in Spatial Databases, LNCS 1262*, pages 298–320, 1997.

[11] P. Fisher. Sorites paradox and vague geographies. *Fuzzy Sets and Systems*, 113(1):7–18, 2000.

[12] F. Gey, R. Larson, M. Sanderson, K. Bischoff, T. Mandl, C. Womser-Hacker, D. Santos, and P. Rocha. GeoCLEF 2006: the CLEF 2006 cross-language geographic information retrieval track overview. In *Working Notes for the CLEF 2006 Workshop*, 2006.

[13] M. Goodchild, D. Montello, P. Fohl, and J. Gottsegen. Fuzzy spatial queries in digital spatial data libraries. In *Proceedings of the IEEE World Congress on Computational Intelligence*, pages 205–210, 1998.

[14] Y. Harada and Y. Sadahiro. A quantitative model of place names as a georeferencing system. In *Proceedings of GeoComputation*, 2005.

[15] L. Hill, J. Frew, and Q. Zheng. Geographic names: the implementation of a gazetteer in a georeferenced digital library. *D-Lib Magazine*, 5(1), 1999.

[16] V. Jijkoun and M. de Rijke. Answer selection in a multi-stream open domain question answering system. In *Proceedings of the 26th European Conference on Information Retrieval, LNCS 2997*, pages 99–111, 2004.

[17] C. Jones, R. Purves, A. Ruas, M. Sanderson, M. Sester, M. van Kreveld, and R. Weibel. Spatial information retrieval and geographical ontologies: an overview of the SPIRIT project. In *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 389–390, 2002.

[18] L. Kulik. A geometric theory of vague boundaries based on supervaluation. In *Proceedings of the International Conference on Spatial Information Theory: Foundations of Geographic Information Science (COSIT 2001), LNCS 2205*, pages 44–59, 2001.

[19] R. Larson and P. Frontiera. Geographic information retrieval (GIR): searching where and what. In *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, page 600, 2004.

[20] D. Montello, M. Goodchild, J. Gottsegen, and P. Fohl. Where's downtown?: behavioral methods for determining referents of vague spatial queries. *Spatial Cognition and Computation*, 3(2-3):185–204, 2003.

[21] R. Purves, P. Clough, and H. Joho. Identifying imprecise regions for geographic information retrieval using the web. In *Proceedings of the 13th Annual GIS Research UK Conference*, 2005.

[22] D. Radev, W. Fan, H. Qi, H. Wu, and A. Grewal. Probabilistic question answering on the web. *Journal of the American Society for Information Science and Technology*, 56(6):571–583, 2005.

[23] I. Reinbacher, M. Benkert, M. van Kreveld, J. Mitchell, and A. Wolf. Delineating boundaries for imprecise regions. In *Proceedings of the 13th European Symposium on Algorithms, LNCS 3669*, pages 143–154, 2005.

[24] N. Sastry, A. Pebley, and M. Zonta. Neighborhood definitions and the spatial dimension of daily life in Los Angeles. In *2002 annual meeting of the Population of Daily Life in Los Angeles, available at http://www.rand.org/labor/DRU/DRU2400_8.pdf*, 2002.

[25] S. Schockaert, M. De Cock, and E. Kerre. Automatic acquisition of fuzzy footprints. In *Proceedings of the Workshop on Semantic-based Geographical Information Systems, LNCS 3762*, pages 1077–1086, 2005.

[26] A. Varzi. Vagueness in geography. *Philosophy & Geography*, 4(1):49–65, 2001.