Predicting Hygiene Inspections Using Online Reviews

Jiannan Chen
University of Massachusetts, Amherst
jiannanchen@umass.edu

Rui Huang
University of Massachusetts, Amherst
rhuang@umass.edu

December 22, 2016

Abstract

Our empirical study examines the correlation between online reviews and the hygienic condition of a restaurant, which can help with the government's health inspections. We predict if a restaurant is hygienic using content-based prediction models which reach an accuracy higher than Naive Bayes models. We also generate a chart of lexical cues extracted from online reviews that provide customers some insights in predicting the hygienic condition. Our study shows the promise of opinion analysis of social media in the government's regulation.

I. Introduction

The hygienic condition of a restaurant is a very important factor that customers consider when choosing where to eat. The government conducts health inspections regularly to motivate restaurants to maintain good hygienic conditions as well as serve as a guide for customers. However, there are still lots of challenges in practice, such as lack of resources to dispatch inspectors, for the Department of Health to carry out inspections that cover most of the restaurants.

With the popularity of online review websites such as Yelp, is it possible to obtain some insights about the hygienic condition of a restaurant by reading online reviews? The paper *Where Not to Eat? Improving Public Policy by Predicting Hygiene Inspections Using Online Reviews* (Kang et al, 2013)[4] proposes that online reviews by customers about the restaurants can be a useful tool to predict the hygienic condition of a restaurant. Our group has conducted an empirical study that demonstrates the correlation between online reviews and the hygienic condition of a restaurant based on the work of Kang et al.[4] We first calculate Spearman's correlation coefficients between penalty scores and different statistics of reviews. Then, we generate a chart of lexical cues extracted from reviews that can provide insights to predict the hygienic condition of a restaurant.

All of our code and dataset can be accessed through Google Drive shared with the instructors(`https://drive.google.com/open?id=0Bxa2Rhi7zdiBNGdyNDBURUFRWjQ`).

II. Related work

The authors indicate that there is no prior work on this specific topic before but there exists some related work in the same field where researchers try to conduct "public health surveillance by measuring relevant textual signals in social media, in particular, micro-blogs" (p.5, Kang et al, 2013)[4]. The papers *Twitter catches the flu: Detecting influenza epidemics using twitter* by Aramaki et al. (2011)[1] and *Modeling spread of disease from social interactions by Sadilek et al (2012)*[9] are two examples. Aramaki et al. proposes a system that exacts influenza related tweets and shows its usefulness at the outbreak and early spread (2011)[1]. Sadilek et al. give the first model that studies the role of people's social interactions in the spread of diseases in a large real-world population (2012)[9]. Both of the works demonstrate the power of NLP in studying the society and helping with real-world problems.

Likelihood Ratio: Kang, Jun Seok, et al.[4] have done a similar research where they use Yelp's dataset to predict the inspection records of Seattle. Our work joins this line but differs in some methods. Firstly, we compare the SVM model with a baseline model (Naive Bayes), demonstrating that SVM is more effective than Naive Bayes under this scenario. Secondly, we use likelihood ratio to find out insightful lexical cues which might indicate hygiene conditions. For Naive Bayes models, McCallum, Andrew, and Kamal Nigam[5] have given us a thorough explanation of Bayesian Network and multinomial model, two most-used Naive Bayes classification model. They also compare the details of the two approaches in the light of implementation and effectiveness. Szoke, Igor, et al.[11] describe several approaches to keyword spotting for informal continuous speech. Even though our dataset is completely different, their work helps us figure out the idea of the likelihood ratio.

