Project no. 34721

TAGora

Semiotic Dynamics in Online Social Communities

http://www.tagora-project.eu

Sixth Framework Programme (FP6)

Future and Emerging Technologies of the Information Society Technologies (IST-FET Priority)

D2.2 First version of Tag-based navigation systems for images and music

---

Period covered: from 01/06/2006 to 31/05/2007
Start date of project: June 1st, 2006
Due date of deliverable: May 31st, 2007
Distribution: Public

Date of preparation: 31/05/2007
Duration: 36 months
Actual submission date: May 31st, 2007
Status: Final

Project coordinator: Vittorio Loreto
Project coordinator organisation name: “Sapienza” Università di Roma
Lead contractor for this deliverable: Sony Computer Science Laboratory

Executive Summary

Sony CSL implemented a tagging system that handles two types of media: images and music. It is designed as an open platform to experiment with new analysis and visualization techniques. In particular, we are evaluating the combination of tags and data analysis to improve the tagging systems. In addition, the system is used to gather data, both tagging information and the browsing history of visitors. The development of this new platform gives us a good understanding of the internals of tagging systems and the Web-based technologies to make them accessible on-line.

Contents

Chapter 1

Ikoru - A Testbed for Collaborative Tagging and Content-Based Analysis

1.1 Motivation

The Ikoru system, developed at Sony CSL, is primarily used to experiment with collaborative tagging and content-based analysis. The current web interface, viewable at `http://www.ikoru.net`, handles both image and music files. Its interface resembles those of existing tagging sites. However, some of the features of Ikoru make it unique:

Content-based tools: The initial motivation of the Ikoru project was to explore the combination of content-based analysis with tagging.

Data Gathering: Ikoru is used as a platform to gather data and explore how the analysis of this data can improve tagging systems. Not only do we store tagging information but also the history of the visitors (visited pages, performed searches). This data may improve analysis techniques that currently only take tagging information in account.

Extendible research platform: Ikoru aims to be an open platform that can be extended with new analysis and visualization tools. As such, researchers can integrate the analysis results in the web site and evaluate the impact of the these tools on the behavior of the users.

Multimedia: Ikoru was initially developed for images but has been extended to handle music files. Future versions will also include video files and text. This multimedia aspect is not incidental. We want to study the tagging behavior for various media types and we seek to use the semantic layer created by the tags as a support for new creative tools.

Small, reusable server: On the technological side, we have designed the Ikoru server as a small, efficient, and stand-alone web component that can be easily reused and integrated in third-party projects.

1.2 System overview

Sony CSL developed a collaborative tagging system, called Ikoru. It is available for public use at `http://www.ikoru.net`. Anyone can register and then upload, tag, and browse photos and music files. A content-based search tool is integrated in the web site and can be used to search for images based on visual similarity (see also Task 4.2).

The web site consists of two components: a server and a web interface. The web interface was implemented using the AJAX paradigm and exchanges XML data with the server using the Ikoru SOAP interface. The web site is one possible instantiation of a client interface. Using the client libraries, it is straightforward to imagine new interfaces.

The server software exports a simple Application Programming Interface (API) that uses the *con-*

text object as a central pillar (see Section 1.2.2). The scripting interface allows external developers to provide analysis tools that are executed on the server.

1.2.1 Web Interface

Through Ikoru's web interface, visitors can browse the public resources (photos, music, a.o.) of the users. The navigation links on the Web site are defined by the relations between tags, people, and resources. Every web page on the site is organized around a *context*. A page displays the photos, tags, and people that are included in the context. The photos are shown as icons. When a visitor clicks on an icon, a larger size version of the photo is shown, including the details of the photo (title, capture date, etc.). By selecting a tag link, only the photos in the current context that are annotated with the selected tag are shown.

Once registered, a person has a personal home page that becomes her default home page for the Ikoru web site. This page lists the icons of most recently uploaded photos and their associated tags. While navigating her photos and music, she can at any time select one or more resources and add new tags to the selection by typing a space-separated list of words in a dedicated text field. Not only can she tag her own photos but also those uploaded by other people. When she tags a photo from someone else's collection it will become accessible in her own collection.

