Imbalanced Classification for Predicting Business Closure

Zijing Di	Zerong Li	Jiangnan Xu	Andrdew Zhang
A14551496	A15689664	A14534652	A15704964
zidi@ucsd.edu	zel003@ucsd.edu	jix209@ucsd.edu	yuz057@ucsd.edu

ABSTRACT

Imbalanced classification has been a widely-studied classification task in real life. In general, it is hard to predict the minor class accurately given the highly imbalanced data.

In this paper, we focused specifically on predicting business closure using the Google Local dataset. We experiment with different classification models, different features such as geographical features, ratings, and compare their effectiveness by using balanced measurement metrics. It turns out that features "*average ratings*", "*reviewText tfidf scores*", "*category scores*" combined and SVM model gives the highest accuracy in terms of F_β-score.

1 DATASET

Google local data contains 11,453,845 reviews on google map/google plus, which covers 3,116,785 individual businesses/places while reviews were made by 4,567,431 individual users all around the world. Local businesses are distributed over five continents, ranging from restaurants, hotels, parks, shopping malls, movie theaters, schools, military recruiting oces, bird control, mediation services, etc.

1.1 Highlight Fields:

Review Data:
- *rating*: a scale number from 0 - 5, higher scale means overall better sentimental reviewing that user leaves for certain business in general.
- *reviewText*: sentences written by users to given business/place.
- *categories*: list if short phrases written by users that describe the categories of the business/place attended.
- *reviewTime*: timestamp of the review (Unix time was also provided)

Business Data:
- *name*: the name of the business/place
- *price*: Yelp styled representation of the price of the place, where "$$$", "$$", "$" are corresponding price from highest to lowest.
- *address*: text address of the business/place
- *hours*: list of open hours for each day in a week.
- *gps*: GPS coordinates of the business/place
- closed: bool value, True if place/business closed down already.

User Data:
- *userName*: name of the user
- *jobs*: list of occupation of user
- *currentPlace*: location provided by user
- *education*: list of education experience of user

1.2 Data Pre-Processing

In order to reduce the complexity of study and narrow down our prediction task, we sampled out all reviews and business data that occurs in California, United States.

To narrow down reviews that occurred in California, we had two potential strategies to use location data to classify and filter out target date set.

Filtering by *GPS* in Business Data: Since precise gps coordinates were provided for most data points in business dataset. A geolocation library geopy[8] was selected to use *GPS* coordinates to get the location of each business. By passing in the coordinates, location information which contains, country, state, city, etc will be returned by API access.

Filtering by *Address* in Business Data : Since most data points in business dataset has descriptive text information on the location of the business/place. We use built-in function of string slicing in Python to get the substring of each location data. Since all addresses of business in the US will have an ending substring of short representation of state name and a fixed 5 digits post number, we used a pattern matching strategy with a key word "CA" to get all business in California. (This strategy could be extended with a dictionary of states in the US to get any target dataset)

After testing with a small sample of original dataset, we found that API call with geopy was too time consuming since error handling caused by connection failing. Rather than getting precise locational data to filter out targeted dataset, which is not sufficient to our prediction task, we decided to use the secondary strategy to narrow down the dataset within an acceptable period of time.

After a series of data cleaning, which includes dropping data with invalid address information, with only including reviews on business in California, we have a total number of 167,758 individual business/places and 778,558 reviews by users. We also decided to drop all user data since most user data have a high ratio of missing fields. Especially for jobs, education and location fields, information was missing for most portions of the original dataset due to privacy and other possible reasons when data originally collected. Also, with consideration on data privacy, we decided to not use any personal information to produce the prediction result.

1.2.1 Imbalanced Data

After reducing the dataset, we found the dataset was extremely imbalanced in terms of closed business versus not closed business. Out of 167,758 businesses, there are only 7,842 closed businesses/places, which means less than 5% of data have a positive label that we aim to predict. Even this distribution was reasonable, we don't expect to see too much closure for businesses, and it shouldn't be, it still increases the difficulty on our prediction tasks. Thus, we planned to perform a series of Exploratory data analysis(EDA) on this reduced dataset.

1.3 Exploratory Data Analysis

1.3.1 Closure rate based on the size of city

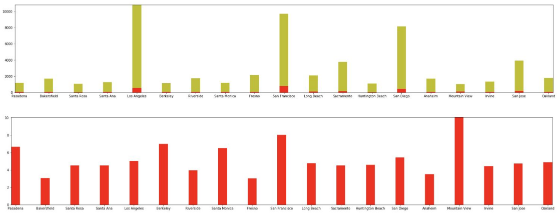
A good way to measure the size of cities that appears in this dataset is to count the number of businesses/places for each city. We applied a similar string slicing strategy that was used to clean data to get all unique cities from business dataset. For each address field, the name of the city is always at the position prior to state name and postal number, where we can slice out each substring of cities in business data points while adding every attendance to a set to keep all unique city names, a roughly representation of cities can be formed effectively. For reduced California dataset, 1754 unique city names were found by this method. After executing a simple filtering method by counting the number of appearances of each city names, it has shown that above 500 cities have more than 20 businesses/places.

Using the same method to explore the business data by changing the threshold of number of businesses. We found that businesses have an unbalanced distribution, which meets the intuition that larger cities have way more business than smaller cities. For instance, 56,960 out of 167,758 businesses are located in 19 major cities out of 500 cities, which means more than 30% of businesses are located in 3% of cities. From this exploration, we decided to compare the closure rate among cities of similar size by ranging number of businesses.

