Ontology Guided Autonomous Label Assignment in Wrapper Induced Tables with Missing Column Names

Mohammad Shafkat Amin Hasan Jamil
Department of Computer Science, Wayne State University, USA
shafkat@wayne.edu, jamil@cs.wayne.edu

Abstract

Formulating and executing queries over distributed, autonomous and heterogeneous resources is an important research area. The advent of the Internet and the Web and their inherent ubiquity have brought forth opportunities to query these information sources in an automated and independent manner. In the domain of information extraction, automatic wrapper generation has been well studied but the efficacy of the current wrappers are limited by the fact that automatic annotation of column names to the extracted tabular data is yet to be perfected. In this paper, we propose a novel annotation system that can assign meaningful column names to the extracted tables for subsequent queries. We enhance our prototype wrapper system FastWrap with this annotator to support fast and autonomous on-the-fly data integration and ad hoc declarative querying.

1 Introduction

In recent years, there has been a flurry of research in the area of web data extraction and wrapper generation. Except a few, most research focused on producing end results for human consumption, and thereby did not address the issue of missing column names. The issue of missing column names arises for two primary reasons. First, typical Web pages often omit labels, which are understood from the context by a human. Second, even if a column name is present, user is forced to use the labels chosen by the Web content providers, which may not be the most appropriate or most descriptive ones. For example, if we take a look at the response produced by a query submitted at a computer vendor site e.g. www.dell.com, we can see that none of the attributes in the resulting records contain a column name. Thus to be able to query the site considering it as a relational data repository, not only the data record will be needed to be extracted but also all the constituent attributes will need to be named. In applications such as comparison shopping where large number of “arbitrary” web sites are searched and information extracted, manual name assignment is not feasible, and an automated procedure is preferred so that results of a query can be used in a follow-up query or can be composed with other queries. Examples of such comparison shopping engines include expedia.com, shopzilla.com, amazon.com, buy.com, and kayak.com. These engines are designed to access specific set of hidden web sites and consolidate information using one single web form so that users need not browse the individual sites themselves manually. While these engines are efficient and have their advantages, there is a need to integrate hidden web databases dynamically and on demand. This is because all needs cannot be specified ahead of time, new databases emerge every day and need to be integrated with the existing engines and in the event of changes in the databases, the search engines need to adapt autonomously to the change. Hence, the importance of automatic column annotation as an integral part of data integration pipeline is undeniable.

2 A Procedure for Column Name Annotation

Wrapper generation process (e.g. FastWrap[1]) from web data sources entails extraction of tabular structure from the input HTML page and its conversion into a relational structure for subsequent query management. This process introduces the challenge of annotating extracted columns from the input HTML text so that the assigned annotations can be used to treat the result as a relational table. The process itself inherently introduces newer complexities regarding domain dependent labeling. In this paper, we propose a novel algorithm to address these issues by analyzing both syntactic and semantic information embedded in the data. In order to impute putative representative column names to be assigned in each of the data fields extracted from the input HTML table, we maintain a hierarchy of Ontological knowledgebase pertaining to the semantic and structural information which is subsequently applied to the column data vector to extrapolate its membership column annotation from a list of input column names. Data table, once ex-
extracted is subjected to a dual scanning mechanism i.e. horizontal and vertical. The vertical scan is performed for each column, where we strive to extract the structural commonality that may exist among the data pertaining to the same column. The structural information is expressed in regular expression in terms of numeric characters, special characters, and string of alphabets. In order to further restrict the scope of membership, any alphanumeric string that appears in each and every row of the column under consideration, is retained in the structural information. The ontology is designed to have a flexible and extendible structure, where a previously unseen structural and semantic information can be easily incorporated in the hierarchical repository. The horizontal scan is performed to mitigate contention between two or more columns that have the probability of being assigned the same column name. As a first step of the processing we initially iteratively process data pertaining to a single column to identify hidden structural patterns associated with it. We first extract the largest common prefix and largest common suffix pertaining to all values in the column vector. We retain those values since they are likely to provide semantic information. Next, we split each and every entry in the column vector in terms of constituent parts of structural components and create a generic regular expression corresponding to that entry. However, it is likely that not all of the entries in a column vector will adhere to the same structural pattern. Hence, we combine all the entries by creating a global regular expression that can satisfy all values in the column vector. We retain those values since they are likely to provide semantic information. Next, we split each and every entry in the column vector in terms of constituent parts of structural components and create a generic regular expression corresponding to that entry. However, it is likely that not all of the entries in a column vector will adhere to the same structural pattern. Hence, we combine all the entries by creating a global regular expression that can satisfy all the elements in the column vector. This is done by vertically aligning the structures using multiple sequence alignment technique. The algorithm for syntactic data extraction can now be written as follows.

Procedure ExtractStructure
Input: ColumnVector C  
Output: Labeled ColumnVector R.  
Given: Regular Expression p.  
begin  
1 $\Pi = \text{LongestCommonPrefix}(C)$  
2 $\Sigma = \text{LongestCommonSuffix}(C)$  
3 for each element $c_i \in C$  
4 Remove $\Pi$ and $\Sigma$ from $c_i$  
5 for each element $c_i \in C$  
6 for each element $\rho_j \in \rho$  
7 if $c_i$ contains $\rho_j$ then  
8 $S = S \cup \text{Replace}(c_i, \rho_j)$  
9 for each element $k \in S$  
10 $S_k = \text{concat}(\Pi, S_k, \Sigma)$  
11 $R = \text{GenerateGenericRegularExpression}($ClustalW$(S))$  
12 return $R$  
end

Once we have obtained the generic regular expression from the column we look up the Ontology that allows traversing its hierarchy based on the query pattern to reduce the search space in the semantic matcher domain. If the structure is absent in the Ontology, the user can incorporate the structure in the ontology. Each Node contains a list of plausible keywords that is associated with the structure. Since the whole wrapper extraction process is driven as a response to a user query (as in Bioflow [3]) where a user submits a list of column names to be extracted, we employ OntoMatch [2] to filter the list of names that can be assigned to a specific column. Although the user expected column names may differ significantly from the actual annotations, OntoMatch can infer the semantic and syntactic relationship between the user expectation and the system assigned annotation.

3 Incorporating Wikipedia as a Knowledge-base

Knowledgebases are the missing links in the information extraction and web intelligence domain that can enhance the capability of current wrapper mediator systems by augmenting it with semantic information needed to facilitate the over all extraction process. In this regard, a comprehensive repository of vocabularies that provides unique identifiers for conceptual elements can play a vital role in resolving semantic coherence pertaining to a group of words. Wikipedia involves community effort to create and maintain its knowledgebase, easily manages numerous comprehensive, informal definitions of terms, each one identified by a URI and categorized by a complex network of categories reflecting contextualized community expectation. Wikipedia contains a mapping between concepts and their inherent categories as perceived by the community. This category-concept mapping can be employed to further extend the efficacy of the column name annotation system.

4 Summary and Future Research

In this paper, we presented a novel wrapper generation system with automatic column name identification capability. The performance of the column name identifier improves monotonically as the number of web sites analyzed is increased. Thus as the ontology and the knowledge base is enriched, the assigned column names will continue to be more accurate and efficient. We regard our proposed technique as a first step toward autonomous annotation of missing column names that warrants more research.

References