1.2.2 Context Explained

The notion of *context* is central in the structure of the Web site and the low-level API. A context is identified by a unique string that is similar to the format of a file path: a sequence of names separated by a slashes ('/'). The first element of the sequence defines the owner of the context, i.e. the name of the user. The plus sign ('+') can be used instead of a user name to indicate that the context covers all users known to the system. The second element defines the media type. It is a literal that should be equal to "photo" or "music". Following the media type, a list of zero or more tags can be given. It is possible to give the identifier of a resource as the last element of the context ID. In that case, the context contains only a single element, namely the specified resource. More formally, a context identifier has the following form:

$$ContextID : /\{PersonID|+\}/\{\text{photo}|\text{music}|...|+\}[/Tag]^*[/ResourceID]$$

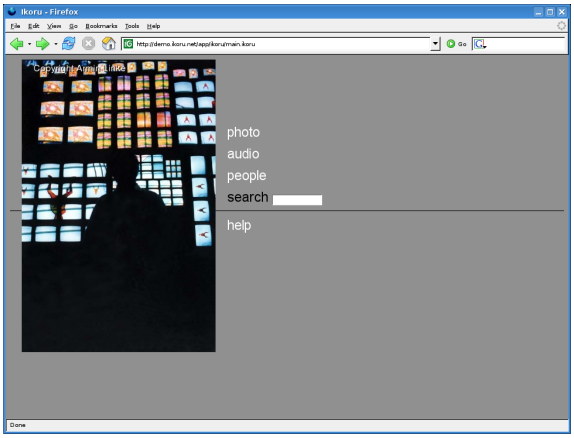
Some examples of valid context identifiers are:

- $/hanappe/photo$: all the photos of user "hanappe".
- $/+/photo/tagora/2007$: the photos of all users tagged with "tagora" **and** "2007".
- $/hanappe/+/1331ba371dd3$: the resource of user "hanappe" with the unique id "1331ba371dd3".

There is a direct mapping between the context identifier and the URL of the Web page to view the context. For example, the URL `http://www.ikoru.net/ikoru/hanappe/photo` will display all photos of user "hanappe".

1.2.3 Similarity-Based Search Tool for Images

Every web page associated with a photo context incorporates a similarity-based search tool. By drag-and-dropping a photo icon on the search tool, the photos of the context will be sorted according to the visual similarity with the example image. Up to three positive and negative example



