NYC Restaurants Data Inspection

Marshall Shen
April, 29th, 2014

1. Introduction

NYC is full of restaurants, but are all the restaurants safe to eat at? We took a dive into NYC restaurants open data and retrieve indications of which restaurants are more likely give your stomach an unpleasant experience.

What did we find? Turns out large-sized restaurants in Manhattan are more likely to have food incidents than small-sized ones, and median-sized restaurants in NY have more food incidents around 2012 than small-sized ones.

All of the claims above are purely based on present data. Let’s see how we reached the implications.

Using NYC open data, we first find the datasets of interests: dataset that describes the sidewalk cafes in the tri-state area, in addition to the dataset of food poisoning from Department of Health and Mental Hygiene.

Original dataset
There are two original datasets extracted from NYC open data:
cafe.csv: https://data.cityofnewyork.us/Business/Sidewalk-Cafes/6k68-kc8u, details on NYC restaurants (restaurant address, square footage, enclosed or unenclosed, etc.)

food_poison.csv: https://data.cityofnewyork.us/Social-Services/food-poisoning/gjkf-etq5, food poison records of NYC since 2012, note that food incidents transpire in various occasion, such as dining out in a restaurant, participating in a food catering event, etc.

Given the two datasets of interest, we then process and combine the datasets to INTEGRATED-DATASET, this allows us to 1) standardize data values across the files 2) have one file from which we can retrieve indication rules from.

2. Retrieve INTEGRATED-DATASET

Here we discuss how we get the INTEGRATED-DATASETS in details, some sample rows of INTEGRATED-DATASET looks like the following:

Restaurant /Bar/Deli/Bakery	occasional incidents	2 AVENUE	fairly recent	MANHATTAN	Enclosed	large	2 AVENUE	Street Match
Restaurant /Bar/Deli/Bakery	frequent incidents	BROADWAY	not recent	MANHATTAN	Enclosed	arge	BROADWAY	Address Match

Let's start by analyzing the schema of the original datasets and the integrated datasets:

The table below is an overview of INTEGRATED-DATASET schema, followed by more detailed explanation for each field. Note that "standardized" indicates whether the data is further processed upon retrieval. Such process is necessary before we apply association rules because: 1) we need to sanitize out any bad data entries 2) we need to transform continuous attributes into categorical attributes 3) some attributes are time-sensitive and we need to process it dynamically to reflect its factual value based on current time (2014).

INTEGRATED-DATASET attribute	**meaning of the attribute**	**standardized?**	**dependent food_poison.csv attribute**	**dependent cafe.csv attribute**
Location Type	Type of the restaurants	No	Location Type	N/A
Incidents Frequency	How often food incidents happen	Yes	Descriptor	N/A
Street Name	Street on which incidents transpire	No	Street Name	Address Street Name
Incident recentness	How recent was the incident reported	Yes	Date	N/A
Restaurant Size	Restaurant size	Yes	N/A	Lic Area Sq Ft
Borough	Restaurant area	No	Park Borough	No
Street Address	Restaurant street	No	N/A	Street Address

Restaurant layout type	Enclosed / Unenclosed	No	N/A	Sidewalk Cafe Type
Match type	Match Street / Match Address	Yes	incident_address	street address

INTEGRATED-DATASET join operation

The join is performed based on “Street Address” column of cafe.csv and “Incident Address” column of food_poison.csv.

Because the “incident address” column of food_poson.csv sometimes include full address, and sometimes only street name. We categorize the matching using “Match Street” and “Match Address”.

Pseudo-code:

```
    if incident_address == restaurant address then:
        join two rows, with label “address match”
    if incident_address partially match restaurant address then:
        join two rows, with label “street match”
```

INTEGRATED-DATASET.location_type

All possible types of restaurant are: "Soup Kitchen", "Restaurant/Bar/Deli/Bakery", "Food Cart Vendor", "Catering Service".

Restaurant type can be a significant indication factor for food incidents, for instance we can assume that restaurant of type “food cart vendor” is more likely to have food incident than that of “retaurant/bar/deli/bakery”.

