University of California, Davis

HW #5: Extracting Text and Features From a Messy Craigslist Dataset

Ana Boeriu
STA 141A
Dr.Nick Ulle
November 13,2018

I. Introduction

Craigslist is a website that allows people to post classified advertisement for free. These posts can span a variety of subjects such as cars, bikes, jobs, and even housing. The goal of this assignment will be to clean and extract features from a messy, expanded version of the Craigslist's dataset. Each apartment posting is in a separate text file. I will be focusing on creating and analyzing a dataset of apartment rental postings in California. I will then use exploratory data analysis along with various visualizations to further analyze any trends and patterns in the data set. This report can be useful for anyone looking to rent an apartment in various locations in California. Because the data used is created by Craigslist, I need to be aware that there can be multiple errors or anomalies, which can skew the data set and cause inaccurate conclusions.

II. Explaining the functions created

In this assignment I have created two functions that aid me in being able to read each file and extract the necessary features from the individual post.

A. Read Post

This function, like the name suggests, allows me to read each post in a specified directory name. Once I can read each post, I then extract the necessary information from each post by using regular expressions. I used the questions to guide me in what features need to be extracted. Furthermore, I also added other features such as region, to better facilitate my graphs. I first started by splitting the title and text of each post which would then correspond to a row in the data frame. Then each feature I extracted from the text would become a column in the data frame. Overall, in this function my goal was to be able to read an individual Craigslist post, and extract variables such as prices, title,text,etc that will help me in my analysis later on.

B. Read All Posts

Once I was able to obtain a function that reads a single file, my next goal was to create a function that would read all the posts in the messy folder, regardless of the directory name. Because I had already done the extractions in the previous function, in read all posts, I just needed to create columns for all the variables that I extracted. Furthermore, I also had to specify the type (character or numeric) of each variable. However, in order to be able to use the previous function, I had to call read post in my read all posts function by using a for loop. Because I used a for loop to call read post, the data frame was not very efficient in loading. To keep track of progress while loading, I printed the name of the directories. So, during the 4-6 minutes, while the data loaded, I knew how many more files needed to be read.

III. Cleaning up the dataset

Before I begin exploring this dataset, I will first begin by observing any outliers or anomalies in this dataset. I will look at all the numeric variables such as price, deposit, bedrooms, bathrooms and pet deposit and check that each variable has a reasonable range. Any outliers or anomalies found in this dataset will be removed, fixed or replaced with na.

A. Prices

When assessing the variable prices, I noticed that some posts did not include a hyphen between the possible range of prices. For example, one post in row 34055 (of the original dataset) included $3,4083,742 as the price for a two bedroom in Mountain View. It is unlikely that a two bedroom would cost 3 million dollars per month. The actual price was a range $3408 - $3742 where the

user did not include a hyphen between the two numbers. Thus, for such instances, I have modified prices for those postings to the largest value in the range.

B. Bedrooms

Next, I will also look at the number of bedrooms that apartments have. The floorplan of the apartments ranges from studio apartments (0 bedrooms) to seven bedrooms. It is highly unlikely that an apartment has seven bedrooms. Normally, a house would be more likely to have seven bedrooms. Upon exploring the variable bedrooms, I noticed that all apartment posts regarding 7 bedrooms are cleaning services. Similarly, all five- and six-bedroom apartments are actually houses. Thus, I will be removing all the posts with 5,6 or 7 bedrooms in order to focus more on apartment rentals.

C. Bathrooms

Furthermore, I can notice that there are some apartments that have more bathrooms than bedrooms. In that case I have set any apartments where number of bathrooms is larger than number of bedrooms and the apartment has more than one bedrooms to na. This will better reflect the data and prevent any unnecessary removal of potentially useful data.

D. Deposit

The variable deposit also had some anomalies and outliers. There were instances when deposit amount in the text varied from the deposit attribute. Thus, I have fixed those values to match. Furthermore, there were some posts that had deposit amounts over $13,000. These posts were in fact 4 bedrooms Mediterranean style houses that I removed to focus on apartment postings.

E. Pet Deposit

When looking at this variable, I noticed that there were cases where the word pet deposit was in between two numbers. A few times, the wrong number would get extracted causing a few apartment postings to have a large pet deposit. I fixed those apartment postings to reflect the correct pet deposit amount.

F. Note

Because this is a very large dataset, there may still be some outliers or anomalies that were not addressed before. I will discuss and fix the values below if necessary.

IV. Getting Familiar with the Dataset

After extracting the necessary features and removing any anomalies or outliers, I will give a brief overview of the data.

