Can Yelp Reviews Signal Health Code Violations?

Iryna Nazirbaeva, Michelle Schaffer, Arnab Sengupta, & Max Voelker

Business Understanding

Though cities dedicate considerable funds to inspecting restaurants for food violations, government budgets are ultimately limited; there can be several months between visits to a given restaurant by inspection officials. In Toronto, routine inspections occur between one and three times a year, depending on the “specific type of establishment, the food preparation processes, volume and type of food served and other related criteria.”[1] Restaurants that are determined to be higher risk by the city are subject to more frequent inspections. Our goal is to develop a model that will help the local government predict which food facilities will have violations. By enabling the government to plan inspections more efficiently, public safety issues could potentially be avoided and money could be saved.   The goal of our project was to create a model that will identify restaurants within a given metropolitan area that are likely to have at least one critical food violation, based on the Yelp reviews, star ratings, cuisine, previous violations, and other attributes of the establishment.

Managers of restaurant chains or restaurant conglomerates are also challenged with complying with food safety standards across locations that are geographically dispersed, which increases the likelihood that violations could go undocumented while

[1] Health, Toronto Public. “DineSafe Inspection and Disclosure System - Toronto Public Health.” DineSafe Inspection and Disclosure System - Toronto Public Health, www.toronto.ca/health/dinesafe/system.htm. Accessed 8 May 2017.

impacting customer service and potentially exposing consumers to health risks. Further, the owners of restaurants are often not aware of violations that are either concealed from management, or difficult to notice until an incident occurs. The model produced from this analysis could be used to assist restaurant managers with identifying potential health code concerns before inspections occur based on the observations captured in the Yelp review text, minimizing risk to the business's reputation as well as customer health.

The business problem we researched has been analyzed in previous studies, including a particularly interesting contest where researchers utilized datasets similar to those in our proposal to predict health code violations in Boston, Massachusetts. Several approaches produced model accuracy indexes of over 80%.[2]   The most successful approaches utilized text feature extraction such as ngram extraction to identify relevant phrases in the review text, and combined these features with other restaurant metadata including average Yelp rating, number of "check-ins" by Yelp users, and restaurant cuisine. These studies also utilized advanced preparation and massaging of the underlying data by removing reviews that may have contained some other bias such as an outlier poor experience at a restaurant, or even fake reviews either meant to improve a restaurant's overall review sentiment and star rating, or to damage an establishment's reputation.

[2] Announcing the Results of Our Keeping It Fresh Competition." DrivenData, 6 Nov. 2016, blog.drivendata.org/2015/11/06/keeping-it-fresh-results/.

Data Understanding

To develop a predictive model, our team merged two datasets that contain historical inspection and review information:

1) Yelp reviews and restaurant metadata for food facilities in Toronto

https://www.yelp.com/dataset_challenge/dataset

This dataset contains selected information about restaurants such as reviews, star ratings, addresses, etc. (Exhibit A). The data is provided in JSON format and was created by Yelp Inc. for the purposes of a data science competition.. The data contains the review information provided by each user (such as a review itself and star ratings), attributes about each establishment such as whether or not alcohol is available, the restaurant’s ambiance etc.; and other characteristics such as the type of cuisine. The review period in this dataset was all available data from January 2014 to date.

2) Food inspection violations in the city of Toronto (Exhibit B)

http://www.toronto.ca/health/dinesafe/system.htm

The dataset contains records of restaurant inspections for the city of Toronto from April 2015 through March 2017. Each inspection can contain many infractions, which are rated on a low to critical scale. The dataset contains records for each inspection event regardless of whether or not any infractions were identified.

Data Preparation

First we built our data science environment infrastructure then sourced and loaded the datasets. Our modeling environment consists of Github (Exhibit A) for code storage, an Amazon RDS postgreSQL database, amazon t2.large ec2 virtual instance for model training and testing, and individual Python IDEs for model construction.

Both of our datasets required extensive cleaning and preparation prior to analysis. The Yelp dataset was provided in JSON format, so we utilized a python script to convert the files to CSV for easier use. Next, these files were loaded into tables in the postgreSQL database. It was at this point that we understood the extent of the data cleaning required. Each field in the Yelp dataset is wrapped in JSON tags that needed to be removed. To do this, we created a view in the database and trimmed the extraneous tags in the view definition.

We performed additional cleaning of the datasets so that similar attributes in both datasets are formatted the same way, which included normalizing street addresses using GIS data, and normalizing text attributes. All attributes have been normalized to remove leading spaces and unreadable characters. Whitespace at the beginning and end of the name and address in each dataset was trimmed, and the strings are converted to uppercase before matching. The matching thresholds were adjusted to increase potential for matching, or decrease false matches.

After initial cleaning of the data, we joined the health code violation data to the Yelp “businesses” table. The “businesses” table contains a hash value primary key for each restaurant that we used to join other tables in the yelp dataset to, such as the "reviews" table that we will mine for n-gram extraction.

We joined these two data sets by matching records in both sources where the restaurant names and street address had levenshtein distances[3] as follows:

- The Levenshtein distance between the normalized restaurant name in each of the two datasets is < 3
- The distance between the normalized address from each dataset is < 4

For each inspection, we identified the previous inspection date for the same restaurant (except in the case of the earliest observation, which was not used in the model). From the Yelp review data, we matched all reviews that were submitted for that restaurant between those two inspection dates. We decided to not focus on minor food inspection violations because it would be unlikely that customers would notice them and remark on them in the Yelp reviews. Examples of these infractions include improper regulatory signage, or the wrong type of non-slip floor mats. To leave low violations out of scope, labeled each observation where at least one critical infraction was identified as belonging to the positive (INFRACTION = TRUE) class. As may be expected, the data are rather unbalanced, with roughly twice as many negative observations as positive.

