Homework 3: Wrapper

Due Date: Wednesday, 2016/09/28 @11:59pm on Blackboard.

Summary

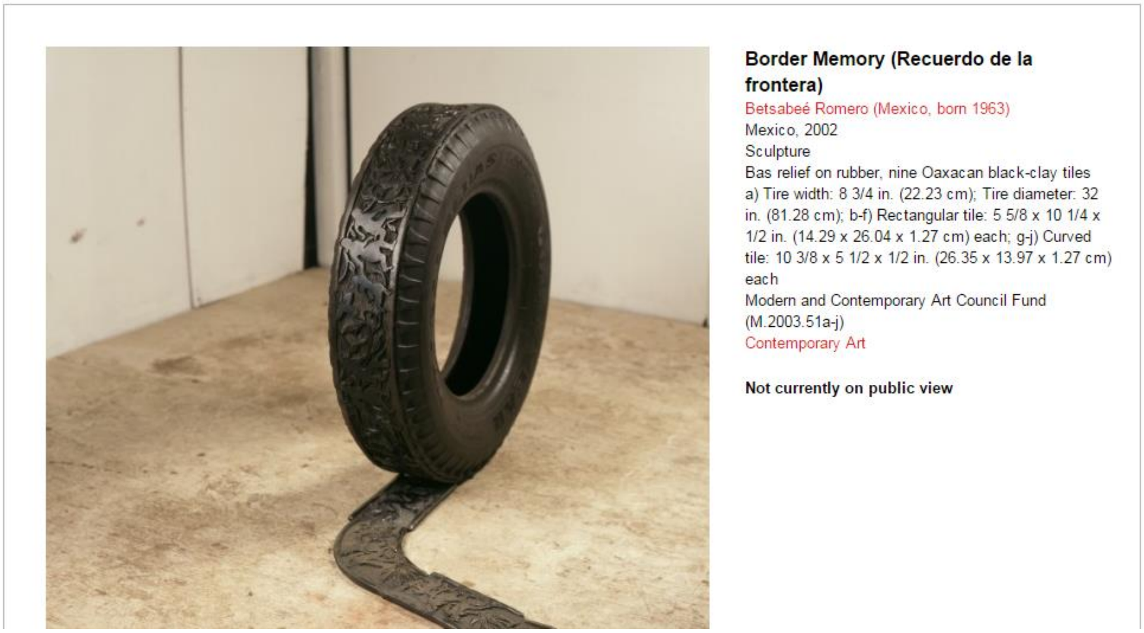
In this homework, you will construct a wrapper to extract data from semi-structured sources.

Task 1 – Data Source

Identify your data source and the data you want to extract.

You should choose a website and scrape **at least 500** webpages, then identify **at least 5** fields that you want to extract from it.

Example:



Border Memory (Recuerdo de la frontera)
Betsabeé Romero (Mexico, born 1963)
Mexico, 2002
Sculpture
Bas relief on rubber, nine Oaxacan black-clay tiles
a) Tire width: 8 3/4 in. (22.23 cm); Tire diameter: 32 in. (81.28 cm); b-f) Rectangular tile: 5 5/8 x 10 1/4 x 1/2 in. (14.29 x 26.04 x 1.27 cm) each; g-j) Curved tile: 10 3/8 x 5 1/2 x 1/2 in. (26.35 x 13.97 x 1.27 cm) each
Modern and Contemporary Art Council Fund (M.2003.51a-j)
Contemporary Art

Not currently on public view

Extracting data about art from LACMA website.

Fields that can be extracted:

- Artist
    - Name
    - Birth year
    - Birth place
- Date of art
- Place art was made
- Type of art
- Materials
- Size of art

Task 2 – Wrapper Construction and Data Extraction

Construct your wrapper using any existing tools available.

For example, one library you can use is BeautifulSoup (http://www.crummy.com/software/BeautifulSoup/) for writing your own scaper. Another example is using the Portia service provided by http://scrapinghub.com/. Yet more examples are provided at the ends of the class slides.

You can either manually construct the wrapper or use a wrapper learner.

Extract at least 5 fields of data from all the webpages, and save them into a file in json format (See submissions for format).

Task 3 – Questions

1. What is the website you are extracting information from (give the base URI and a one-line description about the website)? Name and describe the fields you are extracting (use a table with two columns: field and description. The description should be one line/field). Provide a representative screenshot of the website, and annotate (on the screenshot) the field values that get extracted. **(10 points)**
2. What is the tool that you used to scrape the data? Provide the name of the tool, the link, and between 1-3 sentences about why you decided to pick that tool. **(10 points)**
3. In a single (<=6 sentences) paragraph, describe your wrapper. **Try to be as specific as possible.** We will be looking for details like what kind of wrapper you used (e.g. manual, automatic…?), what is the wrapper model (e.g. HLRT, LR…?), where we can access the wrapper algorithm, if your wrapper is non-manual etc. A good rule of thumb is, can someone familiar

with wrappers read your paragraph and be able to (roughly) replicate your wrapper for themselves? **(20 points)**

Submission Instructions

You must submit the following files (totally 3 files) in a single .zip archive named `Firstname_Lastname_hw5.zip` and submit it via Blackboard:

- **(40 points)** The PDF file `Firstname_Lastname_hw5.pdf` containing the answers to Task 3.
- **(25 points)** The zip file `Firstname_Lastname_hw5.zip` containing the code you have written for your wrapper
- **(15 points)** The file `data.zip` which contains all the raw webpages (at least 500) you scraped.
- **(20 points)** The file `extractions.json` containing the extracted data. Each line in the file must be a **1-level json object** (see below) that contains a URL field with a value being the URL of the webpage on which you ran your extractions, and a field for each extraction. If there are **multiple extractions** for the field (in that webpage), use a list for the value. If the field is nested within another field, use dot delimiters rather than nested objects (this is what we mean by 1-level; see artist.birth-place and artist.name in the example below). See the example below for further clarification:
  - **Example:** Consider the LACMA example from the previous page. Suppose you extracted an artist (with name and birth-place but not birth-year), and 3 type-of-art fields from the (made-up) webpage www.mypage.lacma.com. The corresponding JSON object will look like the following:

  {“URL”: “www.mypage.lacma.com”, “artist.name”: “…”, “artist.birth-place”: “…”, “type-of-art”: [“…”,…, “…”]}

  Be careful about the quotes and caps. Make sure that you do not use newlines inside the JSON, as we will check for a new JSON in each line. There are JSON validators (and also JSON lines-file validators like the one you’re being asked to produce) that you can use to ensure your JSONs correctly match JSON specs. Furthermore, note that it is possible (and perfectly okay) for a field (e.g. type-of-art) to be a list in one JSON but a string in another. Avoid ‘empty’ lists and strings (in other words, if you didn’t extract something (e.g. materials) then that field should not occur as a key in that JSON. FYI, JSON objects are central to what we call *NoSQL* databases that have flexible schemas but lose some of the guarantees of Relational Databases.

Ground Rules

This homework must be done individually. You can ask others for help with the tools, but the submitted homework needs to be your own work. **We will be running sophisticated scripts on all the submissions to detect potential cheaters.**