Final Project: End-to-End Data Cleaning Workflow

Mark Berman, Sanjay Kumar, Jasdeep Duggal

[**​1)​ Introduction and Overview**](#_hsiqbmqd4my8) **2**

[**​2)​ Initial Assessment of the dataset**](#_bbv89j1dh2mv) **3**

[**​3)​ Data Cleaning methods and process**](#_bbv89j1dh2mv) **3**

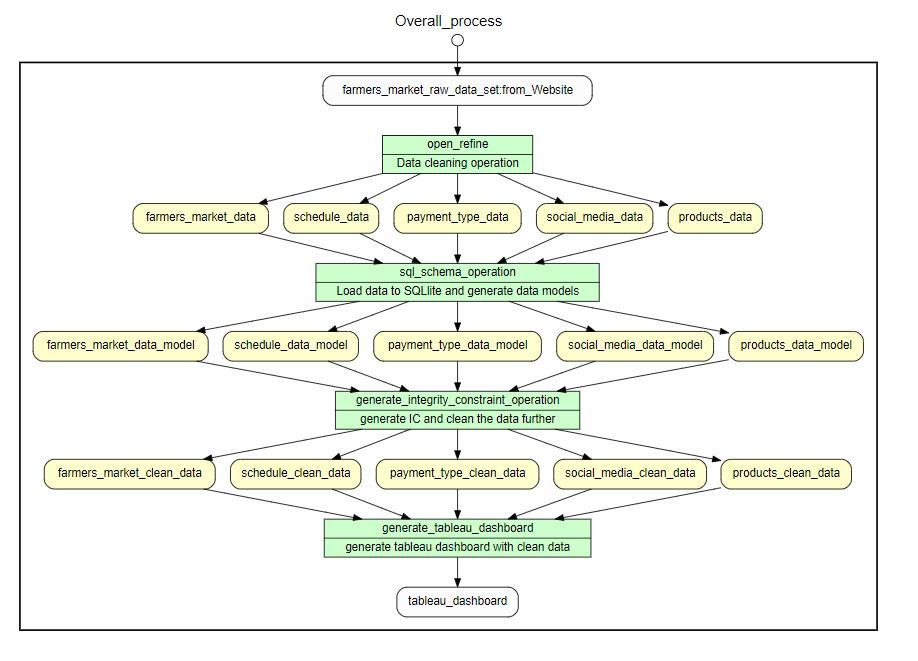
[**​4)​ Data Cleaning Results**](#_bbv89j1dh2mv) **3**

[**​5)​ Conclusions and Future Work**](#_bbv89j1dh2mv) **3**

# ​1)​ Introduction and Overview

Jasdeep Duggal (pending 4C by me)

Data Cleaning

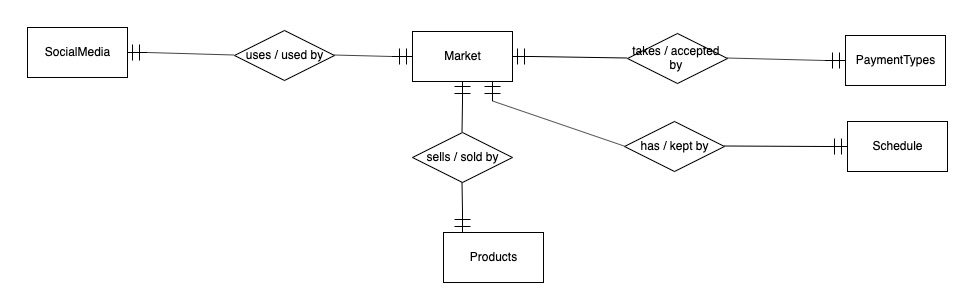


# ​2)​ Initial Assessment of the dataset

The structure of Farmers Market dataset at first glance is trivial -- a single file with 59 columns. However, the dataset can be viewed as five distinct subject areas given the use cases described by this project.

1. Market - contains the each market’s name, address and geo-location.
2. Payment Types - contains the credit based payment types accepted by each market.
3. Products - contains the product types each market sells.
4. Schedule - contains the dates and times when each market is open for business.
5. Social Media - contains the URIs for each market’s social media presence.

The ERD below shows the structure and relationship between the five subject areas.

