Recommending Interesting Activity-Related Local Entities

Jie Tang
University of California, Berkeley
Berkeley, CA 94709
jietang@eecs.berkeley.edu

Ryen W. White
Microsoft Research
Redmond, WA 98052
ryenw@microsoft.com

Peter Bailey
Microsoft Bing
Bellevue, WA 98004
pbailey@microsoft.com

ABSTRACT

When searching for entities with a strong local character (e.g., a museum), people may also be interested in discovering proximal activity-related entities (e.g., a café). Geographical proximity is a necessary, but not sufficient, qualifier for recommending other entities such that they are related in a useful manner (e.g., interest in a fish market does not imply interest in nearby bookshops, but interest in other produce stores is more likely). We describe and evaluate methods to identify such activity-related local entities.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval – *search process, selection process.*

General Terms

Algorithms, Measurement, Experimentation.

Keywords

Local entities, entity resolution, entity-entity matching.

1. INTRODUCTION

People are frequently interested in finding out information about specific local entities, say the Museum of Modern Art (MOMA) in the city of New York, NY. They may be tourists and planning a trip, or residents exploring their city. In addition to retrieving Web documents matching users' search queries for the specific local entity, modern Web search engines may provide a rich and diverse results page with additional information related to the query. Understanding the local user intent behind such users' queries offers an opportunity to surface recommendations about other interesting activity-related local entities. For example, after visiting MOMA, a shopping trip or a visit to a local bar may be common activities. These relationships may not be commutative: an interest in a bar is unlikely to imply an interest in nearby cultural centers. Similarly, an interest in a cupcake shop may mean an interest in other nearby cupcake shops, not some different class of local entity. Lastly, as observed by Jones *et al.* [2], the nature of the entity and/or corresponding activity may imply a different willingness to travel. Thus for suggesting activity-related local entities, an interest in a major city attraction may indicate an interest in other city attractions several miles away, but an interest in a bar may only suggest interest in other nearby bars or cafés.

2. IDENTIFYING LOCAL ENTITIES

This work describes a procedure for identifying related local entities. Xiao *et al.* [6], similarly motivated about local *relatedness*, observed behavior in geographic (local map vertical) search logs and derived mechanisms to identify co-located queries at different geographic region resolutions. Our algorithms are unrelated to theirs, and we focus on identifying *entities*, not queries. Note that although we conduct our study in the context of Seattle, WA, the methods described could be applied to any city of reasonable size.

2.1 Pre-processing and Geocoding

Unlike Wang *et al.*'s approach [4], which works to extract dominant locations (either explicit or implicit) associated with *queries*, we focused on identifying an *entity*'s location. The first step involves the identification of URLs that may be relevant to the location of interest. We obtained clickthrough logs from the Bing Web search engine and mined queries and the URLs that users clicked on for those queries, providing us with implicit associations between queries and URLs. We extracted URLs containing "seattle" in the associated query string, and identified address-based queries and their query-URL mapping. For example, the query [*pike place market, pike street, seattle*] could map to pikeplacemarket.org, describing the local entity Pike Place Market. To remove infrequently-visited sites, we also use browsing logs from a widely-distributed browser plugin, and retained URLs that appear in those logs and the clickthrough logs. This left us with a set of five to ten thousand URLs, potentially describing a local Seattle entity. To determine the physical location for each entity we use the Bing Maps Geocoding API (microsoft.com/maps/developers), which processes queries and returns their geocoded locations. Geocoding results over 100 miles from Seattle were removed from analysis.

2.2 Entity Resolution

We employ entity resolution to get from a URL to other URLs that refer to the same entity. Starting from every URL in the set created using the steps above we find the query which sends the most traffic to that URL (the canonical query). We then look at the top URLs by clickthrough rate for this query, and select those which attract more than 5% of the total click count. These URLs comprise an entity cluster, and we assign as the canonical entity the URL with the highest total click count of the set.

2.3 Entity Recommendation

Entities are represented by their canonical URL from Section 2.2. Entity-entity matching algorithms recommend related entities by processing entity information, location information, and clickthrough or session data, to generate an affinity matrix, A. The matrix is $N \times N$, where N is the number of entities, and the entry $A_{\{i,j\}}$ determines the relatedness between entities i and j

Click graph: Search engine result-page clickthrough data from one year of Bing search logs are incorporated by using the canonical URLs as starting points and performing a two-step graph walk (URL to queries then back out to URLs) to obtain related URLs.

Search sessions: The frequency of URL co-occurrence in search sessions from one-week of browsing logs, comprising search engine queries and post-query navigation (as described in [5]), is used to compute the degree of relatedness between entities.

We also study a number of extensions to each of these methods:

Merged: We combine the two matrices using a convex weight.

Max-flow: We run a max-flow algorithm on the merged affinity matrix to generate a new affinity matrix A', where $A'_{\{i,j\}}$ corresponds to the maximum amount of flow that can be pushed from

SIGIR'11, July 24–28, 2011, Beijing, China.
ACM 978-1-4503-0757-4/11/07.

entity i to entity j [1]. For efficiency, this is restricted to a two-hop max-flow where flow can travel up to two edges.

Hitting time: We run a hitting time algorithm on the merged affinity matrix to create a new matrix A', where $A'_{\{i,j\}}$ is proportional to the expected time before a random walk from i reaches j [3].

3. JUDGING ENTITIES

Evaluating our entity resolution and recommendations requires ground truth data, preferably from human raters, who may be able to determine inter-entity relationships. To this end, we recruited judges from Amazon Mechanical Turk (*turkers*).

