HTML Parser and Support Vector Machine used for Automatic Template Extraction

[1] C.SreeDeepak, [2] S.Kusuma M.Tech (CSE)
[1] Research Scholar, Department of CSE,
Madanapalli Institute of Technology and Science,Madanapalli,A.P. , India.

[2] Assistant Professor,
Department of CSE,
Madanapalli Institute of Technology and Science, Madanapalli, A.P. , India

Abstract- Extracting data from Web pages using wrappers is a fundamental problem arising in a large variety of applications of vast practical interests. There are two main issues relevant to Web-data extraction, namely wrapper generation and wrapper maintenance. In this paper, we propose a novel schema-guided approach to the problem of automatic wrapper maintenance. It is based on the observation that despite various page changes, many important features of the pages are preserved, such as syntactic patterns, annotations, and hyperlinks of the extracted data items. Our approach uses these preserved features to identify the locations of the desired values in the changed pages, and repair wrappers

In this paper, we present novel algorithms for extracting templates from a large number of web documents which are generated from heterogeneous templates. We cluster the web documents based on the similarity of underlying template structures in the documents so that the template for each cluster is extracted simultaneously. We develop a novel goodness measure with its fast approximation for clustering and provide comprehensive analysis of our algorithm. Our experimental results with real-life data sets confirm the effectiveness and robustness of our algorithm compared to the state of the art for template detection algorithms.

Introduction:

World Wide Web is the most useful source of information. In order to achieve high productivity of publishing, the WebPages in many websites are automatically populated by using the common templates with contents. The templates provide readers easy access to the contents guided by consistent structures. However, for machines, the templates are considered harmful since they degrade the accuracy and performance of web applications due to irrelevant terms in templates. Thus, template detection techniques have received a lot of attention recently to improve the performance of search engines, clustering, and classification of web documents.

The Web poses itself as the largest data repository ever available in the history of humankind. Major efforts have been made in order to provide efficient access to relevant information within this huge repository of data. Although several techniques have been developed to the problem of Web data extraction, their use is still not spread, mostly because of the need for high human intervention and the low quality of the extraction results.In this paper, we present a domain-oriented approach to Web data extraction and discuss its application to automatically extracting news from Web sites. Our approach is based on a highly efficient tree

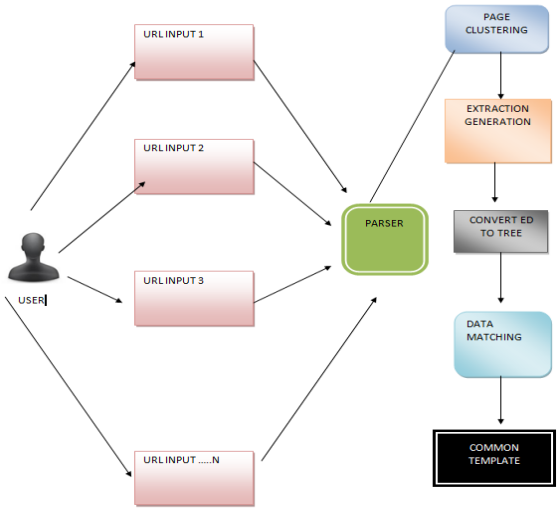
structure analysis that produces very effective results. We have tested our approach with several important Brazilian on-line news sites and achieved very precise results, correctly extracting 87.71% of the news in a set of 4088 pages distributed among 35 different sites.

In this paper, we present novel algorithms for extracting templates from a large number of web documents which are generated from heterogeneous templates. We cluster the web documents based on the similarity of underlying template structures in the documents so that the template for each cluster is extracted simultaneously. We develop a novel goodness measure with its fast approximation for clustering and provide comprehensive analysis of our algorithm. Our experimental results with real-life data sets confirm the effectiveness and robustness of our algorithm compared to the state of the art for template detection algorithms.