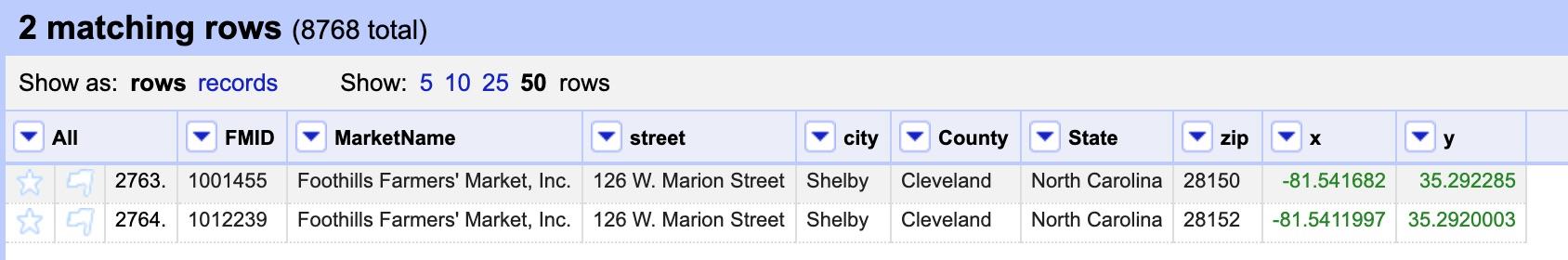


## 

## Observed Data Quality Issues

### Market Subject Area

* There are multiple instances where it appears that the same market appears in the dataset more than once. The following example illustrates this observation. The two market names are almost identical and their longitude and latitude are the same.



* There are multiple instances where geo-location values (e.g., street, city, longitude and latitude) are missing for a market. It is not possible to locate markets for a given geo-location without this data. The following example illustrates this observation.



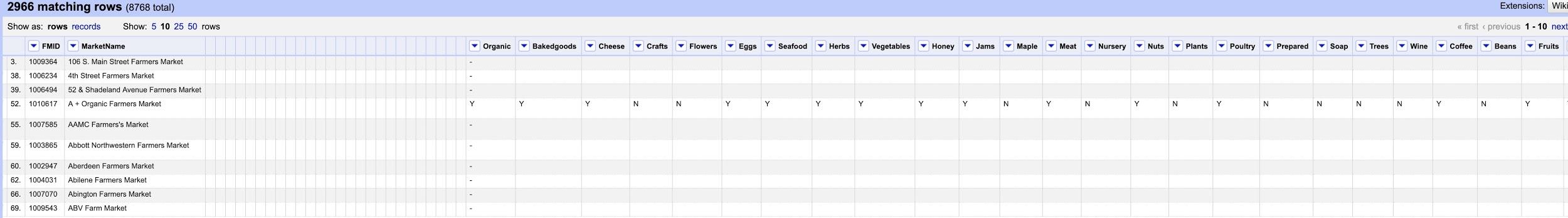
### Payment Types Subject Area

* No observed data quality issues

## 

### Products Subject Area

* There are multiple instances where products sold data is missing for a market. One cannot infer that a missing value for a product type means that type of product is not sold by that market. The following example illustrates this observation.



### Schedule Subject Area

* There are multiple instances where schedule data is missing for a market. Publishing data about a market without the market’s date and times of availability is not very helpful to potential customers. The following example illustrates this point.

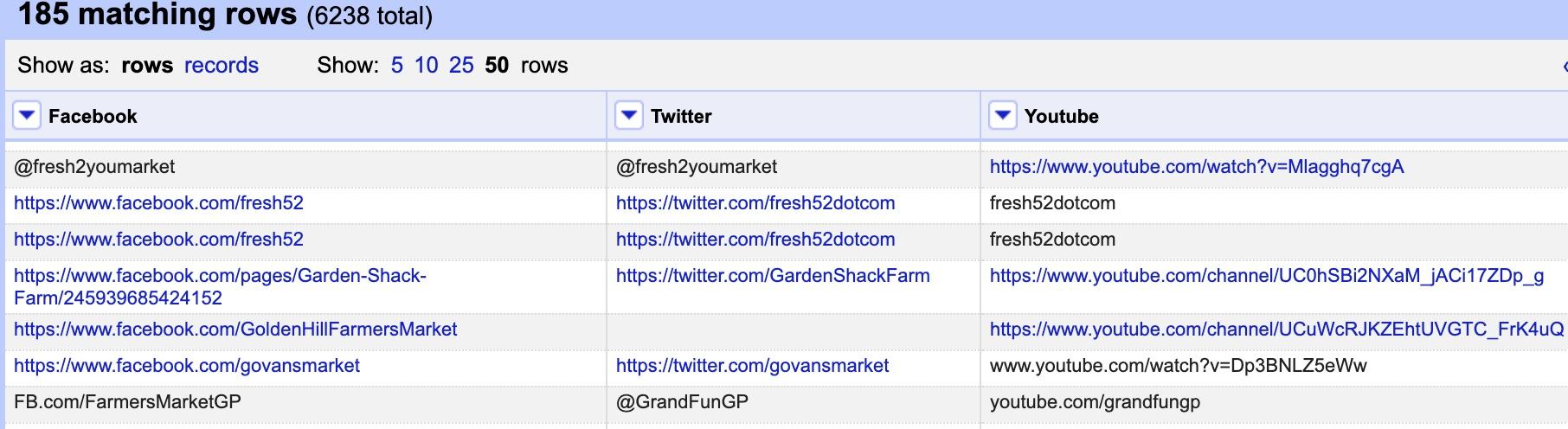


### 

### 

### Social Media Subject Area

* There are multiple instances where Facebook, Twitter and Youtube URIs are inconsistent or invalid. Inconsistency makes it more difficult to programmatically validate the correctness of URIs. Invalid URIs are not helpful to potential customers.



* It should be noted the use cases for this project only depend on the presence or absence of a URI and not on URI consistency or correctness.

# ​3)​ Data Cleaning methods and process

## OpenRefine Based Data Cleansing

The overall data cleaning process is summarized in a YewWorkflow diagram at the following link.

<https://github.com/markcb2/cs513_datacleansing/blob/master/overall.JPG> .

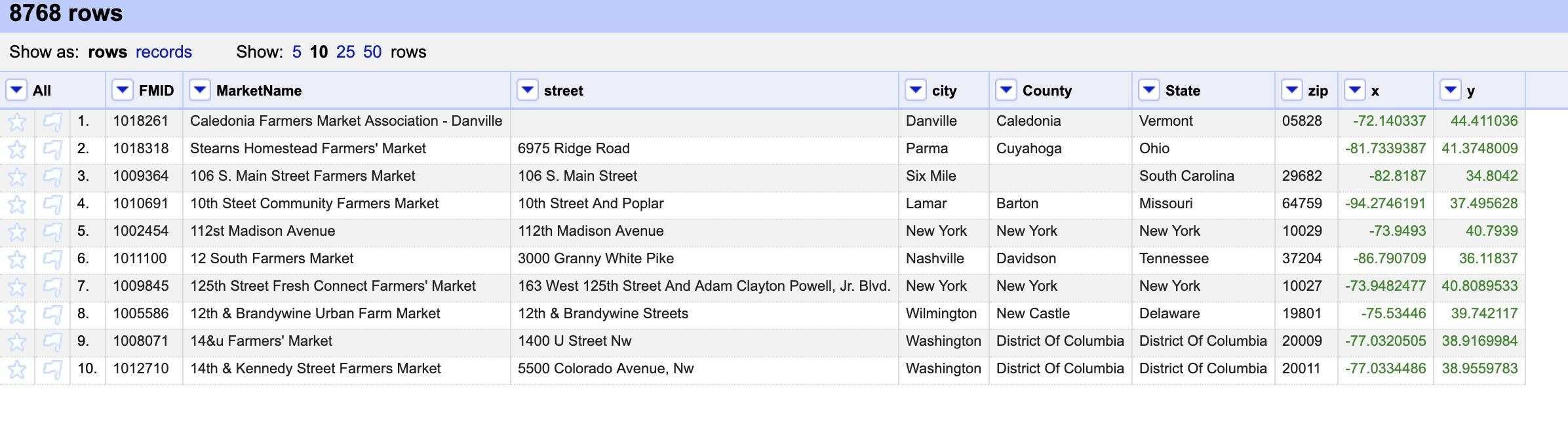
What follows is a summarization of the OpenRefine based data cleansing operations for each of the five subject areas.

### 

### Market Subject Area

|  |  |
| --- | --- |
| **Data Cleansing Operation** | **Impacted Columns** |
| Remove columns | Website, Facebook, Twitter, YouTube, OtherMedia, Remove, Credit, WIC, WICcash, SFMNP, SNAP, Season1Date, Season1Time, Season2Date, Season2Time, Season3Date, Season3Time, Season4Date, Season4Time, Organic, Bakedgoods, Cheese, Crafts, Flowers, Eggs, Seafood, Herbs, Vegetables, Honey, Jams, Maple, Meat, Nursery, Nuts, Plants, Poultry, Prepared, Soap, Trees, Wine, Coffee, Beans, Fruits, Grains, Juices, Mushrooms, PetFood, Tofu, WildHarvested, updateTime |
| Trim and collapse consecutive whitespace | FMID, MarketName, street, city, County, State, zip, x, y |
| Convert to Title Case | MarketName, street, city, County, State, zip |
| Convert to Number | x, y |

The following screen capture shows the Market subject area after the above OpenRefine based data cleansing operations have been performed.



The generated post-refinement OpenRefine recipe for this subject area is viewable at the following link.

<https://github.com/markcb2/cs513_datacleansing/blob/master/farmers_market_base_table_history.txt>

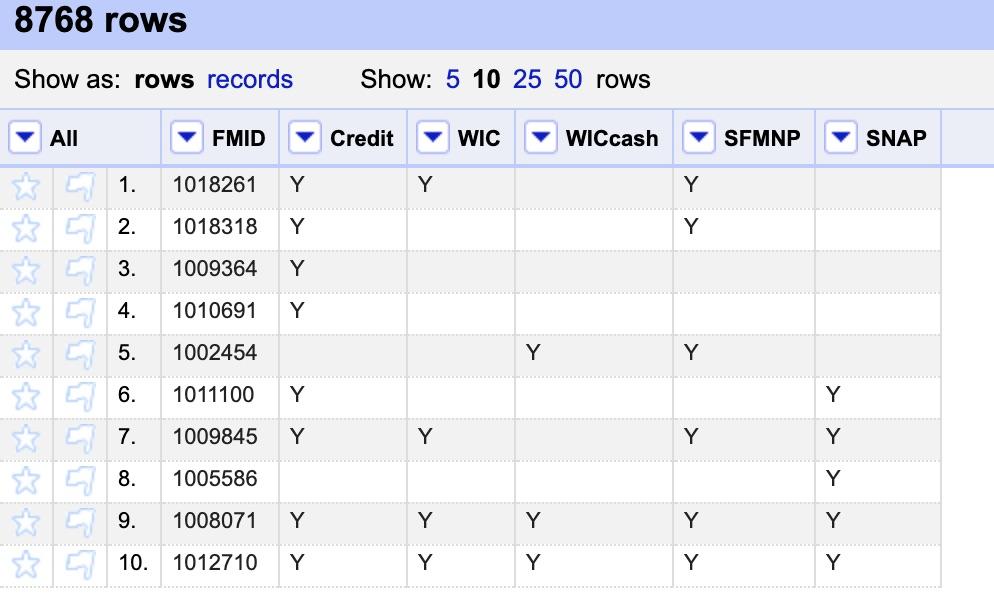
The YesWorkflow linear and parallel diagrams, and YW script that produced them for this subject area are viewable at the following link.

<https://github.com/markcb2/cs513_datacleansing/tree/master/yw_farmers_market_base_artifacts>

### Payment Types Subject Area

|  |  |
| --- | --- |
| **Data Cleansing Operation** | **Impacted Columns** |
| Remove columns | MarketName, street, city, County, State, zip, x, y, Website, Facebook, Twitter, YouTube, OtherMedia, Remove, Season1Date, Season1Time, Season2Date, Season2Time, Season3Date, Season3Time, Season4Date, Season4Time, Organic, Bakedgoods, Cheese, Crafts, Flowers, Eggs, Seafood, Herbs, Vegetables, Honey, Jams, Maple, Meat, Nursery, Nuts, Plants, Poultry, Prepared, Soap, Trees, Wine, Coffee, Beans, Fruits, Grains, Juices, Mushrooms, PetFood, Tofu, WildHarvested, updateTime |
| Trim and collapse consecutive whitespace | FMID, Credit, WIC, WICcash, SFMNP, SNAP |
| Convert to Upper Case | Credit, WIC, WICcash, SFMNP, SNAP |
| Replace N with empty string  Done to facilitate calculation of count metrics for the Tableau visualizations. | Credit, WIC, WICcash, SFMNP, SNAP |

The following screen capture shows the Payment Types subject area after the above OpenRefine based data cleansing operations have been performed.



The generated post-refinement OpenRefine recipe for this subject area is viewable at the following link.

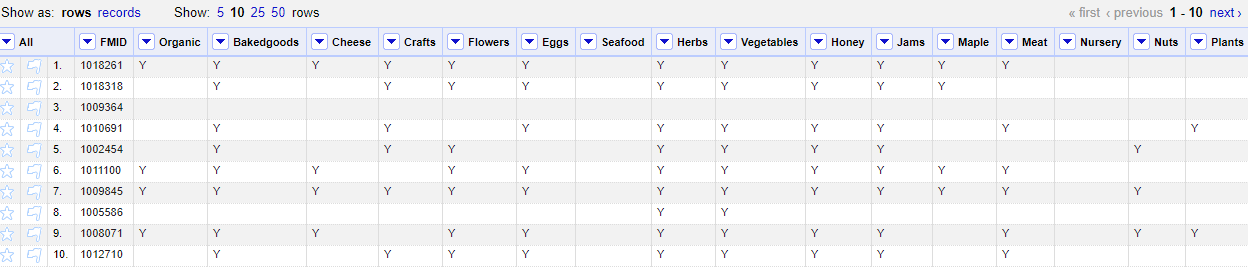
<https://github.com/markcb2/cs513_datacleansing/blob/master/paymentType_history.txt>

The YesWorkflow linear and parallel diagrams, and YW script that produced them for this subject area are viewable at the following link.

<https://github.com/markcb2/cs513_datacleansing/tree/master/yw_paymentType_artifacts>

### Products Subject Area

Cleaning of products type information from the dataset was straight forward. The objective of forming a data model for better data visualization was achieved by reordering and removal of the columns except for the products type column and farmers market unique identifier (FMID) column. Following is the screenshot of the products openrefine project after the data cleaning:



|  |  |
| --- | --- |
| **Data Cleansing Operation** | **Impacted Columns** |
| Reorder Columns | FMID, Organic, Bakedgoods, Cheese, Crafts, Flowers, Eggs, Seafood, Herbs, Vegetables, Honey, Jams, Maple, Meat, Nursery, Nuts, Plants, Poultry, Prepared, Soap, Trees, Wine, Coffee, Beans, Fruits, Grains, Juices, Mushrooms, PetFood, Tofu, WildHarvested |
| Removed Columns | MarketName, Website, Facebook, Twitter, YouTube, OtherMedia, street, city, County, State, zip, Season1Date, Season1Time, Season2Date, Season2Time, Season3Date, Season3Time, Season4Date, Season4Time, x, y, Location, SFMNP |
| Trim and collapse consecutive whitespace | All Columns |
|  |  |
|  |  |

The generated post-refinement OpenRefine recipe for this subject area is viewable at the following link.

<https://github.com/markcb2/cs513_datacleansing/blob/master/productType_history.txt> .

The YesWorkflow linear and parallel diagrams, and YW script that produced them for this subject area are viewable at the following link.

<https://github.com/markcb2/cs513_datacleansing/tree/master/yw_product_artifacts>

### Schedule Subject Area

Refining and extracting schedule information for each farmers market was challenging with open refine. After a series of reordering and extraction of opening and closing seasons, we were able to provide meaningful data to create a data model in SQL.

Following is the screenshot of the schedule openrefine project after the data cleaning:



|  |  |
| --- | --- |
| **Data Cleansing Operation** | **Impacted Columns** |
| Reorder Columns | FMID, |
| Removed columns | MarketName, Website, Facebook, Twitter, YouTube, OtherMedia, street, city, County, State, zip, Season1Date, Season1Time, Season2Date, Season2Time, Season3Date, Season3Time, Season4Date, Season4Time, x, y, Location, SFMNP, All product types |
| New Columns | Season, seasonOpenning, seasonClosing, SeasonTime |
| Trim and collapse consecutive whitespace | All columns |
|  |  |

The generated post-refinement OpenRefine recipe for this subject area is viewable at the following link.