3.1 Selection

We generated sets of URL-URL pairs via a graph walk over A: for every canonical URL, we pull a few URLs from that cluster, a few URLs which are one hop away, two hops away, etc. to form the other half of the URL-URL pair. This ensures that we have pairs covering a spectrum from unrelated to exact matches to be judged.

3.2 Methodology

Judges considered the following scenario: *You are planning an outing in the city and you are searching for ideas for what to do. There are a few locations you have heard about or are already looking to visit, and you would like to find more attractions / restaurants / shopping / etc. to round out your day*. We presented people with URL-URL pairs for each of the suggested URLs in each cluster, and asked them to rate the relatedness between the entity associated with the recommended entity and the reference entity on a four-point scale: 1=*URLs are unrelated*, 2=*URLs are related*, 3=*URLs are obviously related*, and 4=*URLs are the same entity*. Distance data estimating the physical distance between the recommended entity and the reference entity were also provided. Three of the authors judged a set of 20 entities and cluster URLs to establish a gold standard data set that was used in selecting turkers. For example, for Seattle Art Museum (seattleartmuseum.org), visiting the Seattle Aquarium (seattleaquarium.org) may be a good related activity and received an average rating of three, whereas a visit to a local garden and stone company (lakeviewstone.com) received an average rating of one. The Fleiss's kappa (κ) between authors was 0.74, signifying substantial agreement and suggesting that the task was reasonable for remote judges.

We scaled out the study to turkers. To control for the quality of the ratings, we did two things. First, we assigned a qualification test drawn from our gold standard truth data, in which each of the volunteers had to attain over 50% correct to attempt any of our human intelligence tasks (HITs). Next, each HIT contained three questions, one drawn from our gold standard set and two novel questions. Each time a HIT was submitted we verified that the turker answer matched our ground truth. We used this to update the qualification score, taking a running average with a weight of 0.05. This continual assessment strategy was employed to ensure that turkers did not work hard to pass the initial qualification and then not devote similar effort to the post-qualification tasks, whose ratings were important for our study. Turkers with qualification below 50% had their ratings excluded from our test data.

3.3 Judgments

In total, we obtained 3,426 URL-URL rating pairs. These raw numbers include both duplicates and those which raters did not agree on. Fleiss's κ was 0.43, signifying moderate agreement between our raters. Each URL-URL pair rated by three separate turkers, and the rating was used only if all three turkers agreed. We had just over 800 questions where all three raters agreed. We use those questions for the evaluation of the performance of our entity resolution and entity-entity recommendations.

4. FINDINGS

4.1 Entity Resolution

Many different URLs can refer to the same local entity (e.g., for the local entity bethscafe.com, we may also return the yelp.com and citysearch.com pages for the same café). Recommending duplicates may create a poor user experience, and we did not want to reward an algorithm for finding duplicate entities. We use the method for resolving entities described in Section 2.2, and evaluate it using our human labeled data. Two URLs are regarded as the same entity iff the human rating is four. The F_1 score, the harmonic mean of precision and recall, was 0.63, suggesting that we can detect duplicate entities with reasonable accuracy.

4.2 Entity Recommendation

To evaluate entity recommendation, two entities are considered related if turkers assigned a rating of two or three, and unrelated if the rating was one. Any non-zero entry in A was counted as a classification. Table 1 shows the F_1 scores for each of the recommendation algorithms described earlier. We used F_1 since we wanted to weight precision and recall equally.

Table 1. F_1 scores for entity recommendation algorithms.

Algorithm	*Click graph*	*Search sessions*	*Merged*	*Merged Max-flow*	*Merged Hitting*
F$_1$-score	0.39	0.28	0.44	0.48	0.49

Merging the click-graph and search-sessions methods led to improved performance over either method alone (0.44 vs. 0.39/0.26). Enhancements to the merged model using max-flow and hitting time algorithms led to further improvements. One explanation for the poor performance of the search sessions is less data (one week of browsing versus one year of click logs). However, the fact that it can still obtain an F_1 score that is 72% of the click graph score with much less data suggests that sessions may be valuable.

5. CONCLUSIONS AND FUTURE WORK

We studied methods for identifying related local entities and identifying duplicate entities using search and browsing behavior. Our methods performed well in both tasks. Entity recommendations based on a combination of clicks and session data performed particularly well, especially when enhanced with max-flow and hitting time. Future work involves developing new and improved algorithms with more log data, evaluating models with a larger set of judgments, and scaling recommendations to multiple locations.

REFERENCES

[1] Ford, L.R. and Fulkerson, D.R. (1956). Maximal flow through a network. *Canadian Journal of Mathematics*, 399–404.

[2] Jones, R., Zhang, W.V., Rey, B., Jhala, P., and Stipp, E. (2008). Geographic intention and modification in Web search. *J. of Geographical Information Science*, 22(3): 229–246.

[3] Mei, Q., Zhou, D., and Church, K. (2008). Query suggestion using hitting time. *Proc. CIKM*, 469–478.

[4] Wang, L., Wang, C., Xie, X., Forman, J., Lu, Y., Ma, W-Y., and Li, Y. (2005). Detecting dominant locations from search queries. *Proc. SIGIR*, 424–431.

[5] White, R.W. and Drucker, S.M. (2007). Investigating behavioral variability in Web search. *Proc. WWW*, 21–30.

[6] Xiao, X., Wang, L., Xie, X., and Luo, Q. (2008). Discovering co-located queries in geographic search logs. *Proc. LOCWEB*, 77–84.