We then created dummy variables for all of the attribute and category features. We wanted to capture if the attribute is true, false, or not applicable for the restaurant. The category column captures descriptive features such as the types of cuisine served,

[3] Gilleland, Michael. "Levenshtein Distance, in Three Flavors." Levenshtein Distance, Merriam Park Software, 2016, people.cs.pitt.edu/~kirk/cs1501/Pruhs/Spring2006/assignments/editdistance/Levenshtein%20Distance.htm. Accessed 8 May 2017.

and whether or not the restaurant is a bar. The attributes column captures features such as the 'ambiance' of the restaurant, parking, noise level, and other unique features.

There were a couple of obstacles in this step, as the original yelp attribute and category features were provided in the form of nested dictionaries, which required additional clean up to extract those variables as separate features.

We created several engineered features that measure details about the restaurant reviews and measure aspects of the health code violation data including:

Toronto DineSafe

  - “Did most recent inspection have a significant or crucial violation?”
  - Number of total violations for the restaurant in sampling period
  - Number of critical violations in the previous inspection

- Yelp
  - Number of unique reviewers
  - Number of Yelp reviews within the sampling period
  - Number of reviews in each “star ranking” category
  - Average review for the inspection period vs overall

All of the info from the two datasets was collapsed to create a single row in dataframe per restaurant. As part of our preparation for natural language processing, we did text feature extraction, stop word removal, and stemming. We tried to compare lemmatization and stemming and found that lemmatization was too aggressive and removed a lot of unique words that related to food description.

Modeling

We completed n-gram extraction of the Yelp “reviews” data by first merging all reviews for a given restaurant, and then performing the extraction on a per-restaurant basis. This was done so that we are not considering individual reviews for a given restaurant, but rather we treat all reviews within the given inspection period as a single document.

By comparing the results of several vectorizing methods as well as several modeling methods, we determined which combination yields the best performance.

For each class of model, we compared the Term Frequency - Inverse Document Frequency (TFIDF) vectorization on trigrams (i.e. individual, 2 words together, 3 words together) against count vectorization. We found that the TFIDF vectorization method produces the best accuracy when measured in terms of Area Under the Receiver Operating Characteristic (ROC) Curve across all model classes tested.

The first model we used is Logistic Regression, with TFIDFVectorizer. We used grid search to identify optimal hyperparameters for the specific vectorization method being implemented.   To evaluate the performance of the model we used a stratified 5 fold cross-validation and plotted the ROC curves from each fold as well as the mean. It is important to note that using a lower K for the K-fold cross validation method may introduce a pessimistic bias to the resulted estimated by the average ROC curve[4]. Because of the class imbalance in our original data, we selected a stratified validation

[4] Joglekar, Sachin. “Cross Validation and the Bias-Variance Tradeoff (for Dummies).” Sachin Joglekar's Blog, 30 Aug. 2015, codesachin.wordpress.com/2015/08/30/cross-validation-and-the-bias-variance-tradeoff-for-dummies/. Accessed 8 May 2017.

method in order to ensure that each fold was more representative of the class distribution in the overall population.[5]

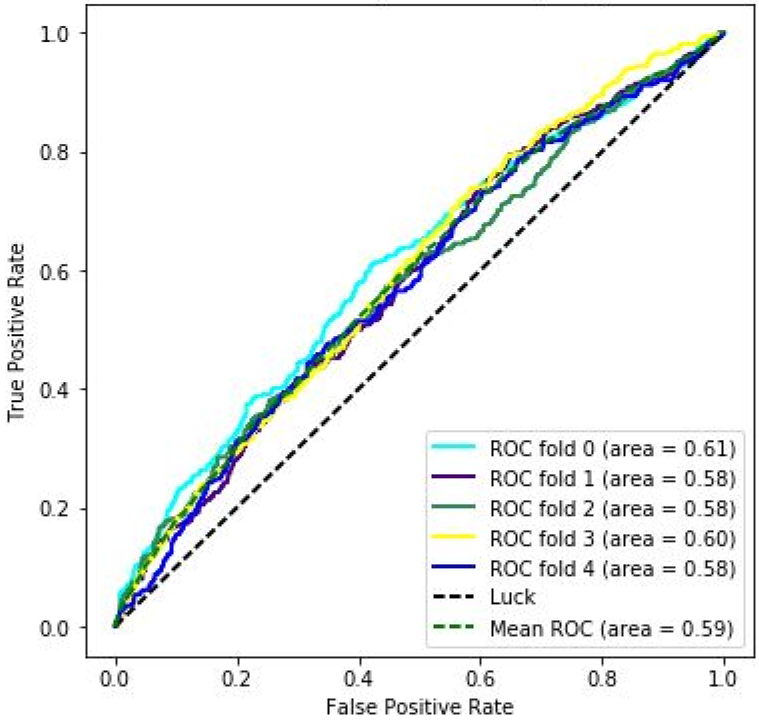
We repeated this process with Random Forest ensemble method in order to determine if more complex learning methods would produce improved results. As with Linear Regression, the TFIDF vectorization method produced a higher mean AUC based on 5-fold stratified cross validation.

Finally, in an attempt to replicate previous research on this topic, we implemented a Support Vector Machine with a linear kernel, which has demonstrated favorable results in text based classification problems[6]. SVMs also provide the benefit of not requiring parameter tuning, which eliminated the need for optimization before cross validation.

