

# Social Tagging for Personalized Location-Based Services

C. Biancalana, F. Gasparetti, A. Micarelli, G. Sansonetti

Department of Computer Science and Automation

Artificial Intelligence Laboratory

Roma Tre University

{claudio.biancalana,gaspares,micarell,gansone}@dia.uniroma3.it

## ABSTRACT

The current generation of location-based services (LBSs) does not provide users with personalized recommendations, but only suggests nearby points of interest (POIs) based on their distance from the user current location. To overcome such a limitation, we have realized a social recommender system able to identify user preferences and information needs, thus suggesting personalized recommendations related to possible POIs in the surroundings. The proposed approach allows users to leverage and assign freely chosen keywords (tags) to resources through collaborative tagging services (folksonomies). Our LBS employs a user-based tag model that derives correspondences between personal tag vocabularies (personomies) and folksonomies. Experimental tests performed on 15 real users enabled us to assess the benefits in terms of performance that our approach is able to provide.

## Author Keywords

Social Recommender System, User Modeling, Ubiquitous Computing

## ACM Classification Keywords

H.3.5 Online Information Services: Web-Based Services

## INTRODUCTION

With the spread of mobile technology, users increasingly resort to location-based services (LBSs) in order to retrieve and share information about nearby points of interest (POIs). To the best of our knowledge, the current generation of LBSs does not provide users with personalized recommendations, but only suggests POIs based on their distance from the user current location. Therefore, they do not consider either user preferences and information needs, or aspects related to the context of use, for example, day of week, time, weather, user activity and means of transport. As a result, users are often overwhelmed with suggestions that may not be of their interest, and/or valid because they are not able to benefit from them, thereby resulting in their dissatisfaction with the overall service. For mobile systems this problem is exacerbated by limitations inherent to the used devices, such as small

screen, restricted input, and limited computing resources available to browse the large number of received recommendations. To address this problem, we propose a social recommender system able to 1. identify user preferences and information needs, 2. take into account the current context of use 3. exploit the wealth of social information available on the Web (e.g., social networks, user reviews, and local search sites). In this paper, we specifically report on the last point. In order to strengthen the social character of the recommender system, our approach allows users to leverage and assign freely chosen keywords (tags) to resources through collaborative tagging services (folksonomies). Folksonomies, which characterize the Web 2.0, represent a valuable source of information for content search, since they result from the aggregation of preferences and interests of large networks of users, and index content through tags. In literature, the advantages of tagging over other categorization methods (e.g., taxonomies) in terms of both flexibility and efficiency have been put forward (see, for instance, [9]). The assignment of tags by users occurs based on their own preferences, current information needs, and language, which decreases the required cognitive effort. However, users have been shown to develop their own tag vocabularies (personomies) over time, which may be highly different from those of other users. As a result, the effectiveness of customized tag recommendation and social search can be drastically reduced. For this reason, our system employs a user-based tag model that relies on tensors and tensor calculus to derive correspondences between personal geo-localized tag vocabularies (geo-personomies) and folksonomies. Such a mechanism facilitates the identification of the real meaning and search intention that users might have associated to the assigned tags in order to take into account their preferences and information need in Web information search. Furthermore, the user-based tag model allows a better prediction of which tags the user may want to assign to new resources. Experimental tests performed on 15 real users allowed us to assess the benefits in terms of performance that our approach is able to provide.

The rest of the paper is organized as follows. In the next section, we illustrate the proposed personalized LBS. Then, we describe our approach to strengthen the social character of the recommender system, and present the results of the experimental evaluation. In the last sections, we review some related work and draw our conclusions on what has been done and what is still to be done.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

Workshop SRS'11, March 19, 2011, Hangzhou, China.

Copyright 2011 ACM 978-1-60558-995-4...\$10.00.

## A PERSONALIZED LOCATION-BASED SERVICE

The proposed social recommender system aims at providing recommendation for location-based queries. Several large scale studies show how location-based queries are a relevant part of all the queries submitted through mobile search interfaces [1, 20, 17, 7]. Smart mobile devices and wireless technologies allow the fast growing number of mobile subscribers to query traditional search tools or mapping service applications, with a view to obtain business listings, events or further information related to a specific location. Besides the advantages of obtaining and processing information during everyday activities, this kind of ubiquitous human-computer interaction has the chance of enhancing traditional search activities according to the location of use or further information such as the collection of nearby people or resources [18]. One of the most popular LBS is Google Maps<sup>1</sup>. According to a patent filed on 2007 [14], Google acquires geo-coded information from one or more sources such as commercial data providers, enterprise Web sites, and/or business directories. Even though the criteria that evaluate and weight each business that satisfies the user query have not been published, according to some practitioners<sup>2</sup>, the most important factors on rankings in the Google<sup>3</sup>, Bing<sup>4</sup>, and Yahoo<sup>5</sup> Local search algorithms are

- the overall distance from the city of search;
- the correspondence between the category chosen by the user and the ones associated to the business;
- keyword matching between the query and the business title, description, url, etc.;
- the availability of additional information (e.g., photos or videos) associated to a business.

