Dynamic Link Service 2.0: using Wikipedia as a linkbase

Patrick Sinclair, Paul Lewis and Kirk Martinez
Electronics and Computer Science
University of Southampton

{pass, phl, km}@ecs.soton.ac.uk

ABSTRACT

This paper describes how a Web 2.0-style mashup approach, reusing technologies and services freely available on the web, have enabled the development of a dynamic link service system that uses Wikipedia as its linkbase.

Categories and Subject Descriptors

H.5.4 [**Hypertext/Hypermedia**]: Architectures

General Terms

Design

Keywords

Wikipedia, Dynamic Link Service

1. INTRODUCTION

In our work on integrating heterogeneous cultural heritage multimedia collections [5], we discovered that the majority of the valuable metadata is stored in an unstructured form: image captions, descriptions and uncontrolled text fields. This raises challenges in both integrating the different collections, and providing powerful search and browsing facilities across the different collections.

In order to overcome these issues we developed a system that would extract information from the free text descriptions and try to identify the respective Wikipedia article describing each entity extracted from the text. To start with we have focused on extracting peoples' names from the text, and our aim was to then retrieve structured information from the Wikipedia article to augment our knowledge base.

As described in Section 2, our approach for identifying the respective Wikipedia article describing each person was extremely simple by design, which allowed us to develop the system in a very short period of time. Whilst our crude approach provided remarkably good results for a large proportion of cases, we felt that it was not accurate enough for us to safely augment the knowledge base with the data from articles it identified.

The issue was that the system could not be trusted to run autonomously without human feedback to validate the results. From this observation, we adapted the system so that rather than extracting information from each identified article, it would simply inject links to the respective Wikipedia page into the text.

The result is a surprisingly powerful dynamic link service [4] that, through a Web 2.0-style mashup approach, uses the rich content from Wikipedia as its underlying linkbase. Unlike traditional link services, where the links in the linkbase are typically defined in advance, the system is able to dynamically add links to any person described on Wikipedia.

Copyright is held by the author/owner(s).
HT'07, September 10–12, 2007, Manchester, United Kingdom.
ACM 978-1-59593-820-6/07/0009.

2. SYSTEM ARCHITECTURE

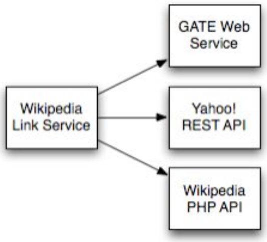


Figure 1. System Architecture.

The system we developed, Wikipedia Link Service, is simple but remarkably powerful. The service is deployed as a RESTful web service that accepts snippets of plain text, such as image captions, and returns that text with the dynamically injected links to the respective Wikipedia page. Figure 1 shows an overview of the system architecture.

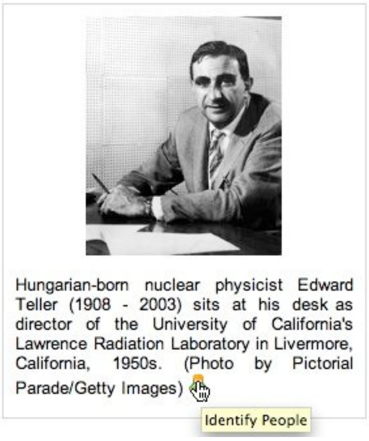
For performing the information extraction task we use GATE, the General Architecture for Text Engineering [2]. For added flexibility, this has also been deployed as a web service that is called by the Wikipedia Link Service. GATE analyses the text and extracts a variety of named entities, including people, places, organizations and so on. For now we have focused on only extracting and identifying peoples' names.

For each person name identified, we use a technique based on [1] to find the respective article on Wikipedia. This approach is extremely simple: using the Yahoo REST API [8] a search for that name is performed but restricted to the Wikipedia domain. This is accomplished by adding *site:en.wikipedia.org* to the query string. The system then returns the first result of the Yahoo query, which will typically be the Wikipedia article describing the person.

We developed a number of techniques to improve the accuracy of the system. Using the Wikipedia PHP API [7] the system checks that the page identified was an article about a person by looking for birth and death date categories. We also developed an alternative search mechanism using birth and death dates to take advantage of the fact that many cultural archives include this information in the text, as can be seen in Figure 2. Using the category information, the system takes an intersection of the set of people born and those who died in the year specified (e.g. 1908 - 2003) and then returns the closest matching name.

3. DEMONSTRATOR

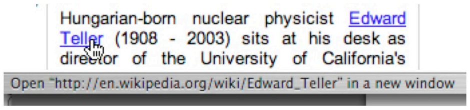
An online demonstration of the system is available at [6], and is illustrated in Figures 2 and 3. A user has reached a web page that contains an image together with its respective caption, and they would like to find out more about the person depicted in the image.