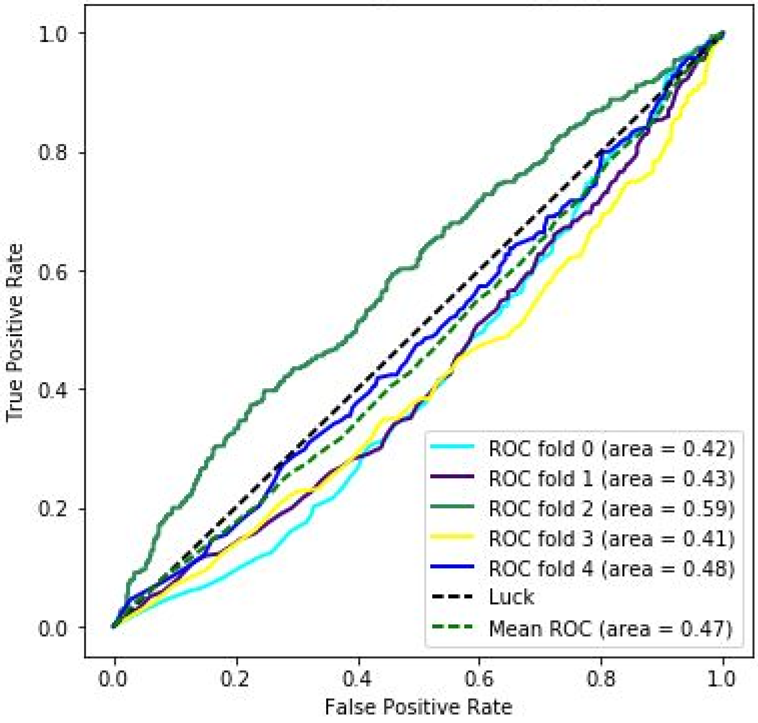
[5] “Cross-Validation: Evaluating Estimator Performance.” 3.1. Cross-Validation: Evaluating Estimator Performance — Scikit-Learn 0.18.1 Documentation, SciKit Learn, 2017, scikit-learn.org/stable/modules/cross_validation.html. Accessed 8 May 2017.

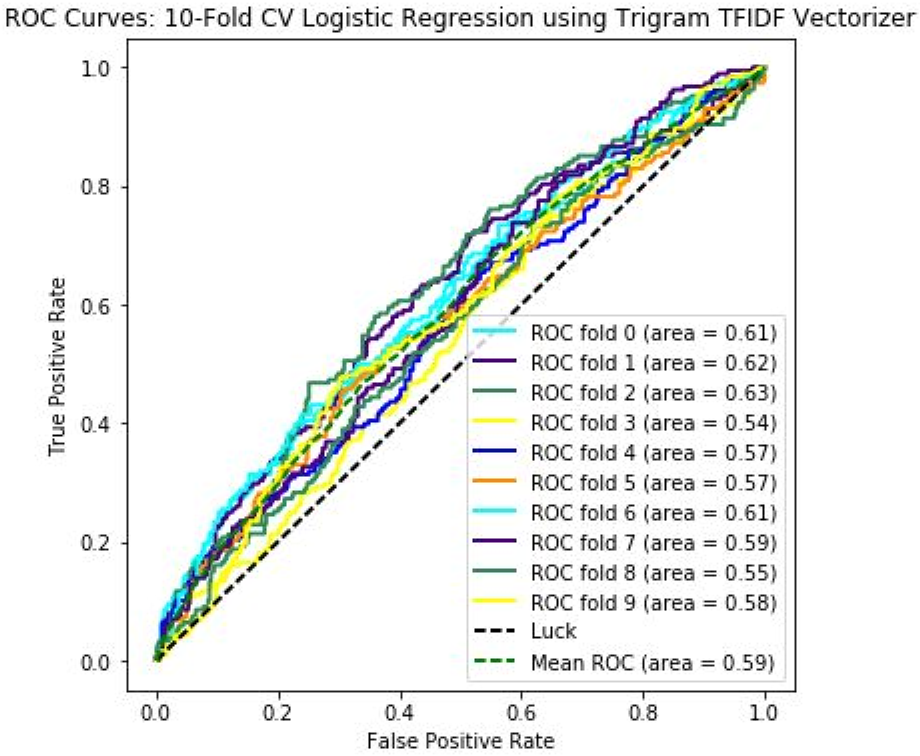
[6] Joachims, Thorsten. "Text Categorization with Support Vector Machines: Learning with Many Relevant Features." Machine Learning: ECML-98 Lecture Notes in Computer Science (1998): 137-42. Web.

ROC Curves: 5-Fold CV Linear Regression using Trigram TFIDF Vectorizer

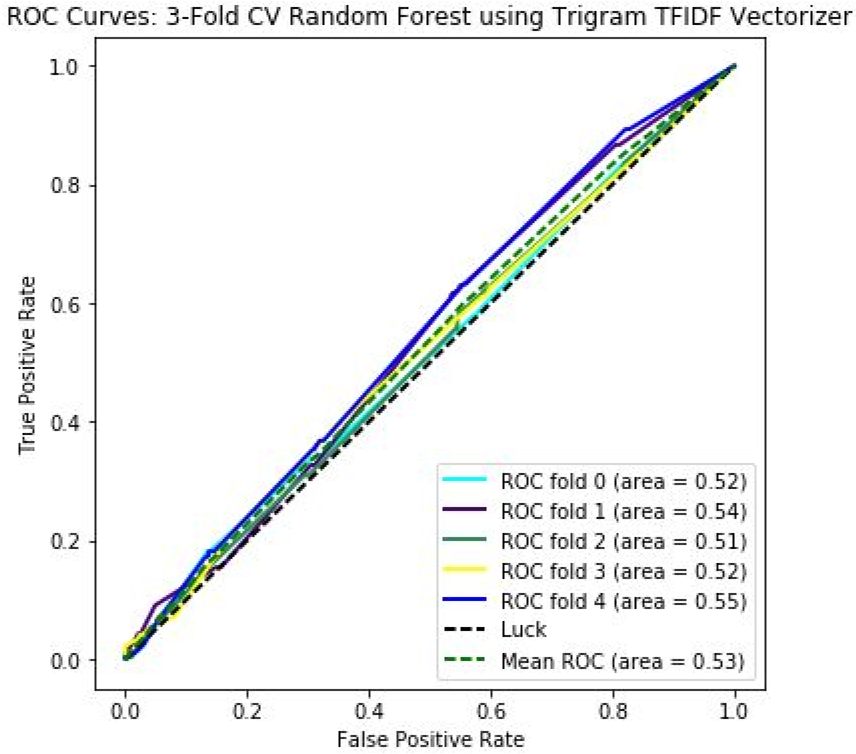


ROC Curves: 5-Fold CV Linear Regression using Trigram Count Vectorizer

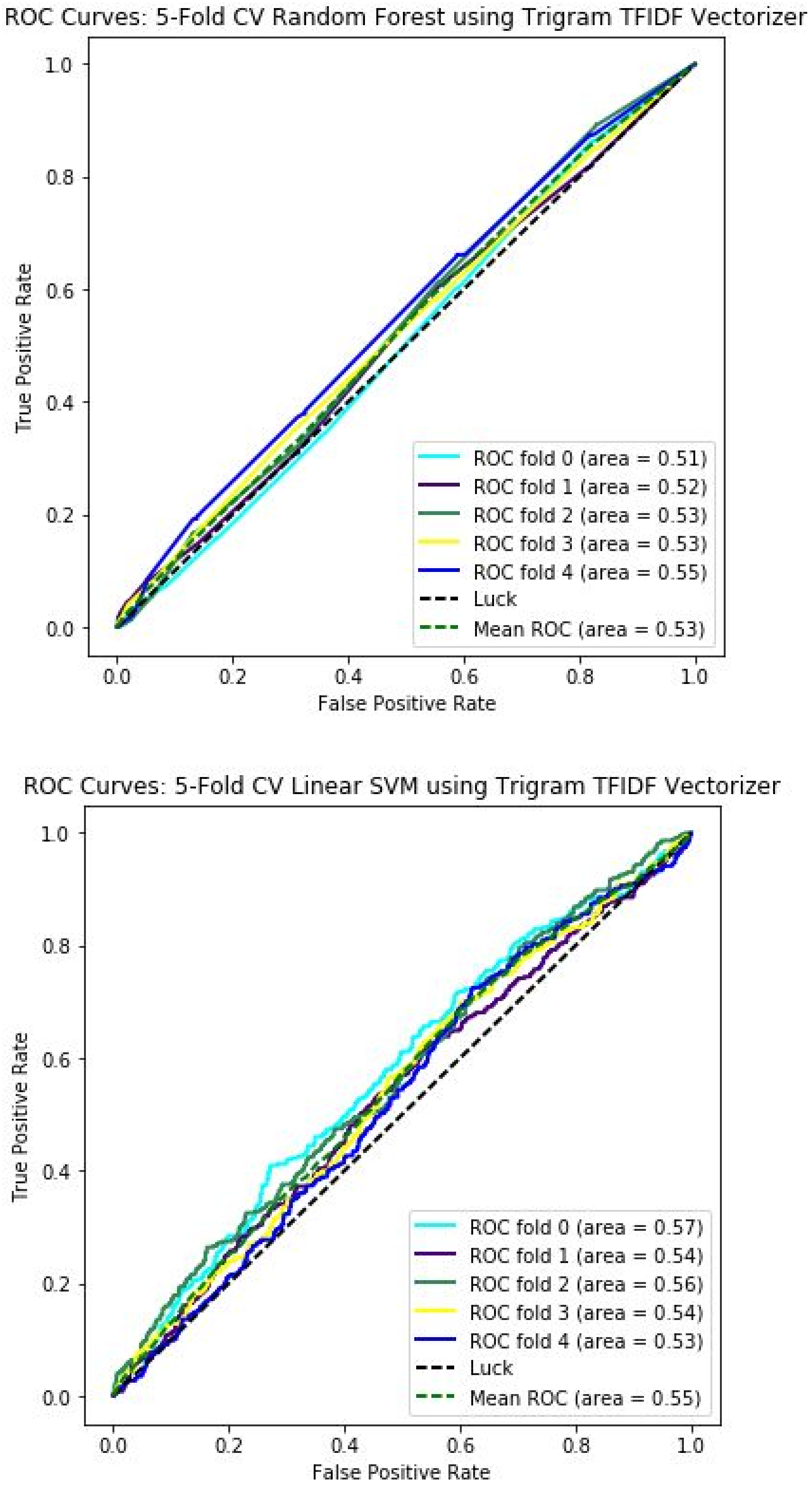




ROC Curves: 10-Fold CV Logistic Regression using Trigram TFIDF Vectorizer



ROC Curves: 3-Fold CV Random Forest using Trigram TFIDF Vectorizer



ROC Curves: 5-Fold CV Random Forest using Trigram TFIDF Vectorizer

ROC Curves: 5-Fold CV Linear SVM using Trigram TFIDF Vectorizer

Table 1: ROC Curves

Linear Regression with optimized parameters of C=.01 and L1 (lasso) penalty, using TFIDF vectorization on the review text produced the highest performance, with an AUC of 0.59.