Main page

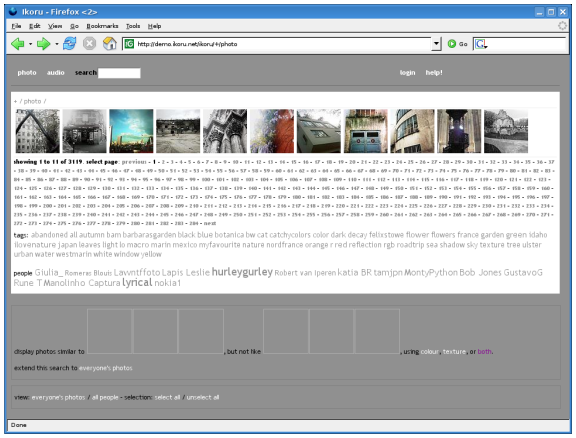


Photo context

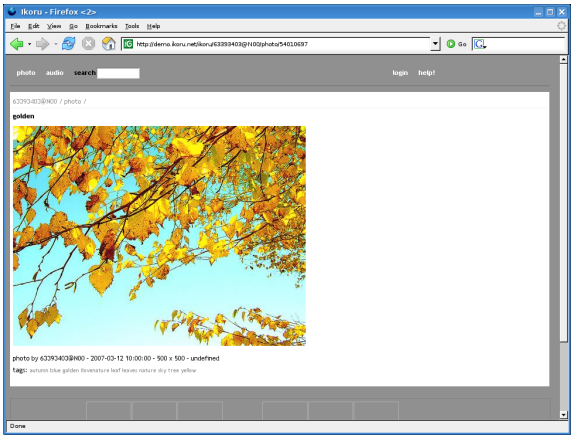


Photo details

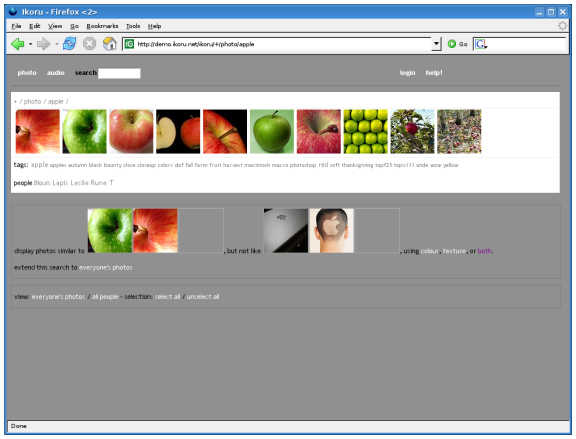
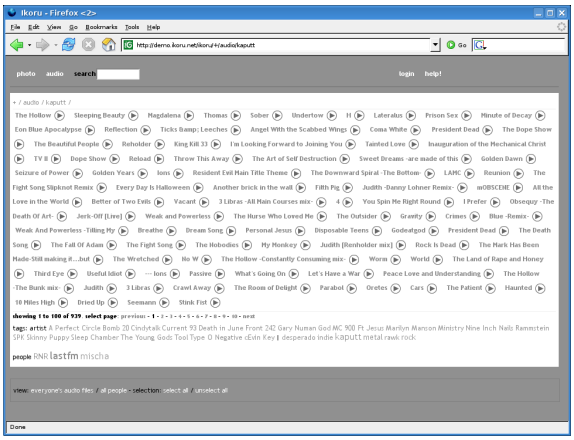
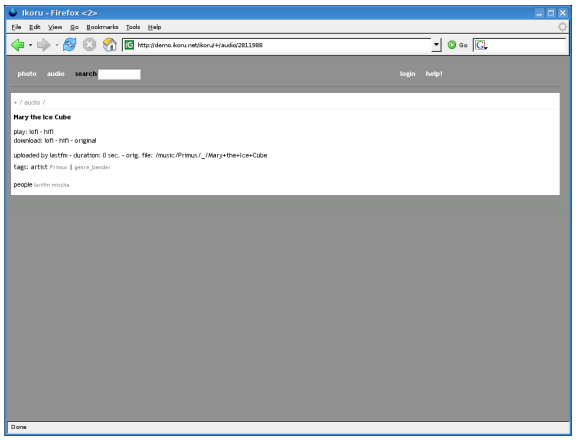


Photo context with classifier



Audio context



Audio details

Figure 1.1: Screenshots taken from http://demo.ikoru.net .

images can be given. The similarity measure can be based on color, texture, or both. To enable the similarity search, the Ikoru server computes two features vectors when a photo is uploaded: one for color and one for texture (Aurnhammer et al., 2006a,b).

1.2.4 Meta-Data for Music

The Ikoru system was initially developed for images but has been extended to handle music files as well. Although Ikoru is targeted at user-made contents, most music tagging sites (e.g. Last.fm) focus on music produced for commercial purposes. The latter already has a fair amount of associated meta-data, such as the artist name or the album title. The challenge was then how Ikoru could maintain a simple tagging concept yet include the existing meta-data. Instead of adopting the MusicBrainz database schema (a often used schema to store music information) we choose to add this information using "machine tags". Machine tags were introduced to add, for example, geographical information to photos. Machine tags have the following syntax: $namespace : predicate = value$. We subsequently adapted the existing meta-data to this format. For example, "music:artist=the_beatles" indicates that the music file is a recording of "The Beatles". Similarly, "music:album=the_white_album" specifies that the music files was is part from "The White Album". The Ikoru server does not make a distinction between "normal" tags and machine tags. Ikoru clients, such as the Ikoru Web site, detect machine tags and display the information appropriately.