Analyzing search engine logs, in particular the subset of geographic queries that employ terms in an attempt to restrict results to a particular location, it is possible to recognize several different categories or topics that may benefit of context-aware and personalization techniques [3]. Examples are the following categories:

- tourism/travel: for example, hotel Manchester UK, rent a car Dublin;
- business: for example, Gucci, food, nail salons, furnitures;
- night life and recreation: pizza in Sydney, bike rentals, casinos;
- arts: art galleries, concerts;

that account for a relevant percentage of the whole local search activity.

<sup>1</sup><http://maps.google.com/maps>

<sup>2</sup>[www.davidmihm.com/local-search-ranking-factors.shtml](http://www.davidmihm.com/local-search-ranking-factors.shtml)

<sup>3</sup><http://www.google.com/>

<sup>4</sup><http://www.bing.com/>

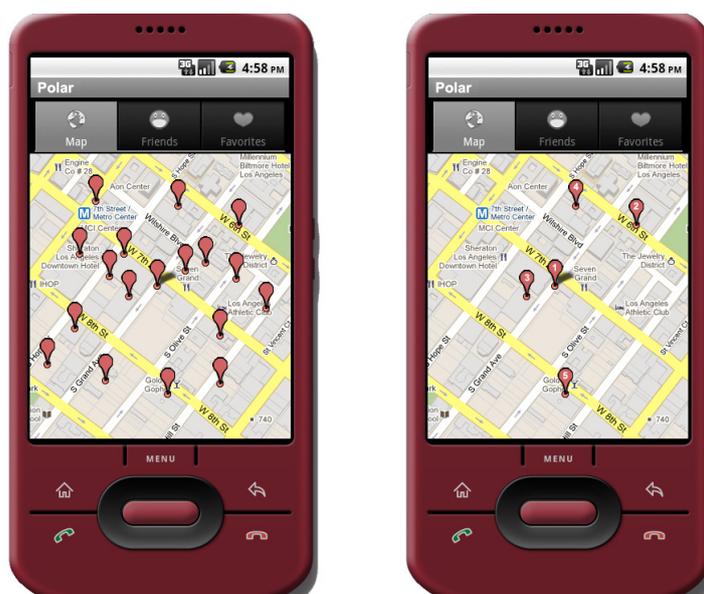
<sup>5</sup><http://www.yahoo.com/>

Context encompasses more than just the user location because further elements in the current situation relevant to an application are also mobile and changing [18]. Of course, it is impossible to exhaustively enumerate all the aspects of the potential situations which are relevant or not, while several of these aspects are even not easy to measure or represent. For example, many authors point out how the presence of other people can be used to characterize the user situation. On holiday, if many people in a group share same interests, it is more likely that particular locations are viewed with more preference than others. Obtaining this information means that all the members of the group want to make their location available to the same system at the same time. Nevertheless, several factors have the chance to heavily influence the ranking of the suggestions recommended. Figure 1(a) shows, for example, how users are not allowed to know if one business is currently open and has a private parking lot in the event they are traveling by car. There are a few attempts to combine the context with preferences and models of the user (see, for instance, [10]). In the example in Figure 1(a), the service suggests several different alternatives close to the user position, but actually the results spread from fast-foods to very expensive restaurants and several different cuisines, for example, Italian, Chinese, Japanese or French. The high density of businesses makes it difficult to pick out that restaurant amongst all the alternatives. As local databases and search engines get richer of geo-coded information with the contribution of Internet users and commercial data providers, this problem becomes prominent. Restricted text input capabilities and small size displays discourage search activities for the acquisition of additional informative elements in order to filter out less interesting businesses. In some cases, the user preferences are even predominant in comparison with the current context. For example, often users are willing to take longer journeys and choose far restaurants if they are on holiday and want to taste the local cuisine, or if they prefer to trust suggestions or reviews from friends and online services. Figure 1(b) shows how the personalized system proposed in this work weights more some restaurants located far from the user current location. The proposed recommender systems addresses all these issues taking into consideration several aspects that is possible to summarize as follow:

- increasing the leverage of location-based services with the vast amount of information from social networking, user review, and local search Web services;
- modeling the user preferences by increasing the recommendation precision of the results;
- defining a richer representation of the context to adapt the results to the current situation, places or, in general, objects relevant to the user current needs.