SVM Model: We use SVM model to classify the reviews. Previous research like Suykens, Johan AK, and Joos Vandewalle's *Least squares support vector machine classifiers.*[10] gives us great help in understanding the idea and algorithm of Support Vector Machine. And Fan, Rong-En, et al.[3] developed the open source library LIBLINEAR, which supports logistic regression and linear support vector machines. With LIBLINEAR, we can conveniently use command-line tools and

library calls to do classification task with support vector machines. When using SVM, we encountered a problem of severely imbalanced dataset. The work of Chawla N V, Japkowicz N, Kotcz A.[2] discusses the influence of dataset imbalance and some useful methods to mitigate such effect. We learn the influence and reform our training set and testing set. Another important issue of SVM modeling is that we need to vectorize the documents for the training and testing. Pedregosa, Fabian, et al. have developed an useful tool called "Scikit-learn"[8] for various machine learning tasks including vectorization. We use this package to do our work.

Deception classifier: One of the contributions of Ott et al. is the first large-scale dataset that contains "gold standard deceptive positive hotel reviews" (p.1, 2013) as well as negative reviews to make up the lack of deception corpus.[6] The dataset also separates deception based on sentiment (positive and negative) to have a better detection rate. They consider a possible relationship between sentiment and deception and provide a new avenue for future research and propose a combination model that uses "psycholinguistically-motivated features and n-gram features" [7] to better detect deception, which is the deception classifier used in our experiments to filter deceptive reviews.

III. Data

Yelp's Academic Dataset (<https://www.yelp.com/dataset_challenge/dataset>) is used in this project. There are several object types in the dataset, we mainly focus on the data of reviews and business. The json-like data structure for reviews is as in figure1, and the json-like data structure for business is as in figure2.

```
{
    'type': 'review',
    'business_id': (encrypted business id),
    'user_id': (encrypted user id),
    'stars': (star rating, rounded to half-stars),
    'text': (review text),
    'date': (date, formatted like '2012-03-14'),
    'votes': {(vote type): (count)},
}
```

Figure 1: *review data structure*

We decide to study restaurants of Las Vegas whose data are the most in Yelp's dataset and obtain the inspection data from the Southern Nevada Health District (<https://www.southernnevadahealthdistrict.org/restaurants/>). The inspection data are in csv formats, a typical inspection data file that contains information about 'inpection_id', 'facility_id', 'inspection_date', 'inspection_time', 'inspection_demerits', 'inspection_grade', 'inspection_result', and 'violations'.

```
{
    'type': 'business',
    'business_id': (encrypted business id),
    'name': (business name),
    'neighborhoods': [(hood names)],
    'full_address': (localized address),
    'city': (city),
    'state': (state),
    'latitude': latitude,
    'longitude': longitude,
    'stars': (star rating, rounded to half-stars),
    'review_count': review count,
    'categories': [(localized category names)]
    'open': True / False (corresponds to closed, not
business hours),
    'hours': {
        (day_of_week): {
            'open': (HH:MM),
            'close': (HH:MM)
        },
        ...
    },
    'attributes': {
        (attribute_name): (attribute_value),
        ...
    },
}
```

Figure 2: *business data structure*

We first collected all information of restaurants of Las Vegas from Yelp's Academic Dataset and created a new file called 'vegas_business.json'. We then downloaded the inspection data from Southern Nevada Health District website and did a cross check within two files to extract all inspection data corresponding to the Las Vegas business file based on 'business_id'. During this process, there was an address mismatch problem since two files had different formats of ad-

dresses. Also, for the same address, there were sometimes multiple inspections within the same inspection period and we suspected that it might be the situation that for example, there were separate inspections for cafe, restaurants, clubs within the same hotel thus at the same address. We decided to count all the inspections as one facility's (like a hotel) instead of separating them into inspections of different places. At last, we collected all the reviews corresponding to the businesses found in the inspection data and created a new file called 'vegas_reviews.json'.

A restaurant usually has several inspection records. We thus define an "inspection period" of each inspection record "as the period of time starting from the day after the previous inspection to the day of the current inspection"(Kang et al, 2013)[4]. Each inspection period corresponds to an instance in the training or test set. We merged all reviews within an inspection period into one document when creating the feature vector. We also applied filters to the dataset to remove dubious data and noise and got two new datasets: "filtered_noise_reviews.json" and "filtered_deception_reviews.json". The statistics of datasets are shown in table1.