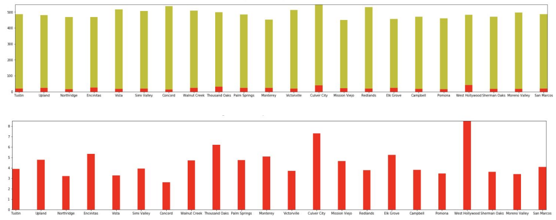
Figure Explanation:

	Top Figire	**Bottom Figure**
Yellow Bar:	Total number of businesses	N/A
Red Bar:	Number of closed businesses	Ratio of closed businesses out of all business

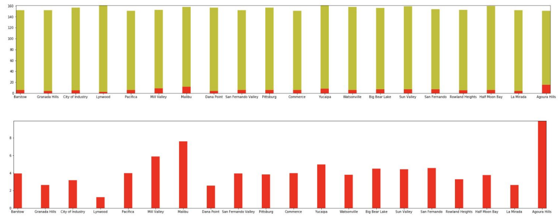
Large cities: selected cities with more than 1000 businesses.



Middle cities: selected cities with a number of businesses within range 450 to 550.



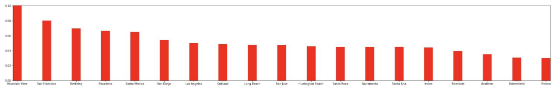
Small cities: selected cities with a number of businesses around 150.



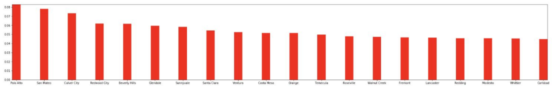
As shown in the bar plot, we can see that larger cities tend to have slightly higher closure rate than small cities, however, the size of cities are not a sufficient factor of affecting closing.

Similarly, we used this method to visualize cities with top closure rate, where red bar in figures represents ratio of closed businesses out of all business.

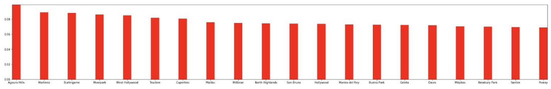
Large cities: selected cities with more than 1000 businesses.



Middle cities: selected cities with a number of businesses within range 450 to 550.



Small cities: selected cities with a number of businesses around 150.



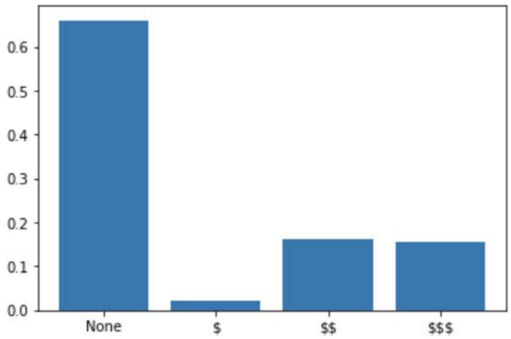
As shown in the bar plot, we can see that larger cities have larger variance in closure rate while smaller cities tend to have lower variance. In a larger picture, we can tell most cities have similar closure rate on businesses.

1.3.2 Prices

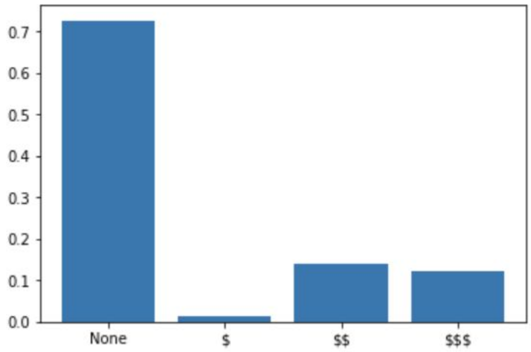
In real life intuition, prices might be a key factor for businesses in sense of continuity. A pattern could be matched from the *price* in business data. We sampled out all 4 categories of prices: “$”, “$$”, “$$$”, and None from the reduced dataset and performed visualization to catch any possible patterns.

Figure Explanation:

X-axis:	Labels of each price class
Y-axis:	Ratio over each label



Prices Pattern for Closed Businesses



Prices Pattern for Opening Businesses

As shown in the bar plot, prices pattern over closed and opening business are extremely close. A fair and intuitive explanation is the dataset are equally distributed on all businesses/place of different categories, which prices could have less affection on closing. It conflicts with our assumption, which might be more precise for businesses like restaurants and cafes.

1.3.3 Hours

There are detailed open hours for each business/place provided in the *hours* field in business data. We made an assumption that open hours of closed business might differ from opening businesses.

Since the *hours* field represents open hours in a user friendly way, which uses a 12-hour clock(time with am/pm). We parsed all hours from 12-hour clock to a 24-hour clock in float number form.

