Project book

Overview and Motivation

There are two main districts that travelers to Las Vegas visit: the Downtown or Strip Las Vegas. The Downtown Las Vegas (DTLV) is also often called as Old Las Vegas. As the later name implies, it is the historic center of Las Vegas, Nevada. All the casinos and hotels are relatively close together, mostly in walking distance. They are also less expensive that those in the Strip Las Vegas as it requires less minimum bet for different table games. For this reason, DTLV is "for the budget minded and the serious gambler who do not like all the glitz [and] glamour of the Strip [Las Vegas]".

The Las Vegas Strip, on the other hand, features mega resorts that one often sees in television shows or movies. It has "Disney-like attractions and nothing less than 5 star hotels" All the famous casino and resorts like the Bellagio, Wynn, Venetian, and Mandalay Bay could be found in the Strip Las Vegas. Thus sightseers or less serious gamblers would prefer to stay on the Strip.

We are mainly interested in comparing Yelp reviews, particularly those related to restuarants, in DTLV and the Strip. We will use a vareity of different data visualization techniques in order to conduct conpartive analysis.

Research Questions

1. How restaurants are visually distributed in the map in Las Vegas?

2. How high/low rated restaurants in Old and New Las Vegas differ in terms of their business attributes?

3. Would the sentiment toward key business attribute differ in two regions, due to different preferences of the traveler groups in each region?

Data

We plan to use the Yelp's academic dataset. This dataset has 4.1 million reviews by 1 million users for 144,000 businesses. It provides dataset for different cities like Edinburgh, U.K; Karlsruhe, Germany; Montreal and Waterloo, Canada; Pittsburgh, Charlotte, Urbana-Champaign, Phoenix, Las Vegas, Madison, Cleveland, USA. The dataset also includes 1.1 million business attributes like hours, parking availability, and ambience.

It consists of two datasets. The first is business dataset where it includes Name of business (with business ID), Categories, Attributes, Star ratings, and Review Count. The other is review Dataset which includes review text, star rating, and user ID.

For our project, we are only interested in the restaurant data in Las Vegas in the United States. Thus we plan to use subset of this large dataset.

Data Wrangling

For our project, data wrangling was the most challenging part. Since the Yelp Challenge dataset is very large with 4.1 M reviews and 1.1 M business attributes, it took us a great amount of time–almost forever– to do the data wrangling. We both used R and python in order to extract and clean the datasets that we are particularly intersted in. If you are intersted, you can see the code in our gihub repository.

For interactive geospatial analysis, we followed the following steps:
- Download Yelp dataset from: https://www.yelp.com/dataset_challenge
- Download Zillow Nevada Shapefile dataset from: https://www.zillow.com/howto/api/neighborhood-boundaries.

htm
- Create a data folder containing four empty folders named external, interim, processed and raw under the local repo
- Unzip Yelp dataset and put .json files into data/raw folder
- Run src/data/process_raw_data.py
- We post a final subset of data in this final submission repo

Again, all the source code is accessible through github repo.

The following is the last data wrangling process we went through for text and sentient analysis.

```
# bring in the final dataset that was cleaned using python
## this dataset contains business attribute files
data <- read.csv("final_data.csv")
# bring in the final review dataset that was cleaned using python
## this dataset contains text review files, business ids, and star reviews
review <- read.csv('data_r.csv', fileEncoding = "latin1")

# exclude non-ASCII texts
review$text <- gsub("[^\x20-\x7E]", "", review$text)

# merge attribute data frame and review text file
total <- merge(data, review, by = "business_id", all = FALSE)

# only keep unique texts (delete duplicated ones)
total_u <- distinct(total, text, .keep_all = TRUE)

# choose only downtown and The Strip
total_s <- total_u %>% filter(neighborhood == "Downtown" | neighborhood == "The Strip")
```

The Corpus cleaning process that we then conducted is pretty much same as what we did in class.

```
##### clean text #####
total_s$text <- as.character(total_s$text)
str(total_s$text)
text <- as.data.frame(total_s$text)

df_source <- DataframeSource(text[1])
df_corpus <- VCorpus(df_source)
df_corpus

# check the contents
df_corpus[[1]][1]
df_corpus[[13312]][1]

# Text cleaning
df_corpus <- tm_map(df_corpus, content_transformer(tolower))
df_corpus <- tm_map(df_corpus, content_transformer(removeWords), c(stopwords("english")))
# df_corpus <- tm_map(df_corpus, content_transformer(removeWords), c("list", "w/"))
df_corpus <- tm_map(df_corpus, content_transformer(removeNumbers))
df_corpus <- tm_map(df_corpus, content_transformer(removePunctuation))
# df_corpus <- tm_map(df_corpus, content_transformer(replace_abbreviation))
# df_corpus <- tm_map(df_corpus, replace_symbol)
# delete non-english characters
```

```
## stem document
df_corpus_stem <- tm_map(df_corpus, stemDocument)
df_corpus_stem <- tm_map(df_corpus_stem, stripWhitespace)

# check stemmed document
df_corpus_stem
df_corpus_stem[[1]][1]
df_corpus_stem[[2]][1]

# givr row names to data frame to match neighbhorhood in tdm and dtm
total_s$row.names <- 1:nrow(total_s)

# dtm and tdm
tdm <-TermDocumentMatrix(df_corpus_stem)
tdm_td <- tidy(tdm)
tdm_td$neighborhood <- total_s[match(tdm_td[['document']], total_s[['row.names']]), 'neighborhood']
head(tdm_td)

dtm <- DocumentTermMatrix(df_corpus_stem)

dtm_td <- tidy(dtm)
head(dtm_td)
dtm_td$neighborhood <- total_s[match(dtm_td[['document']], total_s[['row.names']]), 'neighborhood']
head(dtm_td)
```

The following is the code for the data wrangling process we went through for the sentiment analysis. The main challenge here was to merge two datasets.

```
pos <- read.table("positive-words.txt", as.is=T)
neg <- read.table("negative-words.txt", as.is=T)

library(quanteda)

sentiment <- function(words=c("really great good stuff bad")){
  require(quanteda)
  tok <- quanteda::tokenize(words)
  pos.count <- sum(tok[[1]]%in%pos[,1])
  neg.count <- sum(tok[[1]]%in%neg[,1])
  out <- (pos.count - neg.count)/(pos.count+neg.count)
  return(out)
}

Strip <- total_s %>% filter(neighborhood == "The Strip")
Downtown <- total_s %>% filter(neighborhood == "Downtown")

Sent_s <- data.frame(matrix(0, ncol = 1, nrow = nrow(Strip)))
colnames(Sent_s)[1] <- "sent"
Sent_s$text <- Strip$text
Sent_s$star <- Strip$stars

Sent_d <- data.frame(matrix(0, ncol = 1, nrow = nrow(Downtown)))
Sent_d$text <- Downtown$text
colnames(Sent_d)[1] <- "sent"
```

```
Sent_d$star <- Downtown$stars

for(i in 1:nrow(Strip)) {
  Sent_s[[i, 1]] <- sentiment(Strip[[i, 495]])
}

for(i in 1:nrow(Downtown)) {
  Sent_d[[i,1]] <- sentiment(Downtown[[i, 495]])
}

sentiment(Strip[[7, 495]])
sentiment(Downtown$text)

Sent_d$neigh <- "Downtown"
Sent_d$neigh <- factor(Sent_d$neigh)
Sent_s$neigh <- "The Strip"
Sent_s$neigh <- factor(Sent_s$neigh)
```