## The Architecture

A way of improving the mobile users satisfaction during the interaction with LBSs is to adapt the contents and presentation of the service to each individual user and his current context of use. In this way, the user interaction is minimized and users have quick access to the information or service of



(a)

(b)

**Figure 1.** A snapshot of the Google Maps GUI during a search for restaurants on a mobile phone (1(a)), and restaurants suggested by our social recommender system where numbers are associated to the most important results showed on the same GUI (1(b)) (2011 Google - Map data ©2011).

interest. In this work, we focus on LBSs able to interact with users through some kind of application installed on their mobile devices, that is, a software developed with one of the Software development kit provided by the popular brands of mobile devices such as Apple iOS, Android or the open source Symbian OS. This is required to collect additional information for building user contexts (e.g., current location or location history) that traditional mobile browsers are not able to communicate to remote services. An application on the mobile device monitors, discovers and keeps track of these data and transmits them to the remote services along with the submitted query. The general architecture of the proposed system is shown in Figure 2. The user interacts with the application on mobile phone that encodes the current information related to the context, such as location, time and speed, and sends it along with the query to the LBS. The social recommender engine follows different stages. In the first place, a local database is populated with information from popular data sources on the Web. For example, business listings, phone numbers and addresses are extracted from Yellow Pages and traditional LBSs such as Google Maps, Yelp <sup>6</sup>, Zagat <sup>7</sup>, etc. Data extraction algorithms are able to associate metadata to each entry in the form of weighted keywords from a dictionary defined during the learning phase. The extraction basically covers the review posted in social Web sources, such as the above-mentioned LBSs but also forums, blogs and social networking sites, for example, Facebook <sup>8</sup>. The data extraction is able also to automatically extract structured data on Web pages that are related to par-

<sup>6</sup><http://www.yelp.com/>

<sup>7</sup><http://www.zagat.com/>

<sup>8</sup><http://www.facebook.com/>

ticular features of the objects. For example, Yelp shows for each restaurant business information such as price range or wheelchair accessibility. These kind of data is stored in the local db preserving the related semantic sub-category. People sharing their reviews with others are evaluated according to several factors in order to weight the reliability of the retrieved review data. Through adaptation a system can adapt its behavior instead of providing the same results in all the situations. The recommender system monitors the current situation and the user environment, that is, the contextual factors, computes adaptation decisions, and provides the personalized suggestions to mobile devices. Personalized recommendation is performed in two phases. Users are associated to profiles, each one representing their interest in one of the possible macro categories. During the first phase, this profile is matched to the items in the local db to rank the recommended objects according to the user preferences. Afterwards, the current context of use is evaluated to re-rank the objects that are rated more appropriately for the situation. Depending on the current user location, additional information is retrieved by public services, such as weather and traffic report, for augmenting the contextual data. Activity recognition algorithms recognize further elements that are associated to the context. The final rank presents information in such a way that it is relevant to a user context and preferences, for example, by only retrieving Chinese or Italian restaurants within 10 minutes walk of a user location. The references of recommended objects are presented in such a way that their relevance to a user context is expressed as numbers associated to their pinpoints (see Fig. 1(b)).

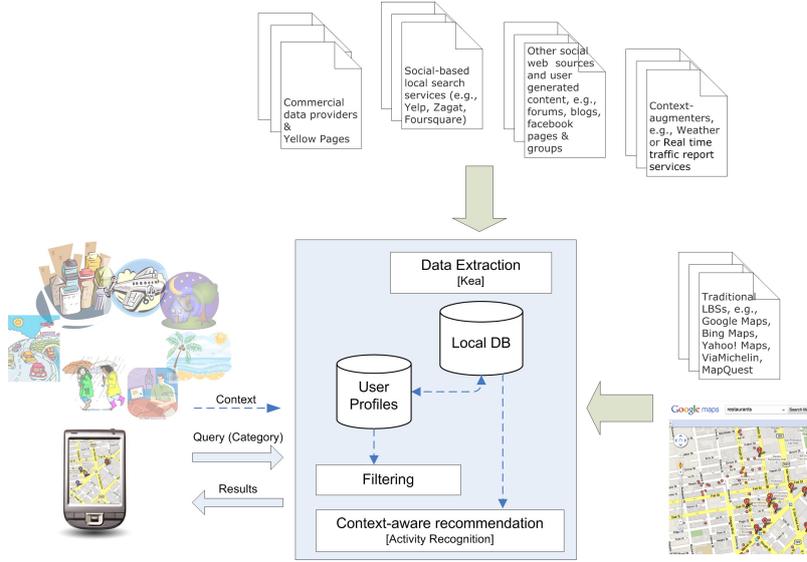


Figure 2. The general architecture of the social recommender system.

