Analysis of Yelp dataset

Team "The Data Analytics Group"

Akshay Karki (avk1063@rit.edu)
Gaurav Gawade (gdg6776@rit.edu)
Shristika Yadav (sy2109@rit.edu)

Keywords

yelp, dataset, businesses, analysis, check-ins, time, plot.ly, api, json, python, pandas, matplotlib, scipy, users

ABSTRACT

Yelp ratings are often seen as a respectable metric for businesses. But, how can we better something which is already good? Wouldn't it be cool if we could predict the number of stars a business would get just by looking at the number of stars it gets on each review? Or we could predict the busiest hours a business should expect based on the frequency of people? These are some of the questions that we hope to tackle with this project.

1. INTRODUCTION

With the increasing dependence on technology, more and more businesses of all kinds are looking at data analysis as an avenue they have to venture into, to keep up with others. Many businesses are trying to find out patterns and trends at a larger scale to find out when they might be attracting customers. This forms the basis for their business model. In this report, we look at the Yelp dataset to analyze the different businesses and when they are busy with the customers over the span of the entire week.

The dataset contains about 1.6M records. The dataset is spread over multiple JSON files and the data in all the files is interlinked. The task involved in this project will be to parse the JSON files using Python and Pandas library. The parsed data will then be stored in a dictionary and then further used for visualizing trends and analyzing patterns.

There is a review file where multiple users have reviewed a business by giving it some ratings as well as comments. A business file consists of all the demographic information for that business while a user file gives information about the reviewer. A check-in file tells which time the business is usually the busiest at.

With the given dataset, some of the important questions which we will be able to answer are:

- How well a business is doing in terms of check-ins
- Is the business one of the best in its city
- Depending on user reviews what is the next business he will review
- What are the best businesses in a given category
- Where is the business located and how does it compare with other businesses near the same location

The report is divided into the following sections. Firstly we talk about why we wanted to do this Yelp analysis project in our 2 section. Then we describe the tools used and the process that we implement in the 3 section. In the next section 4 on lessons learned, we talk what were the important lessons that we learned along the course of this project. We finally talk about our plans for the future and conclude this report in the 5 and 6 sections respectively.

2. PROJECT MOTIVATION

The project was undertaken keeping in mind that it is of utmost importance for businesses to understand their customers. It is critical that businesses plan according to how many customers they are likely to encounter and during what time of the day in a given week.

Yelp [3], being one of the most popular application for businesses, generates a lot of data. We used this data to understand the different ways in which businesses attract customers. In this report, we also see how widespread the businesses are located across the world depending on their geographical location.

3. IMPLEMENTATION

The implementation part of the project will mainly consist of three stages such as dataset collection/assembly, clean data and analyze, and data mining and visualization. Our main focus will mainly be on cleaning the data as much as possible in order to have optimum analysis result [2] [1]. The data mining algorithm that will be used will be KMeans and visualizations will be done using the Python libraries.

.

3.1 Tools used

In this section, we will consolidate the data collected from various sources for Yelp [3]. Given the dataset and how it could be analyzed, we used Python to code. We also used Python libraries like SciPy, Matplotlib, and Pandas for data visualization. We used the Plot.ly API too for visualizing our data on a map.

We used Weka for clustering purposes, but given the size of the dataset, it took a long time to load the dataset and perform clustering on the same. So, we switched over to Python and wrote the code to visualize our data. We used Weka only to run the KMeans algorithm since it saved us the time of writing the algorithm.

3.2 Data Cleaning and Analysis

The data collected from [3] had to be cleaned a lot since it had missing data and outliers. We wrote a Python code and used the 'json' library to convert the JSON files in the dataset into CSV files. The files generated were:

1. 'checkins.csv', which had the business names and the count of check-ins for each day.
2. 'time.csv', which had the business names and the count of check-ins for each time range in the entire day.
3. 'rating.csv', which had the business name and the count of ratings given to that business.
4. 'business.csv', which had the business ID, business name, state, city and major category of each business.

We had lots of check-ins with 0 values. But, that is not a missing value since a day can have 0 check-ins for a given day. So, we decided to see if we had any records with 0 values over a period of five days. About 28K records out of the given 140K records had such values in the 'checkin.json' file. So, we got rid of these records since we felt that those records could not sufficiently contribute to the analysis.