A. Rows

The rows correspond to the number of observations or, in this dataset, the number of apartment rental posts in the data. There are a total of 45,757 apartment posts corresponding to different regions within California.

B. Columns

For each observation recorded there are 18 variables (columns in the dataset). The variables contain information such as the title of the post, text of the post, rental price, date that post was published on, size of the apartment, number of bedrooms and bathrooms, and any many more. All these variables were created from common features that needed to be extracted from each individual post.

C. Span of rental posts

All the apartment listings were posted between September 30, 2018 and November 1, 2018.

V. How Rental Price and User Price Compare

In this section I will focus on further analysing the variables price and user price. I will be looking for any trends, patterns, and anomalies and discuss them below. I will also answer all the questions stated below. Unless otherwise specified in this section, price refers to the rental price in the title and user price is the price attribute in the text.

A. Do all of the titles have prices? How do these prices compare to the user-specified prices (the price attribute)?

Out of all 45757 apartment posts, there are 45,581 apartments postings that have rental price stated in the title. Similarly there are 45,577 apartment postings that have a user price specified in the text. From this I can see that it is more common for users to state prices in the title rather than the text. In order to be able to compare these variables, I will only look at the values that are different. There are only twelve apartment posts that do not have the same rental price and user price. To further condense the table I will only look at the unique differences. In other words, any identical price difference that occurs more than once will only be recorded once in the table. This reduces the table from twelve to eight values.

Price (rental price)	User Price (price attribute)
2995	2925
2350	2550
2625	2425
1995	2045

Price (rental price)	User Price (price attribute)
2325	2425
2375	2425
2000	2200
2395	2495

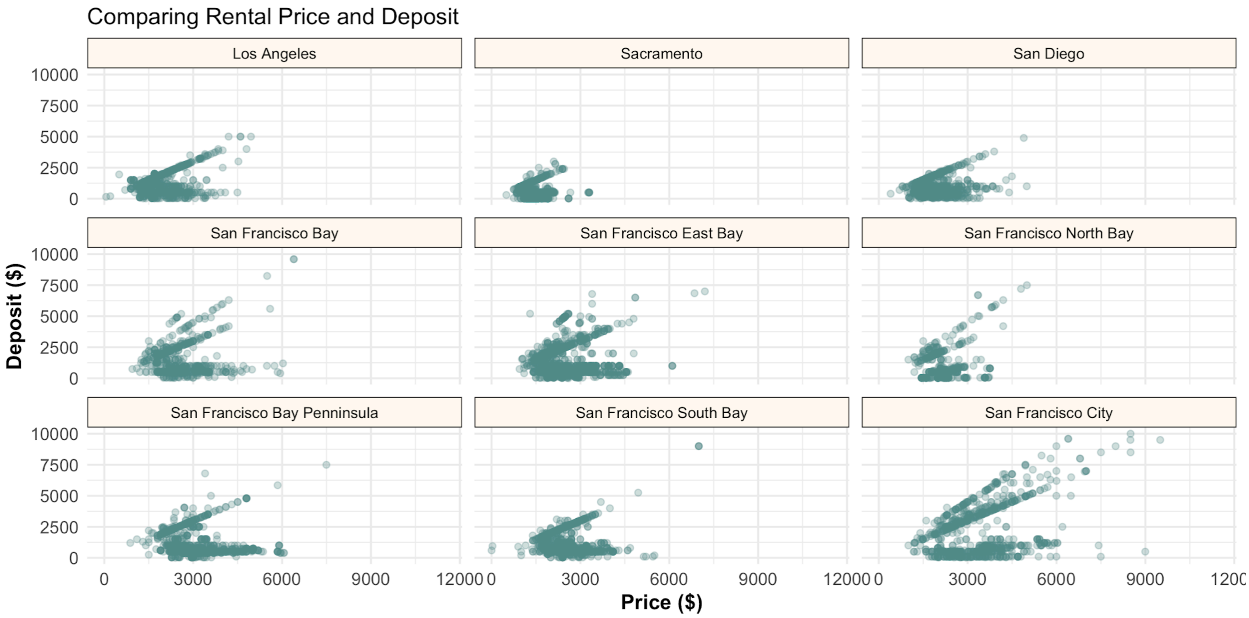
Furthermore, the average difference between rental price and the price attribute is $-25.833. This means that on average, the rental price stated was less than the price attribute by $25.833. So users made the rental price in the title seem cheaper to get more people to view the post when in reality, the price attribute was more expensive.