### ENHANCING THROUGH FOLKSONOMIES

In order to provide users with personalized recommendations, our system resorts to a user-based tag model that relies on tensors and tensor calculus to derive correspondences between user personomies and folksonomies. This mechanism enables the system to recognize the real meaning and search intention that users may have associated to the assigned tags in order to take into account their preferences and information needs in Web information search. Moreover, the user-based tag model allows a better prediction of which tags the user may want to assign to new resources (see Fig. 3). We can define the problem through the formalism introduced in [5]: a folksonomy can be described as a tuple:

$$F = (I, T, U, Y)$$

where  $I$ ,  $T$ ,  $U$  represent finite sets of items, tags, and users, respectively;  $Y$  represents the following ternary relation:

$$Y \subseteq I \times T \times U$$

(i.e., the item  $i$  has been tagged with the tag  $t$  by the user  $u$  at the time  $k$ ), whose elements are named *tag assignments*. Based on this structure, we have realized and evaluated two different systems: a tag recommender and a tag-based search engine. The tag recommendation problem can be formulated as follows: given an item  $i$  (belonging to the folksonomy) and a user  $u$ , provide  $u$  with a list of tags (belonging to the folksonomy) ranked based on their relevance to both the personomy of user  $u$ , and the position of  $i$  in the folksonomy (see the next pseudocode for details).

#### Algorithm 0.1: RECOMMEND\_TAGS( $user, item$ )

**comment:** Vector of all ones with length equal to the number of users;  
 $users\_vector.length \leftarrow COUNT\_USERS();$   
 $users\_vector \leftarrow [1, 1, 1, 1, 1, \dots, 1];$

**comment:** Co-occurrence matrix as aggregation of the folksonomy tensor;  
 $co\_occurrence\_tag\_item\_matrix \leftarrow$   
 $TENSOR\_PRODUCT(user\_mode, folksonomy,$   
 $users\_vector);$

**comment:** Tensor product;  
 $transposed\_item\_normalized\_co\_occurrence\_tag\_item\_matrix \leftarrow$   
 $TRANPOSE($   
 $ITEM\_NORMALIZE(co\_occurrence\_tag\_item\_matrix));$   
 $translation\_tensor \leftarrow TENSOR\_PRODUCT($   
 $item\_mode, folksonomy,$   
 $transposed\_item\_normalized\_co\_occurrence\_tag\_item\_matrix);$

**comment:** Current user translation tensor;  
 $user\_translation\_matrix \leftarrow$   
 $GET\_USER\_TRANSLATION\_MATRIX($   
 $translation\_tensor,$   
 $user);$

**comment:** Vector with length equal to the number of items;  
 $item\_vector.length \leftarrow COUNT\_ITEMS();$   
 $item\_vector \leftarrow [0, 0, 0, \dots, 1 \text{ at } GET\_INDEX(item), \dots, 0];$

**comment:** Tag distribution for the current item;  
 $community\_tag\_distribution\_by\_item\_vector \leftarrow$   
 $TENSOR\_PRODUCT($   
 $item\_mode,$   
 $transposed\_item\_normalized\_co\_occurrence\_tag\_item\_matrix,$   
 $item\_vector);$

**comment:** Mixture ratio parameter;  
 $alpha \leftarrow [0..1];$

**comment:** Translation matrix for folksonomy tags;  
 $retrieved\_tags\_from\_translation \leftarrow$   
 $NORMALIZE(TENSOR\_PRODUCT($   
 $user\_tag\_mode, user\_translation\_matrix,$   
 $community\_tag\_distribution\_by\_item\_vector));$

**comment:** Set union operation;  
 $result\_tag\_vector \leftarrow$   
 $alpha \cdot retrieved\_tags\_from\_translation +$   
 $(1 - alpha) \cdot community\_tag\_distribution\_by\_item\_vector;$

**return** ( $result\_tag\_vector$ );

The tag-based search problem can be formulated as follows: given a user  $u$  and a query  $q$  in which each term can be interpreted as a tag, provide  $u$  with a list of items (belonging to the folksonomy) ranked based on their relevance to both the personomy of user  $u$ , and the position of terms (tags) of the query in the folksonomy (see the next pseudocode for details).