	Vegas business (original)	Vegas business (corresponding to inspection)	Vegas reviews	Filtered noise review	Filtered deception review	Inspection data
Source	Yelp Academic Dataset	Generated from cross checking	Filtered	Filtered	Yelp Academic dataset	Southern Nevada Health District
Document count	4650	1862	181033	129103	123360	90270
Sentence count	N/A	N/A	1725163	1266301	1195225	N/A
Word count	N/A	N/A	20063193	14756653	13849829	N/A

Table 1: *statistics of dataset*

IV. Methods

i. Correlations between Demerits and Different Factors

The feature "demerits" is the penalty score the Nevada Health District gives to each restaurant in each inspection. The higher the demerits is, the more severe a restaurant's violation to the hygienic rules is. Thus we think it is necessary and useful to research the correlation of demerits and factors like volume of reviews and sentiment of reviews. We calculate the Spearman's correlation coefficients between demerits and factors.

We first calculated several statistics of the following factors with respect to demerits with different thresholds [10,15,20,25]:

1. Volume of reviews: count of all reviews; average length of all reviews.
2. Sentiment of reviews: average review rating; count of negative reviews.
3. Deceptiveness of reviews: count of deceptive reviews.

We used deception classifiers based on linguistic patterns (Ott et al, 2011)[7] to filter the deceptive reviews. We obtained the training dataset from Dr. Ott's website (<http://myleott.com/op_spam/>). Since there is no deception corpus of restaurants reviews, reviews of hotels are used as training set in this case. There are 20 truthful and 20 deceptive reviews of each sentiment (positive and negative) for each of 20 Chicago hotels. Truthful reviews are from websites such as TripAdvisor, Expedia, Hotels.com, etc,. Deceptive reviews are from Mechanical Turk.

Before calculating the correlations, we also filtered the database based on two criteria. We first removed noise reviews that are too far away from ($\delta \geq 2$) the average review rating, which resulted in filtrering out one fifth of the data. We then applied the deception classifiers to filter deceptive reviews, which didn't decrease the size of the dataset significantly.

ii. Likelihood ratio

The likelihood-ratio may be the most commonly used statistical test in the analysis of keywords. It is often the case that the counts of common words are not discriminatively enough to give lexical cues of each class. Based on the probability of appearances in both positive class and negative class, likelihood-ratio can give us more discriminative cues. There are two versions of the calculations of

likelihood-ratio. We use the positive likelihood-ratio in our experiments.

The positive likelihood-ratio is calculated as:

$$LR+ = \frac{Pr(T+|D+)}{Pr(T+|D-)}) \tag{1}$$

A likelihood-ratio greater than one indicates that the tested word is associated with positive class, and a likelihood-ratio less than one indicates that the word is associated with negative class. To get an intuitive perception of lexical cues for each class, we can exchange the position of the two classes in the formula.

iii. Support Vector Machines

The goal of support vector machines is to find out the optimal hyperplane which can maximize the margins between classes in the training data. The hyperplane can be shown as in figure3. Given a set of labeled data, SVM algorithms can train a model from these data and assign new examples to one category or the other.

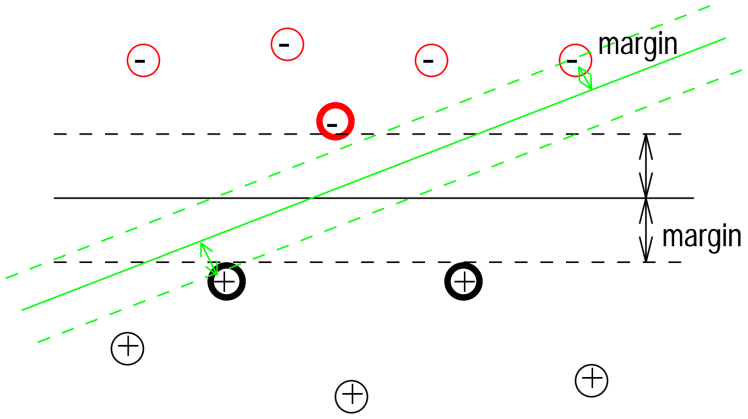

Figure 3: *SVM hyperplane*

Support vector machine has some advantages. For example, it is especially effective in high dimensional spaces. It is still effective even when the number of dimensions is greater than the number of samples. What's more, since the solution to SVM is global and unique, it will not suffer from local minima. Some prior works have shown the effectiveness of using SVM to solve text classification problems, so we also use SVM in our experiment.

