CS513: Theory & Practice of Data Cleaning

Suruchi Sinha – suruchi3@illinois.edu
Pradeep Sharma – pks6@illinois.edu
Upul Gunasena – uig2@illinois.edu

July 2021

CS513: THEORY & PRACTICE OF DATA CLEANING

Github: https://github.com/upulindika/CS-513-Team75

Abstract- This 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 opensource software and tools to clean and organize the dataset: OpenRefine, SQLite, and YesWorkFlow, Creately for creating ER diagrams

Index Terms- Data Cleaning, Provenance, OpenRefine, YesWorkflow, SQLite

1. INTRODUCTION

We are going to select the New York public library historical menus dataset. The data set NY menus is a collection of more than 45000 historical menus from the 1850s to the 2000s. Most data were organized by Frank E Buttolph [5] around 1900-1921. The dataset was digitized in 2011 via the "What's on Menu?" project and so far, 17500 historical menus have been digitized. We will try to clean this data using the tools and the techniques we have learned in the CS513 Theory and Practice of Data cleaning.

2. DATA SET INTRODUCTION OVERVIEW AND INITIAL ASSESSMENT

2.1 THE DATASET

The New York Public Library Rare Book Division holds over 45,000 historical menus. About half of these were collected and curated by Frank E. Buttolph between 1900 and 1921. The menus date from the 1850s to the present and include menus from the restaurant, railroad, and steamship companies, as well as a range of other organizations. Beginning in 2011, menus from the NYPL's collection were digitized and transcribed with the help of thousands of volunteers. Through the NYPL's What's on the Menu? [4] the project, volunteers looked at digitized copies of the menus and typed in the many pieces of information included on each one, such as restaurant names, locations, dishes, prices, and dates.

The detailed fields information about the fields in the file is described as below:

MenuItem.csv
Id: unique id for this menu
menu_page_id: id of the menu page that this item appears.
price: the price of the smallest amount of the item
high_price: the price of the max amount of item
dish_id: menu item referring dish id
created_at: the date when it was created
updated_at: last updated
xpos: x-axis position of the item on the scanned image
ypos: y-axis position of the item on the scanned image

MenuPage.csv
Id: id of the menu page
menu_id: id for menu
page_number: page number of menus
image_id: unique id of the image
full_height: height of the menu
full_width: width of menu
uuid: uniqueid of page/image

Dish.csv
Id: id of dish
name: Name of the dish
description: Description of the dish
menus_appeared: dish appearing on a number of menus
times_appeared: number of times this dish appears in additional section also
first_appeared: year this dish first appeared
last_appeared: year this dish last appeared
lowest_price: lowest price of the dish
highest_price: the highest price of the dish

Menu.csv
Id: id for this menu
name: name on menu/restaurant or blank
sponsor: sponsor is the name of the restaurant:
event: name of the meal or the event the menu was created for
venue: location where the food is served
place: includes city/state/country/address or name of venue
physical_description: paper stock, dimension, color, design of the menu
occasion: special occasion, holiday, daily or blank
notes: additional details about the menu
call_number: number within NYPL collection
keywords: keywords on menu
language: language the menu is printed in
date: date menu was collected
location: menu used location
location_type: type of location
currency: money type charged
currency_symbol: symbol for currency
status: digitization of the menu (complete/under review)
page_count: number of pages on the menu
dish_count: number of dishes on the menu

2.2 USE CASES

U1: We can clean the data to answer the below use cases.

- How the food preferences changed over time based on the event, years, and location in the menu? After cleaning, we can ask this question, and the solution can be visualized as a trend analysis.
- Is there any specific type of dish whose price change has been greater than or less than the average change over time?
- How has the median price of dishes changed over time? Are there particular types of dishes whose price changes have been greater than or less than the average change over time?

U0: As the data set is really messy for any practical use case, but still we can use the data to gather some statistics which can be done without cleaning the data. Some of those use cases can be as below.

- We can get the information about the popular dishes based on how many times it appeared in menus using the dish dataset.
- What is the max/min of height and width of the Menu structure of previous years using the menu Page dataset?
- Using the help of Menu Page and Menu item together we can get information about a particular menu item as to when and where it appeared on a menu.

U2: What we cannot answer even after data cleaning and steers our thoughts towards the arguments that data cleaning should be done with a purpose in mind.

- How we would predict anything about a dish's price based on its name or description as it requires modeling
- There's been some work on how the words used in advertisements for potato chips are reflective of their price: is that also true of the words used in the name of the food?
- Based on French or Italian words in the name, does the price of the dish become expensive?