**Algorithm 0.2:** TAG\_SEARCH( $user, query$ )

```

co_occurrence_tag_item_matrix ←
  GET_CO_OCCURRENCE_TAG_ITEM_MATRIX();
user_translation_matrix ←
  GET_TRANSLATION_MATRIX(user);
alpha ← [0..1];

comment: Query as tag vector;
query_tag_vector ←
  PARSE_QUERY_AS_TAG_VECTOR(query);

comment: Weight normalization;
transposed_tag_normalized_co_occurrence_tag_item_matrix ←
  TRANSPOSE(
    TAG_NORMALIZE(co_occurrence_tag_item_matrix));

comment: Computation of item distribution for query tags;
community_items_distribution_by_tag_vector ←
  TENSOR_PRODUCT(
    tag_mode,
    transposed_tag_normalized_co_occurrence_tag_item_matrix,
    query_tag_vector);

comment: Query translation in the user personomy;
translated_query_tag_vector ←
  TENSOR_PRODUCT(user_tag_mode,
    user_translation_matrix, query_tag_vector);

comment: Tensor product;
retrieved_items_from_translation ← NORMALIZE(
  TENSOR_PRODUCT(community_tag_mode,
    transposed_tag_normalized_co_occurrence_tag_item_matrix,
    translated_query_tag_vector));

comment: Set union operation;
results ← alpha · retrieved_tags_from_translation +
  (1 - alpha) · community_tag_distribution_by_item_vector;

return (results);

```

In [23], tag-based search only relies on the user personomy, that is, it allows only queries whose terms are employed by the user who performs the search. But what does it happen when the user is interested in something else? In our approach we removed this limitation by adding the contribution of search only to the co-occurrence matrix tag-item. This is because we think it is more appropriate (especially during the cold start of the system) to consider that users may be interested not only in what they prefer, but also in what they are not familiar with, according to the system. As in perfect Web 2.0 style, our approach allows users to interact and collaborate with each other by assigning freely chosen tags to resources. The system prompts the user to enter any number of new tags either defined by himself or chosen

among the recommended ones. The grade of recommendation for each tag is expressed as size of the related label (see Fig. 3).

## EMPIRICAL EVALUATION

In order to assess the effectiveness of the proposed approach, we carried out an exploratory field study. The specific goal we pursued was to evaluate the real benefits the user can derive by the personalization of the search when using the system. A major problem that this kind of evaluation poses is the unavoidable subjectivity of the test: the relevance of a POI for a user is a factor impossible to objectively quantify, thus it was necessary rely on real testers for evaluating the performance of the system. Hence, we recruited 15 participants among the college students of Roma Tre University, Italy. In the system evaluation, we chose to restrict the domain of POIs to restaurants, in order to focus the analysis on a category of business most users are familiar with. Furthermore, this domain is rich and diversified, thus enabling us to evaluate the full potential of the system. The experimental evaluation took place in two different stages. In the first stage, it was necessary to proceed with the training of the system and the creation of a user model for each tester. To that end, each of the 15 testers was presented with a selection of 30 restaurants in New York, randomly chosen from the data extraction engine. For each restaurant, the user was required to consider all the received information related to that particular POI and to give a rating from 1 (“I do not like it at all”) to 5 (“I like it very much”), which reproduces the explicit feedback mode during normal system operation. Based on these evaluations, the system built a user model for each tester who participated in the experimental trials. For each restaurant testers were shown all the details found on Yelp, and some tags extracted by the system through analyzing user reviews on the Web site. Any tester was also given the link to the specific Web page of Yelp where to look for further details about the restaurant, such as the photo gallery as well as the full list of reviews, which could be useful if the proposed information were not sufficient. In the second stage, we assessed the effectiveness of the system in proposing POIs in different cities based on the user preferences, which had been included in the user profile in the first stage. Testers were presented with five sets, each consisting of ten restaurants located in five different cities of the United States. For each set, users were asked to rank the restaurants by composing a list according to their preferences, with the most preferred restaurant placed in the first position and the less appreciated in the last one. Los Angeles, San Francisco, Chicago, Seattle, and Boston, were selected as test cities, since all of them are large American metropolises that provide a wide variety of restaurants like New York, so as to provide consistent and reliable test results. Once we received the feedback from testers, the experimental evaluation kept drawing up two lists for each of the five sets: the first in which POIs were ordered only according to the original ranking given by Yelp, the second returned by the system based on user profiles defined for each tester in the first phase of the evaluation. Then, we evaluated how much the feedback proposed by testers was similar to the two aforementioned lists in order to find out if the per-



**Figure 3.** The system allows the user to enter any number of new tags defined by himself or chosen among the recommended ones. The grade of recommendation for each tag is expressed as size of the related label.