iv. Naive Bayes

Naive Bayes classifiers are a set of probabilistic classifiers which apply Bayes' theorem with Naive independence between features. The 'naive' independence means that we assume each feature F_i is independent of every other feature F_j ($j \neq i$), for a category C_k it means:

$$p(x_i|x_{i+1}, ..., x_n, C_k) = p(x_i|C_k) \tag{2}$$

The common decision rule of Naive Bayes model is MAP (maximum a posterior) decision rule. The corresponding Naive Bayes classifier is a function that assigns a class label for certain k as:

$$\hat{y} = \underset{k \in 1,...,K}{argmax}\ p(C) \prod_{i=1}^{n} p(x_i|C_k) \tag{3}$$

Naive Bayes classifiers are simple to implement and their performance are usually not bad. So it's appropriate to use them as our baseline model.

V. Results

i. Correlation

We calculated the Spearman's correlation coefficients of demerits and different factors. These results are shown in figure4, 5, 6, 7, 8.

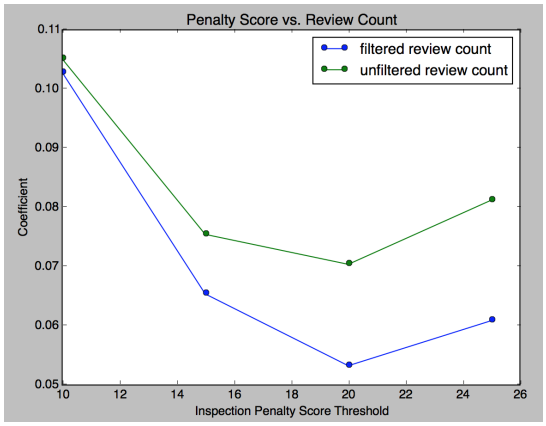

Figure 4: *Penalty Score vs. Review Count*

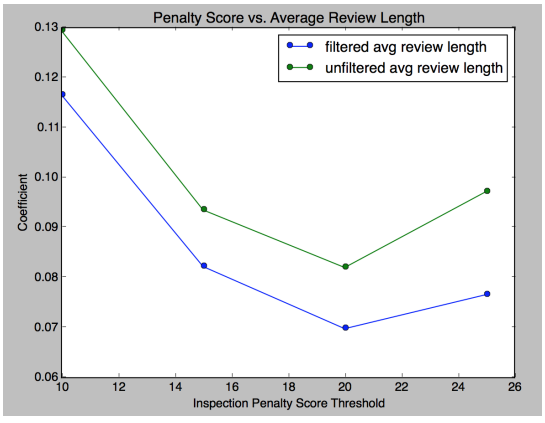

Figure 5: *Penalty Score vs. Average Review Length*

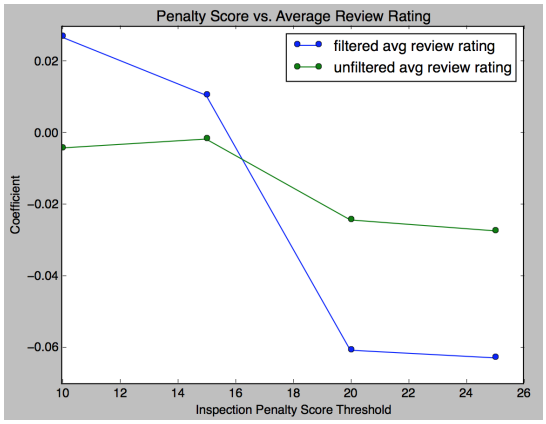

Figure 6: *Penalty Score vs. Average Review Rating*

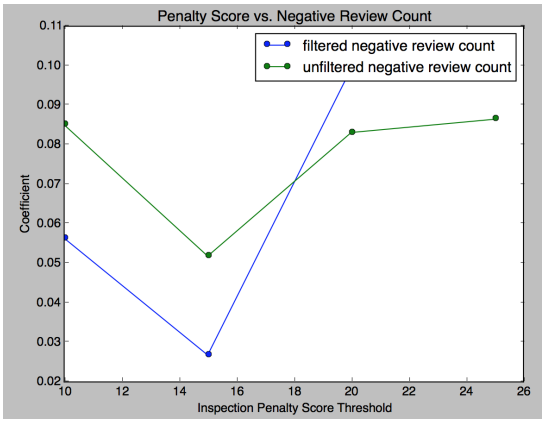

Figure 7: *Penalty Score vs. Negative Review Count*

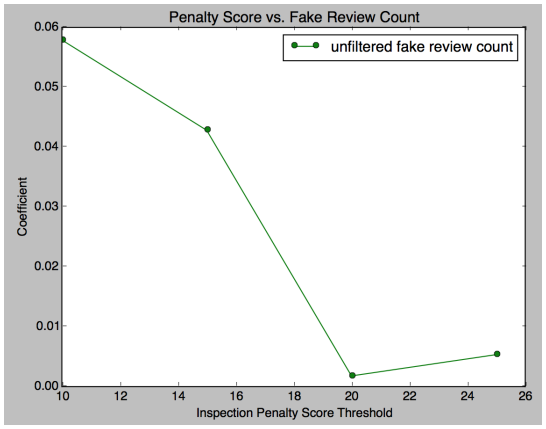

Figure 8: *Penalty Score vs. Fake Review Count*

Figure4 to 8 show the Spearman's correlation coefficients with respect to different factors, filtered (blue line) and unfiltered (green line), at different thresholds of demerits 10, 15, 20, 25.