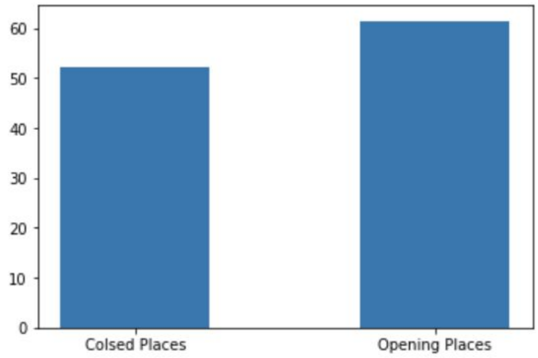
For instance, an input hours list in json object of hours is:

```
[['Monday', [['11:30 am--2:00 pm'], ['5:00--10:00
pm']]],
['Tuesday', [['11:30 am--2:00 pm'], ['5:00--10:00
pm']]],
['Wednesday', [['11:30 am--2:00 pm'], ['5:00--10:00
pm']], 1],
['Thursday', [['11:30 am--2:00 pm'], ['5:00--10:00
pm']]],
['Friday', [['11:30 am--2:00 pm'], ['5:00--10:00
pm']]],
['Saturday', [['12:00--2:30 pm'], ['5:00--10:00
pm']]],
['Sunday', [['12:00--2:00 pm'], ['5:00--9:30
pm']]]]
```

Our API will transform it into a better format for further operation, which the output is:

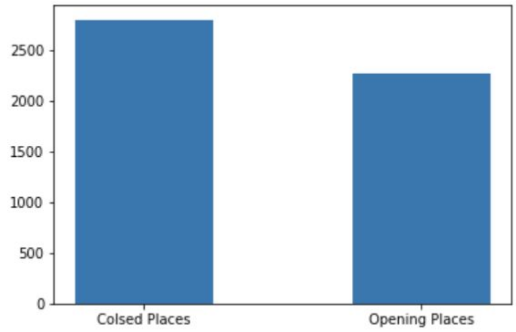
```
{1: [(11.5, 14.0), (17.0, 22.0)],
2: [(11.5, 14.0), (17.0, 22.0)],
3: [(11.5, 14.0), (17.0, 22.0)],
4: [(11.5, 14.0), (17.0, 22.0)],
5: [(11.5, 14.0), (17.0, 22.0)],
6: [(24.0, 14.5), (17.0, 22.0)],
7: [(24.0, 14.0), (17.0, 21.5)]}
```

And the total open hours in this case would be *27.5* hours. In addition to the above, edge cases like `['Open 24 hours']` will be transformed to `[(0,24)]` and `['Closed']` will be transformed to `[(0,0)]`. With infrastructure built, we visualized average open hours between closed and opening businesses/places. and the result shows that it could be a sufficient feature that we should try when doing prediction tasks.

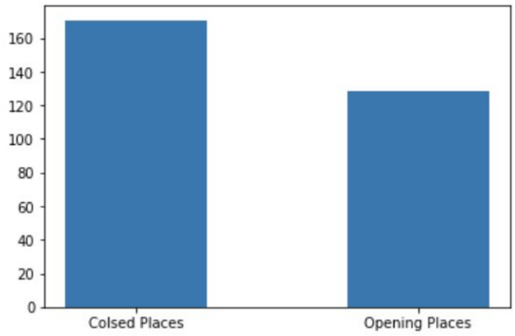


Avg Hours: Closed vs. Opening

1.3.4 Surrounding Situation
We sampled cities with different sizes in section 2.3.1, which doesn't provide a sufficient result we need in order to use locational wised information on our prediction. We choose to explore locational data in a reversed way compared to the process in 2.3.1. An assumption we made was that the surrounding situation could be a key factor affecting closure. Intuitively, if a certain amount of business closed surrounds other businesses in the same area, then a business situation/pattern could be summarized. We use city names as keys to search backwardly for other businesses around one business to find this pattern. We sampled out the average number of surrounding businesses/closures to visualize.



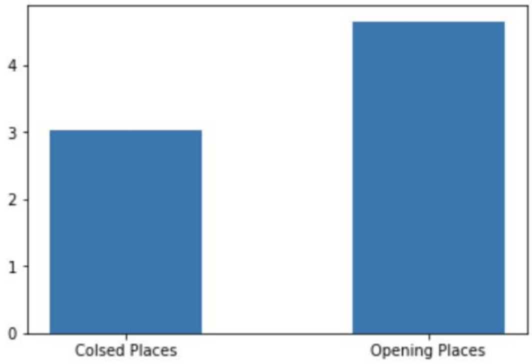
All Business Surrounded: Closed vs. Opening



Closed Business Surround: Closed vs. Opening

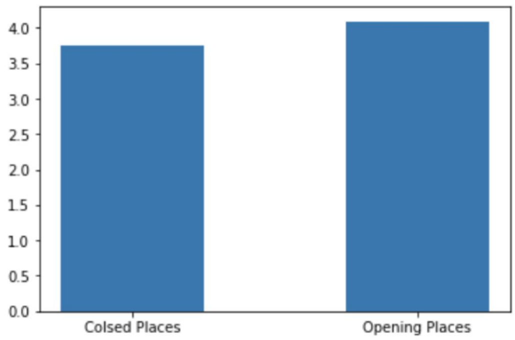
As shown in the bar plot, a strong pattern was matched in both figures, where closed businesses tend to be locational wised closed to other closures. Also, the first figure confirms that cities that have more businesses tend to have higher closure rate, which was also mentioned in section 2.3.1.

1.3.5 Number of Reviews
Simply count the number of reviews of each business and take an average between opening and closing businesses. This figure shows that closed businesses tend to have less reviews.