Hungarian-born nuclear physicist Edward Teller (1908 - 2003) sits at his desk as director of the University of California's Lawrence Radiation Laboratory in Livermore, California, 1950s. (Photo by Pictorial Parade/Getty Images)

Image courtesy of Getty Images

Figure 2. User clicks on the 'Identify People' icon to start the link injection process.



Hungarian-born nuclear physicist Edward Teller (1908 - 2003) sits at his desk as director of the University of California's

Open "http://en.wikipedia.org/wiki/Edward_Teller" in a new window

Figure 3. A link to the Wikipedia article describing the person in the image is added to the caption.

To both minimize access to the system and any undesired loading times for the user, the link injection is not automatic. Instead, the user is required to click the "Identify People" icon that has been placed after the caption (Figure 2). Once they do that, an AJAX request is made to the Wikipedia Link Service, which extracts peoples' names and attempts to identify the respective article on Wikipedia using the approach described in Section 2. The links are added to the text, which updates automatically when the service completes its actions.

Using Greasemonkey, a plugin [3] for Mozilla Firefox that allows users to install scripts that make on-the-fly changes to specific web pages, the system can be applied to any web site. We have developed scripts so that for a given picture archive, such as Getty Images, the 'Identify People' icon shown in Figure 2 can be added to the image captions on that site. This allows users to invoke the service as they are browsing the picture archive web site, without modifying the existing website structure.

4. CONCLUSIONS AND FUTURE WORK

The simple approach described in this paper has resulted in a dynamic link service that uses the whole of Wikipedia as its linkbase. Whilst most dynamic link service approaches might define a fixed linkbase, our system has a completely dynamic linkbase composed of the massive Wikipedia knowledge base.

We have found that due to its incredibly wide coverage of subjects Wikipedia is a fantastic resource for such a system. In contrast, more established resources such as the Getty ULAN and AAT thesauri would only be applicable to specific museum collections covering art and fine art. With Wikipedia we have been able to apply the system to diverse image archives such as Getty Images, which cover subjects from art to current events and photos of celebrities.

Another advantage is that Wikipedia is constantly growing and evolving, so if the user comes across a notable person who is not yet on Wikipedia they would be able to add a new article. Once Yahoo indexes the article, the system will be able to offer that link to other users.

The Web 2.0-style mashup approach we have taken to build this service, using resources such as Wikipedia and techniques such as the Yahoo REST API and Greasemonkey, has allowed us to revisit open hypertext ideas, such as dynamic link services, and deploy them on the web.

In terms of future work, we are considering expanding the system to link to other types of information besides people such as places and organization names. We would also like to improve the quality of the matching system by using more information taken from the caption to help disambiguate the person being searched for. We are also considering using the DBpedia project, which has made all of the structured information on Wikipedia available as a machine processable format (SPARQL), instead of using the limited Wikipedia PHP API.

5. ACKNOWLEDGMENTS

This research has been supported by the eCHASE project which is co-funded by the European Commission, DG Information Society, under the contract EDC 11262, within the "European digital content for the global networks" programme.

REFERENCES

[1] Biddulph, M., *Using Wikipedia and the Yahoo API to give structure to flat lists*, http://www.hackdiary.com/archives/000070.html

[2] Cunningham, H., Maynard, D., Bontcheva, K., Tablan, V. *GATE: A Framework and Graphical Development Environment for Robust NLP Tools and Applications*. Proceedings of the 40th Anniversary Meeting of the Association for Computational Linguistics (ACL'02). Philadelphia, July 2002

[3] Greasemonkey, http://www.greasespot.net/

[4] Pearl, A. *Sun's Link Service: A Protocol for Open Linking*, In Hypertext '89 Proceedings, Pittsburgh, 1989, pp. 137 – 146.

[5] Sinclair, P., Lewis, P., Martinez, K., Addis, M. and Prideaux, D. (2006) Semantic Web Integration of Cultural Heritage Sources (Poster). In Proceedings of 15th World Wide Web Conference, Edinburgh, Scotland.

[6] *Wikipedia Identifier Demonstrator*, http://multimedia.ecs.soton.ac.uk/semanticintegration/wikipedia-identifier

[7] *Wikipedia Query API* http://en.wikipedia.org/wiki/User:Yurik/Query_API

[8] Yahoo!, *Web Search Documentation for Yahoo! Search Web Services* http://developer.yahoo.com/search/web/V1/webSearch.htm