In the 'business.json' file, we encounter businesses with multiple categories. So, we first select 100 major categories from all the given categories and assign it to the business which contains one of these major categories. Then, we remove the categories which contain no or a minimal number of businesses since we feel that it will not be useful for the overall data mining procedure.

In the 'time.json' file, we have counts for check-ins for the businesses and the time they were made during the day. We merge together the times to make 6 distinct time ranges. We name these ranges as Late night, Early morning, Early afternoon, Afternoon, Evening, Late evening. So, counts of check-ins from 0:00 am to 4:00 am will be added together and stored in the Late night time interval.

We did not face the problem of incorrect column names or useless attributes since the CSV file was generated by us and we made sure to keep these points in mind. We do not remove duplicate records even if all the days have exactly same check-ins. This is because it is probable for two different business IDs to have the same order of check-ins.

3.3 Data Mining and Visualization

Initially, we decided to see which days might be attracting a lot of customers. We plotted a bar graph which helped us to see which day might have the highest number of check-ins in the given dataset. The figure 1 shows the bar graph plotted using the Matplotlib library.

We see that Saturday has the highest count of check-ins. This is unsurprising since usually it is a holiday and so people may want to get out of their houses. This could be said regarding Sunday too but since the next day is usually working, it might cause a slightly reduced count in check-ins.

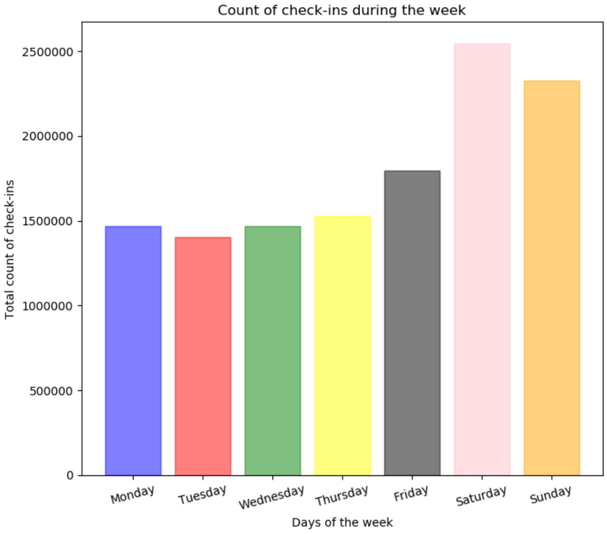

Figure 1: Count of check-ins during the week

We were also interested in finding out what time of the day might be attracting the most number of customers for businesses. A bar graph was generated for the same showing the six-time ranges that we have created spanning the entire day and the count of check-ins during that time for all businesses combined.

We see that the time range for late night, which is 0:00 am to 4:00 am, has the highest count of check-ins. This is kind of surprising since people would usually be at home during that time. The lowest count of check-ins can be seen during the early afternoon period, which is 8:00 am to 12:00 pm. which could be put down to the fact that most people might be busy in offices or schools.

The highest count if check-ins is almost eight times the lowest count of check-ins during the day. This is quite a significant increase and is shown in figure 2.

From the generated *checkins.csv* file, we extract the list of business names and the sum of all its corresponding check-ins for the entire week. We sort this list and extract the list of top ten businesses in terms of most check-ins. This helps us to see how the businesses compare to each other in terms of check-ins over the period of a week. This is shown in the figure 3.

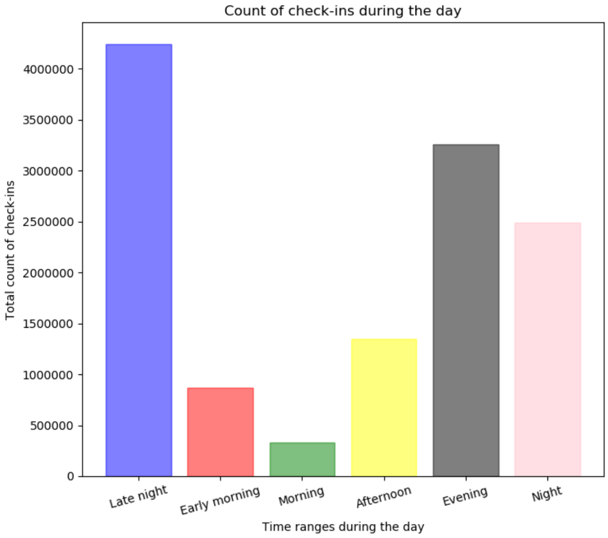

Figure 2: Count of check-ins during the day

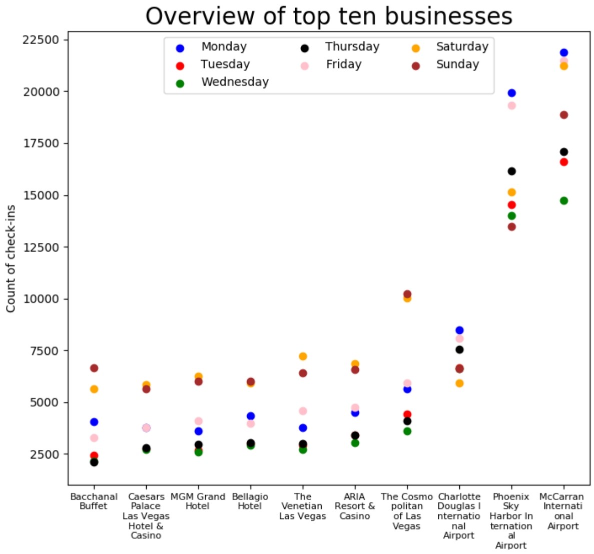

Figure 3: Count of check-ins for top ten businesses for the entire week

We also individually compare the businesses over each day and see which business out of the top ten have a higher check-in count for that particular day. Unsurprisingly, we find that businesses with higher check-in counts dominate over the entire week for any given day. As expected, the two busiest businesses have higher check-ins all throughout the day than other businesses and is shown in the figure 4.

From the *checkins.csv* file, we also extract the list of business names and the sum of all its corresponding check-ins for the entire day. We use the list of top ten businesses that we compiled earlier and see how these businesses compare to each other in terms of check-ins over the entire day in figure 5. The two most popular businesses are unsurprisingly airports, Pheonix Sky Harbor and McCarran International.

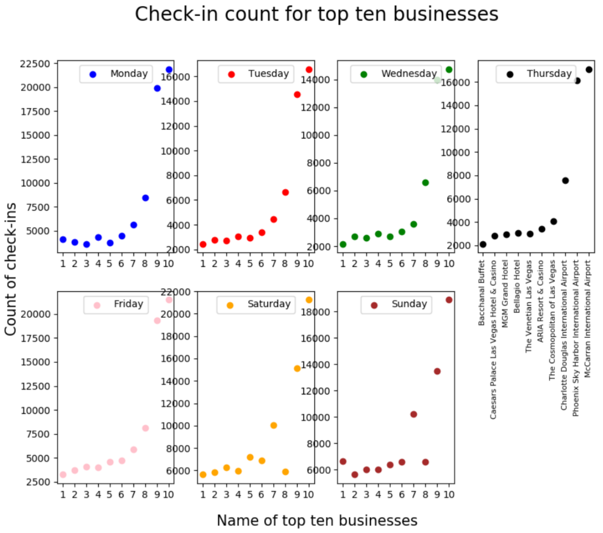

Figure 4: Individual analysis of each business on a given day

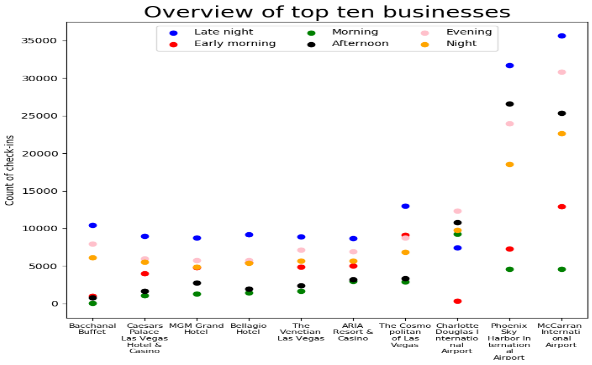

Figure 5: Count of check-ins for top ten businesses during the day

We also individually check which business has a higher check-in count for a particular time range. As was the case for the entire week, businesses with higher check-in counts dominate over the entire day for any given range. The top two businesses have the highest check-ins in the entire day during the late night time period, as shown in figure 6.

In order to identify the best value of k in K-means clustering, we implemented the KMeans algorithm in Python3 using the scikit library and used the elbow method to determine the best value of K (i.e. the number of clusters). In figure 7, the sum of squared errors (SSE) is plotted against the various values of k for values two to ten.

The SSE is basically the squared distance between the data points and centroids of a particular cluster. The algorithm was run iteratively on both the dataset generated which is the business.CSV file which contains check-ins for

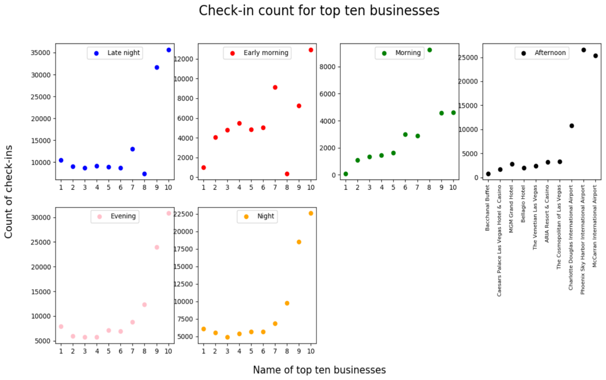

Figure 6: Analysis of each business for each time range

business on each day of the week and time.CSV file which contains check-ins for the different sets of time ranges. The elbow point is the point where there is a significant drop in the SSE and this occurs at value 4.

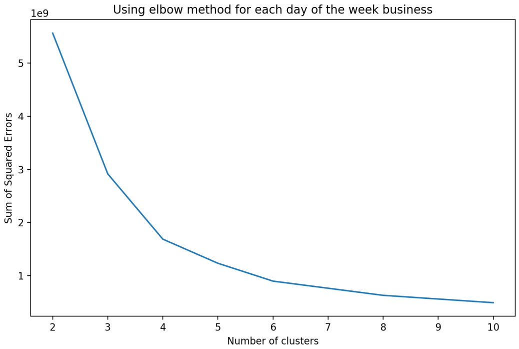

Figure 7: Plotting the SSE vs number of clusters for each day of the week

In the figure 8, we can see that the elbow points occurs at around 4 clusters too. This tells us that if we have around 4 clusters, we will have the most optimal results as per our given data. It is surprising that the same value of elbow points occurs for both time and day analysis.

3.4 Using Plot.ly API

For enhancing our analysis on the Yelp dataset, we also integrated our Python code with the Plot.ly API. The API required us to create an account on their official webpage. Then, once we were registered, we were given an API key to be used in our code. This key enabled us to create a Plot.ly map using our data. This can be seen in the figure 9 which shows all our data points over the entire world.

We had already found out the list of top 500 businesses based on their check-in counts. Using the *business.csv* file, we also extracted the latitudes and longitudes of these businesses. The lists of latitudes and longitudes was passed as input to the Plot.ly API's map integration. This enabled us to visualize our data on a geographical map which opened up as a HTML file on the web browser.

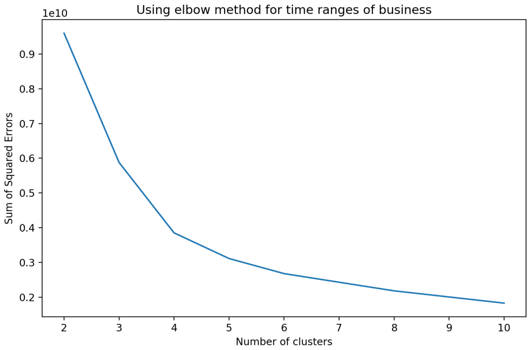

Figure 8: Plotting the SSE vs number of clusters for different time ranges

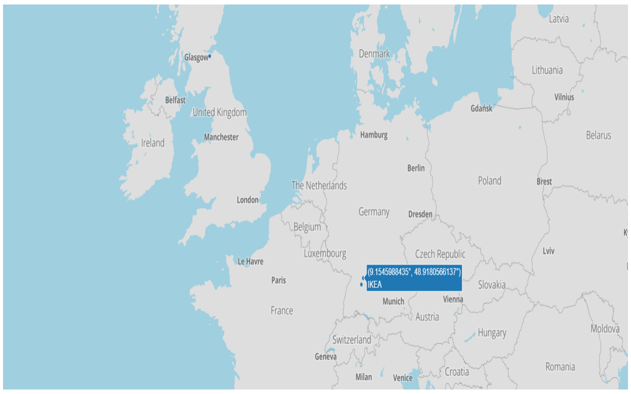

Figure 9: Plotting our data on a map using the Plot.ly API

We can see that USA is the country with the most busiest businesses having the highest number of check-ins for its businesses. This can be seen in figure 10. There are lots of busy businesses among the top states in the country like New York, Illinois and Arizona.

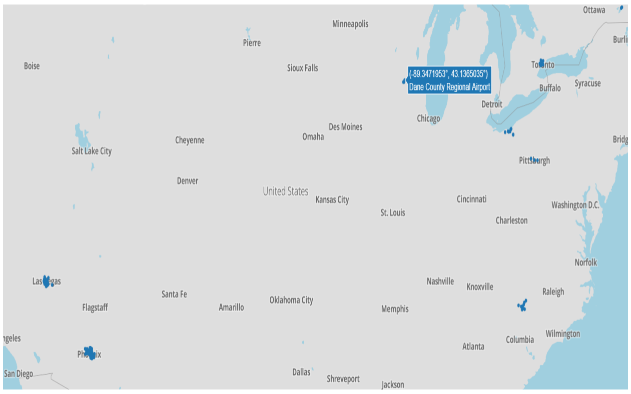

Figure 10: The country with the most busiest businesses

The HTML file that opened up in the browser could be zoomed in or out based on the mouse scroll. The users could look at which data point it was by hovering over the data point. This displayed the name of the business at that location. This was done by passing the list of the business names as an input along with the latitudes and longitudes. We see that in figure 11 that most busiest businesses belong to the state of Las Vegas. This state also has the most busiest business, which is the McCarran International Airport.

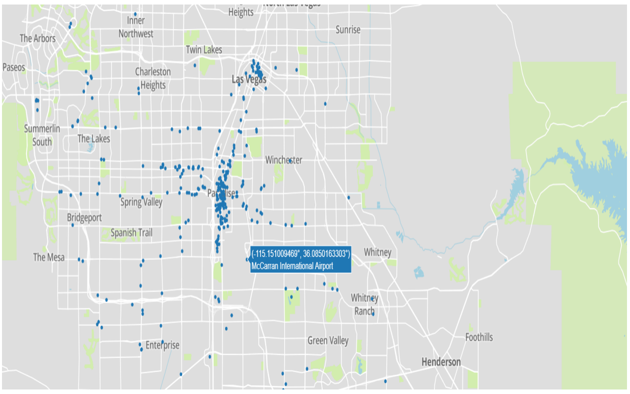

Figure 11: Busy businesses in Las Vegas

4. LESSONS LEARNED

The most important lesson that we learnt was that it is always a good idea to understand what your data is about before trying to do anything with it. Keeping this in mind, we cleaned the data as much as we could. The challenge here was to convert all the JSON files into CSV files and then work on them.

Another lesson that we learned was that usually general knowledge can be represented by data tool. This happened in our case since we saw that most businesses had the highest check-in counts during the weekend. We also understood about the elbow point method in greater detail.

5. FUTURE WORK

We can go one step further than what we have done until now in terms of data cleaning. We can perform text mining on the generated *ratings.csv* file. This would help us to understand whether a given review was good or bad for a business. We can also perform analysis on the *business.csv* to show which businesses have the most check-ins in a given city or state. Another thing we could try, given time constraints is finding out which businesses from a particular major category gathered most reviews and check-ins.

6. CONCLUSION

To sum up our report, we perform analysis on the Yelp dataset consisting of 1.6 million records in different JSON files. The first step was to convert it to CSV files using Python. Then we analyzed these files from different perspectives and applied different kinds of visualization techniques on it. The main observation that we found from our analysis was that the two most busiest businesses of all the businesses given in the dataset were airports and the busiest days for these businesses were during the weekend.

We also visualized our dataset using the Plot.ly API to see how spread out our data was. We find that the businesses present in the dataset consisted of almost all the continents and that each continent had atleast a few businesses in the list of our top 500 businesses. This list was generated based on the total count of check-ins for a particular business over the entire week.

References

[1] M. M. Hamad and A. A. Jihad. “An Enhanced Technique to Clean Data in the Data Warehouse”. In: *2011 Developments in E-systems Engineering.* Dec. 2011, pp. 306–311. DOI: `10.1109/DeSE.2011.32`.

[2] Dr. Arvind Kalia Sapna Devi. “Study of Data Cleaning and Comparison of Data Cleaning Tools”. In: *International Journal of Computer Science and Mobile Computing* 4.3 (Mar. 2015), pp. 360–370.

[3] *Yelp Dataset.* 2013. URL: `https://www.yelp.com/dataset%5C_challenge/`.