Average Number of Reviews: Closed vs. Opening

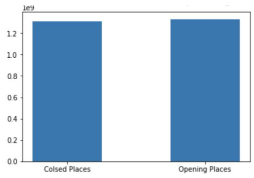
1.3.6 Rating
Intuitive reason for the closure of any business would be lower overall ratings, which is also confirmed by this figure.



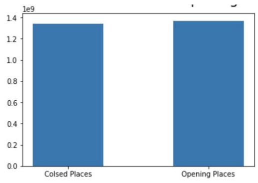
Average Rating: Closed vs. Opening

1.3.7 Review Time

All review time was provided in *reviewTime* field in review data, and it is possible that some pattern could be summarized from either the timestamp of earliest review, latest review, and also the time interval between those two. Since a nice Unix time format was provided in the original dataset, we built an API which iteratively finds the timestamp of earliest and latest review of each business.

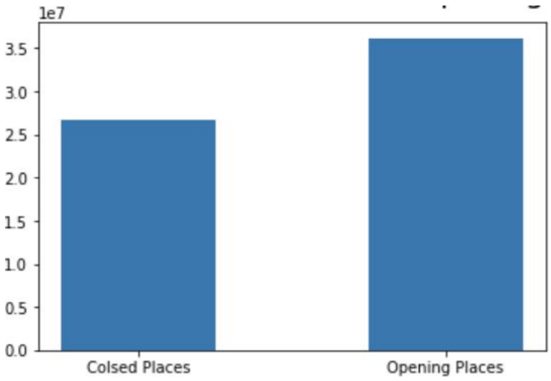


Earliest Timestamp



Latest Timestamp

It is reasonable that the earliest and latest timestamp for reviews turns out to be similar, because the original review dataset was equally distributed each year. However,figures of time intervals are confidently shown that closed businesses tend to have shorter time between the first and last reviews.



Timestamp Intervals: Closed vs. Opening

1.4 Interesting findings

1.4.1 Inaccurate Descriptions

In section 2.3.2, we found that prices were not sufficient to our prediction tasks. Also, from distribution/pattern that shown in figure in 2.3.2, we noticed that the *price* field of more than 60% of businesses are *None*. So we look into this part of data and find couple unreal/inaccurate descriptions:

```
'name': 'Walmart Neighborhood Market','price': '$$$'
'name': 'Taco Bell','price': '$$$'
'name': 'Wingstop','price': '$$$'
'name': 'Quiznos Sandwich Restaurants','price': '$$$'
'name': 'Burger King', 'price': '$$$'
```

Which shows that the *price* field in this dataset seems to be messed up in some certain way, and is not reflecting the actual situation of some businesses.

1.4.2 Highest Closure Rate

City with the highest closure rate is Mountain View, which is also where Google's headquarters is. This is an interesting finding because this dataset is provided by Google and some field names in this dataset start with “gplus” which refers to Google Plus, it was not too hard to imagine that part of this dataset was collected from Google Plus platform.

Google Plus is a social network platform which was completely shut down in Apir, 2019.

Indeed, it was because of Mountain View.

2 PREDICTIVE TASK

2.1 Task and Baseline Description

Our task is to predict whether a certain business will close based on users’ and business’ information from the *Google Local* dataset. More specifically, our prediction is based on past average rating (0.0-5.0), business category, and review text. For the baseline, we only take average rating into consideration. The reason why we choose average rating as prediction baseline is that it represents the popularity of a business. Intuitively, businesses with higher ratings are more popular and thus are more likely to last longer. Thus, using average ratings to predict the possible shut-down is a simple, reasonable way.

2.2 Features Selection and Description

After our EDA process, we found several features that differ dramatically between closed and opening businesses: closure rate based on the size of cities, prices, opening hours, surrounding situations, number of reviews, ratings, review time, and they are the possible candidates of our model. We try different feature combinations to our model and compare their accuracy, precision, FPR and F-Beta score to select the best combination. We finally choose to incorporate average rating, business category and review text as features into our model. The basic idea of extracting those three features from dataset is described as following: average rating for each business is calculated by first putting the business and all its corresponding reviews (including ratings) into a dictionary, and then calculating the average rating for each business; business category is obtained by extracting keywords from review texts using TF-IDF, and then take the average of TF-IDF vector for each business; review text is also obtained by utilizing TF-IDF and taking the average TF-IDF vector of each business. As it is shown, the features we choose are compatible with our EDA results. EDA shows that the average rating of closed business is lower than that of still-opened business, with 3.75 out of 5.0 for the closed business and 4.09 out of 5.0 for the opening business, which means that average rating is a factor that we should consider in our prediction.

2.3 Model Selection

Possible candidates for our business open/closure classifiers are Naive Bayer's, Logistic Regression, and Support Vector Machine(SVM). Naive Bayers only works when the features are conditionally independent; Logistic Regression minimize logistic loss and it is sensitive to outliers. Since the features of our data are not independent with each other, for example, the keywords from review text are somehow dependent on the business category, we cannot use Naive Bayes here. Compared to the two models, SVM works better because SVM can apply on highly dimensional data and tries to find the separating hyperplane that maximizes the margin between the closet support vectors and thus it would find a solution that will keep the two categories as far as possible. Here, we choose SVM as our classifier and set C = 10, because we want to reduce misclassification. When we want to extract important keywords from review text, the possible tools we can apply are Bag of Words and TF-IDF. Bag of Words uses the most frequently-occurred words to represent the text however it might not be representative since number of the word occurrences doesn't necessarily represent the importance of the words. Compared to Bag of Words, TF-IDF is a more appropriate tool, because it highlights the most unique and frequent words for a single document and thus we choose TF-IDF to extract keywords from review text.