sonalization of recommendation through the introduction of user profiles in ranking POIs deliver real benefits. To this aim, we computed the distance  $d(u, y)$  between the list provided by each tester and the one returned by Yelp, and then the distance  $d(u, s)$  between the list provided by each tester and the one ranked by the system based on the user model. To compute the distance between two ordered lists of preferences we resorted to the Kendall  $\tau$  coefficient. If  $P$  is the number of preferences that agree and  $Q$  is the number that disagree, the Kendall's  $\tau$  is defined as [2]:

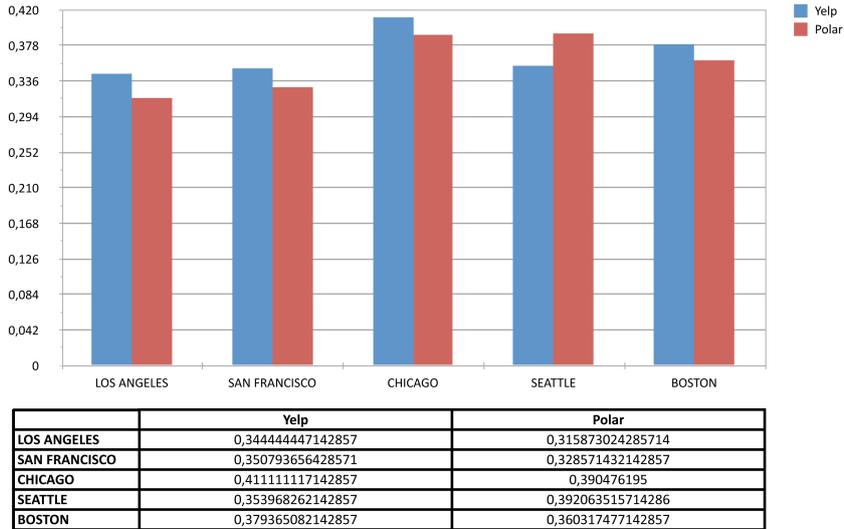
$$\tau = \frac{P - Q}{P + Q}$$

This coefficient varies between 1 (when all preferences agree) and -1 (when they all disagree). Such coefficient is ideally suited to measure which list is closer to that one expected by the user. Preliminary findings are encouraging: based on user profiles, the proposed system provided a personalized list of POIs so as to result in a decrease of the distance from the list expected by the user. The average improvement was found to be 10%. For each city, the graph reported in Figure 4 shows the average distances between the lists returned by Yelp and the ones expected by users, and the average distances between the lists returned by the system (we called *Polar*) and the ones expected by users.

#### RELATED WORK

In this paper we report on a SRS able to identify user preferences and information needs in order to provide useful recommendations concerning possible POIs in the surroundings of the user current location. As far we know, there is no contribution in literature that proposes similar approaches. However, some systems exhibit a few similarities with the

one herein described. In [11] the authors point out that much research has been conducted on the personalization of Web and desktop search, but less work has been devoted to the local search. According to the authors, a single user profile is not effective in local search, so they propose a model to customize the results from local search engines. This model relies on a combination of profile-based modeling and click-through data modeling. The approach described in [11] shares some aspects with our system, but is not able to exploit the wealth of information from social networking and user reviews. In other words, it does not incorporate Web 2.0 participative features, unlike our approach. In [12] the implementation of a LBS on a normal mobile phone with minimal requirements is described, which enables users to exploit public sources from the Internet that can be associated with geographic locations. Although there are some similarities with our approach, the differences are substantial. As in [11], this approach does not take advantage of the information from users willing to collaborate and to share their opinions and experiences online. Furthermore, the system is not able to adapt its output to the user. Another system that shares some similarities with our approach is presented in [13]. The authors propose a map-based recommendation system able to take user preferences into account through a model based on Bayesian networks. The system collects user request and information about the context of use (e.g., location, time or weather) from the mobile device. Then, it leverages the user profile to display the most relevant POIs on the map. The major difference with our system is that it does not extract information from Web sources, nor social networking. Specifically, the system in [13] selects the most relevant POI for the user from among a limited number of POIs. In the system presented herein, we used the algorithm of tag recommendation proposed in [23]. Although tag recommendation is a rather recent topic, in literature there are several contributions. As for non-personalized tag recom-



**Figure 4.** The average distances between the lists returned by the system and the lists expected by users are lower than those between the lists provided by Yelp and the users' ones.

