GuideMe! The World of Sights in Your Pocket

Sergej Zerr, Kerstin Bischoff, Sergey Chernov

L3S Research Center, Leibniz Universität Hannover
Appelstrasse 9a, 30167 Hannover, Germany
{zerr, bischoff, chernov}@L3S.de

***Abstract*—Web 2.0 applications are a rich source of multimedia resources, that describe sights, events, whether conditions, traffic situations and other relevant objects along the user's route. Compared to static sight descriptions, Web 2.0 resources can provide up-to-date visual information, which has been found important or interesting by the other users. Some algorithms have been suggested recently for the landmark finding problem from photos. Still, if users want related videos or background information about a particular place of interest it is necessary to contact different social platforms or general search engines. In this paper we present GuideMe! - a mobile application that automatically identifies landmark tags from Flickr groups and gathers relevant sightseeing resources from various Web 2.0 social platforms.**

I. INTRODUCTION

When planning a trip people often use various social services to search for photos and videos of representative landmarks near their travel destination. This is possible due to the user generated tags, which are especially valuable for recommending, searching and browsing multimedia resources shared by other users. However, users often do not know the main sights in the city of interest in advance, such that they cannot search for them directly. In such situations the system should automatically identify the landmarks and find corresponding resources (e.g. photos or videos) for a given city name. We call the task of automatic selection of representative landmarks for a given location the *landmark finding* problem. Recent research on landmark finding was focused on creating landmark photo summaries [1], [2], [3], [4], [5]. Some working prototypes for landmark finding are available online, for example, in the World Explorer application[1] a user can enter a location name and browse through the landmark related tags and the corresponding photos. However, the photos alone do not provide a complete overview of the location. A tourist might need some background information regarding the landmarks, video guides, recommendations from fellow travellers, etc. As this information is spread across different social platforms, one has to search for landmarks on each service separately. For example, a user plans to visit Hanover to attend ICDE 2011 and searches for resources tagged with "Hanover". She wants to see a concise and representative view of the city with a few photos and videos related to the famous landmarks such as Royal Gardens of Herrenhausen, the new Town Hall or the Church. She might also need a couple of web pages with a historical overview and travel tips. Such informational need could be satisfied with results from different Web 2.0 services, including photos retrieved from Flickr[2], videos from Youtube[3] and bookmarks from Del.icio.us[4], like in Fig. 1.

Fig. 1: Landmark resources for "Hanover".

This scenario requires several searches on different social platforms, which might be particularly inconvenient if the user is already travelling and looks for the sights of interest using her mobile phone.

In this work we demonstrate a mobile application GuideMe! that retrieves landmark-related resources from various Web 2.0 platforms. First, GuideMe! extracts representative tags from landmark pictures according to the algorithm which we earlier described in [5]. These landmark-classified tags are used to query a number of social sources like Flickr, Youtube, Delicious, Slideshare, etc. Finally, the retrieved results of the federated search are fused and ranked according to their relevance, popularity, ratings and the number of comments. Contributions of this demonstration include: (i) an algorithm for efficient extraction of landmarks from Web 2.0 sources for a given location; (ii) a system for federated search of Web 2.0 resources related to these landmarks; (iii) a ranking strategy to

[1]http://tagmaps.research.yahoo.com/worldexplorer.php

[2]www.flickr.com

[3]www.youtube.com

[4]www.delicious.com

978-1-4244-8960-2/11/$26.00

provide users with a representative overview of the up-to-date sightseeing information.

We present the system architecture in Section II, describe our method for landmark finding/tag extraction and the dataset used in Section III. In Section IV we present an overview of the demonstration. Section V provides a conclusion by summarizing our main contribution and addressing open questions.

II. System Architecture

This section describes the three main components of the GuideMe! system architecture, illustrated on Fig. 2: the mobile client, the landmark seeker web service and InterWeb - a mash up network which integrates a number of Web 2.0 tools like Flickr, YouTube, Slideshare, etc.

