**CS513: Theory & Practice of Data Cleaning Project**

End-to-End Data Cleaning Workflow

MCS-DS program at the University of Illinois at Urbana-Champaign – Fall 2018

Abstract: The report describes a data cleaning workflow using the New York Public Library Rare Books Division historical menus dataset has an example to demonstrate various cleaning techniques. This project will use the following software and tools to clean and organize the dataset: OpenRefine, SQLite, and YesWorkFlow.

**Team members**

Brad Ballard - bjb3@illinois.edu

Dhanendra Singh - disingh2@illinois.edu

Jacob Rettig - jrettig2@illinois.edu

**1. Dataset Overview and Initial Assessment**

We use tools introduced in CS513 to clean and prepare a sample dirty dataset (NY menus). Along the way we will document the steps from dirty to clean. The sections that follow describe the specific steps in detail: (2.) Data cleaning with OpenRefine, (3.) Develop Rational Database Schema, and (4.) Create a Workflow Model.

An initial assessment of the New York Public Library dataset (referred to now forward as NY menus dataset) is over 45,000 historical menus. The majority of these were organized by Frank E. Buttolph (ref1) around 1900-1921. The dates on the menus range from the 1850s to 2010s. The data contains information on restaurant menu, but also other organizations like railroad or shipping companies. The dataset was digitized in 2011 via the “What’s on the Menu?” project (ref2). So far 17,500 the libraries’ historical menus have been digitized.

The data is split into four files: Menu, MenuPage, MenuItem, and Dish. The ‘Menu.csv’ file has the unique id number, location information, venue, currency used, and other description-based information. The ‘MenuPage.csv’ file contains the id, plus an additional unique mean\_id, image\_id, height, width, and another with other image related information. The ‘MenuItem.csv’ file contains the id, plus an additional\_page\_id, dish\_id, and other price related information for each dish. The ‘Dish.csv’ file contains the id, name of the dish, description, first/last appearance, and various price information.

Some hypothetical use cases for a dataset like this should be – How has the composition of restaurants changed over time? The density, clusters, or price changes? Can a model be made to predict food prices in certain areas? How have consumer food preferences changed over time? Can a model be made to estimate food prices based on ingredients/description? The problem with answering all these questions is - the dataset is quite messy. For example, the date columns seem full of repeat or missing dates. So, how plan does the dataset need to be? You can sort the columns and get some sense of the composition of the data, but there will need to be some prep work to answer practical questions. Having the data broken into multiple files makes it hard to compare across sheets. However, this format should be a straight forward import into the SQL database.

Progress throughout his project can be tracked at our github repository (ref3) or this associated webpage (ref4). This tool was mostly used to aid in our team collaboration, but also provided a platform to share our data cleaning techniques with the wider world.

**2. Data cleaning with OpenRefine**

OpenRefine, formal called Google Refine, is an open source desktop application the has many help helpful features for data cleaning. It behaves like a database with rows of data and cells under columns which are similar to relational database tables. An OpenRefine project itself consists of one table. I plan to clean each of the four data file separately.

The major OpenRefine feature that will be helpful in cleaning our dataset is the clustering feature. The option allows the user to cluster similar text and replace it with a more standardized description. The common problem this solves the many variants of spelling but which, in fact, reference the same object.

The following subsections will describe the step-by-step process from input file to output file. As mentioned above, each file will be loaded into OpenRefine individually. We will use UTF-8 encoding.

***File: Menu.csv***

*For column: sponsor*

(1) Trim leading and trailing white spaces

(2) Collapse consecutive white spaces

(3) Convert all column values to upper case

(4) Remove special characters using GREL (%, #, !, /, (, ), [, ], ?)

(5) Replace “;” with a space instead and then trim leading/trailing white spaces

(6) Make a facet and perform the cluster operation using the key-collision method and fingerprint function. Next merge the selected clusters.

(7) Repeat the previous step with n-gram, fingerprint, meta-phone3, and cologne-phonetic methods.

(8) Create a new facet. Next, use the cluster operation nearest neighbor method and levestein distance function. Then, merge the selected clusters.

(9) Create a new facet. Next, use the cluster operation nearest neighbor method and PPM distance function. Then, merge the selected clusters.

[INSERT SCREENSHOTS]

*For column: physical\_description*

(1) Slit the columns using ‘;’

(2) Then rename the first column: physical\_description\_type

(3) Use GREL to join ‘physical\_description 2’, ‘physical\_description 3’, and ‘physical\_description 4’ into one column named: physical\_description\_additional.

(4) Use ‘ - ‘ to separate the values from the different columns (Remember the space before and after the dash). Note: if the cell in the column is empty, leave that value as a blank space. For example, if a column is blank, make the new column in ‘physical\_description\_additional’ blank also.

[INSERT SCREENSHOTS]

*For column: date*

(1) Convert date format to YYY-MM-DD

(2) Remove date outliers, where the year is either less than 1851 or greater than 2012. Make these flagged cells blank.

*For column: call\_number*

(1) Trim leading/trailing white spaces

(2) Collapse consecutive white spaces

Unchanged columns: id, name, keywords, language, status, page\_count, dish\_count

***File: MenuPage.csv***

File not cleaned in OpenRefine

Unchanged columns: id, menu\_id, page\_number, image\_id, full\_height, full\_width, uuid

***File: MenuItem.csv***

*For column: created\_at*

(1) Convert date format to YYY-MM-DD

(2) Remove date outliers, where the year is either less than 1851 or greater than 2012. Make these flagged cells blank.

*For column: updated\_at*

(1) Convert date format to YYY-MM-DD

(2) Remove date outliers, where the year is either less than 1851 or greater than 2012. Make these flagged cells blank.

File: Dish.csv

For column: name

For column:

(1) adafd

(2)

(3)

(4)

For column:

(1) adafd

(2)

(3)

(4)

**3. Develop Rational Database Schema**

here

**4. Create a Workflow Model**

here

**4. Conclusion**

here