An Architecture for Automatic Reference Linking

Donna Bergmark and Carl Lagoze
Cornell Digital Library Research Group*

Abstract. Along with the explosive growth of the Web has come a great increase in on-line scholarly literature, which is often more current than what appears in printed publications. The increasing proportion of on-line scholarly literature makes it possible to implement functionality desirable to all researchers – the ability to access cited documents immediately from the citing paper. Implementing this direct access is called “reference linking”. The Cornell Digital Library Research Group employs value-added surrogates as a generalizable mechanism for providing reference-linking behavior in Web documents. This mechanism exposes reference linking data through a well-defined API, permitting the construction of reference linking services by external clients. We present two example reference linking applications buildable on this API. We also introduce a performance metric; currently we are (automatically) extracting reference linking information with more than 80% accuracy.

1 Background and Motivation

The variety of on-line scholarly literature has grown along with the web. Informal on-line archives are repositories for papers and technical reports. Proceedings are increasingly often published on the Web. The collection of on-line journals is growing. The increasing proportion of on-line scholarly literature (nearly 10,000 on-line journals in 1999 [11]) makes it possible to implement a function desirable to all researchers – the ability to access cited documents immediately from the citing paper. Implementing this direct access is called “reference linking”.

Reference linking is actually an old idea. Classical reference linking arose from a desire to study citation patterns among scholarly articles. The Science Citation Index 5 founded by Eugene Garfield in the 60’s was invented to do just that, and was a spectacular success. Staff captured a paper’s bibliographic data, and then went to the reference section and did the same for each reference to a journal covered by the SCI.

As a result, one could look up the links and build a graph as shown in Fig. 1(a). The nodes in this graph represent scientific papers, and directional arcs have two contextual meanings. Outgoing arcs, with respect to a specific node, lead to references of that paper. Incoming arcs, with respect to a specific node, originate from that paper’s citations. From the graph we can observe that Paper

* NSF Digital Libraries Initiative, Grant # IIS-9907892
C has 4 references and that Paper A has two citations. The links in the graph are explicitly contained in the SCI.

1. Paper C has 4 references.
2. Papers C, D, and G have been analyzed.
3. Paper A has 2 citations.

Fig. 1. (a) Classical Reference Linking vs. (b) Current Linking on the Web

We then fast-forward some 40 years to the current time, where there is a growing amount of scholarly literature on-line. In many cases, the authors of this literature have inserted references to other works on the Web. These URLs allow efficient movement from the referencing to the cited work, but do nothing to help maintain an account of a paper’s citations. Figure 1(b) shows a sub-graph of interlinked Web papers. From the fragment shown here, we can deduce that document C has four links in it to other Web documents and document A has at present two links to it.

Our reference linking project aims to bridge the classical view and what exists today on the Web. We wish to make the web graph explicit for selected repositories, as well as automatically supply additional links where possible. While a variety of uses exist for explicit hyperlink graphs in themselves [4,6], we are aiming higher. We aim to create a reference linking layer on the Web that provides sufficient data for a variety of value-added reference linking applications. Some of these applications may be targeted at human use, consisting of a user interface for navigating the reference linked graph. Others may be middleware, massaging data for use by other applications, such as to provide a citation database. Our main objective is to provide this layer automatically, without the direct participation from authors or publishers.

Figure 2 is an example of a human-oriented reference linking application that exploits the data from a reference linking layer.

Implementing the functionality shown in Fig. 2 requires a number of steps. The reference linking layer must: