Project I

Extracting Information from Web: A Case Study Using Banner Transcript Page

Hui Chen

Release on Feburary 17, 2016
Lastly revised on February 17, 2016
Revision:

Contents

1 Introduction 1

2 Objective 1

3 General Advice 2

4 Submission 3

1 Introduction

Data often appear unstructured and nonetheless contain useful information. A few examples that retrieve useful information from unstructured or semi-structured data are in [1–3,8]. An important step in those work is to convert the unstructured or semi-structured data to structured data that data analysis programs can recognize and process. Lexical and syntax analysis is an important tool for this endeavor. In this project, you will use lexical and syntax analysis knowledge and skills to extract useful information from web pages that we consider as semi-structured data.

2 Objective

Many universities use Banner to manage students' data. Virginia State University is one of those institutions. You can view your transcript using Banner.

The objective is in this project is as follows,

- to automatically retrieve the transcript data by writing one or more programs,
- and to efficiently answer the following queries in one or more programs,
  - what classes do I need to take to meet the requirement of a given academic curriculum, e.g., a computer science curriculum?
  - what is the cumulative GPA of my courses in a subject area, e.g., mathematics, computer science, general education?
  - what is the cumulative GPA of my major courses, and that of non-major courses?
  - what is the cumulative GPA of last two years, or last three years?

3 General Advice

As indicated in [6],

> " WebDriver is a remote control interface that enables introspection and control of user agents. It provides a platform- and language-neutral wire protocol as a way for out-of-process programs to remotely instruct the behaviour of web browsers. "

It is a tool that you may use to automate the data retrieval from Banner. A well-known implementation of the WebDriver is *Selenium WebDriver* [5] that has been considered as a reference implementation [7]. Selenium WebDriver provides binding for Java, C#, Ruby, Python, and Javascript (Node.Js) [4].

To parse the transcript web page in Banner, one should weigh a few factors, *correctness*, *reliability*, *robustness*, and *complexity*.

- Correctness. The data retrieved from the web page of course needs to be correct and demonstrated by test cases.
- Reliability. Despite the factor the data retrieved appear to be correct, how likely will they remain correct given a great number of student transcripts that contain various variations and conditions that have never been encountered before ?
- Robustness. After all, Banner is a software product we do not have any control. When Banner changes the format of the web page, will the data retrieved remain correct?
- Complexity. One may consider various factors, such as the items above, in designing and writing the programs. One may easily become two ambitious or too timid when approaching this project. Although no one can tell what the balance is, you will always consider how to strike a balance. Simply put, neither is too ambitious and too complex of an approach good, nor is too timid and too simple one.

We can model an academic curriculum as a formal language. The transcript of an actual student is an instance in the language. The instructor will provide a copy of the Computer Science curriculum at Virginia State University.

4 Submission

The submission deadline is *March 4, 2016*.

Include the following items in your submission,

- a well-written, but concise report containing motivation, design including both design on lexical and syntax analyzers and design of the applications, a discussion on implementation, a discussion on test procedure, and a list of test cases and results.
- and a public code repository containing the outcomes of the project including source code and the report

References

[1] Charu C. Aggarwal and ChengXiang Zhai. *Mining Text Data*. Springer Publishing Company, Incorporated, 2014.

[2] Jiawei Han and Chi Wang. Mining latent entity structures from massive unstructured and interconnected data. In *Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data*, SIGMOD '14, pages 1409–1410, New York, NY, USA, 2014. ACM.

[3] Matthew A. Russell. *Mining the Social Web: Analyzing Data from Facebook, Twitter, LinkedIn, and Other Social Media Sites*. O'Reilly Media, Inc., 1st edition, 2011.

[4] SeleniumHQ. Selenium Client & WebDriver Language Bindings. http://www.seleniumhq.org/download/, retrieved on February 17, 2016.

[5] SeleniumHQ. Selenium WebDriver. http://www.seleniumhq.org/projects/webdriver/, retrieved on February 17, 2016.

[6] Simon Stewart and David Burns. WebDriver – Living Document. https://www.w3.org/TR/webdriver/, retrieved on February 17, 2016.

[7] Simon Stewart and David Burns. WebDriver – W3C Working Draft 17 January 2013. https://www.w3.org/TR/2013/WD-webdriver-20130117/, retrieved on February 17, 2016.

[8] Stephen W. Thomas, Ahmed E. Hassan, and Dorothea Blostein. *Evolving Software Systems*, chapter Mining Unstructured Software Repositories, pages 139–162. Springer Berlin Heidelberg, Berlin, Heidelberg, 2014.