Evaluation

Using the Yelp Data Science Challenge (Exhibit A) dataset and the Toronto historical health code violation dataset (Exhibit B), we trained a model to classify restaurants based on whether or not there could likely be at least one critical violation in the next inspection.

Linear regression produced the best results in accurately identifying restaurants that have high or critical violations. Interestingly, random forests do not appear to perform well on this task. This is likely because tree based classifiers do not perform well on highly dimensional and sparse data, of which the Yelp dataset is a classic example. Many features only apply to a handful of restaurants, and the yelp review data contain a high number of unique words, such as phrases or food types that are specific to a certain cuisine.

Due to the popularity of Linear SVMs as a solution in text classification problems, we predicted that this method would perform best out of all the methods tested, however Linear Regression performed better than the SVM in many trials.

Based on the results we believe that further research, model tuning, and experimentation may yield improved results. In order to make our model more effective we would suggest further processing and treatment of the review text. We found that reviews tend to be very polite overall, which could be a function of cultural norms in Canada in contrast to the US, where previous research on this topic has been performed. Notably, the previous iteration of this experiment was done using review text from the Northeast United States, which, anecdotally, generally has a different level of conversational “politeness” than what might be expected of a Canadian city. The histogram of average star ratings for restaurants in Toronto shows a negative skew towards higher ratings, which supports this theory.

With more time to improve the model, we would use more sentiment analysis of the review text to normalize for it being overly positive. Another potential issue is that Yelp data could contain some fake reviews; either fake positive ones generated by the restaurant itself or fake negative ones created by users with personal issues with the restaurant owners.

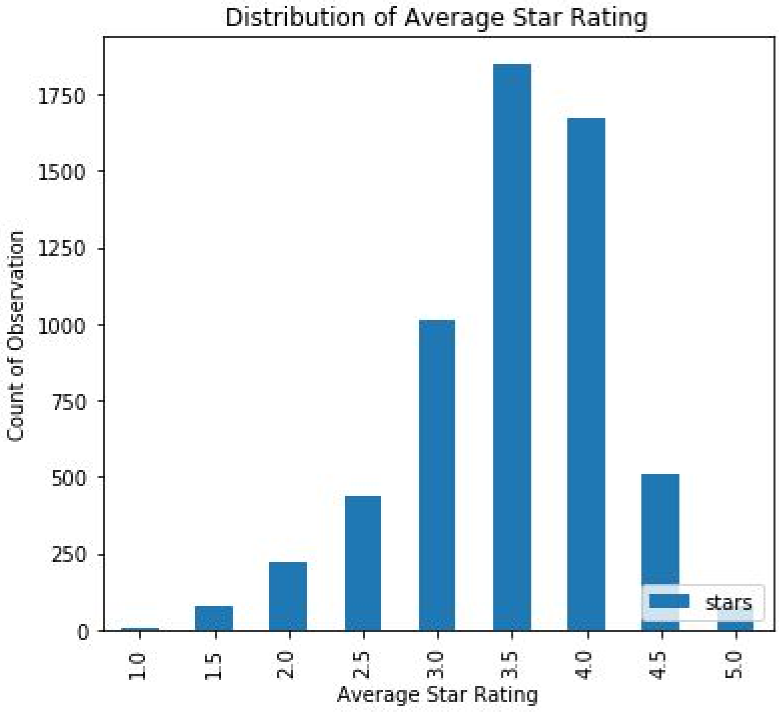


Table 4: Distribution of Average Star Rating

The model would likely be significantly better if there was more data available. As outlined in Table 2 below, most restaurants had only a few reviews in scope to use for a given inspection period. This may be a function of the limited data availability for the Yelp challenge; more data may be available through their various API options. Additionally, the limited lookback period of both the inspection data and yelp data implies that we may not be observing changes in a restaurant's inspection results (and theoretically the review sentiment) over time.

Finally, this paper explores the application of a limited number of model classes to the problem of predicting health code infractions. While several of the models we explored were optimized using a grid search method, further optimization of the models employed is likely to improve the model accuracy.

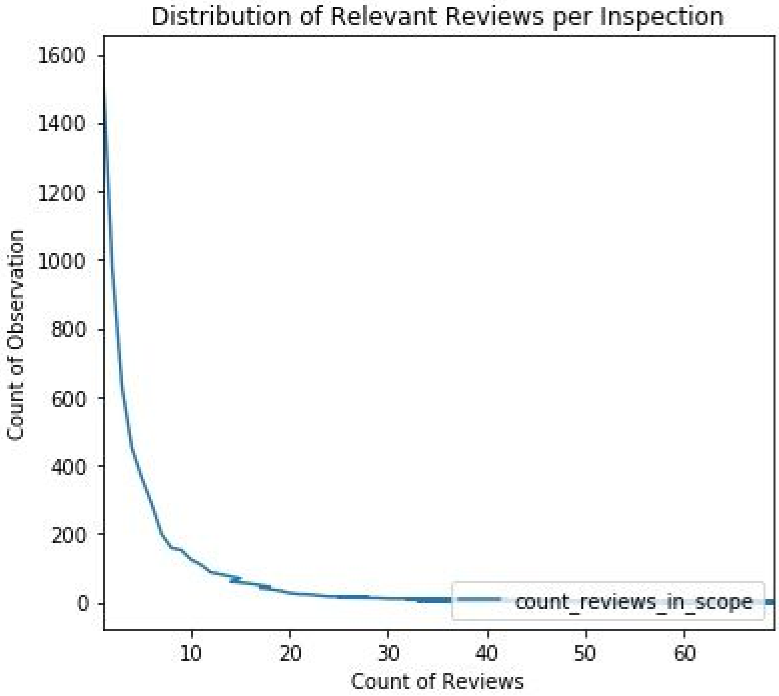


Table 3: Distribution of Relevant Reviews per Inspection

We also noticed that there is an imbalance as there are half as many inspections in the Toronto data flagged as “critical” or “significant” which affects modeling (see Table 3). While we used a stratified cross validation approach to account for this class imbalance, other solutions such as upsampling have yet to be assessed.