2.4 Measurement of Performance

In the stage of exploratory data analysis, we explored the factors that influence the opening and closing of restaurants. To analyze how good our modes are, we analyze and compare their F_β score, precision, False Positive Rate (FPR), and accuracy.

F_β score: we set β to 0.5, this is because precision is more important and to weight precision, we need to pick a β value in the interval $0 < \beta < 1$.

$$F_\beta = (1+\beta^2)\,\frac{precision * recall}{\beta^2 * precision + recall}$$

Precision: attempts to answer what proportion of positive identifications was actually correct. In other words, it shows what proportion of businesses that are predicted to be closed are actually closed. The higher the precision, the better the model is.

$$Precision = \frac{TP}{TP+FP}$$

FPR: Label that is predicted as True Positive means the business is predicted as "closed" where it is actually closed. Labels that are predicted as False Positive means the business is predicted as "closed" where it is actually opened. We choose to use False Positive Rate over False Negative Rate because the goal of our classifier is to predict whether a business is closed or not and False Positive Rate directly shows how many businesses that are predicted to be closed are correct. In addition, in our model, false negatives are nuisances but false positives are disastrous. And the lower the FPR, the better the model is.

$$False\ Positive\ Rate\ (FPR) = \frac{FP}{(FP+TN)}$$

Accuracy: indicates the proportion of correct results among the total number of cases examined.

$$Accuracy = \frac{(TP+TN)}{(TP+TN+FP+FN)}$$

We didn't choose FNR because it is more harmful for a business to be incorrectly predicted to be shut-down (FPR) than it is incorrectly predicted to be not (FNR), and we need a more insightful standard to measure our model.

3 MODEL

First, We develop a baseline model for our classifier. For the baseline model, we only consider the factor of average rating because the average rating of opening business is 4.09 out of 5.0 while the average rating of closed businesses is 3.75 out of 5.0. Data is splitted into 80% of the training set and 10% of the validation set. Each index in the feature vector contains the average rating of that business and y is a vector containing labels that represents whether the business is actually closed or opened. After two vectors are built, Logistic Regression is used to predict the result. As a result, the validation set of our baseline model is scored with precision of 0.068, recall of 0.492, FPR of 0.325, accuracy of 0.667 and F_{β} score of 0.852.

Portion/Statistics	Training Set	Validation Set
Precision	**0.065**	**0.068**
FPR	**0.323**	**0.325**
Accuracy	**66.7%**	**66.7%**
F_{β}	**0.849**	**0.852**

Comparison Between Training Set and Validation Set of the Baseline Model

Adding more related terms to the feature vector is a way to optimize the performance of the model. Therefore, categories of the business and review text of each business can be considered.

For modeling the categories, we first collect all categories across all businesses in the dataset and use TF-IDF to pick the most 100^{th} popular words of categories. Since there can be more than one review for a business, every word of the category in a review of the same business is reviewed and mapped to the TF-IDF vector of that business. After all review of the same business is reviewed, the average of the accumulated TF-IDF vector is calculated.

For modeling the review text, we use TF-IDF again to pick the most popular words out of all reviews. Similar to how categories are modeled, the average is taken for the accumulated TF-IDF vector with all reviews of a business.

As a result, the "best model" we have built is adding the average rating, the category TF-IDF vector and the review text TF-IDF vector of a business to the feature vector. This time, SVM model is used to make prediction rather than Logistic Regression because SVM model tends to optimize the number of mistakes and the result shows that the SVM results in higher precision, higher accuracy and higher F_{β} score than using Logistic Regression with the same penalty parameter C = 1.0. Again, the y is a vector containing labels that represents whether the business is actually closed or opened. We fit our newly built model to Linear SVM with penalty parameter C = 10. As a result, the training set of the "best model" is scored with precision of 0.267, FPR of 0.364, accuracy of 0.931 and F_{β} score of 0.931. And the validation set of the "best model" is scored with precision of 0.230, FPR of 0.04, accuracy of 0.927 and F_{β} score of 0.929.

Model/Statistics	Baseline	Best Model
Precision (higher is better)	**0.068**	**0.230**
FPR (lower is better)	**0.325**	**0.04**
Accuracy (higher is better)	**66.7%**	**92.7%**
F_{β} **(higher is better)**	**0.852**	**0.929**

Comparison Between Baseline Model and the Best Model

3.1 Issues:

There is a problem of overfitting in most models we tried along the way. We are trying hard to solve the problem of overfitting. We know that overfitting is caused by making a model more complex than necessary, and the relationship between penalty parameter C and the lambda regularizer is

$$C = \frac{1}{\lambda}$$

Therefore, one way to solve overfitting is decrease C (increase λ, trade-off accuracy versus complexity). However, results indicate that decreasing C did not solve overfitting for our models (more potential models will be described later). We also attempt to improve the model by modifying the feature vector.

3.2 Other Unsuccessful Attempts and Comparison:

Candidate A: Using the business operating hours weekly because the average business hours of opening businesses is 61.42 hours per week while the average business hours of closed businesses is 52.18 hours per week. In addition, the number of closed businesses around is another factor as well because the average closed businesses around a closed business is 171 while 129 closed businesses for opening business. Furthermore, the number of reviews left in a business is also considered because closed businesses received 3 reviews per business while opening businesses received 5 reviews per business. That is, using the average rating, total business operating hours weekly, the number of closed businesses around, the number of reviews a business received and categories.

Candidate B: Inspired by the lecture of text mining, we think that review text could dominate the decision of predicting whether a business is closed or not. Therefore, we concatenate all reviews of the same business into one long review text and apply TF-IDF with unigram and bigram for the whole review text without picking a bag of words.

Model/Statistics	Baseline	Best Model	A	B
Precision (higher is better)	0.068	0.230	0.172	0.103
FPR (lower is better)	0.325	0.04	0.012	0.139
Accuracy (higher is better)	66.7%	92.7%	94.5 %	83.6 %
F_β (higher is better)	0.852	0.929	0.924	0.904

Comparison Between Baseline Model, the Best Model, Candidate A and Candidate B

As the table shown above, Candidate A wins in terms of accuracy and False Positive Rate. However, the difference between Candidate A and the "best model" is relatively small. In addition, the "best model" tends to have a higher precision across our attempts. As we can conclude from the table that Candidate A has a lower precision score and F_β score compared to the "best model" we have obtained. Furthermore, we prefer the "Best mode" since it has less features and is simpler. Candidate B did not perform as expected.

4 RELATED WORK

4.1 Google Local Dataset Usage

The dataset we used, Google Local dataset, is collected by Professor Julian McAuley and can be directly accessed through his website https://cseweb.ucsd.edu/~jmcauley/datasets.html. The dataset is used to experiment the effectiveness of models *TransFM* and *TransRec*. *TransFM* and *TransRec* are used for sequential recommendation techniques. Unlike common recommender systems, sequential recommender system takes the sequence of user's interactions into account, i.e. it predicts the user's next interaction according to the trend of previous interactions.

Since Google Local Dataset contains a large amount of user infoes, business ratings and reviews, and geographical information, it is used to test the models with various feature settings. For *TransFM*, the authors evaluate businesses from six U.S. states (Pasricha and McAuley, 2018) and predict the user's next buKyle Carbon, Kacyn Fujii, and Prasanth Veerina. 2014. Applications of machine learning to predict Yelp ratings. Stanford Univ., Stanford, CA.siness to visit according to previous visit records. For *TransSec*, the authors preprocess the data by discarding users/businesses with less than 5 interactions associated with them and taking all existing ratings as positive feedback. The dataset is used for the same predictive task for both models, and it shows that *TransFM* surpasses *TransSec* with more content features.

4.2 Other Similar Datasets

A very similar dataset to Google Local Dataset is the Yelp Dataset. It can be accessed through Yelp website https://www.yelp.com/dataset and is updated regularly for Dataset Challenge. Yelp Dataset also contains subsets of users, businesses and reviews. The file *business.json* contains businesses in selected cities and each business has attributes *"business_id", "name", "address", "city", "state", "postal code"*, etc. where most of them contain similar information about each business as in Google Local. *review.json* has "*stars*", similar to "*rating*" in Google Local, "date", "text", etc. *user.json* contains more information about the user, such as "*friends*", "*review_count*", "*elite*", which do not appear in Google Local and hence Yelp Dataset can inspire more recommendation tasks utilizing social networks or user similarity. With highly similarities between Yelp

Dataset and Google Local Dataset, there are tasks performed to predict business closure using Yelp Dataset with slight variation.

For example, Lu et al. predict future success of Yelp restaurants by predicting closure in next year (2018). If a restaurant was open during 2016, it should be predicted to succeed if the restaurant was still open during 2017. The authors first preprocess the dataset and only keep businesses and related users, reviews if the businesses were open during 2016. Then in the next step, the authors randomly sampled 1047 restaurants that were open during 2016 and open during 2017 as the positive samples, and sampled another 1047 restaurants that were open during 2016 but closed during 2017 as the negative samples (Lu et al. 2018). They encountered the same imbalanced data issue as us, i.e. there are much more open businesses than closed businesses. But instead of using balanced measurement metrics like us, they resampled the data to be balanced. They separated features into Text Features and Non-Text Features. Below is a table of features they used.

Text Features	**Unigram**	**Good**
		Bad
	Bigram	**Sanitation**
		Location
		Service
		Taste
Non-text Features	**Trend**	**Review Star Loss**
	Bussiness	**Review Count**
		Chain Restaurant
		Return Guest Count
		Restaurant Type
	Location	**Nearby Restaurant Comparison**
		City Economic Status

As there are multiple reviews for each business, the authors process text data in the same way as we did in one of the models using "*reviewText*" --- They concatenate all reviews towards the same business and treat it a single review. Then they use unigram to extract good/bad words, and use bigram to extract pairs of words with certain representations (2018). For non-text features, they evaluate multiple trends in terms of time and different locations.

For model selection, the authors choose logistic regression as well for the same reason that we are performing a binary classification task. Furthermore, they split train/test in 90/10 ratio which is similar to ours 80/10/10 ratio for train/val/test.