INTEGRATED-DATASET.incidents_frequency

It describes how often food incidents happen. ‘Descriptor“ column from food_poison.csv has only two possible values (“1 or 2”, “3 or More”). This attribute is one of the targeted attribute from which we want to draw indication rules from. We further processed incidents frequency using the following:

food_posion.csv	INTEGRATED-DATASET.incidents_frequency
1 or 2 (food poison happened 1 or 2 times)	occasional incidents
3 or More (food poison happened 3 or more times)	frequent incidents

INTEGRATED-DATASET.street_name

The main street on which the restaurant is located. This can be a significant indicator for food incidents, for instance, we may retrieve indication that restaurants on Columbus Ave. appears to have more food incidents reported (It is totally hypothetically, we will let data speaks itself later on)

INTEGRATED-DATASET.restaurant_size
This attribute is retrieved from “Lic Area Sq Ft” column of “cafe.csv”. This categorized attributes includes 3 possible values: “small”, “median”, “large”. We transform the numerical values of square footage into a categorical value by comparing the square footages within the rows: sorted by square footage, the top 30% smallest restaurants are considered small, the next 30% restaurants are considered median, and the rest are considered large.

INTEGRATED-DATASET.incident_recentness
This attribute is retrieved from “Created Date” column of food_poison.csv, the “Created Date” is the date which the food incident case upon the restaurant was first opened. Given a date, we further categorize them as follows:

Date on food_poison.csv	INTEGRATED-DATASET.incident_recentness
2014/01/01 - present	‘recent’
2013/01/01 - 2013/12/31	‘fairly recent’
before 2013/01/01	‘not recent’

INTEGRATED-DATASET.borough
It is the neighborhood in which the restaurant is located. All possible values are: “BRONX”, “BROOKLYN”, “MAHATTAN”, “QUEENS”, “STATEN ISLAND”

This is another targeted attribute from which we want to draw indication rules from, the value is directly retrieved from the “Park Borough” column of food_posion.csv

INTEGRATED-DATASET.street_address
Full street address of the restaurant, it is directly retrieved from “Street Address” column of cafe.csv

INTEGRATED-DATASET.restaurant_layout_type
Directly retrieved from column “Sidewalk Cafe Type” of cafe.csv, there are two possible values: “enclosed” and “unenclosed”. An enclosed area on the public sidewalk in front of the restaurant that is constructed predominantly of light materials such as glass.

3. Extract association rules from INTEGRATED-DATASET

3.1 A-priori algorithm

A general overview of what the algorithm does

Input: a collection of items i_1, i_2, … i_n

Output: 1) a collection of frequent item(s) that are likely to be appear in the dataset

2) a collection of indication rules that specifies “If A, then B” indication rules.

Note that in the context of restaurant data, the items are categorical values of restaurant traits, such as restaurant size (“large” or “small”), borough (“Manhattan”? “Brooklyn”?), and so forth.

The algorithm has two major steps:

1) Figure out frequent itemsets with set size = k, where k in the range of (1, n), and n is the number of individual items in the collection.

2) Check all the frequent itemsets, figure out possible indication rules (“If A, then B”) format. Note that A can be an itemset with size greater than or equal to 1, but B can only an itemset with size 1.

Fetch all the frequent itemsets: get SupportData

We define SupportData as a hash, the key is the tuple of frequent items, the value is the support of the itemsets. Support = (times of itemset appearance) / (total number of data rows), given we have a minimum support value, the algorithm runs the following steps:

- initial step: scan through data, find all the single items that pass minimum support value, call the collection C1
- iterative step: (where k >=1)

  Assign C_(k+1) as a “permutation” from C_k: for each list L_i in C_k, append one more item into C_k that does not originally belong to L_i.

  Iterate through each list C_(k+1), scan data and remove any list L_j in C_(k+1) that doesn’t have minimum support.

  Add elements in C_(k+1) into frequent itemsets.

  Repeat until C_(k+1) is an empty set.

Find all the association rules using SupportData

The next step is to find out all possible association rules using SupportData. We do the following processing for each frequent itemset in SupportData:

- *Given the format that the right hand side can only have one element*, we first generate all possible association rules inside one frequent itemset. For instance, if i_1, i_2, i_3 are present in the dataset, all possible association rules are:

  [i_1, i_2] => i_3, [i_2, i_3] => i_1, [i_1, i_3] => i_2

- Scan through data, find confidence for each possible association rule. For instance, the confidence for [i_1, i_2] => i_3 = support([i_1, i_2, i_3]) / support([i_1, i_2])
- Output all association rules that pass minimum confidence.