Related work:

The template extraction problem can be categorized into two broad areas. The first area is the site-level template detection where the template is decided based on several pages from the same site. Crescenzi et al. studied initially the data extraction problem and Yossef and Rajagopalan introduced the template detection problem. Previously, only tags were considered to find templates but Arasu and Garcia-Molina observed that any word can be a part of the template or contents. We also adopt this observation and consider every word equally in our solution. However, they detect elements of a template by the frequencies of words but we consider the MDL principle as well as the frequencies to decide templates from heterogenous documents. Vieira et al. suggested an algorithm considering documents as trees but the operations on trees are usually too expensive to be applied to a large number of documents. Zhao et al. concentrated on the problem of extracting result records from search engines

A novel approach of Proposed system:



Newcomers to Perl often want to know how to parse HTML. For instance, to extract the text between between and tags, or to extract content by assembling and following hyperlinks. HTML is treacherous in that in looks as though it could be handled with just a few regular expressions. Even when you slurp the whole file and work on large strings, sooner or

later regular expressions won't be enough.The HTML::Parser module provides powerful mechanisms for extracting content, tags and tag attributes from any html stream.

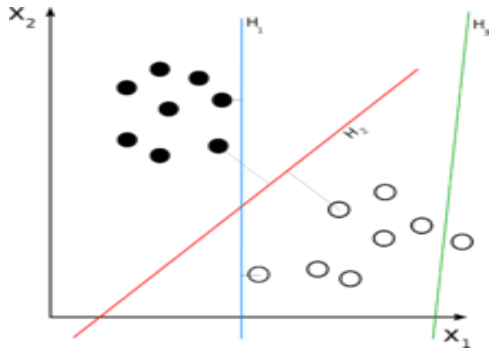
A support vector machine (SVM) is a concept in statistics and computer science for a set of related supervised learning methods that analyze data and recognize patterns, used for classification and regression analysis. The standard SVM takes a set of input data and predicts, for each given input, which of two possible classes comprises the input, making the SVM a non-probabilistic binary linear classifier. Given a set of training examples, each marked as belonging to one of two categories, an SVM training algorithm builds a model that assigns new examples into one category or the other. An SVM model is a representation of the examples as points in space, mapped so that the examples of the separate categories are divided by a clear gap that is as wide as possible. New examples are then mapped into that same space and predicted to belong to a category based on which side of the gap they fall on.

Hypothesis Formulation and Variable Selection:

To keep the computational load reasonable, the mapping used by SVM chosen to be linear combinations with parameters α_i of images of feature vectors that occur in the data base. With this choice of a hyper plane, the points x in the feature space that are mapped into the hyper plane are defined by the relation:schemes are designed to ensure that dot products may be computed easily in terms of the variables in the original space, by defining them in terms of a kernel function $K(x,y)$ selected to suit the problem.[1] The hyperplanes in the higher dimensional space are defined as the set of points whose inner product with a vector in that space is constant. The vectors defining the hyperplanes can be

$$\sum_i \alpha_i K(x_i,x) = constant$$

Note that if $K(x,y)$ becomes small as y grows further from x, each element in the sum measures the degree of closeness of the test point x to the corresponding data base point x_i. In this way, the sum of kernels above can be used to measure the relative nearness of each test point to the data points originating in one or the other of the sets to be discriminated. Note the fact that the set of points x mapped into any hyperplane can be quite convoluted as a result allowing much more complex discrimination between sets which are not convex at all in the original space.



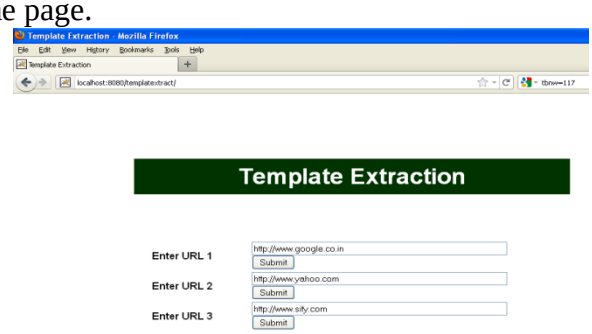
H3 (green) doesn't separate the two classes. H1 (blue) does, with a small margin and H2 (red) with the maximum margin.