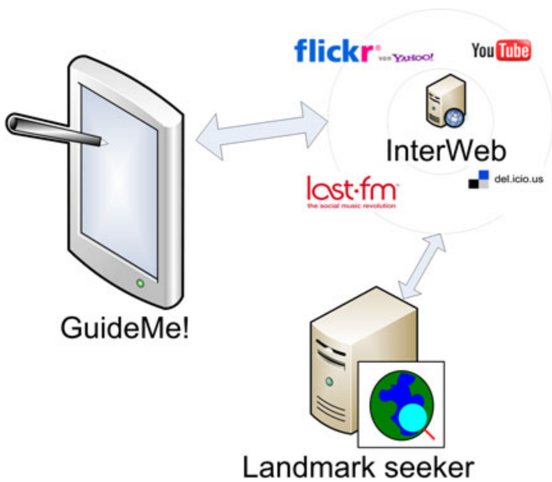

Fig. 2: GuideMe! System architecture

A. Mobile Client

The GuideMe! application can be installed on any mobile device with Android[5] operating system. The graphical user interface consists of the explorer page and the settings page, see Fig. 3. The explorer page contains a search field and a list of results displayed after a successful query execution. At the settings page the user can define how many resources should be displayed in the result list and select their preferred data types. The available data types are:

- Videos (from YouTube, Vimeo and Ipernity)
- Pictures (from Flickr and Ipernity)
- Presentations (from Slideshare)
- Bookmarks (Delicious)

Each result can be previewed in its original context using the integrated browser. In order to interact with Landmark Seeker Service and other application components, GuideMe! uses an internet connection over WLAN or GSM, dependent on their availability.

[5]http://en.wikipedia.org/wiki/Android

Fig. 3: GuideMe! Graphical user interface

B. Landmark Seeker Webservice

Landmark Seeker is a SOAP[6] based web service that retrieves landmarks for a given location using the algorythm described in section III. The web service interface consists of the method *getTopKcityTags* with the mandatory (String) parameter *cityName*.

The response of the web service is an XML-formatted list of sightseeing labels corresponding to the given location name. For example the query *hanover* could return the following list: "herrenhausen", "nordlb", "rathaus", "cityhall", and "marktkirche". The web service implementation is Java based (JDK 6) by using the Apache CXF[7] Framework.

C. InterWeb

InterWeb is a web service which integrates a number of different Web 2.0 tools like YouTube, Flickr, Ipernity, Slideshare, and Delicious. Most of the Web 2.0 applications and their orchestrations focus on finding resources related to the users informational needs. Portals like iGoogle and Netvibes also help to locate information distributed across different information sources. However, such portals typically do not provide any facilities for the actual linking of information obtained from them.

InterWeb web application provides a rich set of functions and a seamless overview of the entire set of distributed Web 2.0 resources. In this manner InterWeb serves as a Meta-Web 2.0 service. It provides a uniform interface for basic functionalities such as search and sharing of resources.

III. Landmark Data Extraction

To provide users with a set of landmark resources we first need to identify the tags associated with landmarks. Our

[6]http://www.w3.org/TR/soap/
[7]http://cxf.apache.org/

method for extracting landmark information from Web 2.0 exploits tags and social groups from *Flickr*. The details of the landmark tags extraction can be found in [5]. Here we provide a brief summary of our approach. One important difference to the original paper is the change of classifier. While in [5] we used the SVM-Light package, here we chose the *NaiveBayesMultinomial* classifier from WEKA[6] as it builds models faster and delivers comparable classification results.

A. Step 1: Classification of Landmark Photos

In the beginning we select a set of photos related to a particular city. Here we rely on a simple heuristic of having the city name as a tag associated with a photo. From the set of pictures containing a city tag, we want to select photos representing landmarks. For training the classifier we pick photos from existing landmark-related Flickr groups. The idea is that some of the *Flickr* groups like "Landmarks around the world" can serve as positive examples, while arbitrary general groups, like "Birds" or "Airplanes" represent negative examples. For the current demonstration we used about 50.000 photos from Flickr groups about landmarks and 100.000 photos from groups about other topics. A classifier assigns each photo to either the "landmark" or the "non-landmark" category given the input feature vectors. Each tag gets a weight equal to its normalized tag frequency $TF_r(t)$. In the following definitions U represents the set of users, T stands for the set of tags, and R is the set of resources. Let T_r denote the set of tags assigned to a resource r. Since in Flickr each tag can appear only once per image the normalized tag frequency $TF_r(t)$ of a tag t in a resource r is defined as follows:

$$TF_r(t) = \frac{1}{|T_r|} \tag{1}$$

In an experimental evaluation we achieved 98.92% accuracy on training data (0.999 AUC/ROC) and 97.37% accuracy (0.996 AUC/ROC) when applying 10 fold cross-validation.