<https://github.com/markcb2/cs513_datacleansing/blob/master/schedule_history.txt>

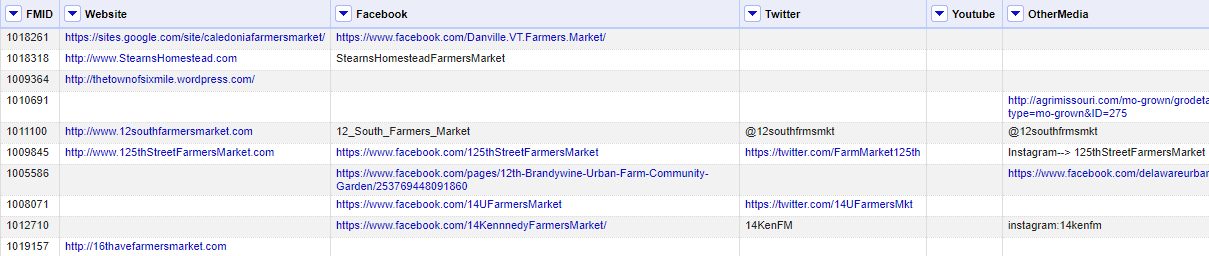
The YesWorkflow linear and parallel diagrams, and YW script that produced them for this subject area are viewable at the following link.

<https://github.com/markcb2/cs513_datacleansing/tree/master/yw_schedule_artifacts>

### Social Media Subject Area

Cleaning of social media information from the initial dataset was straight forward too. The objective of forming a data model for better data visualization was achieved by reordering and removal of the columns except for the columns containing social media information and farmers market unique identifier (FMID) column. As part of the cleaning, rows with no social media data were removed. Post refinement, it was found that there are **6238** farmers market which has a social media presence across USA.

Following is the screenshot of the social media openrefine project after the data cleaning:



|  |  |
| --- | --- |
| **Data Cleansing Operation** | **Impacted Columns** |
| Reorder Columns | FMID,Website, Facebook, Twitter, YouTube, OtherMedia |
| Removed columns | MarketName, street, city, County, State, zip, Season1Date, Season1Time, Season2Date, Season2Time, Season3Date, Season3Time, Season4Date, Season4Time, x, y, Location, SFMNP, All product types |
| Trim and collapse consecutive whitespace | All columns |
|  |  |

The generated post-refinement OpenRefine recipe for this subject area is viewable at the following link.

<https://github.com/markcb2/cs513_datacleansing/blob/master/socialMedia_history.txt>

The YesWorkflow linear and parallel diagrams, and YW script that produced them for this subject area are viewable at the following link.

<https://github.com/markcb2/cs513_datacleansing/tree/master/yw_socialMedia_artifacts>

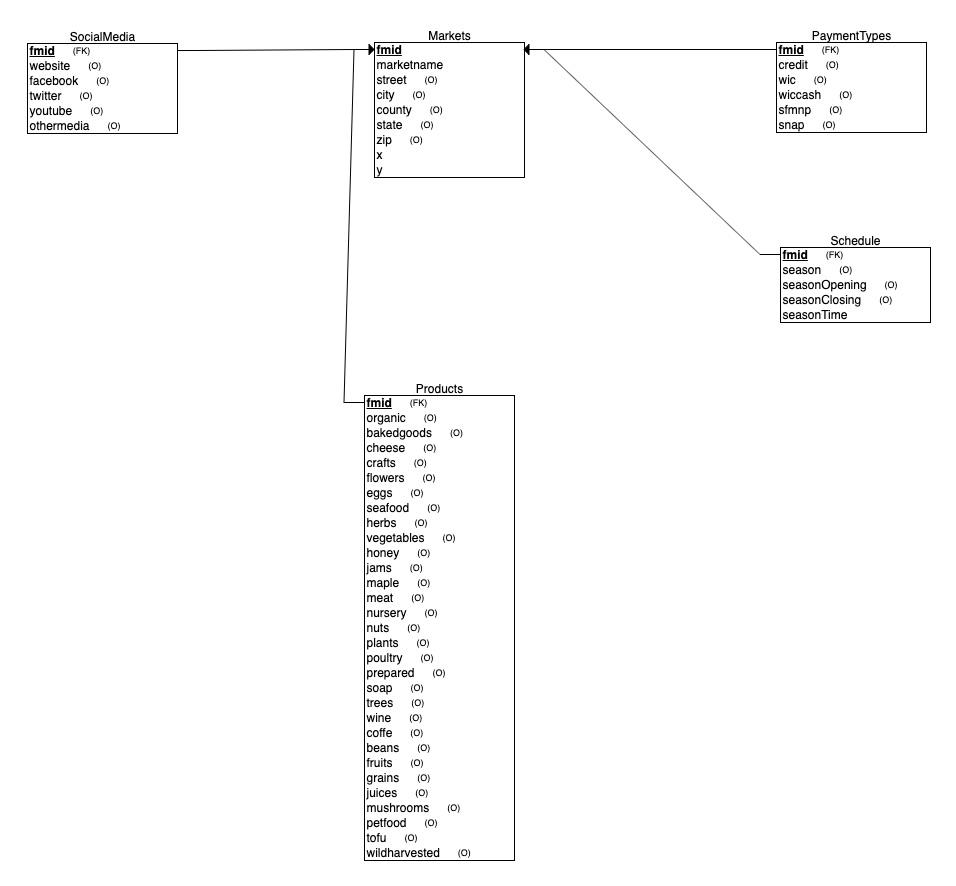
## OpenRefine Data Cleaning Limitations and Alternative Approaches

OpenRefine is best suited for “column at a time” data cleansing. OpenRefine is not ideal for finding duplicate records or cases where multiple columns in a record need to be evaluated. Seventeen different integrity constraint checks were performed using SQL queries against a database schema implemented using SQLite. There were seven integrity constraint violations that were identified. These integrity constraint violations were remediated using SQL ***“Create Table <TableName> as select”*** statements. The integrity constraint checks and the remediation steps are described in the **Data Cleaning Results** section.

# ​4)​ Data Cleaning Results

## Relational Schema and Integrity Constraint Remediation.

OpenRefine was used to generate the SQL schema. The SQL schema along with its cleansed data was exported from OpenRefine and loaded into SQLite. A relational data diagram of the schema and the SQL commands used to generate the schema are shown below.



|  |  |
| --- | --- |
| DROP TABLE IF EXISTS markets;  CREATE TABLE markets (  FMID VARCHAR(10) NOT NULL,  MarketName VARCHAR(100) NOT NULL,  street VARCHAR(100) NULL,  city VARCHAR(50) NULL,  County VARCHAR(50) NULL,  State VARCHAR(50) NULL,  zip VARCHAR(10) NULL,  x NUMERIC(12) NULL,  y NUMERIC(12) NULL  ); | DROP TABLE IF EXISTS paymentTypes;  CREATE TABLE paymentTypes (  FMID VARCHAR(10) NOT NULL,  Credit VARCHAR(1) NULL,  WIC VARCHAR(1) NULL,  WICcash VARCHAR(1) NULL,  SFMNP VARCHAR(1) NULL,  SNAP VARCHAR(1) NULL  ); |
| DROP TABLE IF EXISTS products;  CREATE TABLE products (  FMID VARCHAR(10) NOT NULL,  Organic VARCHAR(1) NULL,  Bakedgoods VARCHAR(1) NULL,  Cheese VARCHAR(1) NULL,  Crafts VARCHAR(1) NULL,  Flowers VARCHAR(1) NULL,  Eggs VARCHAR(1) NULL,  Seafood VARCHAR(1) NULL,  Herbs VARCHAR(1) NULL,  Vegetables VARCHAR(1) NULL,  Honey VARCHAR(1) NULL,  Jams VARCHAR(1) NULL,  Maple VARCHAR(1) NULL,  Meat VARCHAR(1) NULL,  Nursery VARCHAR(1) NULL,  Nuts VARCHAR(1) NULL,  Plants VARCHAR(1) NULL,  Poultry VARCHAR(1) NULL,  Prepared VARCHAR(1) NULL,  Soap VARCHAR(1) NULL,  Trees VARCHAR(1) NULL,  Wine VARCHAR(1) NULL,  Coffee VARCHAR(1) NULL,  Beans VARCHAR(1) NULL,  Fruits VARCHAR(1) NULL,  Grains VARCHAR(1) NULL,  Juices VARCHAR(1) NULL,  Mushrooms VARCHAR(1) NULL,  PetFood VARCHAR(1) NULL,  Tofu VARCHAR(1) NULL,  WildHarvested VARCHAR(1) NULL  ); | DROP TABLE IF EXISTS socialMedia;  CREATE TABLE socialMedia (  FMID VARCHAR(10) NOT NULL,  Website VARCHAR(256) NULL,  Facebook VARCHAR(256) NULL,  Twitter VARCHAR(256) NULL,  Youtube VARCHAR(256) NULL,  OtherMedia VARCHAR(256) NULL  ); |
| DROP TABLE IF EXISTS schedule;  CREATE TABLE schedule (  FMID VARCHAR(10) NOT NULL,  season VARCHAR(50) NULL,  seasonOpenning VARCHAR(50) NULL,  seasonClosing VARCHAR(50) NULL,  seasonTime VARCHAR(100) NULL  ); |  |

The following table lists the seventeen integrity constraint checks that were performed and the seven places where there were violations. The actual integrity constraint SQL queries and the result sets returned by these queries are available at the following link.