VI. Looking at Deposit and Rental Price

In order to gain a better understanding of this dataset, I will be looking at how the rental price and security deposit amount compare and if there are any trends, or outliers. I will be focusing on answering the questions stated below.

A. Is there a relationship between rental price and deposit amount?

I would expect that for some apartments, deposit amount depends on the rental price. For example, some apartments have a security deposit of first and last month's rent. On the contrary, for some other apartments, security deposit will not depend on rental price. Unless otherwise specified in this section, price will refer to rental price of the apartments.



In this graph there seems to be two main clusters. One cluster is linear where deposit increases with price and the deposit depends on the rent. The other cluster is flatter which tells me that deposit doesn't depend on price. Furthermore, deposit is half the monthly rent. Lastly this type of relationship is called a piecewise. This means that the function is defined by multiple sub functions, each sub function applying to a certain interval of the main function's domain. This can be seen with prices greater than about $2800. Furthermore, out of all the regions, Sacramento seems to be the least expensive place to live while San Francisco City is the most expensive.

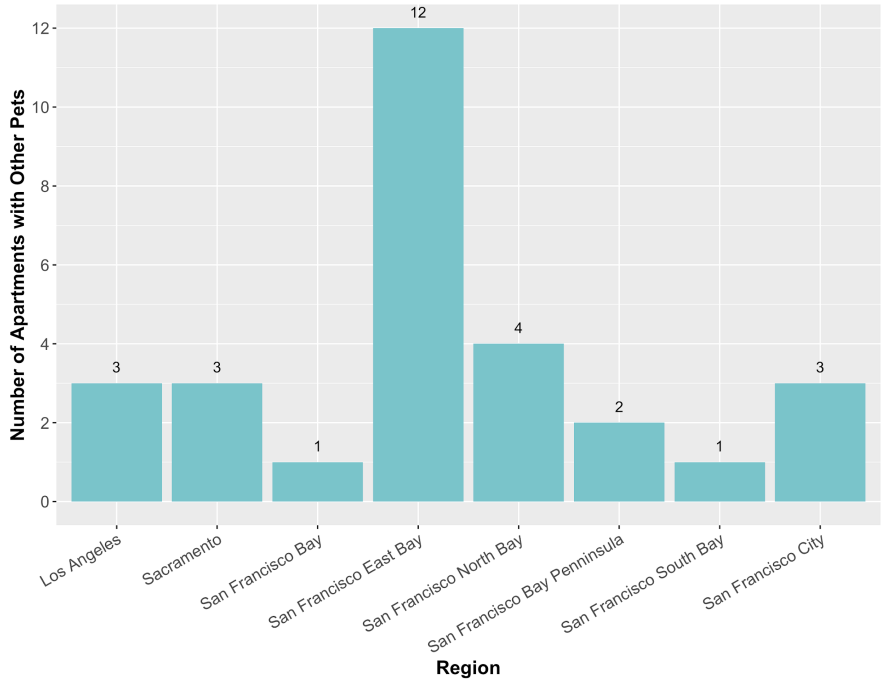
VII. Analysing Pet Policy of Apartments

Here, I will be looking at the categorical variable pets. I will focus on answering the questions stated below. For this section I have defined other pets to be either: a hamster, guinea pig, rabbit, turtle, chickens, lizards, amphibian, reptile, or fish.

A. Are there any apartments that allow some other kind of pet?

Both	Cats	Dogs	None	Other
11689	284	18	5915	29

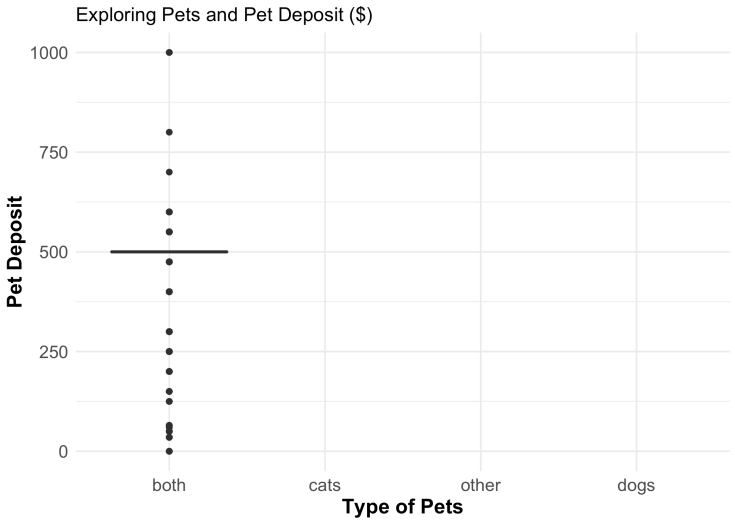
From the table we can see that there are 29 apartments that allow other pets. This means that out of the 29 "other" pets, 41% are from San Francisco East Bay.