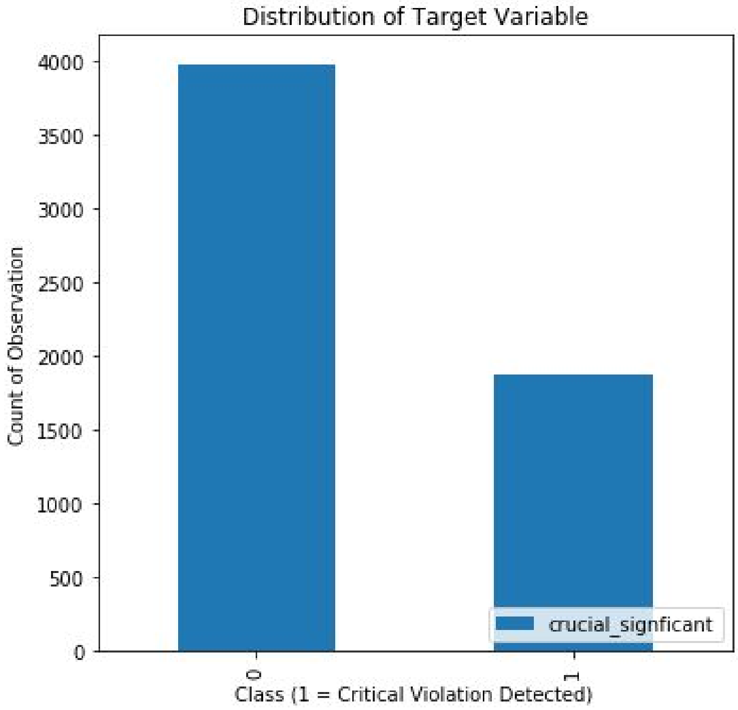


Table 4: Distribution of Target Variable

The results of the model could be used by either restaurant management to take preventative actions to address any potential violations, or it could be used by the health department to identify higher risk restaurants.

To deploy, maintain and enhance this model, the city and / or restaurant executives must invest in resources with technical expertise in data science. As this paper demonstrates, one must venture beyond basic data science techniques in order to generate results that are sufficiently sound and accurate enough to be used in decision making. Though expenses can be minimized by hiring contract staff as needed to work on the model itself, managers should be trained to effectively hire and supervise these resources.

The long term benefits include inspection funds being used more efficiently. Based on the unique records from the Toronto data, there are approximately 8,130 food-related businesses in Toronto; in light of this, inspections by the city require considerable resources from budgets that are already likely constrained.

In addition to potentially realizing direct savings, cost avoidance is a major factor. The cost of restaurant chains not addressing food violations can be enormous. The troubles faced by Chipotle serve as a disturbing case study and a cautionary tale. As highlighted in a recent article in Fortune, “the outbreak of foodborne illnesses that broke out coast to coast for five months in 2015 has permanently impaired the profitability of its business model. Simply put, the company’s lack of investment in basic food safety procedures violated the first rule of running a restaurant: Don’t get the customer sick.”[7] The effects of this crisis are still felt today; even now in 2017, “customers still think of Chipotle as the place that has E. coli.”

Deployment

[7] Penney, Howard, and Shayne Laidlaw. “Why Chipotle May Never Make a Big Comeback.” |Fortune.com, Fortune, 4 May 2017, fortune.com/2017/05/04/chipotle-stock-earnings-hack-data-breach-mcdonalds/. Accessed 8 May 2017.

The model is intended to be deployed by the city of Toronto as well as the leadership teams of restaurant chains for use in proactively detecting food violations.

One obstacle to overcome with deployment is access to Yelp and inspection data that extends beyond the timeframe used in this paper. Though our group leveraged the data available through Yelp Challenge, according to the terms of use, it is only intended to be for academic purposes.[8] In addition, the current challenge is only due to run through June 30th.[9] The regular Yelp API is unfortunately limited in certain critical ways so those looking to deploy our model will likely need to negotiate with the Yelp Partnerships team over payment terms for Premium Access.[10] According to the FAQ, the free Yelp API "does not return full review text; three review excerpts of 160 characters are provided by default." [11]

Another obstacle is that these organizations need to have people able to access to gather the relevant data and run the models. As outlined in this paper, familiarity with data handling (e.g. JSON, APIs) plus data querying and modeling (e.g. SQL, python) is required.