<https://github.com/markcb2/cs513_datacleansing/blob/master/ic_queries.sql>

|  |  |
| --- | --- |
| **Integrity Constraint Check** | **Violation Occurrences** |
| IC1: Markets records where there are at least two rows having the same ID, but different column values. | No violations. |
| IC2: Missing market names in the Markets table. | No violations. |
| IC3: Duplicate markets in the Markets table. | 11 violations. |
| IC4: Records with Invalid US longitude or latitude in the Markets table. Cannot execute the geo-location use cases on those records that violate this constraint. | No violation. |
| IC5: Records with missing longitude or latitude in the Markets table. Cannot execute the geo-location use cases on those records that violate this constraint. | 28 violations. |
| IC6: Social Media records where there are at least two rows having the same ID, but different column values. | No violations. |
| IC7: Invalid websites in the Social Media table. Valid web sites must have at least one character between "http(s)://" and the "." and at least two characters after the dot. | No violations. |
| IC8: Payment Type records where there are at least two rows having the same ID but different column values. | No violations. |
| IC9: Records with invalid payment type indicator values (valid values are 'Y' or null) in the Payment Types table. (‘N’ values are converted to empty strings as part of OpenRefine based data cleansing.) | No violations. |
| IC10: Schedule records where there are at least two rows having the same ID, but different column values. | No violations. |
| IC11: Schedule records where there is no or incomplete schedule information. Cannot execute the use case that displays schedule information for those records that violate this constraint. | 3205 violations |
| IC12: Product records where there are at least two rows having the same ID, but different column values. | No violations. |
| IC13: Product records with no product information. | No violations. |
| IC14: Social Media records where its foreign key not found in Markets table after the Markets table was purged of integrity constraints. | 37 violations. |
| IC15: Payment Types records where their foreign keys are not found in the Market table after the Markets table was purged of integrity constraints | 39 violations. |
| IC16: Product records where their foreign keys are not found in the Market tables after the Markets table was purged of integrity constraints. | 39 violations. |
| IC17: Schedule records where their foreign keys are not found in the Markets table after the Markets table was purged of integrity constraints. | 39 violations. |

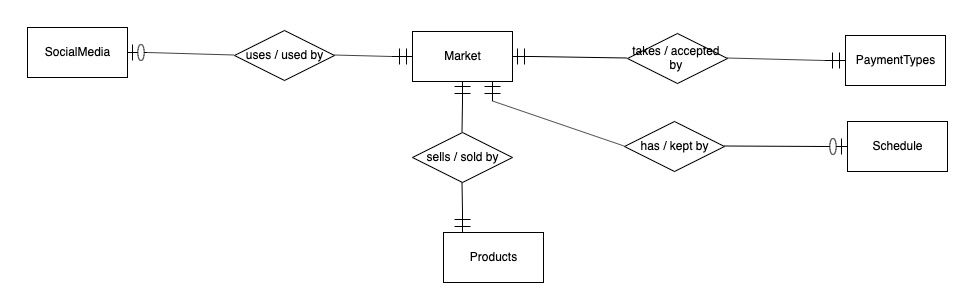
The following table shows the record count for all five subject areas before and after integrity constraint remediation.

|  |  |  |
| --- | --- | --- |
|  | Records Count Before IC Remediation | Record Count After IC Remediation |
| Markets | 8768 | 8729 |
| Payment Types | 8768 | 8729 |
| Products | 8768 | 8729 |
| Schedule | 8768 | 5551 |
| Social Media | 6238 | 6201 |

The **“Create table <TableName> as select”** SQL statements used to remediate the integrity constraint violations are available at the following link.

<https://github.com/markcb2/cs513_datacleansing/blob/master/ic_queries.sql>

The combination of OpenRefine based data cleansing and integrity constraint remediation changes the cardinality of from 1:1 to 0:1 on two of the relationships of the subject area ERD. The following diagram illustrates the cardinality changes of the Markets to Schedule relationship and the Markets to Social Media relationship.



# ​5)​ Conclusions and Future Work

# Sanjay