The dataset is quite messy and needs to be organized. For example, the date columns seem full of repeat or missing dates. However, some files are cleaner than others. We are not choosing to clean MenuPage.csv, as it is clean enough. As because we are getting towards the final outcome as trend analysis, we want to remove the data with outliers. We observed that most of the data has been recorded between 1850 to 2014 so we will remove the data for the rest of the dates.

3. DATA CLEANING STEPS

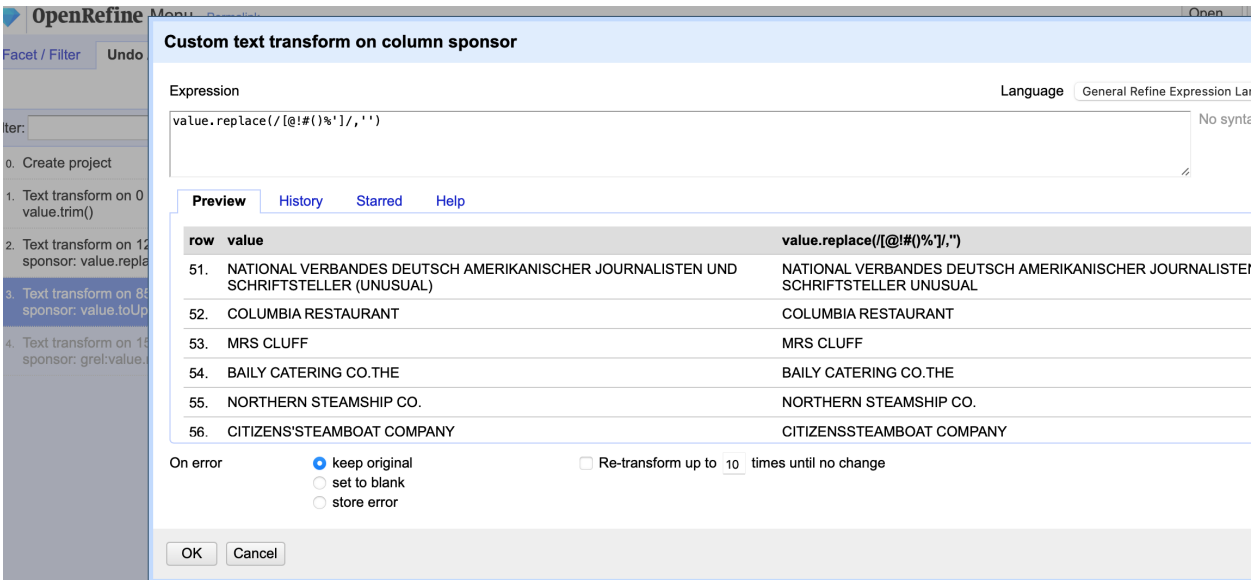
Based on the initial view of the data in the four files we can do the following sort of cleaning for all the files separately and then we can use SQL lite to create tables and put constraints:

Menu.csv:

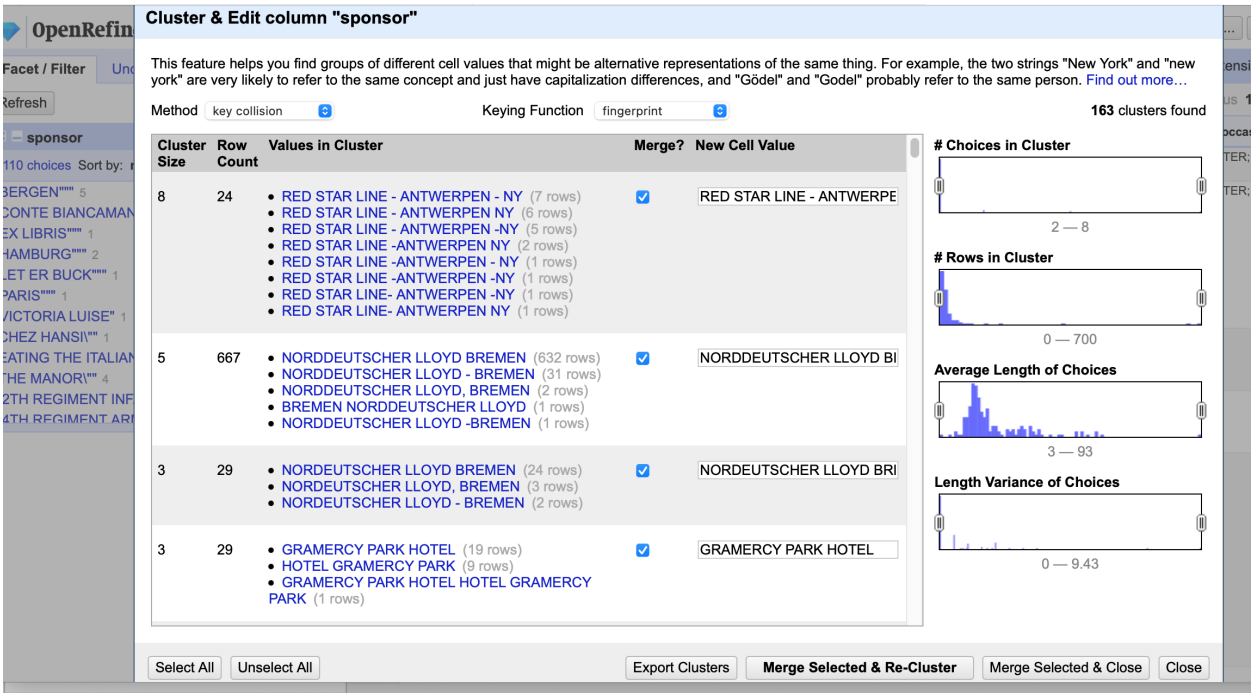
Generally, for any data cleaning activity for the file below things can be done which will make data suitable for conveying information and making it more presentable.