B. Step 2: Ranking of Landmark Tags

Once we have selected a set of city photos and filtered only landmark-related ones, we want to see those tags representing landmarks specific to a particular city. We would like to give low scores to common tags and high scores to city-specific tags. The assumption is that representative landmark tags appear in landmark photos, but are not very common among the whole collection of images. Let R_t denote the set of resources which contain tag t and U_t be a set of users who used tag t. Similar to the notion of Inverse Document Frequency in Information Retrieval we define Inverse Resource Frequency and Inverse User Frequency as follows:

$$IRF(t) = log(\frac{|R|}{|R_t|}) \tag{2}$$

$$IUF(t) = log(\frac{|U|}{|U_t|}) \tag{3}$$

We compute $IRF(t)$ (Eq. 2) and $IUF(t)$ to penalize popular tags. If a tag is frequently used to tag photos in the dataset, it has a low $IRF(t)$ value and vice versa. Similarly, if a tag is globally very common amongst users, it must be scored low. This is achieved by computing $IUF(t)$ (Eq. 3).

After defining global scoring factors, we come to local measures computed on that part of the collection with landmark photos only. When considering the dataset containing only pictures associated to a particular city and classified as landmarks, our assumption is that common tags should be scored high. Let us represent the set of landmark-related photos selected for a city as R_c and the corresponding tag set as T_c. If a tag is common among the photos for a particular city as well as among many users, probably this tag represents some feature of the city, e.g. some museum, or an old and famous building. Let $U_c(t)$ be a set of users using a tag t for the landmark photos for a city c. We compute the normalized *City Tag Frequency*, CTF, using (Eq.4):

$$CTF(t) = \frac{|U_c(t)|}{MAX(|U_c(t')|)} \tag{4}$$

We combine all the above mentioned factors that affect the ranking of the tags and compute a representativeness score for each tag t occurring along with the resources classified as landmarks of a city c. Here, the final scoring function as proposed in [5] has been adapted to better fit the new, though, similar dataset and demo scenario, e.g. with respect to time constraints (i.e. the number of pictures queried for from Flickr). The representativeness score of each tag for a city c is computed as follows:

$$SCORE(t) = IRF \cdot IUF \cdot CTF \tag{5}$$

Therefore, each city is assigned a ranking of landmark related tags and this information is stored in a database. Both Step 1 and Step 2 are performed offline and do not affect system performance.

C. Step 3: Collecting Landmark Resources

When calling the web service, first a database with previously learned city tags is queried for the given city name. If there are city tags – already learned according to the steps described above – in the database, the top-k city tags will be returned in descending order. If no city tags have been learned for the city yet, a search request with the name will be sent to the Flickr API. This request returns all relevant photos that have been tagged with the given city name. Then the classification and tag ranking procedures presented above are applied on the returned set of photos. The top 20 city tags will be saved to the database and the top 5 of them will be send to the client. Employing this method [5] obtained 12% improvement in precision over the World Explorer system.

D. Step 4: Ranking Landmark Resources

Using the set of landmark tags returned by the seeker service, InterWeb combines them into a query and executes it on available Web 2.0 services. In order to provide a representative and diverse overview of sights in Hanover on a small display, the results have to be ordered before presenting them to the user. The list of sightseeing results returned by InterWeb is first divided into sub-lists, one for each landmark. We sort each of the lists by relevance to the corresponding landmark and the top-ranked result from each list is displayed to a user. If some result is relevant to several landmarks it is copied to each corresponding list, see Fig. 4.