menders, which for a given item suggest the same tags to all users, some interesting works are described in [4, 19]. However, in [16] the authors point out that personalization is important because users tend to select different tags for the same suggestions. In addition, in [15] it has been empirically demonstrated that the main state-of-the-art personalized tag recommenders allow us to achieve better performance than any non-personalized tag recommender. Among the most relevant personalized tag recommenders we mention FolkRank, an adaptation of PageRank, which was introduced in [5]. In [6] one of the early graph-based tag recommender in folksonomies is proposed. The authors claim that the FolkRank method slightly outperforms co-occurrence-based tag recommenders. Recently, some factorization models based on Tucker Decomposition (TD) have been employed in tag recommendation. In [21] Higher Order Singular Value Decomposition is used, which is a TD model optimized for square loss. Even better performance are reported in [15] when resorting to methods of ranking-oriented tensor factorization. The approach outlined in [8] is more similar to the one proposed in [23]. This system adopts a lexicon-based approach in order to weight the tags of user personomies in comparison with community tags. As regards hybrid recommender systems, which usually integrate content-based recommendation with collaborative filtering, one noteworthy approach is presented in [22]. The authors put forward a hybrid system that combines tag-based models and collaborative filtering, thus providing better performance than standard collaborative filtering approaches.

## CONCLUSIONS

In this paper, we have illustrated a social recommender system that takes advantage of the correspondences between user personomies and folksonomies to recognize their preferences and search intentions. Experimental evaluations carried out on 15 real users allowed us to assess the benefits of the proposed approach. We are currently investigating a few areas of future work related to the LBS prototype. The current version of the system represents interests according with the keywords chosen by users for tagging a given resource. Hence, the relevance of a keyword in tag recommendation and tag-based search is proportional to its occurrence in the user vocabulary. Even though this is a popular approach for user profiling in recommender systems, there are scenarios where this additive feedback might negatively affect the accuracy of the results. Some kinds of events have a delimited validity or relevance: for example, a visual artist exhibition or fashion catwalks, or the user interest could drop after a certain period of time. In the first scenario, users will likely tag more than one resource related to a particular event and this occurs in a relatively short period of time. The system starts suggesting resources related to the given event; nevertheless, it is not able to alter the profile of interests after the end of the event. The recommended resources have the chance to include false positives, namely, items that are not interesting for users. Analogously, in the second scenario the system does not adapt the user profile according to the change of interests, thus equally weighting interests distant

in time from each other. This phenomenon is often named *interests flattening*. One possible solution for this problem involves the tensor model previously discussed. In particular, extending the definition of folksonomy with the inclusion of the time factor, we have:

$$F = (I, T, U, K, Y)$$

where  $I, T, U, K$  represent finite sets of items, tags, users, and timestamps, respectively;  $Y$  represents the following quaternary relation:

$$Y \subseteq I \times T \times U \times K$$

(i.e., the item  $i$  has been tagged with the tag  $t$  by the user  $u$  at the time  $k$ ), whose elements are named *tag assignments*. In other terms, the representation of the tensor model is enhanced with the additional time dimension. Tensor calculus allows the system to distinguish tags that are subjected to a short period of validity from tags that concern long-term interests during the tasks of tag recommendation and tag-based search.

#### ACKNOWLEDGMENTS

We would like to thank Andrea Flamini and Samuele Millevolte for their support to this work.