Exploratory data analysis

Many exploratory was done on the text files.

```
# The bar graph of frquent words used in review of each neighbor
# frequent terms in general
tdm_td %>% group_by(term) %>%
  summarise(n = sum(count)) %>%
  arrange(desc(n)) %>%
  top_n(n = 20, wt = n)  %>%
  mutate(term = reorder(term, n)) %>%
  ggplot(aes(term, n)) +
  geom_bar(stat = "identity", fill = "skyblue") +
  xlab(NULL) +  coord_flip() + theme_tufte() +
  ggtitle("The Words Most Frequently Used in Both Neighborhood") +
  theme(plot.title = element_text(lineheight=.8, face="bold")) +
  ylab("Term Frequency")

# Bind the TF,DF, and IDF frequency
# of a tidy text dataset to the dataset
tf_idf <- tdm_td %>%
  bind_tf_idf(term, document, count)  %>%
  arrange(desc(tf_idf))

tf_idf

tf_idf2 <- tdm_td %>%
  bind_tf_idf(term, document, count)  %>%
  arrange(desc(tf))

# frequent words by used in reviews of each neighborhood
### The Strip
#### by tf
tf_idf2 %>% filter(neighborhood=="The Strip") %>%
  arrange(desc(tf)) %>%
  top_n(n = 10, wt = tf)  %>%
  ggplot(aes(x = term, y = tf)) +
  geom_bar(stat = "identity", fill = "skyblue") +
  xlab(NULL) + ylab("Term Frequency") + coord_flip() + theme_tufte() +
  ggtitle("Top 10 Frequent Words Used in Reviews of The Strip") +
  theme(plot.title = element_text(lineheight=.8, face="bold"))

#### by tf_idf
tf_idf %>% filter(neighborhood=="The Strip") %>%
  top_n(n = 10, wt = tf_idf)  %>%
  ggplot(aes(x = reorder(term, tf_idf), y = tf_idf)) +
  geom_bar(stat = "identity", fill = "skyblue") +
  xlab(NULL) + ylab("TF-IDF") + coord_flip() + theme_tufte() +
  ggtitle("The Frequent Words Used in The Strip") +
  theme(plot.title = element_text(lineheight=.8, face="bold"))


### Downtown
#### by tf
tf_idf2 %>% filter(neighborhood=="Downtown") %>%
  #arrange(desc(tf)) %>%
  top_n(n = 10, wt = tf)  %>%
  ggplot(aes(x = term, y = tf)) +
  geom_bar(stat = "identity", fill = "skyblue") +
  xlab(NULL) + ylab("Term Frequency") + coord_flip() + theme_tufte() +
  ggtitle("Top 10 Frequent Words Used in Reviews of Downtown") +
  theme(plot.title = element_text(lineheight=.8, face="bold"))

#### by tf_idf
tf_idf %>% filter(neighborhood=="Downtown") %>%
  arrange(desc(tf_idf)) %>%
  top_n(n = 10, wt = tf_idf)  %>%
  ggplot(aes(x = reorder(term, tf_idf), y = tf_idf)) +
  geom_bar(stat = "identity", fill = "skyblue") +
  xlab(NULL) + ylab("TF-IDF") + coord_flip() + theme_tufte() +
  ggtitle("The Frequent Words Used in Downtown") +
  theme(plot.title = element_text(lineheight=.8, face="bold"))

# comparison
plot <- tf_idf %>%
  group_by(neighborhood) %>%
  top_n(n = 10, wt = tf_idf) %>%
  mutate(key = 1:10)
plot %>% ggplot() +
  geom_bar(aes(x = term))

####################### wordcloud #######################

# wordcloud by neighborhood
library(wordcloud)
# Create purple_orange
purple_orange <- brewer.pal(10, "PuOr")
# Drop 2 faintest colors
```

```r
purple_orange <- purple_orange[-(1:2)]

## The Strip
term_frequency_s <- tf_idf %>% filter(neighborhood=="The Strip")
set.seed(100)

# Create a wordcloud for the review in The Strip
wordcloud(term_frequency_s$term, term_frequency_s$tf,
          max.words = 200,
          colors= purple_orange)

### includes words "overpriced"

## Downtown
term_frequency_DT <- tf_idf %>% filter(neighborhood=="Downtown")
set.seed(213)

# Create a wordcloud for the review in Downtown
wordcloud(term_frequency_DT$term, term_frequency_DT$tf,
          max.words = 100, colors = purple_orange)

#### includes the world "cheap"
```

Evaluation

1 More spatial clusters of restaurants in New Vegas
2.High-End, expensive restaurants are preferred and exist more in New Vegas
3. Overall sentiment score for two regions are similarly distributed and somewhat popular
4. However, consumers in the Strip are more likely to get negative sentiment in low rated restaurants.
5. Casual restaurants are welcomed in both areas
6. Up-scaled/romantic restaurants are welcomed in both areas

All in all, consumers in Old Vegas is more budget-minded than those of New Vegas Consumers in New Vegas are more likely to be the traveler for specific events, funs, and traveling.