Average review rating is negatively related to demerits and negative review counts are positively related to demerits, as expected. However, review count, average review length and fake review count are all negatively related to demerits and have a change of tendency at threshold 20, which is the opposite of what Kang et al.[4] obtained in their experiments. In addition, interestingly, the unfiltered dataset usually has stronger correlation with demerits than filtered dataset, which is also opposite to what Kang et al.[4] got. We will analyze it in details in the section Discussion and Future Work.

ii. Content-based prediction

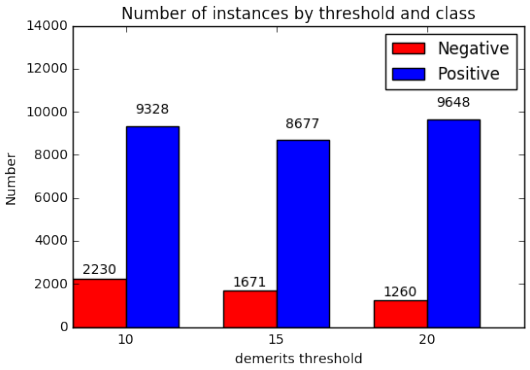

Figure 9: *Number of instances v.s. demerits threshold*

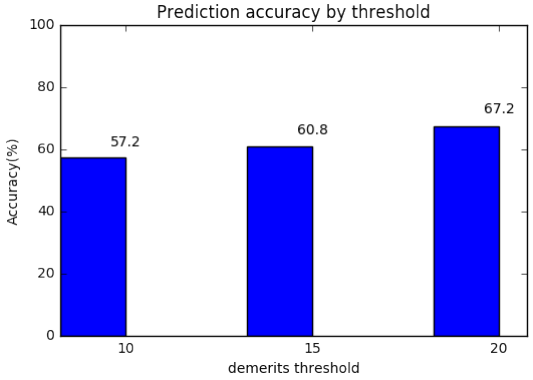

Figure 10: *Prediction accuracy v.s. demerits threshold*

Before the prediction, we create instances from the review dataset. As shown in figure9, we will get different numbers of instances if we choose different demerits thresholds. In the analysis of correlation, we find that threshold = 20 is a turning point. And the Nevada Health District also sets threshold = 20 as an assessment criteria about whether a restaurant severely violates the hygiene rules. What's more, the prediction result using unigram as in figure10 also shows that the accuracy goes up as the threshold increases. Consequently, we use a threshold of 20 to create our training set and testing set.

