Bootstrapping Semantic Annotation for Content-Rich HTML Documents

Saikat Mukherjee I.V. Ramakrishnan Amarjeet Singh
Department of Computer Science
Stony Brook University,
Stony Brook, NY 11794, U.S.A.
{saikat,ram,amarjeet}@cs.sunysb.edu

Abstract

Enormous amount of semantic data is still being encoded in HTML documents. Identifying and annotating the semantic concepts implicit in such documents makes them directly amenable for Semantic Web processing. In this paper we describe a highly automated technique for annotating HTML documents, especially template-based content-rich documents, containing many different semantic concepts per document. Starting with a (small) seed of hand-labeled instances of semantic concepts in a set of HTML documents we bootstrap an annotation process that automatically identifies unlabeled concept instances present in other documents. The bootstrapping technique exploits the observation that semantically related items in content-rich documents exhibit consistency in presentation style and spatial locality to learn a statistical model for accurately identifying different semantic concepts in HTML documents drawn from a variety of Web sources. We also present experimental results on the effectiveness of the technique.

1 Introduction

Semantic Web envisions a next-generation information network where content providers define and share machine processable data on the Web. A primary aspect of Semantic Web documents is that they contain metadata to express the meaning of their content. But an enormous amount of extant semantic data (such as product descriptions and pricing information, different categories of news, etc) is still being encoded in "plain" HTML documents. Although RDF/XML has been widely recognized as the standard vehicle for representing semantic information on the Web, we can extend the reach of Semantic Web to HTML documents by identifying and annotating the (implicit) semantic concepts that are present in their content.

Early solutions [15, 10] to this problem were based on hand-crafted ontologies and graphical ontology/annotation editors that facilitated manual mapping of unlabeled document segments to ontological concepts. From an automation standpoint they are at the "low-degree-of-automation" end of the solution spectrum. The technique in [7] as well as our previous work [22] cover the middle ground wherein document segmentation is done automatically and assignment of semantic labels to these segments is done with manually-crafted ontologies and knowledge bases.

Such a labeling process, via hand-crafted ontologies, is expensive and time-consuming. This begs the question: *Can this semantic labeling step be automated?* We address this question in this paper. Specifically we couple our structural analysis technique, described in [22], for partitioning Web pages into unlabeled segments (consisting of "semantically related" items) with a highly automated labeling step, to provide a "high-degree-of-automation" solution to the problem of semantic annotation of HTML documents.

Overview of our Approach: Our approach to structural analysis was based on the simple but useful observation that semantically related items in a HTML page normally exhibit *consistency in presentation style* and *spatial locality*. This is particularly true of content-rich Web sites that update frequently such as news portals, education and e-commerce sites, because these sites are typically maintained using content management software that create HTML pages by populating templates from backend databases.

Figure 1(a) and (b) are two examples of such Web sites. Note the consistency in presentation style of items in the news taxonomy in the left corner in Figure 1(a). The main taxonomic items, "NEWS", "OPINION", "FEATURES", ..., etc., are all presented in bold font. All the subtaxonomic items (*e.g.*, "International", "National", "Washington", ..., etc.) under a main taxonomic item (*e.g.*, "NEWS") are hyperlinks. This kind of consistency in presentation style has a very strong manifestation in the Document Object Model (DOM) tree of an HTML document. For example, Figure 2(a) is a fragment of the DOM tree for New York Times home page shown in Figure 1(a). The root-to-

Proceedings of the 21st International Conference on Data Engineering (ICDE 2005)
1084-4627/05 $20.00 © 2005 IEEE

IEEE COMPUTER SOCIETY