There are a few risks associated with using this model. For example, false positives can result in wasting city or corporate resources on inspections that are not necessary. False negatives mean that the city or restaurant management aren't aware of sites that have problems. While violations go undetected, health code problems can result in customer illnesses or injuries. To mitigate these risks, it is important to

[8] "Yelp Dataset Challenge." Yelp, www.yelp.com/dataset_challenge/. Accessed 8 May 2017.
[9]Yelp Dataset Challenge User Agreement. www.yelp.com/html/pdf/Dataset_Challenge_Academic_Dataset_Agreement.pdf+. Accessed 7 May 2017.
[10] "Introducing the Yelp Fusion API." Yelp Fusion, www.yelp.com/developers. Accessed 8 May 2017.
[11] "FAQ." Developers - Frequently Asked Questions - Yelp Fusion, www.yelp.com/developers/faq. Accessed 8 May 2017.

fine-tune the model to minimize both false positives and negatives. Since false negatives can result in health impacts, it is preferable to err on the side of caution and reduce those where possible.

Other third parties, such as Yelp Inc. or another advertising review site, would likely be interested in which restaurants are indicating a higher risk of foodborne illness or health code violations as a feature to attract traffic to their site. Customers are likely to place value on knowing which restaurants to avoid based on this information.

It is important when deploying data science models to consider the ethical implications. For example, given the stigma associated with food violations, it is critical that the analysis is properly annotated so that the results are not misinterpreted. In light of the sensitivity of the information provided by the results, it is important that the results are only shared with key stakeholders until there is reasonable confidence that the model will perform as expected in production, and that the audience is made aware of the assumptions and limitations of the model.

Exhibits

Exhibit A: Project Github Link

https://github.com/ms682/DataScience

Exhibit B: Yelp Dataset Models (JSON)

yelp_academic_dataset_business.json

```
{
  "business_id":"encrypted business id",
  "name":"business name",
  "neighborhood":"hood name",
  "address":"full address",
  "city":"city",
  "state":"state -- if applicable --",
  "postal code":"postal code",
  "latitude":latitude,
  "longitude":longitude,
  "stars":star rating, rounded to half-stars,
  "review_count":number of reviews,
  "is_open":0/1 (closed/open),
  "attributes":["an array of strings: each array element is an attribute"],
  "categories":["an array of strings of business categories"],
  "hours":["an array of strings of business hours"],
  "type": "business"
}
```

yelp_academic_dataset_review.json

```
{
   "review_id":"encrypted review id",
   "user_id":"encrypted user id",
   "business_id":"encrypted business id",
   "stars":star rating, rounded to half-stars,
  "date":"date formatted like 2009-12-19",
  "text":"review text",
  "useful":number of useful votes received,
  "funny":number of funny votes received,
  "cool": number of cool review votes received,
  "type": "review"
}
```

Exhibit C: Health Department Inspection Violations Data Model

http://www.toronto.ca/health/dinesafe/regulation.htm

- ROW_ID - Represents the row number
- ESTABLISHMENT_ID – Unique identifier for an establishment
- INSPECTION_ID - Unique identifier for each inspection
- ESTABLISHMENT_NAME – Business name of the establishment
- ESTABLISHMENTTYPE – Establishment type, i.e. restaurant, mobile cart
- ESTABLISHMENT_ADDRESS – Municipal address of the establishment
- ESTABLISHMENT_STATUS – Pass, Conditional Pass, Closed
- MINIMUM_INSPECTIONS_PERYEAR – Every eating and drinking establishment in the City of Toronto receives a minimum of 1, 2, or 3 inspections each year depending on the specific type of establishment, the food preparation processes, volume and type of food served and other related criteria
- INFRACTION_DETAILS – Description of the infraction
- INSPECTION_DATE – Calendar date the inspection was conducted
- SEVERITY – Level of the infraction, i.e. S – Significant, M – Minor, C – Crucial
- ACTION – Enforcement activity based on the infractions noted during a food safety inspection
- COURT_OUTCOME – The registered court decision resulting from the issuance of a ticket or summons for outstanding infractions to the Health Protection and Promotion Act
- AMOUNT_FINED – Fine determined in a court outcome

Exhibit D: Final Model Attributes

crucial_signficant [TARGET]