3.2 Apply association rules

Given 3343 rows of restaurant data, we treat the dataset as a historical transactions and each row as one transaction. Using association rule, we are trying to draw implications, for example:

Food Cart Vendor, Broadway -> “recent”, “occasional incidents”

Such an association can be interpreted as given a food cart along Broadway, it is more likely that a recent 1 or 2 food incident happened in those type of restaurants.

3.2.1 Adjust minimum support and minimum confidence.

Take a close look at INTEGRATED-DATASET, we notice that the dataset is skewed by restaurant area: there are many more data on restaurants in Manhattan than any other neighborhoods. Therefore we must carefully calibrate minimum support and minimum confidence.

3.2.2 Algorithm run and analysis

Run 1: min_support = 0.8, min_conf =0.8

The result isn’t interesting because it doesn’t draw any interesting indication. “Street Match -> occasional incidents” or “occasional incidents -> Street Match” says that there are lots of data rows of food incidents with address that only includes a street name rather than a full address.

Note that :

```
['MANHATTAN'] support: 0.820520490577
['Restaurant/Bar/Deli/Bakery'] support: 0.815734370326
['Unenclosed'] support: 0.804965599761
```

These frequent itemset confirms that the dataset is not evenly distributed: the majority rows are Manhattan restaurants (82% of data records), the majority of the restaurants are categorized as “Restaurant/Bar/Deli/Bakery” (81% of data records, and other options include “Food Cart Vendor”, “Delivery Service”), and the majority of the restaurants are unenclosed (80% of the data records).

Because we want to expose more interesting patterns, and the data is skewed, our strategy is to lower the support but maintain a high or even higher confidence rate: if the appearance of a certain set of traits is relatively slim but the traits are always tend to appear together, such set of traits are interesting to explore.

Run 2: min_support = 0.5, min_conf =0.8 [INTERESTING FREQUENT ITEMSETS]

Lowering the support gives us more frequent itemsets and rules. The following are a portion of outputs that appear to be interesting:

```
==Frequent itemsets (min_sup=0.5)
['occasional incidents', 'MANHATTAN', 'Unenclosed']         support: 0.615016452288
['not recent', 'Restaurant/Bar/Deli/Bakery', 'MANHATTAN']  support: 0.504935686509
….
==High-confidence association rules (min_conf=0.8) # top 2 results
['not recent', 'MANHATTAN', 'Street Match']) --> ['occasional incidents']
          support: 0.56476218965 confidence: 0.939303482587
['not recent', 'MANHATTAN'])--> ['occasional incidents']
          support: 0.613819922226 confidence: 0.936131386861
```

The frequency items indicate that there are many occasional food incidents took place in Manhattan, and many food incidents in Manhattan are not recent.

The indication rules aren't that interesting: a food incident in Manhattan that is not recent indicates it is an occasional incidents. This indication reflects the nature of the data as 91% of the food poison cases were reported as "occasional".

Run 3: min_support = 0.2, min_conf = 0.8 [INTERESTING ASSOCIATION RULES]

As the minimum support dropped, we see more interesting rules emerge, below are some retrieved interesting patterns:

```
==Interesting High-confidence association rules (min_conf=0.8)
          ['large', 'MANHATTAN'])-->['occasional incidents']
          support: 0.273107986838 confidence: 0.939300411523
          ['smalll', 'MANHATTAN']-->['occasional incidents']
          support: 0.22913550703 confidence: 0.902237926973
          ….
```

The comparison of the two rules above indicates that a large restaurant in Manhattan is more likely to have food incidents happen than a small one (93.9% confidence vs. 90.2% confidence)

```
          ['not recent', 'median']-->['occasional incidents']
          support: 0.237810349985 confidence: 0.914844649022
          ['not recent', 'smalll'] -->['occasional incidents']
          support: 0.220161531558 confidence: 0.904176904177
          ….
```

The comparison of the two rules above indicates that a median restaurant in NYC area has more occasional incidents happen in the past (2 years ago).