The Yelp Dataset is used for other predictive tasks as well. For example, many researchers are interested in predicting business ratings so that they can recognize which features lead to higher ratings (Carbon et al. 2014). Carbon et al. experiment with SVM, logistic regression, Naive Bayes, etc. They extract the importance of each feature using across feature selection. It turns out that non-text features such as "location, price range, and the option of take-out have significant predictive power." and text-feature such as text sentiment has an even stronger effect on the prediction accuracy (Carbon et al. 2014). They draw a conclusion suggesting that if a business can improve a feature with stronger prediction effect, it can improve its rating as well.

4.3 Other Similar Imbalanced Predictive Tasks

Imbalanced data issues often arise in bankruptcy prediction which is pretty close to our closure prediction (although many businesses were not closed due to unsuccess while most businesses bankrupted due to unsuccess). Many researchers deal with issues using oversampling, undersampling, and other advanced techniques.

In "Performing of corporate bankruptcy prediction models on imbalanced dataset: The effect of sampling methods", the author compares six sampling methods including ROWR, SMOTE, RU, UBOCFNN, etc and five quantitative models including SVM (Zhou 2013). The author also uses measurements F_β score for imbalanced data. In the end, the author concludes that which sampling performs the best depends on the number of bankruptcies in the training set, but in general SVM performs well regardless of types of samples (Zhou 2013).

4.4 State-of-the-Art Methods

To deal with imbalanced data, in general researchers implement sampling to train on a more balanced dataset, or use advanced models such as neural networks, clustering to overcome the problems.

4.4.1 Random oversampling with replication (ROWR). This is an oversampling technique in order to produce a balanced dataset with equal number of data in majority and minority class. The basic idea is to replicate data from the minority class until we gain an evenly split dataset.

4.4.2 Synthetic Minority Over-sampling Technique (SMOTE). This is also an oversampling technique. It is slightly different from ROWR in the way that SMOTE doesn't oversample the minority class by replicating, but by synthesizing minority class example using k nearest neighbor algorithm.

4.4.3 Random Undersampling (RU). This is an undersampling technique. Opposite to oversampling technique, it produces a balanced dataset by randomly selecting majority class samples until the majority class sample size is the same as the minority class sample size.

4.4.4 Undersampling Based on Clustering from Nearest Neighbor (UBOCFNN). This is an undersampling technique. It reduced the size of majority class sample using clustering such that only one representative in each cluster is selected and the number of clusters is the same as the minority sample size.

4.5 Comparison

In "Should I Invest it? Predicting Future Success of Yelp Restaurants.", the authors use Yelp dataset and conclude that non-text features play an more important role in predicting Restaurants success (i.e. openness until the next year) than text features. (Lu et al. 2018). Furthermore, features "*chain restaurant*" (whether a restaurant appears more than two times), "*nearby restaurant comparison of star ratings*", and "*city economic status*" are found out to be the most important ones (Lu et al. 2018). In contrast, we find out that text features are more important than others for Google Local Dataset. The best model so far incorporates "*average rating*", "*category TF-IDF scores*" and "*review text TF-IDF scores*" which emphasizes that text-features are very important to predict business closure. We didn't try to utilize "*chain business*" feature and it is worth trying in the future work. We did try to compare businesses in the same city but the result is not ideal, and we think that is due to the overwhelming effect of the majority class samples such that few nearby businesses contain decisive information about the minority class samples.

Additionally, the way we sample the dataset is different from the existing work. We did not use any undersampling or oversampling techniques, rather we keep the train/validation/test datasets imbalanced which requires us to use balanced measurement metrics instead of accuracy because always predicting "false" can lead to incredibly high accuracy given very few negative instances. In contrast, Lu et al. preprocess the data using undersampling technique (2018). They intentionally choose a certain number of positive samples to produce a balanced dataset. In this case, they can directly use accuracy to verify model effectiveness. However, predicting on an undersampling dataset may not work in real life settings.

Furthermore, our models are different. With imbalanced data, we choose SVM because it is designed to minimize misclassification instead of merely ensuring accuracy. Lu et al. instead choose logistic regression to perform the predictive task.

5 CONCLUSION

5.1 Summary of model election

In all, we split our whole dataset into 80% training set, 10% validation set and 10% test set, and we want to predict whether a certain business will close or not based on its current situation. According to our EDA results, we tried three different models, the first model is using the average rating, total business operating hours weekly, the number of closed businesses around, the number of reviews a business received and categories and fitting those features into Linear SVM (penalty C = 10). The second candidate model we tried is concatenating all reviews of the same business into one long review text and apply TF-IDF with unigram and bigram for the whole review text and then put that TF-IDF matrix. The third model we tried is fitting average rating, business category TF-IDF vector and review text TF-IDF vector into linear SVM (penalty C = 10). After comparing their performance based on accuracy, precision, FPR and F-Beta score, we choose the third model as our final model. Compared to baseline with 0.068 as precision, 0.325 as FPR, 66.7% as accuracy and 0.852 as F-Beta Score(we only use use average rating as a single feature), our model achieve 0.23 as precision, 0.04 as FRP, 92.7% as accuracy, and 0.929 as F-Beta Score.