- The sponsor column should trim leading and trailing white spaces and replace consecutive spaces.
- Convert columns to uppercase
- Remove special characters
- Replace; with space when it is inside the text of column value and may create separate columns with values that might need to be updated in ER. Saw the value in the physical description
- Convert similar values of the important columns and making in a standardized format using facets
- Make standard date formats
- Remove null columns after the processing

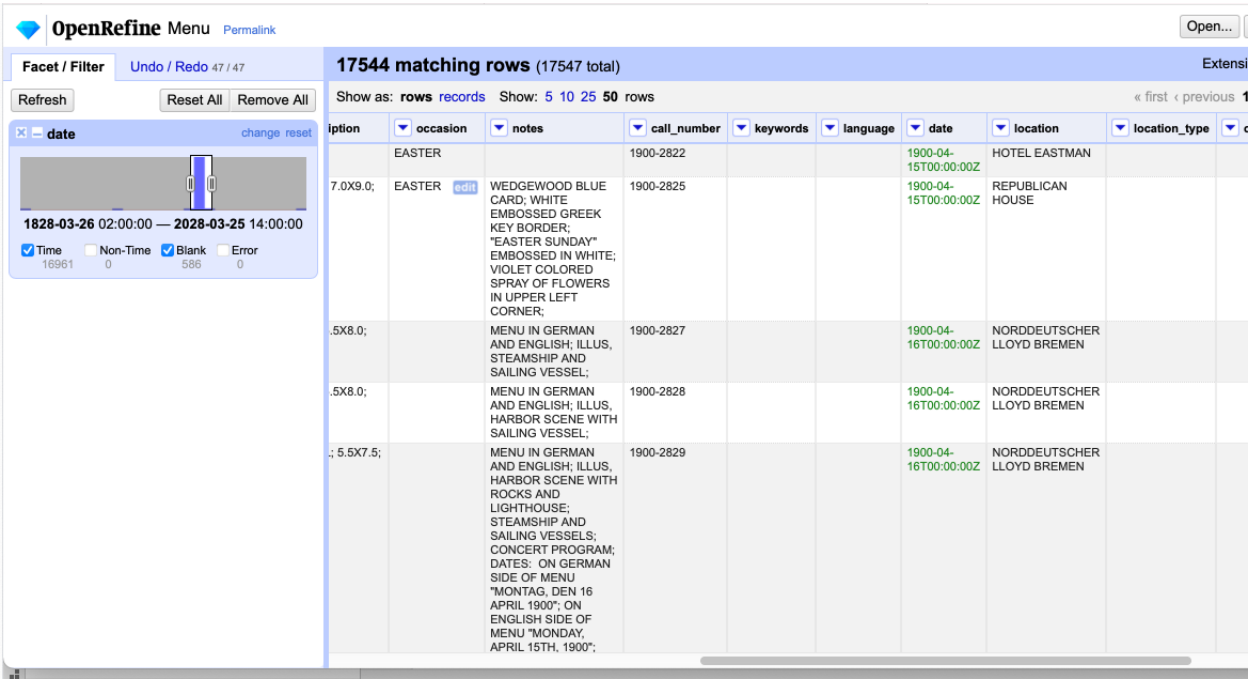
To be more specific we observed the below data issues in the sponsor column with special characters leading and trailing white spaces.



