DeepDish

Rayan Elalamy, Édouard Ghaleb, Adrian Hamelink, Marc Ourfali
École Polytechnique Fédérale de Lausanne
Applied Data Analysis CS-401
October 28th, 2019

***Abstract*— To ensure food quality and safety to their fellow citizens, the Chicago Department of Public Health (CDPH) conducts regular inspections of every establishment providing food. We perform an analysis of food violations in Chicago and analyze results of these inspections to assess the most critical violations. Afterwards, we take a look at the most representative comments arising from failed inspections through NLP. Then, we forecast the evolution of failed inspections for the next year. Moreover, we complement the dataset with data scrapped from Yelp.**

I. Introduction

In this project, we consider the Chicago Food Inspection dataset available on Kaggle [1].

This dataset contains information about inspections of restaurants and other food establishments in Chicago from January 1, 2010 to November 18, 2019. Our first aim is to provide a geographical representation of the distribution of the data. We then look at different predictions we can infer on the r esults of the inspections using features from the data. Finally we use the external Yelp dataset (LINK YELP) and assess its complementary capacity for our predictions. As a first step, we study the features provided in the data. We then visualize the violation and the risk of each establishment through different choropleth maps as well as heatmaps (LINK GITHUB). We then try to predict the result for each inspection using classification algorithms and important feature from the violations using NLP. We will also consider some forecasting for the results using time features from the dataset and explain how we can predict the result using exponential smoothing methods.

II. Exploratory Data Analysis

The dataset contains 196,030 inspections, with variable such as the inspection date, the name of the establishment inspected (*AKA name*), its geographical location (*Latitude, Longitude, Zip code*) and a risk rating (*Risk*). It also contains the inspection outcome (*Results*) and the violations that were found (*Violations*) as well as other variables. A detailed explanation of the variables in the dataset can be found here.

The majority of establishments in this dataset are restaurants with two thirds of the data. Other significant establishments type are store with 13% of the dataset and schools and day care centers both representing 5 % of the data.

The risk rating consist of three levels and is tied to the frequency of inspections. This rating is an appreciation of how likely the establishment poses a risk to public health. As Fig.1 shows, the higher the risk attributed to an establishment, the more likely it is to get inspected.

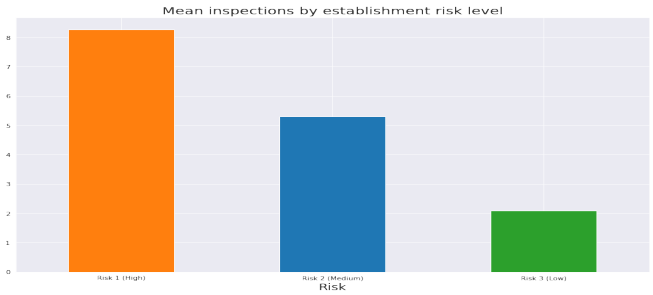

Fig. 1. Mean number of inspections by establishment risk level. This plot shows that higher risk establishments are more likely to get inspected.

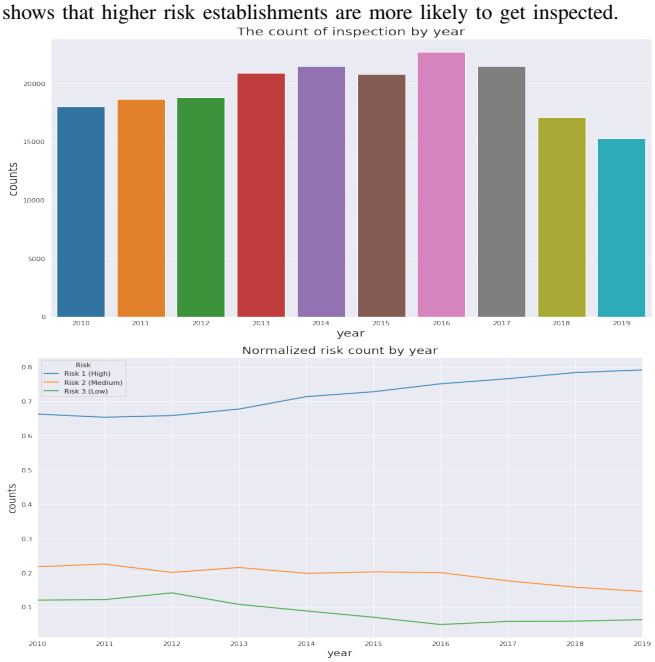

Fig. 2. The count of inspections per year increases from 2010 to 2016 and then decreases. The second plot shows an upward trend in the high risk ratings since 2012.

Inspections can be of different types, the main one being *canvass* inspections which are unannounced to the establishment. The number of inspections has increased steadily from 2010 to 2016 and then decreased since 2016 as shown in the barplot in Fig.2. Fig.2 also shows the normalized risk count per year indicating an upward trend in the ratio of high risk rating of establishments since 2012.

The *Violations* variable is a string containing the violations that were raised during an inspection. There are 45 possible violations types, each of these has a code (1 through 44

and 70) and comments about the violations that were found are also included. From January 1st 2018 onward, the city of Chicago has however changed its definition of violations. Looking at the data before this date, the violations with code 1 to 14 were considered *critical* and violations 15 to 29 were considered *serious*. We then added a variable *Violation risk* with value 3 if a *critical* violation was found, 2 if a *serious* violations was found, 1 if neither was found and 0 if not violation is specified. This variable proved to have a high correlation (0.8) with the results of the inspection and was subsequently used in our analyses.

Moreover, various plots regarding the geographical distribution of the risk and the results of the inspections over time were made. These interactive plots are accessible for visualisation on our python notebook here, under part C or on our GitHub repository [5].

We originally intended to use the dataset proposed by Yelp for their Dataset Challenge. Unfortunately, it did not contain any information about establishments located in Chicago. After emailing their support team, they advised us to use theirs public API to request the information we needed. We were limited however to 5'000 calls per day, an upper bound Yelp unfortunately didn't want to raise in our scenario.

We first needed to get unique establishments from the dataset. Using the Business Match endpoint, we could query Yelp to obtain the identifier they used for that specific business. This only gave geographical information we could compare to ours. Then using the same id, we could get more details such as price range, food categories, open status, and the user rating.

We did not have enough time to query the reviews left by users, but these were limited to three per establishment, and cropped to less than 200 characters.

In total, we were able to match about 36% of all unique establishments, giving us 12'000 businesses on Yelp.

III. Feature Importance

In this section, we try to determine the most critical violations in an inspection. Due to a change in the data collection procedure in July 2018, we only consider data up to this date. More precisely, we want to verify the assertion of the dataset description on Chicago's website which states that the most serious violations are the ones linked with number 1 to 14. We first fit a random forest [3] classification model on failed and passed inspections and then extract the most important features of the model, we consider inspections with "Pass" and "Pass with conditions" results as passed and only inspections with 'Fail' result as failed. In order to predict the binary results, we encode these outcomes as 0 and 1 respectively. We can first see a clear imbalance between categories (78% of Pass and 22% of Fail) thus we first upsample the failed inspections. In We obtain the following results :

Accuracy	Precision	Recall	F-score
92%	96%	88%	92%

TABLE I

Results for Random Forest Algorithm.

We see that our model performs well on the data and thus justifying extracting its most important features. As we can see on Table II it seems that the violations 1-14 are not dominant in the top 5 important violations. To confirm these results, we do a Chi-squared [4] test for features selection and we obtain the following results:

	Violation int	Violation
1	18	no evidence of rodent or insect outer openings...
2	29	previous minor violation(s) corrected.
3	19	outside garbage waste grease and storage area;...
4	24	dish washing facilities: properly designed, co...
5	16	food protected during storage, preparation, di...

TABLE II

Features importance from random forest model

	Violation int	Violation
1	18	no evidence of rodent or insect outer openings...
2	29	previous minor violation(s) corrected.
3	19	outside garbage waste grease and storage area;...
4	24	dish washing facilities: properly designed, co...
5	16	food protected during storage, preparation, di...

TABLE III

Features importance from Chi-squared test

As we obtain the same results we can say that violation 18 seems to be the most important violation and that violations 1-14 are still outside of the top 5 (and barely present in the top 10).

IV. NLP Insights

For each violation found, inspectors can leave a comment to give a little more details about the situation. Then we want to extract these comments to analyze words that are very representative of failed inspections and get a better insight into the true reasons of the outcome of inspections. First we process each text, we remove special characters, stopwords, numbers and we lemmatize each word so that we can consider "flies" to be the same word than "fly". We rank the words by using frequency difference among classes:

$$\text{freqdiff}(word) = \frac{|\{word \in category_{Fail}\}|}{|\{category_{Fail}\}|} - \frac{|\{word \in category_{Pass}\}|}{|\{category_{Pass}\}|}$$

Then we obtain that the words which are the most linked with failed inspections.

We notice that many of these words are linked with rodents which highlights results obtained in the feature importance section. At the end of the paper there is a representation of the distribution of all the significant words in the dataset. As it seems that rodents and insects are a serious problems in inspections, we can visualize the presence of these on a map of Chicago. Hence on Figure 3 we can see in yellow

	Top Fail Words
1	mouse droppings
2	mouse
3	minimize eliminate
4	control service
5	droppings

TABLE IV

TOP 5 FAIL WORDS EXTRACTED FROM COMMENTS.

facilities with a presence of mice and on figure 4 we can see in yellow facilities with a presence of fly. We can see that there is a strong link between geolocalisation and type of rodent/insect (which can be explained by several factors like the lake shore). If we had more time we would have studied the spatial dependency of comments in more details.

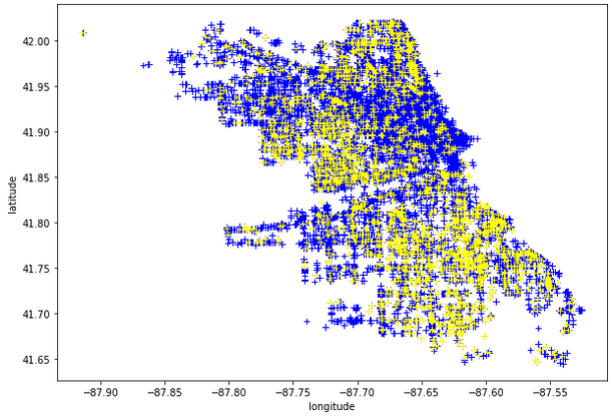

Fig. 3. Inspections with 'mouse' in comments (in yellow) and blue else.

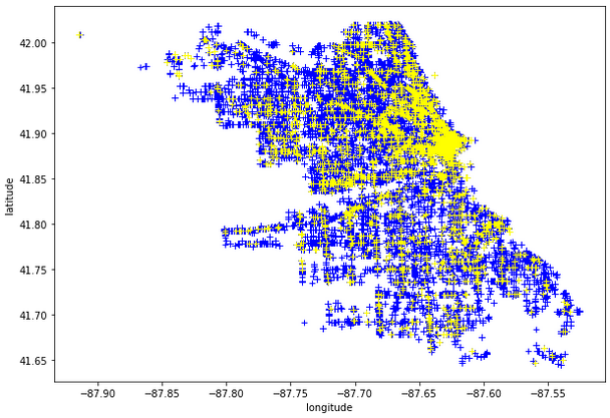

Fig. 4. Inspections with 'fly' in comments (in yellow) and blue else.

V. FORECASTING

Another interesting results would be to forecast the percentage of failed inspections in the next year. We thus look in more details at the number of passed and failed inspections by month and by year. We can see in Fig. 5, that the number of fail is slowly decreasing in the last years but as is the number of inspections. Furthermore the data shows a clear seasonality with higher inspection number between March and May and in July, percentage of the inspections that are getting passed are lower compared to any other month.

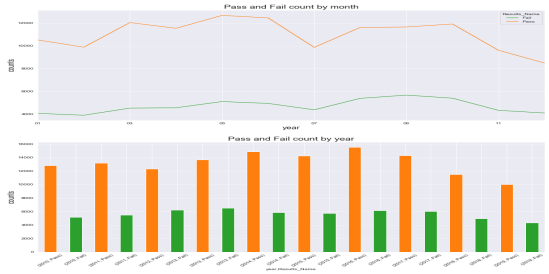

Fig. 5. Count of pass and Fail by month and year

We thus create a time series containing the percentage of failed inspection in each month from January 2010 till November 2019. We can use a seasonal decomposition to look in deeper at the trend and the seasonality of the data. Fig. 6 below shows us that our data seems cyclic with cycle of period $p = 12$ and do not seem to have a important movement in the trend. We will need to find a model that can capture these results.

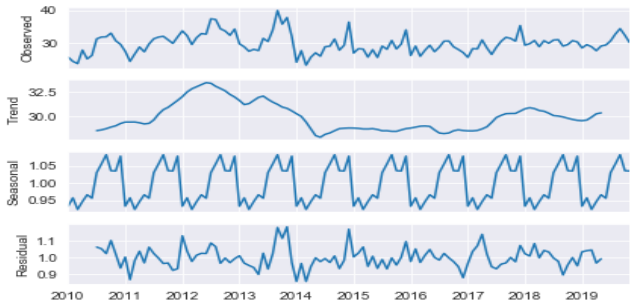

Fig. 6. STL Decomposition

To do so we will look at exponential smoothing method [2], contrary to classical time series method they consider a weighted averages of past observations to forecast new values. The goal is to give more importance to recent values in the series. As observations get older, the importance of these values get exponentially smaller. Exponential Smoothing Methods combine Error, Trend, and Seasonal components in a smoothing calculation. Each term can be combined either additively, i.e. $Y[t] = T[t] + S[t] + \varepsilon[t]$, multiplicatively, i.e. $Y[t] = T[t] * S[t] * \varepsilon[t]$ or be left out of the model. From Fig. 6 we will consider methods that have seasonal periods, and compare the result obtain with additive or multiplicative trend and seasonality. We can see by comparing their AIC value that the best model is with a seasonal period $p = 12$ no trend component and a multiplicative seasonality. This model can be written as

$$Y_t = (l_{t-1})S_{t-p} + \varepsilon_t$$
$$l_t = l_{t-1} + \alpha\varepsilon_t / S_{t-m}$$
$$S_t = S_{t-p} + \gamma\varepsilon_t / l_{t-1}$$

where α, γ constant and $\varepsilon \sim \mathcal{N}(0,1)$. Fitting this model and forecasting the percentage of fails in the next 12 months as well as getting their 95% confidence yields the result outlined in Fig. 7. We have that our model seem to fit well enough the data capturing its seasonality but has some difficulties to get the very high picks in the percentage. This comes from the fact that these results are older and have less weights in our predictions. We can see that the prediction should remain between $[27.5\%, 35\%]$ for the following year with a decrease in the beginning of the year and the a pike during July and August, which is very similar from the result from 2018 and 2019.

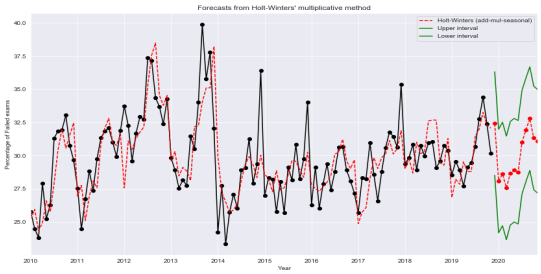

Fig. 7. Prediction for our time series in 2020

In order to understand the goodness of fit of our model we can look at its residuals. We have that a good metric is to consider the Q-Q plot of the residuals, we should have that our residuals are normally distributed. The plot is given in Fig. 8, we can see that the residuals falls in the 45° line but with some outliers in the tail of the distribution. We can asses that the model do indeed fit well the model as these outliers come from high picks from the years between 2010-2015 which have lower weights in our model and can thus be discarded for the predictions.

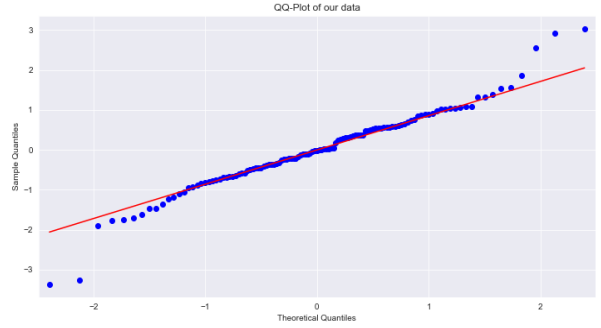

Fig. 8. Q-Q plot of the residuals from our model

VI. Yelp

We first examine whether the score of a restaurant can be used as a predictor for the result of an establishment inspection result. For this we needed to filter out the dataset. Because the score of establishments dates from December 2019, we only analysed the results of inspections from the last year. Moreover, we decided to only look at the result of inspection of the type *canvass* which would give us a more representative score. For inspections of the other types, establishments would be able to prepare for them and results in more passes, even though they might not have passed at other times. We also removed businesses that had been declared out of business. We then merge these entries with the Yelp dataset, again filtering out some results. We only consider the entries for which there are more than 5 reviews to get a more representative score. Our first analysis looks at the score of establishments in groups that have passed inspection (passed or passed with conditions), as well as their price range given by the following table. We collapse \$ with \$\$ and \$\$\$ with \$\$\$\$. For establishments in the \$ 0 to \$30 range, the mean score is actually lower when the inspection is passed (-0.05), but higher in pricier places (+0.43). Applying a t-test comparing the mean for each groups tells us the there 6.76% and 0.06% chance respectively that this difference in means is due to random chance. Only the pricier establishments satisfy a 95% confidence interval. This could be explained by the fact that Yelp reviewers are more critical of the state of an establishment when they are paying a higher price. For lower priced establishments it is harder to give an explanation.

\$	\$\$	\$\$\$	\$\$\$\$
under \$10	\$11 to \$30	\$31 to \$60	above \$61

TABLE V
PRICE RANGE CATEGORIES FROM YELP.

When trying to predict the inspection result of a restaurant using the Yelp rating and price category though, we ran into issues with the amount of failed inspections which was much lower that the amount that passed. We accounted for that, however we did not find any conclusive results when using a logistic model.

VII. Conclusion

Overall, the dataset lends itself to analyses from different methods. Our attention was focused on the nature of the violations and the insight they provide as well as the periodic nature of the data. Using feature importance, we are able to identify a new set of violations that have high predictive capability. Furthermore NLP insights allows us to extract geographical patterns for certain types of violations, especially those concerning pest control. The seasonality of the data lends itself to a time series analysis permitting us to infer on predictions of the percentage of successful inspections for the year to come. Finally, scores for Yelp could be used to motivate the scheduling of new inspections for pricier etablishments rated below the average of their category.

REFERENCES

[1] Chicago Food Inspections From City of Chicago Open Data. [Online; accessed December 19th 2019]
https://www.kaggle.com/chicago/chicago-food-inspections

[2] Innovations state space models for exponential smoothing. [Online; accessed December 19th 2019]
https://otexts.com/fpp2/ets.html?fbclid=IwAR17xsjUPVbSplv2ddHUSu3ITpToRRvnfEMAfU0t9Us4Yegy61f1am5NTcI

[3] Random Forest method from pyton package sklearn. [Online; accessed December 19th 2019]
https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestRegressor.html

[4] Chi-squared feature selection method from pyton package sklearn. [Online; accessed December 19th 2019]
https://scikit-learn.org/stable/modules/generated/sklearn.feature_selection.chi2.html

[5] GitHub Repository for the project. [Online; accessed December 19th 2019]
https://github.com/adr1anh/DeepDish

APPENDIX

A.