I can notice that San Francisco east bay has the most number of other pets. Upon further inspection of these apartments in the East Bay region, the most common "other" pets are chickens, rabbits, and hamsters. However, rabbits and chickens aren't usually kept inside an apartment. These animals reside in a common large garden within the apartment complexes. Furthermore, some more "other" pets are reptiles, and amphibians. However, some apartment posting provided a breed restrictions list of all the animals that were not allowed. Reptiles and amphibians were on that list. Thus these should not be considered as "other pets".

B. For apartments that allow pets, make a graphic that shows how pet deposits are distributed and discuss what the graphic suggests about pet deposits.

Many people looking at apartments and who want to bring a pet would be interested to know how pet deposit compares to the type or number of pets.



This graph only has data points in the both category. This is because there were no pet deposits for apartments that allowed only dogs or only cats or other pets. A cause of this could be that my regular expression function was not specific enough or that there simply weren't any deposits for cats or dogs. To provide an accurate analysis of this graph, I will look at the median because, compared to the mean, it is less affected by any skewness in the dataset. The median dollar of amount pet deposit for apartments that allow both dogs and cats is $500. Thus, apartments that allow just a single pet tend to not have a pet deposit whereas apartments that allow both pets tend to have a deposit amount that is much larger.

VIII. Looking at HVAC

In this section I will be looking at heating and air conditioning of apartments. I will be discussing any outliers, anomalies and trends in the dataset. Furthermore, I will also be discussing any questions stated below. To clarify, anytime I use HVAC , I am referring to heating and air conditioning

A. Is air conditioning more common than heating?

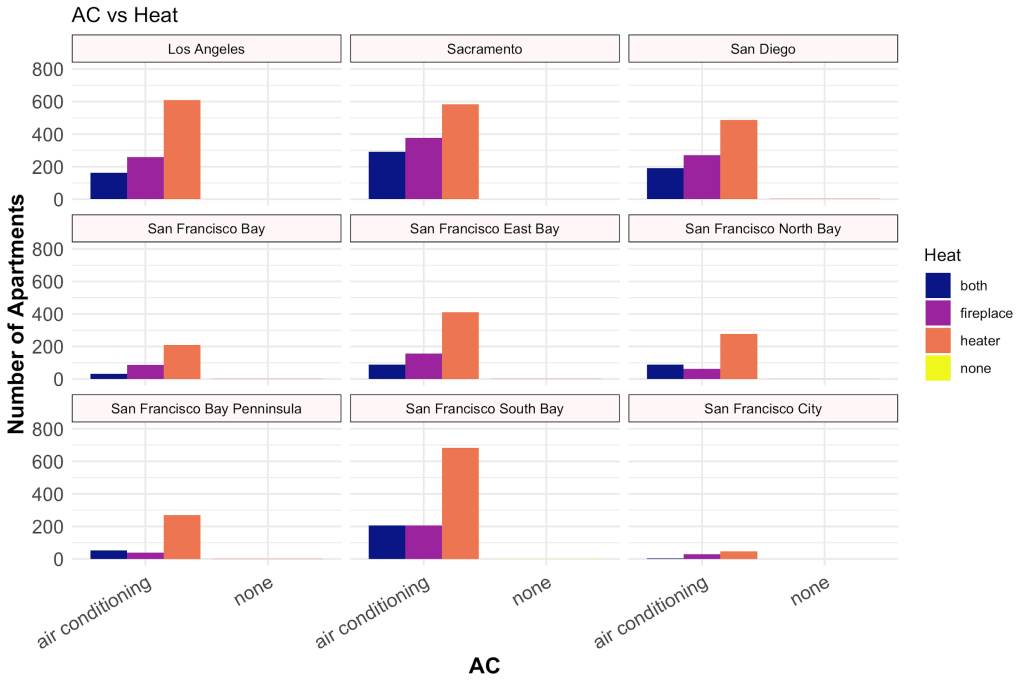
Here I will be looking at how common it is for an apartment to have heating and air conditioning. I will also be looking at how different areas within California influence HVAC. Furthermore, to able to better tell what type of system is more common, I need to exclude the posts that include both a heating and cooling system. I will be looking at apartments that have just air conditioning or just heating.