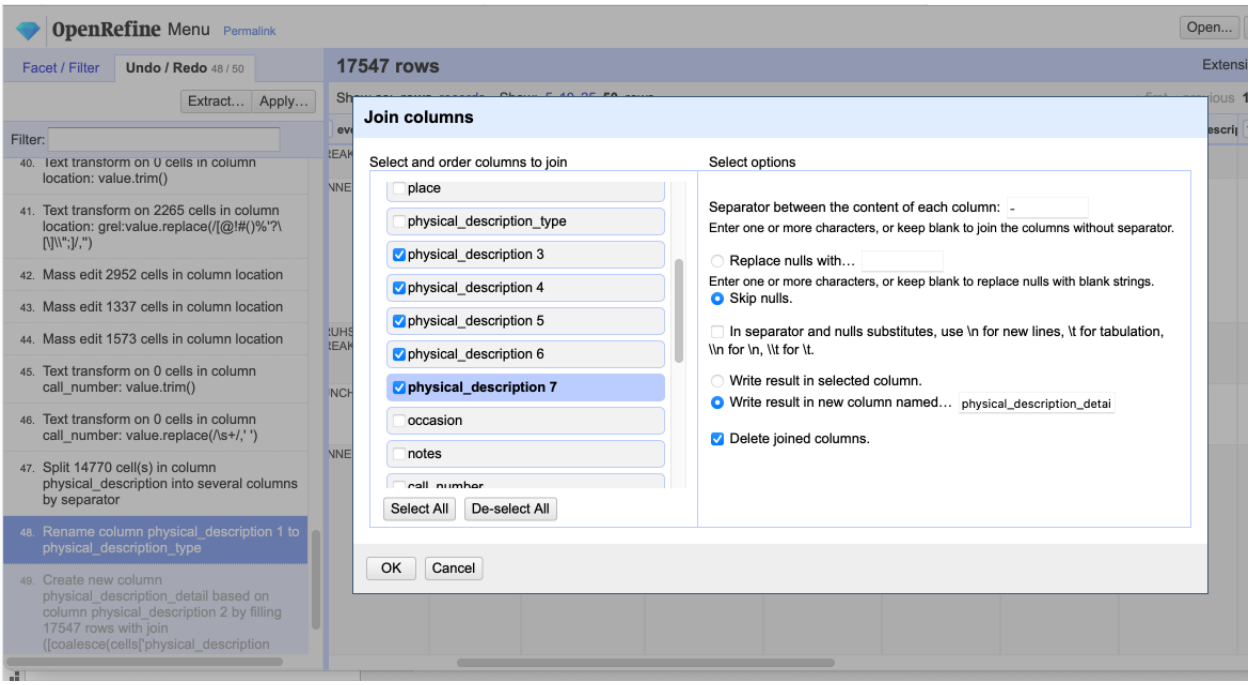
As we see many data items having multiple variations of the same fields, we will be using OpenRefine different clustering facets to standardize sponsor for display purposes.



We are going to do similar processing for columns: event, venue, name, place, location, occasion. For the date column, we will remove outliers as below.



For the physical description column, we will split it into multiple columns to deduce more meaningful information about the menu sizes and structures.



We will not do any kind of cleaning activities for the below columns excepts uppercases and removing spaces making a suitable for viewing in the trend analysis tying to our use case U1:

ID, Keywords, Language, Status

No changes are needed for page count and dish count.

MenuPage.csv:

looks to be clean. As per the initial set up looks like it might not need cleaning or a bit of cleaning.

MenuItem.csv:

Convert all the dates to a standard format for columns ***created_at*** and ***updated_at***.
Check for the special characters and remove if any.

Dish.csv:

