CS839 Project Stage 2

Group members: Yiwu Zhong, Zhihan Guo, Jing Liu

Aril 7, 2019

1 Web Data Source

We extracted the movie data from **Rotten Tomatoes** and **IMDB**. They are both online data sets of movies, which contain the movie meta data and the review scores provided by audiences. Also, each website is already structured in some certain format so that we can apply our rule-based wrapper to crawl the data.

2 How to Extract Data

Since the websites are well-structured, we construct rule-based wrapper to search the tags and extract the text content inside those tags. More specifically, we first identify the special tags which are always accompanied with the data we need and use Beautiful Soup 4 to locate these tags. Then we use regex expression to extract the plain text inside the tags and clean them to the format we want.

3 Description of Output

We extract **3000** instances for Rotten Tomatoes and **3003** instances for IMDB. We extract the following data for each movie: **'name', 'year', 'score', 'tomatoter', 'audience', 'runtime', 'genre', 'certificate', 'gross', 'director', 'star', 'writer'**. Each table contains the **meta data** of movies, such as the 'name' and 'runtime', and the **review scores**, such as 'score' from IMDB, and 'tomatoter' and 'audience' from Rotten Tomatoes. Some of them are **strings**, like 'name' and 'director', and the others are **number**, like "year", "tomatoter" and "gross".

4 Tools

- Request: given a url of a website, we use request to obtain the HTML content of that website.

- Beautiful Soup 4: Beautiful Soup 4 is a parsing library that can use different parsers. A parser is simply a program that can extract data from HTML and XML documents. We use this tool to locate tags and extract the content within the tags.

- CSV, Pandas: we use them to format the data we extract to a table.