1.2.5 Implementation Details

The Ikoru server was developed as a stand-alone server written in C++. We aimed to make the server compliant with Web standards. It is developed and maintained on Linux but is easily portable to Windows and UNIX variants. It consists of about 40000 lines of code. It handles the following tasks:

- manage network connections and thread pool,
- reply to HTTP and SOAP requests,
- manage persistent sessions (using HTTP cookies),
- assure the security (login, validate user input, permissions),
- store/retrieve tagging and user data in a persistent storage,
- receive the uploaded files (testing file type, conversion to Web formats, feature extraction),
- search data based on the text and content (for images),
- generate pages dynamically using key/value substitutions, and
- run the analysis scripts

The choice of writing a complete Web server was motivated by several goals. We mainly wanted to make it easy to deploy Ikoru on a wide range of platforms. In particular, we wanted to keep the option that users install Ikoru for personal use on their computers to manage their media files. This, in the long run, may allow Ikoru to be deployed in peer-to-peer systems. We wanted to retain the possibility to use Ikoru on embedded devices, such as Network Attached Storage (NAS) devices, game consoles, or even mobile devices. For this reason, the amount of resources used by Ikoru had to be controllable. Finally, we wanted to facilitate the integration of the system in larger projects. As a result, Ikoru was developed with a self-contained "Web component", based on SOAP/HTTP and optimized for collaborative tagging.

Ikoru publishes the API of its services on the Web using the Web Service Description Language (WSDL). Client libraries can use the WSDL description to build interface classes that allow third

party applications to communicate with Ikoru. Client libraries for JavaScript, Java, and PHP are available, in addition to a partially implemented C++ client library.

At its core, the Ikoru server mainly processes tag assignments (defined in (Hotho et al., 2006)). In our implementation, every tag assignment has also an associated resource type (photo, audio, . . .) and a resource index. People can use this index to manually change the order of the resources associated with a tag.

Most queries to the Ikoru server start with obtaining all the tag assignments for the requested context. It then checks the access permissions (public/private) associated with the resources. In the next step, the assignments are filtered using the active content-based classifier, if there is one. Depending on whether the client requested the list of people, tags, or resources associated with the context, only the distinct tag assignments for that data field are retained. Next, the tag assignments are sorted according to a client-defined criterion (post date, tag size, alphabetic order, . . .). Finally, the client can request to obtain a specific interval from this ordered set[1], such as the first ten tag assignments. This sorting and slicing can be applied several times. A client can request, for example, to pick the 50 biggest tags (i.e. with the most tag assignments) and then sort this list alphabetically.

1.2.6 Analysis Interface

Ikoru has built in support for new analyze and visualization techniques for tagging, possibly in combination with the browsing history of a user or content-based analysis. To this extent, Ikoru has a server-side scripting interface that allows third parties to execute analysis scripts on the server in a similar way to the Common Gateway Interface (CGI). The advantages of the open platform are that it speeds up the testing of a hypothesis, that the analysis can be applied to user-defined contexts, and that the results can be easily shared with others. Finally, the analysis can be integrated in the web interface to study its impact on the behavior of the users.

For example, Figure 1.2 shows the visualization of a Principal Component Analysis of the color feature for a selected context. This is part of work in progress that takes a look at the relation between principal components of the image features and the co-occurring tags in a given context. Both images can be viewed live at `http://demo.ikoru.net/analysis/pca/feature-colour//63393403@N00/photo/autumn` and `http://demo.ikoru.net/analysis/pca/tag-colour//63393403@N00/photo/autumn`.