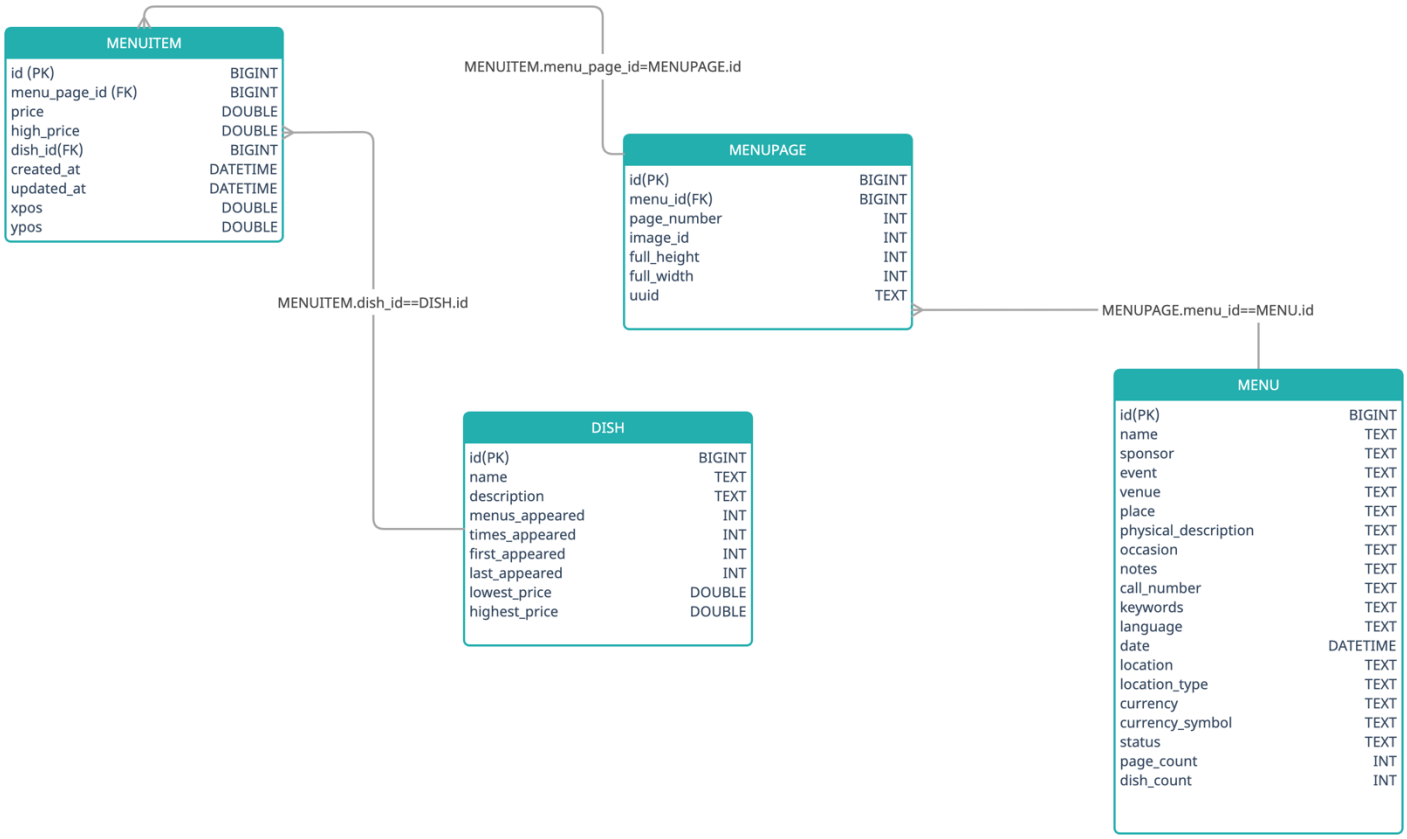
For columns ***name*** and ***description*** using key collision to cluster values and not nearest neighbor because it was computationally expensive. Check for the special characters and remove them while cleaning.

After we clean the data, we will run through the integrity constraints to check the data for the fields in the table.
We will use SQLite [3] for this. We can do a second round of cleanup for the data missing from OpenRefine [1].

4. SCHEMA

4.1 ER DIAGRAM

Below is the ER diagram of the files.



4.2 INTEGRITY CONSTRAINT

The checks that we will perform on the data for all the files will be as below:

1. Not null checks
2. Dates within a range
3. Primary keys are not null and empty as well as unique
4. Null foreign keys will be deleted
5. Low price should below the highest price
6. None of the columns should have only NULL as the value
7. Update date should be greater than the created date.

5. CONCLUSION

We will discuss our observations and provide more information in phase II.

CONTRIBUTIONS

Cleaning raw data files using OpenRefine: Suruchi Sinha, Pradeep Sharma
Creating tables and integrity constraints checks: Suruchi Sinha, Upul Gunasena, Pradeep Sharma
YesWorkflow diagrams: Pradeep Sharma, Upul Gunasena
Github uploadings: Suruchi Sinha, Pradeep Sharma, Upul Gunasena

ACKNOWLEDGMENT

We would like to thank Prof. Bertram Ludaescher and TAs for their guidance to work on this project.

REFERENCES

[1] "Openrefine: A free, open-source, powerful tool for working with messy data," http://openrefine.org/.
[2] B. L. et al, "Yesworkflow," https://github.com/yesworkflow-org/.
[3] "Sqlite," https://www.sqlite.org/.
[4] N. Y. P. Library, "What's on the menu?" http://menus.nypl.org/
[5] "Frank e. buttolph," https://en.wikipedia.org/wiki/Frank E.Buttolph.