5.2 Explanation of Feature Selection

The features in other models such as opening hours, review time, and prices don't work well might be because those factors are not correlated with business closure. For example, opening hours is not a good indicator for business closure even though EDA shows that closed business generally have shorter opening hours. This is because the business category can highly affect opening hours. Our dataset contains different business categories such as natural park, night bar, museum which have different business hour standards. In this case, we can say that it is the business category but not opening hours that actually affects business shut-down. That's why we include categories but not opening hours as one of the useful features into our final model, and when we try incorporate business hours into our model, we get a lower precision and a lower F-beta Score. For similar reasons, we discarded other useless features and after we tried different features combinations, we finally selected average rating, business category TF-IDF vector and reviewed text TF-IDF vector to train our model.

5.3 Parameter Interpretation

We use linear SVM as our final model, and the parameters are: C=10, class_weight='balanced', random_state=0, tol=1e-5. In SVM, we consider two parts: a hyperplane with the largest minimum margin, and a hyperplane that correctly separates as many instances as possible. C represents the penalty parameter of the error term, which defines the trade-off between misclassification and margin. A large value of C means that we want to increase the penalty for misclassification, and pay less attention on choosing a smaller-margin hyperplane. For the 'class_weight', we set it as 'balanced'. The 'balanced' mode uses the values of y to automatically adjust weights inversely proportional to class frequencies in the input data as n_samples/(n_classes * np.bincount(y)). We set the 'class_weight' as balanced because our dataset was extremely imbalanced in terms of closed business versus not closed business. Out of 167,758 businesses, there are only 7,842 closed businesses/places, which means less than 5% of data have a positive label that we aim to predict. Thus, we use the 'balanced' parameter as class weight to balance our dataset in order to make a more accurate prediction. We then set random_states as 0. A fixed value guarantees that the same sequence of random numbers are generated each time we run the code, and the results produced will always be the same. Tol represents tolerance for stopping criteria.

5.4 Application of predictions

Our prediction task is predicting possible future shut-down based on current business situation. If our prediction can achieve even higher accuracy, we could estimate risk for a bank to loan to a certain business owner. Additionally, we can provide improvement suggestions for businesses that are highly likely to close by comparing them with the ones that are not predicted to be shut-down.

5.5 Limitations and Feature Work

However, our Precision is not ideal, which is only 0.23, which means that among all the businesses we predicted as closed, only 23% of them are closed in reality. It might be the case that our precision is affected by the limitations of our dataset. Firstly, since we are predicting possible future shut-down based on current business situation, we don't have actual true labels (future business status) to test our performance, and thus we could only use the current business status as our 'temporary' true labels, which might lead to lower precisions. Secondly, unlike Yelp dataset which has universal categorical terms defined for business, each user in Google Local can put their personalized defined categorical term for the same business. And since we use category as one of our features, we can only extract categories from review text, which can be less accurate. Thirdly, lack of 'reviews per user' limits our ability to improve the performance of our model. We planned to do the prediction using Jaccard similarity between closed business and the target business. However, we find that less than 10000 users have more than 5 reviews, which makes it hard to utilize user-based similarity between business to make the prediction.

Thus, to make further progress, we need to collect more relevant data and add more reasonable features to our model. We are thinking about adding more background information like the economic environment and population size of the city that the businesses are in by using API, because the big environment can have significant effect on the business shut-down. Also we could add time series analysis, like business popularity changes with time based on review and ratings in timeline.

REFERENCES

[1] Rajiv Pasricha and Julian McAuley. 2018. Translation-based factorization machines for sequential recommendation. In *Proceedings of the 12th ACM Conference on Recommender Systems (RecSys '18). ACM, New York,* NY, USA, 63-71. DOI: https://doi.org/10.1145/3240323.3240356.

[2] Ruining He, Wang-Cheng Kang, and Julian McAuley. 2017. Translation-based Recommendation. In *Proceedings of the Eleventh ACM Conference on Recommender Systems (RecSys '17). ACM, New York,* NY, USA, 161-169. DOI: https://doi.org/10.1145/3109859.3109882.

[3] Xiaopeng Lu, Jiaming Qu, Yongxing Jiang and Yanbing Zhao. 2018. Should I Invest it? Predicting Future Success of Yelp Restaurants. In *PEARC '18: Practice and Experience in Advanced Research Computing, July 22-26, 2018, Pittsburgh,* PA, USA. ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3219104.3229287.

[4] Kyle Carbon, Kacyn Fujii, and Prasanth Veerina. 2014. Applications of machine learning to predict Yelp ratings. *Stanford Univ., Stanford,* CA.

[5] Ligang Zhou. 2013. Performance of corporate bankruptcy prediction models on imbalanced dataset: The effect of sampling methods. *Know.-Based Syst. 41 (March 2013), 16-25. DOI=http://dx.doi.org/10.1016/j.knosys.2012.12.007*.

[6] Tuong et al. 2018. A Cluster-Based Boosting Algorithm for Bankruptcy Prediction in a Highly Imbalanced Dataset.

[7] Myoung-Jong Kim, Dae-Ki Kang, and Hong Bae Kim. 2015. Geometric mean based boosting algorithm with over-sampling to resolve data imbalance problem for bankruptcy prediction. *Expert Syst. Appl. 42, 3 (February 2015), 1074-1082.* DOI=http://dx.doi.org/10.1016/j.eswa.2014.08.025

[8] MIT License (MIT) https://pypi.org/project/geopy/ https://github.com/geopy/geopy/blob/master/LICENSE