Alcohol_full_bar

Alcohol_none

Alcohol

Ambience_casual

Ambience_classy

Ambience_hipster

Ambience_intimate

Ambience_romantic

Ambience_touristy

Ambience_trendy

Ambience_upscale

BikeParking

BusinessAcceptsCreditCards

BusinessParking_garage

BusinessParking_lot

BusinessParking_street

BusinessParking_valet

BusinessParking_validated

Caters

ByAppointmentOnly

CoatCheck

DogsAllowed

DriveThru

GoodForDancing

HappyHour

Music_background_music

WiFi_free

WiFi_no

WiFi_paid

WiFi

is_open_0

is_open_1

is_open

BestNights_friday

BestNights_monday

BestNights_saturday

BestNights_sunday

BestNights_thursday

BestNights_tuesday

BestNights_wednesday

RestaurantsCounterService

Smoking_no

Smoking_outdoor

Smoking_yes

Smoking

WheelchairAccessible

stars

RestaurantsAttire_casual

RestaurantsAttire_dressy

RestaurantsAttire_formal

RestaurantsAttire

RestaurantsDelivery

RestaurantsGoodForGroups

last_insp_cs

stars_1

stars_2

stars_3

stars_4

stars_5

review_text

all_review_count

count_reviews_in_scope

count_unique_users

RestaurantsPriceRange2_1

RestaurantsPriceRange2_2

RestaurantsPriceRange2_3

RestaurantsPriceRange2_4

RestaurantsPriceRange2

RestaurantsReservations

RestaurantsTableService

RestaurantsTakeOut

Alcohol_beer_and_wine

Music_jukebox

Music_karaoke

Music_live

Music_no_music

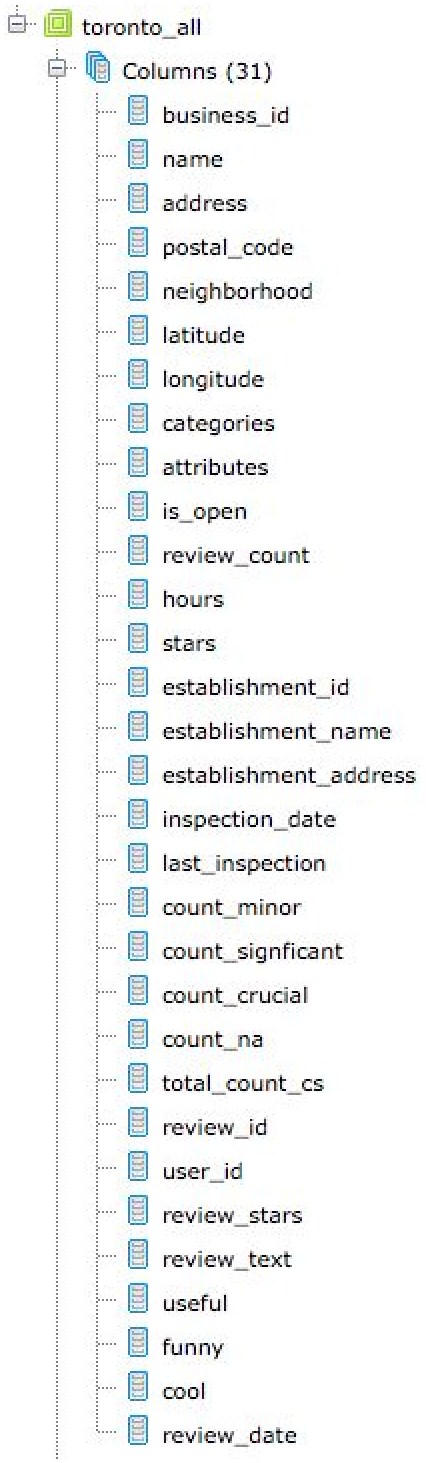
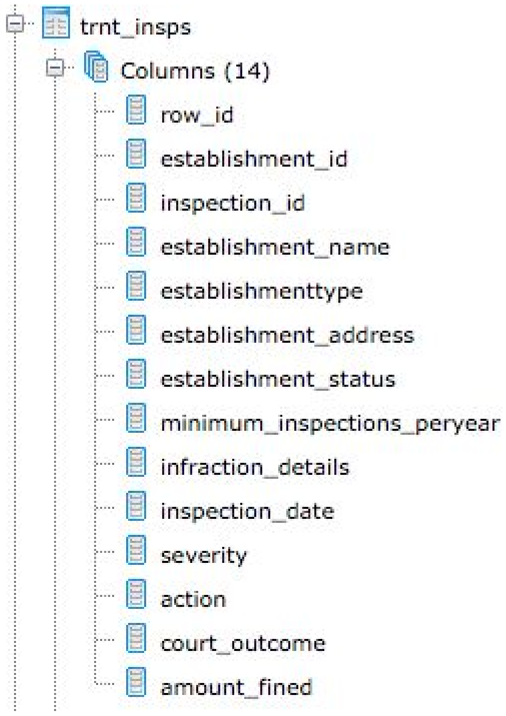
Music_video

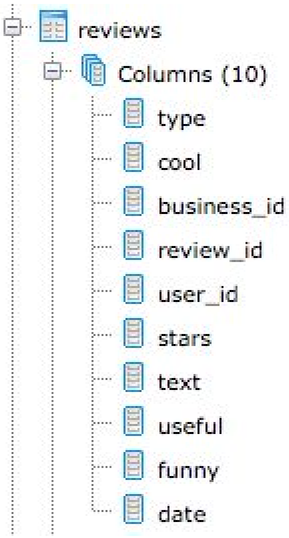
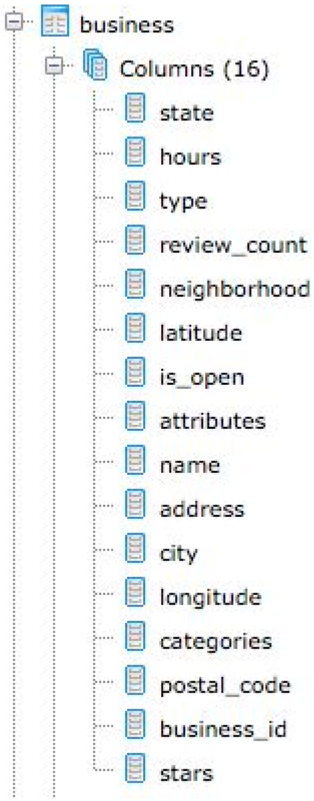
Open24Hours

Music_dj

Restaurant Categories (dummy-ized cuisine and restaurant type data)

Exhibit E - Database Information

Combined Inspection and Yelp View	Toronto Inspection Table
 toronto_all Columns (31) business_id name address postal_code neighborhood latitude longitude categories attributes is_open review_count hours stars establishment_id establishment_name establishment_address inspection_date last_inspection count_minor count_signficant count_crucial count_na total_count_cs review_id user_id review_stars review_text useful funny cool review_date	 trnt_insps Columns (14) row_id establishment_id inspection_id establishment_name establishmenttype establishment_address establishment_status minimum_inspections_peryear infraction_details inspection_date severity action court_outcome amount_fined

Yelp Reviews Table	Yelp Business Table
 reviews Columns (10) type cool business_id review_id user_id stars text useful funny date	 business Columns (16) state hours type review_count neighborhood latitude is_open attributes name address city longitude categories postal_code business_id stars