As figure9 shows, the number of samples in positive class(hygienic) severely outcomes the number of samples in negative class(unhygienic). In our primary test, we find that the imbalanced dataset made the SVM model simply classify almost all samples as positive to reach a high accuracy. For unhygienic class, this led to a high precision but a low recall. To avoid such problems, we reduced the negative dataset, fed the SVM model with 1260 positive samples and 1260 negative samples.

We utilize the following features:

Features based on customer's opinions:

1. Aggregated opinion: averaged review rating.
2. Content of the reviews: unigram, bigram.

Features based on restaurant's metadata:

3. Cuisine: e.g., Chinese, Italian, Indian listed on Yelp.
4. Non-negative review count.
5. Review count
6. Inspection history: a boolean feature ('0' or '1' for previous inspection record 'hygienic' or 'unhygienic')

Features	Accuracy
—	*50.0%
Review Count	*50.0%
Cuisine	62.4%
Non-Positive Review Count	*53.8%
AVG. Rating	*55.2%
Inspection history	*60.5%
Unigram	67.2%
Bigram	*52.9%
All	**98%
Baseline Model(Naive Bayes)	*58.5%

Table 2: *Classification result. [*]: means statistically significantly different from the results of Unigram (p<0.05). [**]: for the combined model, we only use 400 training samples and 80 testing samples.*

Classification Result:

Our baseline method is a Naive Bayes model. This model simply treats each word in the document as a feature and makes use of term frequency and probability. Considering its simple idea, its performance is not too bad actually, reaching an accuracy of 58.5%, higher than random guessing (50%).

We use LIBLINEAR's SVM with L1 regularization and 10 fold cross validation. As said before, we have filtered reviews which are classified as 'noise'or 'deceptiveness'. We also run Support Vector Regression using LIBLINEAR and the results are shown in figure10. We can see that as the threshold increases, the accuracy improves. Table2 shows the results running the SVM with each feature and the combined features at threshold=20.

As shown in the table2, some of the metadata information of restaurants are useful but some are not. Information about reviews such as non-negative review count, review count alone doesn's perform very well. They are not much better than random guessing (50%) while information about restaurants like cuisine

alone can reach an accuracy of 62%.

Some unexpected outcomes appear with aggregated opinion. We expect average review rating to be an effective indicator of hygienic conditions because we suppose that people will give low ratings to restaurants with poor hygienic conditions. However, it seems to perform not much better than chance (55.2%). This result suggests that sentiment cues are not effective in hygiene prediction.

Time factor shows good prediction power in the table2; more specifically, history inspection reaches an accuracy above 60%. This suggests the past performance is relevant to future performance and we might take more time factors into account in the future to improve our model. The results from textual content of the reviews are a little confusing. Unigram model shows better performance than all other features, reaching an accuracy of 67.2% while bigram model is no better than random guessing, achieving an accuracy of only 52.9%. Our guess is because of lack of optimization for bigram model, which leads to bad features for training.

In the end, we combine all features together for training and testing. The feature matrix becomes so large that it is impossible to run on our computers. We wanted to utilize the Swarm cluster for large-scale computing and applied for a user account a week before the deadline. However, we still have no replies from the department. As a result, we reduced the dataset and run our model with 400 training samples and 80 testing samples, achieving an accuracy up to 98%. We expect this abnormally high accuracy to go down when our dataset is larger, and we wish to do that in our future work.

VI. Insightful Cues

We think that there might be some insightful cues in the reviews which indicate whether a restaurant is hygienic or not. An insightful cue can be a word which is common in one class but rare in another. We use the idea of Likelihood Ratio to find them.

Hygienic	clearing, dishes, undercooked
Basic ingredients	octopus, noodle, cucumbers, curry
Cuisine	Vietnamese, Chinese, Japanese, Asian, Indian, Thai, sushi, tempura
Sentiment	dying, inexpensive, disappoints, bothered, inconsistent, ridiculously, good, amazing

Table 3: *Lexical cues - Unhygienic(negtive)*

Basic ingredients	breadsticks, ribs, alfajores, mints, mousse, lamps
Cuisine	Argentinian, McDonald, Domino's, Arby's, Italian, pizza
Sentiment	careful, glorified, brilliantly, delicious, crazy

Table 4: *Lexical cues - Hygienic(positive)*

Instead of splitting the dataset into training set and testing set, here we simply use the whole positive class and negative class to find lexical cues. Table 3 and 4 show representative lexical cues for each class excerpted from the reviews.

Hygiene: An interesting phenomenon is that hygiene related words are mostly seen in negative class, such as 'clearing', 'dishes', 'undercooked'. This suggests that reviewers do complain about the hygienic environment when the restaurant is indeed dirty, but they are unwilling to praise or mention the hygienic condition when the environment has no obvious problems with cleanliness. When the environment is satisfactory, they would focus on other positive aspects such as the details of the food, the atmosphere etc,.

Whom & When: If the reviewers talk about the details of their social occasions, it seems like a good sign of cleanliness which makes sense because people usually become more selective when they decide where to spend their date or weekend.

Sentiment: Sentimental words can be good signals of the classes of restaurants. We can see that in the negative class, people feel 'dying', 'inexpensive', 'disappoints', 'bothered', 'inconsistent', and 'ridiculous' while in the positive class, the reviewers feel 'careful', 'glorified', 'brilliant'. It's also noticeable that people tend to express their feelings when they are not satisfactory with the environment. Another interesting phenomenon is that the hygienic conditions seem to have no influence on the evaluation about the taste of food. We can see words indicating good taste such as 'good', 'amazing', 'delicious', and 'crazy' in both classes.

Basic ingredients: We expect to see little difference between positive class and negative class when it comes to basic ingredients. However, we actually find ingredients like 'octopus', 'noodle', 'curry' to be a bad sign of cleanliness, while 'breadsticks', 'ribs', 'lambs' seem to indicate a good hygienic condition. And the difference may also indicate different cuisine styles which we talk about next.

Cuisine: Like said in the part of content-based classification result, cuisine style is also an effective indicator of hygienic conditions. This is consistent with what we find with likelihood ratio. In general, East Asian restaurants are more likely to have hygienic problems than restaurants with American and European styles. We guess it is relevant to the difference of cooking styles. Asian foods, especially Chinese foods, use lots of pepper, black pepper, canola, and Chinese prickly ash so the cooking process may make the environment smoky or sticky. Another interesting discovery is that fast-food chain restaurants have better hygienic conditions in general. For example, McDownald's, Domino's, and Arby's are all good signs of cleanliness. We think this is because those chains follow a set of standards so they are unlikely to have problems with environment.

VII. Discussion

We have used SVM algorithms to train models and test based on the information and features extracted from review dataset and inspection record dataset. In our experiment, some features alone can reach a good accuracy much higher than random guessing, such as cuisine, inspection history, and unigram. And we also discover that a combined model may bring out better prediction results and we plan to test that on a larger scale in the future. Naive Bayes classifier is proved to be less effective in doing text classification like in our project than support vector machines classifiers because the result of the former is significantly worse than the results of the latter. However, there are still something worth discussing in our experiment.

First of all, we suspect that the inconsistency between our correlation coefficients and those of Kang et al.[4] results from the different datasets used as well as the different cities to study. Kang et al.[4] studies Seattle while we choose Las Vegas. Also, the dataset used in our experiments is only a subset of Yelp's Academic Dataset while Kang et al.[4] "scraped entire reviews written for restaurants in Seattle from Yelp" (p.2, 2013). It's very likely that our dataset is not representative of all reviews of restaurants in Las Vegas, which thus generates biased results.