1.3 Future Work

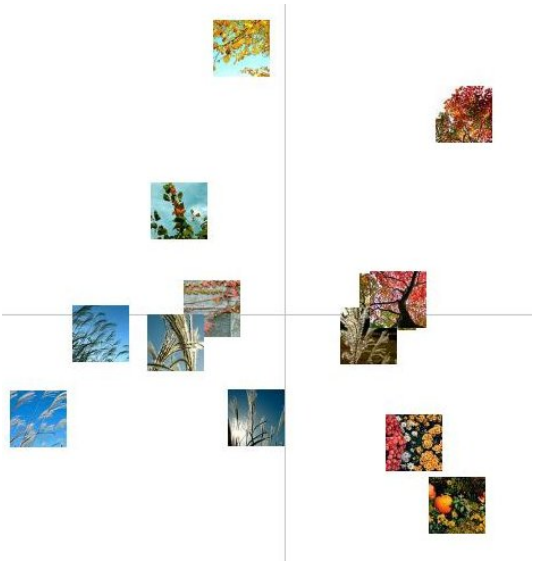
In the short run, we continue testing the Web site for shortcomings. Slowly, we will try to attract users by promoting it through mailing lists.

More content-based analysis tools will be integrated into the Web site. In particular, we will try to integrate the work on music analysis (Aucouturier et al., 2007; Pachet and Roy, 2007).

We will add new media types such as video and texts and we already took some steps to make Ikoru usable as an intuitive content management system (CMS) in which people can compose their Web pages. This would give Ikoru a competitive edge over other tagging systems.

[1] We designed the comparison function between tag assignments such that it introduced a total order on the tag assignments.

a)



b)

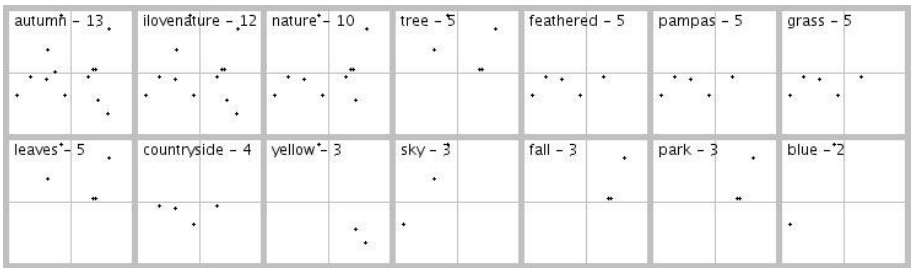


Figure 1.2: a) Visualization of the principal component analysis of the color feature for the “autumn” context, visible at `http://demo.ikoru.net/ikoru/63393403@N00/photo/autumn`. b) Subsets of the same analysis results, grouped by tags that co-occur with the “autumn” tag.

Bibliography

J.-J. Aucouturier, F. Pachet, P. Roy, and A. Beurivé. Signal + context = better classification, 2007. Submitted to the International Conference on Music Information Retrieval (ISMIR).

M. Aurnhammer, P. Hanappe, and Steels. L. Integrating collaborative tagging and emergent semantics for image retrieval. In *Proceedings WWW2006, Collaborative Web Tagging Workshop*, May 2006a. http://www.csl.sony.fr/downloads/papers/2006/aurnhammer-06b.pdf.

M. Aurnhammer, P. Hanappe, and L. Steels. Augmenting navigation for collaborative tagging with emergent semantics. In *International Semantic Web Conference*, 2006b. http://www.csl.sony.fr/downloads/papers/2006/aurnhammer-06a.pdf.

Andreas Hotho, Robert Jäschke, Christoph Schmitz, and Gerd Stumme. BibSonomy: A social bookmark and publication sharing system. In *Proceedings of the Conceptual Structures Tool Interoperability Workshop at the 14th International Conference on Conceptual Structures*, pages 87–102, 2006. ISBN 87-7307-769-0. URL http://www.kde.cs.uni-kassel.de/hotho/pub/2006/iccs_tools_ws_final.pdf.

F. Pachet and P. Roy. Exploring billions of audio features. In Eurasip, editor, *Proceedings of CBMI 07*, 2007. http://www.csl.sony.fr/downloads/papers/2007/pachet-07a.pdf.