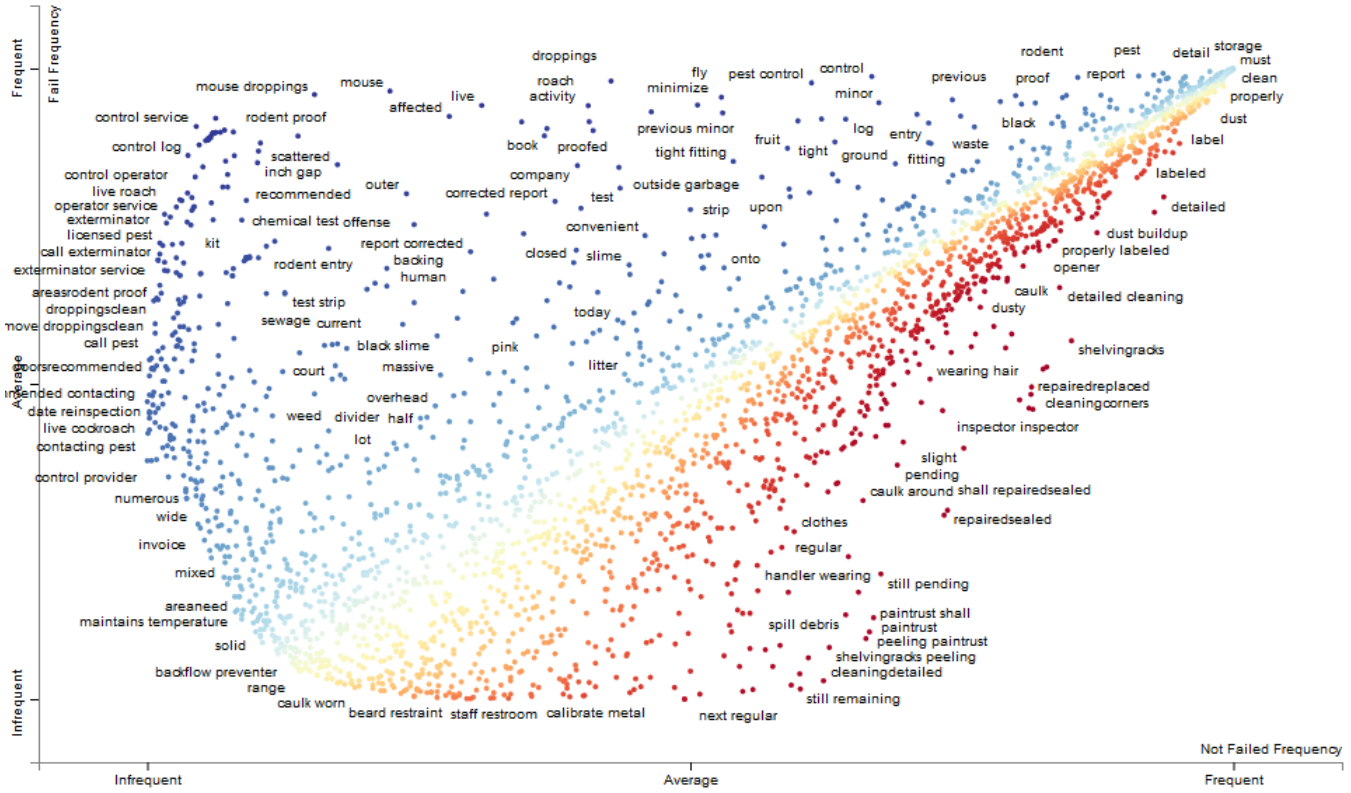

Fig. 9. Representation of the distribution of all the significative words for the two classes.