Classifying data is a common task in machine learning. Suppose some given data points each belong to one of two classes, and the goal is to decide which class a *new* data point will be in. In the case of support vector machines, a data point is viewed as a p-dimensional vector (a list of p numbers), and we want to know whether we can separate such points with a $(p-1)$-dimensional hyperplane. This is called a linear classifier. There are many hyperplanes that might classify the data. One reasonable choice as the best hyperplane is the one that represents the largest separation, or margin, between the two classes. So we choose the hyperplane so that the distance from it to the nearest data point on each side is maximized. If such a hyperplane exists, it is known as the *maximum-margin hyperplane* and the linear classifier it defines is known as a *maximum margin classifier*; or equivalently, the *perceptron of optimal stability*.

Result analysis:

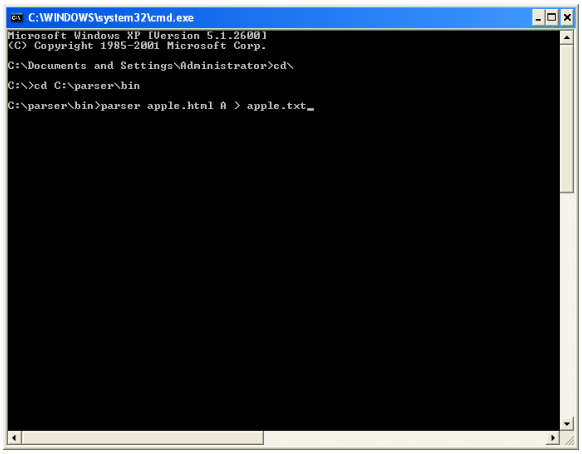
The screen shots of the paper is displayed below pictures.

1) the front end of the page.

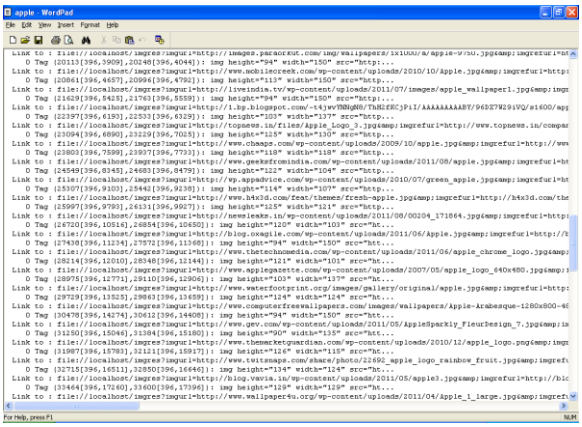


2)got the template from web pages

3)execution



4)



Conclusion:

We introduced HTML parser for extracting the template from the input data. Extracting and maintaining the data from the web pages is a big problem, so we used the parser for the easy extraction. We introduced a novel approach of the template detection from heterogeneous web documents. We employed the Support vector machine method a non-probabilistic linear classifier, its both the clustering and classification technique that improves the efficiency in extracting the common template.

References:

[1]Collective Extraction from Heterogeneous Web Lists Ashwin Machanavajjhala_ Arun Iyer† Philip Bohannon_ Srujana Merugu_

[2]Heterogeneous Web Data Extraction using Ontology Hicham Snoussi Laurent Magnin Jian-Yun Nie

[3]Collective Extraction from Heterogeneous Web Lists Ashwin Machanavajjhala_ Arun Iyer† Philip Bohannon_ Srujana Merugu_

[4]Experiences with Content Extraction from the Web Mira Dontcheva1;2 Sharon Lin1 Steven M. Drucker3 David Salesin1;2 Michael F. Cohen4

[5]Information Extraction from Heterogeneous Sources Using Domain Ontologies Waqas-ur-Rehman Chaudhry, Farid Meziane

[6]Automatic Web News Extraction Using Tree Edit Distance Davi de Castro Reis1 2 Paulo B. Golgher2 Altigran S. da Silva3 Alberto H. F. Laender1

[7]Automatic Web Data Extraction based on Genetic Algorithms and Regular Expressions David F. Barrero and David Camacho and Maria D. R-Moreno