

Community Cataloging Application: Description of V1.0 Prototype

Tom Leonhardt, July, 2005

Overview

The initial CCA prototype tool provides a basic web interface which allows the public to annotate a collection of images with their own text descriptors. Through this process images become associated with sets of terms that can then be used to support search functions. Both annotating and searching actions are logged and the resulting data is made accessible to researchers. An administrator is able to manage access by annotators and researchers.

Front End Functionality

Once logged in, a contributor is presented with a random image which has not been previously viewed. A text field is provided for them to input unstructured text describing the image in their own words. They also have the option to ignore the image. After submitting the information another random image is presented for cataloging, and so on until one chooses to quit the process or no further unviewed images remain. Users can continue the annotation process through multiple sessions.

Users who are not logged in are able to search the image collection using unstructured text input. Search results are displayed in a thumbnail 'lightbox' view with results listed in order of number of matches occurring between search terms and terms associated with images. Clicking on a thumbnail opens a new window containing a large view of the image with museum-supplied information such as title, author, date, etc. Image-term associations are created by a post-processing script invoked by the tool administrator.

User Management

The prototype requires a login identity for all users wishing to contribute or extract system data. Personal identity is verified through a registration process whereby a system-generated password is sent to the email address specified by the user. Supplying a registered password and identity combination allows the user to immediately begin contributing data to the system. If a user forgets their password, a new one is generated and sent to their email address upon request. This scheme provides a moderate level of security against malicious or automatic data entry.

Two additional user levels provide backend access to the system data. An administrator level has complete access to the database; allowing for common tasks such as user management, data importing and export data filtering among others. A researcher level allows access to the export of data from relevant database tables into delimited text files.

Community Cataloging Application: Description of v1.0 Prototype

(cont.)

Data Ingestion

Initial data is ingested into the tool by an administrator using sets of image and related text data files supplied by the museum. The images are tied to the museums content management system through ID content embedded in the data files. Additional image data such as author, title, credits, etc are optionally included in the data files. Image files are supplied at the largest required scale and resized dynamically for alternate views such as thumbnails.

Data Validation

Both term and search input is logged in its raw form along with a timestamp, user ID and image ID for later analysis. A plug-in type architecture was developed for post-processing the input data to allow flexibility in experimenting with different validation algorithms. Custom functions that adhere to standard input/output requirements can be uploaded to a script library as separate files. Once registered in the tool database, these functions can be run by an administrator as an ordered set of processes.

A basic function was supplied which breaks raw term input into words, removes non-alphabet characters, converts all characters to lower case, and uses only words which are longer than a specified length. The final output of the validation process is a list of word-image associations which include the number of associations found. This list is used to find images with the built-in search function, and can also be exported. Each time the validation process is run a completely new term list is generated based on the functions used and the input data available at the time.

Data Export

Data stored in various tables of the database can be exported as delimited text files for further analysis. Exportable data includes logs of raw term and search text, validated terms, registered users, and image data as supplied by the museum. The content of these files can be cross referenced using ID key fields which associate data across tables.

Technical Features

This version of the prototype tool is written in PHP (v5); a widely used, Open Source web applications scripting language available on most server platforms. All persistent data is stored in an SQL database, with the current implementation using MySQL - a popular Open Source product. The use of a database abstraction layer (ADODB - Open Source) makes it possible to run the prototype with numerous other SQL databases such as Oracle, msSQL, etc. The web interface presentation layer of the system is separated from its functional and data logic through the use of Smarty; a popular Open Source template engine written in PHP. This approach lets an interface designer work on different files from those the application programmer is concerned with, thereby facilitating team development.