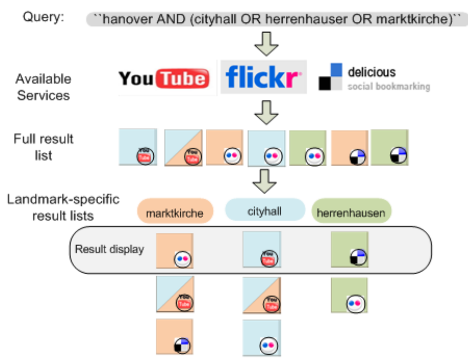

Fig. 4: Collecting and Ranking Landmark Resources

The relevance depends both on the content of the resource and its popularity at the source service. The popularity can be derived from the number of views, comments and ratings of the resource. Thus the relevance can be computed as a weighted sum of these factors. Let us define $Sim(r,t)$ as the textual similarity between resource r's description and tag t, V_r as the number of views of resource r, C_r as the number of comments assigned to r, R_r as a rating assigned by users, and P_r as the position of r in results ranking returned by a particular social service. We rank all resources according to their relevance to a landmark using the Equation 6:

$$RELEVANCE(r) = \alpha \cdot Sim(r,t) + \beta \cdot V_r + \gamma \cdot C_r + \\ + \delta \cdot R_r + \epsilon \cdot P_r, \quad (6)$$

where α, β, γ, δ and ϵ are coefficients used for system tuning.

IV. Demostration Overview

In this demonstration we will primarily show how GuideMe! works and retrieves sightseeing resources of different data types. We will demonstrate the resource discovery process using a mobile device. Additionally we will explain the complete query process including landmark extraction and resource aggregation at InterWeb. First the user selects the preferred data types such as images, videos, or presentations at the options page and saves these settings. She types the search terms into the text field at the exploration page and starts the query execution. Results are shown as a ranked list from which the user selects a resource of interest and views it using the browser integrated in the mobile device. The demonstration package can be downloaded from our page[8] and can be installed on any mobile device with android operating system or an android emulator available for PC.

[8]http://l3s.de/∼zerr/guideme/

V. Conclusions and Future Work

Web 2.0 resources and meta data are very valuable for answering the diverse and complex information needs users have. In this demonstration we focus on a landmark finding scenario. We identify and extract landmark information from multiple social platforms and compile a representative summary for a given city. We present a mobile search interface which retrieves landmark resources from social sites like Flickr, YouTube, Delicious, etc. and presents the user with the overview of the available resources. Our contributions include an algorithm for efficient extraction of landmarks from Web 2.0 sources, a system for federated search of Web 2.0 resources related to these landmarks and a ranking strategy to provide a user with a representative and diverse overview for sightseeing.

One of the open questions is what is the best type of resource for a particular landmark. Some static objects like buildings or paintings look good on photos, while objects like church bells call for a video representation. In our future work we would like to explore how well different types of resources are suited to visualize specific types of sights.

Acknowledgments

This work was partially supported by the GLOCAL project funded by the European Commission under the 7th Framework Programme (Contract No. 248984) as well as by the FP7 EU Project Sync3 (contract no. 231854) and NTH School for IT Ecosystems. NTH (Niedersächsische Technische Hochschule) is a joint university consisting of Technische Universität Braunschweig, Technische Universität Clausthal, and Leibniz Universität Hannover. We would like to thank Nicole Ullmann and Philipp Kemkes for their contributions in implementing this demo application.

References

[1] S. Ahern, M. Naaman, R. Nair, and J. H.-I. Yang, "World explorer: visualizing aggregate data from unstructured text in geo-referenced collections," in *JCDL '07: Proc. of the 7th ACM/IEEE joint conference on Digital libraries*. ACM, 2007, pp. 1–10.

[2] T. Rattenbury, N. Good, and M. Naaman, "Towards automatic extraction of event and place semantics from flickr tags," in *SIGIR '07: Proc. of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 2007, pp. 103–110.

[3] A. Jaffe, M. Naaman, T. Tassa, and M. Davis, "Generating summaries and visualization for large collections of geo-referenced photographs," in *MIR '06: Proc. of the 8th ACM international workshop on Multimedia information retrieval*. ACM, 2006, pp. 89–98.

[4] L. Kennedy and M. Naaman, "Generating diverse and representative image search results for landmarks," in *WWW '08: Proc. of the 17th International World Wide Web Conference*, 2008.

[5] R. Abbasi, S. Chernov, W. Nejdl, R. Paiu, and S. Staab, "Exploiting flickr tags and groups for finding landmark photos," in *ECIR 2009: Proc. Advances in Information Retrieval, 31th European Conference on IR Resea*, 2009, pp. 654–661.

[6] I. H. Witten and E. Frank, *Data Mining: Practical Machine Learning Tools and Techniques*, 2nd ed. Morgan Kaufmann, 2005.