Second, we notice that bigram model performs pretty bad compared to unigram model and it is even worse than training with history inspection. It is contradictory to our expectation, and we think this results from the fact that we don't optimize the features for the bigram model.

Third, our SVM classifiers are not as effective as those in some prior works. For example, the prediction accuracy of models using unigram and history in-

spection is obviously worse than those in Kang et al.'s work[4]. There are several explanations for this phenomenon. The first is as said above, due to different datasets. Next, our dataset for training is limited in size due to the lack of strong computation power and large memory. Last but not least, we might have missed some steps to preprocess our dataset and make the features more representative and useful.

VIII. Future work

As mentioned before, our first task in future work is to train and test the model using combined features on a larger dataset. We may use the power of clusters or computers with larger memory to achieve that. We expect the accuracy that we have reached on the reduced dataset to decrease but still higher than the accuracy of models using single features. In addition, our experiment with history inspection shows the potential prediction power of time factors. In future, we may take more time factors into account such as the time since last hygienic violation, the sum of past violations etc,. The model may achieve a better accuracy with more time factors. Lastly, during our experiments, we find that the data we collect are severely imbalanced. The number of samples in positive classes is much more higher than the number of samples in negative classes. It is possible that support vector machines will suffer from this kind of imbalance. It's better to provide the SVM with equal samples in both classes. If we can solve the imbalance problem, we can make use of more data to train our model.

Our project is an empirical study of the utility of online reviews to predict the hygienic condition of a restaurant. It shows the promise of opinion analysis of social media on government inspections. It can be a valuable topic in NLP to explore, may be applied into predicting different social aspects and helps the government with a better regulation.

Reference

[1] Eiji Aramaki, Sachiko Maskawa, and Mizuki Morita. Twitter catches the flu: detecting influenza epidemics using twitter. In *Proceedings of the conference on empirical methods in natural language processing*, pages 1568–1576. Association for Computational Linguistics, 2011.

[2] Nitesh V Chawla, Nathalie Japkowicz, and Aleksander Kotcz. Editorial: special issue on learning from imbalanced data sets. *ACM Sigkdd Explorations Newsletter*, 6(1):1–6, 2004.

[3] Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, and Chih-Jen Lin. Liblinear: A library for large linear classification. *Journal of machine learning research*, 9(Aug):1871–1874, 2008.

[4] Jun Seok Kang, Polina Kuznetsova, Michael Luca, and Yejin Choi. Where not to eat? improving public policy by predicting hygiene inspections using online reviews. In *EMNLP*, pages 1443–1448. Citeseer, 2013.

[5] Andrew McCallum, Kamal Nigam, et al. A comparison of event models for naive bayes text classification. In *AAAI-98 workshop on learning for text categorization*, volume 752, pages 41–48. Citeseer, 1998.

[6] Myle Ott, Claire Cardie, and Jeffrey T. Hancock. Negative deceptive opinion spam. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Short Papers*, Atlanta, Georgia, USA, June 2013. Association for Computational Linguistics.

[7] Myle Ott, Yejin Choi, Claire Cardie, and Jeffrey T. Hancock. Finding deceptive opinion spam by any stretch of the imagination. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 309–319, Portland, Oregon, USA, June 2011. Association for Computational Linguistics.

[8] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12(Oct):2825–2830, 2011.

[9] Adam Sadilek, Henry A Kautz, and Vincent Silenzio. Modeling spread of disease from social interactions. 2012.

[10] Johan AK Suykens and Joos Vandewalle. Least squares support vector machine classifiers. *Neural processing letters*, 9(3):293–300, 1999.

[11] Igor Szöke, Petr Schwarz, Pavel Matejka, Lukás Burget, Martin Karafiát, Michal Fapso, and Jan Cernockỳ. Comparison of keyword spotting approaches for informal continuous speech. In *Interspeech*, pages 633–636. Citeseer, 2005.