#### REFERENCES

1. S. Asadi, C.-Y. Chang, X. Zhou, and J. Diederich. Searching the world wide web for local services and facilities: A review on the patterns of location-based queries. In P. Brusilovsky, A. Kobsa, and W. Nejdl, editors, *6th International Conference, WAIM 2005*, volume 3739 of *Lecture Notes in Computer Science*, pages 91–101, Berlin, Heidelberg, and New York, 2005. Springer Berlin, Heidelberg.
2. B. Croft, D. Metzler, and T. Strohman. *Search Engines: Information Retrieval in Practice*. Addison-Wesley Publishing Company, USA, 1st edition, 2009.
3. Q. Gan, J. Attenberg, A. Markowetz, and T. Suel. Analysis of geographic queries in a search engine log. In *LOCWEB '08: Proceedings of the first international workshop on Location and the web*, pages 49–56, New York, NY, USA, 2008. ACM.
4. P. Heymann, D. Ramage, and H. Garcia-Molina. Social tag prediction. In *SIGIR '08: Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 531–538, New York, NY, USA, 2008. ACM.
5. A. Hotho, R. Jäschke, C. Schmitz, and G. Stumme. Information retrieval in folksonomies: Search and ranking. In *The Semantic Web: Research and Applications*, volume 4011 of *Lecture Notes in Computer Science*, pages 411–426, Heidelberg, June 2006. Springer.
6. R. Jäschke, L. B. Marinho, A. Hotho, L. Schmidt-Thieme, and G. Stumme. Tag recommendations in folksonomies. In J. N. Kok, J. Koronacki, R. L. de Mntaras, S. Matwin, D. Mladenic, and A. Skowron, editors, *Knowledge Discovery in Databases: PKDD 2007, 11th European Conference on Principles and Practice of Knowledge Discovery in Databases, Warsaw, Poland, September 17-21, 2007, Proceedings*, volume 4702 of *Lecture Notes in Computer Science*, pages 506–514. Springer, 2007.
7. M. Kamvar and S. Baluja. A large scale study of wireless search behavior: Google mobile search. In *CHI '06: Proceedings of the SIGCHI conference on Human Factors in computing systems*, pages 701–709, New York, NY, USA, 2006. ACM.
8. M. Lipczak. Tag Recommendation for Folksonomies Oriented towards Individual Users. In *Proceedings of ECML PKDD Discovery Challenge (RSDC08)*, pages 84–95, 2008.
9. P. Mika. Ontologies are us: A unified model of social networks and semantics. In Y. Gil, E. Motta, V. R. Benjamins, and M. A. Musen, editors, *The Semantic Web - ISWC 2005, Proceedings of the 4th International Semantic Web Conference, ISWC 2005, Galway, Ireland, November 6-10*, volume 3729 of *Lecture Notes in Computer Science*, pages 522–536. Springer, 2005.
10. M. F. Mokbel and J. J. Levandoski. Toward context and preference-aware location-based services. In *MobiDE '09: Proceedings of the Eighth ACM International Workshop on Data Engineering for Wireless and Mobile Access*, pages 25–32, New York, NY, USA, 2009. ACM.
11. P. O'Brien, X. Luo, T. Abou-Assaleh, W. Gao, and S. Li. Personalization of content ranking in the context of local search. In *WI-IAT '09: Proceedings of the 2009 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology*, pages 532–539, Washington, DC, USA, 2009. IEEE Computer Society.
12. M. Pannevis and M. Marx. Using web-sources for location based systems on mobile phones. In *Workshop on Mobile Information Retrieval (MobIR'08) 24 July 2008, Singapore*, 2008.
13. M.-H. Park, J.-H. Hong, and S.-B. Cho. Location-based recommendation system using bayesian users preference model in mobile devices. In J. Indulska, J. Ma, L. Yang, T. Ungerer, and J. Cao, editors, *Ubiquitous Intelligence and Computing*, volume 4611 of *Lecture Notes in Computer Science*, pages 1130–1139. Springer Berlin / Heidelberg, 2007.
14. E. Pasztor and D. Egnor. Generating structured information, September 2006.
15. S. Rendle, L. B. Marinho, A. Nanopoulos, and L. Schmidt-Thieme. Learning optimal ranking with tensor factorization for tag recommendation. In *KDD '09: Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 727–736, New York, NY, USA, 2009. ACM.

16. S. Rendle and L. Schmidt-Thieme. Pairwise interaction tensor factorization for personalized tag recommendation. In *WSDM 2010: Proceedings of the 2010 ACM International Conference on Web Search and Data Mining*. ACM, 2010.
17. M. Sanderson and J. Kohler. Analyzing geographic queries. In *In Proc. of the ACM SIGIR Workshop on Geographic Information Retrieval*, pages 1–2, 2004.
18. B. N. Schilit, N. Adams, and R. Want. Context-aware computing applications. In *In Proceedings of the Workshop on Mobile Computing Systems and Applications*, pages 85–90. IEEE Computer Society, 1994.
19. Y. Song, L. Zhang, and C. L. Giles. A sparse gaussian processes classification framework for fast tag suggestions. In *CIKM '08: Proceeding of the 17th ACM conference on Information and knowledge management*, pages 93–102, New York, NY, USA, 2008. ACM.
20. A. Spink and B. J. Jansen. A study of web search trends, 2004.
21. P. Symeonidis, A. Nanopoulos, and Y. Manolopoulos. Tag recommendations based on tensor dimensionality reduction. In *RecSys '08: Proceedings of the 2008 ACM conference on Recommender systems*, pages 43–50, New York, NY, USA, 2008. ACM.
22. R. Wetzker, W. Umbrath, and A. Said. A hybrid approach to item recommendation in folksonomies. In *ESAIR '09: Proceedings of the WSDM '09 Workshop on Exploiting Semantic Annotations in Information Retrieval*, pages 25–29, New York, NY, USA, 2009. ACM.
23. R. Wetzker, C. Zimmermann, C. Bauckhage, and S. Albayrak. I tag, you tag: Translating tags for advanced user models. In *WSDM '10: Proceedings of the Third ACM International Conference on Web Search and Data Mining*, pages 71–80, New York, NY, USA, 2010. ACM.