AC and no Heat	Heat and no AC
0	8

Apartments that have only a heating system are more common than apartments that have only an air conditioning system. Similarly, regions such as San Francisco Bay, San Diego, San Francisco East Bay, San Francisco North Bay all have some apartment that fall under the heat and no AC category. I can further analyse this by saying that 13.5% of apartment posts specify the type of HVAC system as having both air conditioning and heat.

B. Do apartments with air conditioning typically have heating?

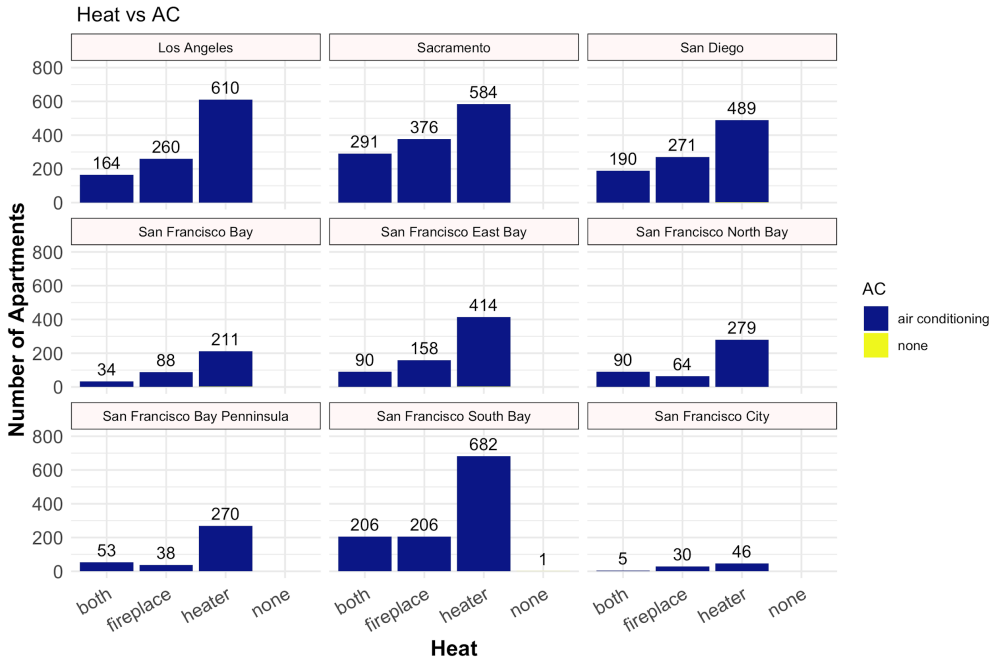
Here I will be looking at if apartments with air conditioning imply that they also have heat.



Apartments with air conditioning typically have heating. This graph uniquely shows that for every apartment, in the specified region, that has air conditioning, the apartment also has some type of heating system. Only San Francisco South Bay has one apartment that has no air conditioning and no heat. This observation is the same apartment in Overall, apartments that have air conditioning also have some other heating system.

C. Do apartments with heating typically have air conditioning?

Similar to above I will be checking the converse is true. In other words, I will be looking at if the phrase, apartments with a heating system will also have air conditioning, is true. Even though I found that apartments with air conditioning also have heating, I cannot assume nor say that apartments with heating will also have air conditioning.



The majority of apartments that have a heating system already installed also have an air conditioning system. Furthermore, San Francisco South Bay has an apartment that has no heater and no AC. This apartment is most likely the same apartment that was noted above. Given this information I can conclude that if an apartment has a heating system it will also have an air conditioning system and vice versa.

IX. Hiding Contact Information

Craigslist offers an option where user can hide their contact information from the post as a security precaution. Thus, I will be looking at how many users use this optional security feature. I created a regular expression that will extract the phone numbers and email from each post. I then summed up the total number of posts that had a contact information present.

Number of Posts that have Phone Numbers	Number of Posts that have Email
1303	0

Some Craigslist users prefer to be contacted via phone rather than email. Those who want to be contacted through the Craigslist website probably have used the optional feature to hide their contact information. Upon further inspection of the phone numbers, I realized that some phone numbers begin with a zero. These numbers were actually part of a website URL link. For the future, I would probably modify my regular expression to be a bit more accurate.

X. R Appendix

```
library(stringr)
setwd("~/Documents/UC Davis/Year 3/STA 141A/assignment 5")
directory="messy"
files=list.files(directory, full.names = T)
file=files[10]
readLines(file)


read_post= function(file){
  raw_text= readLines(file)
  #if(length(raw_text)==0) return(NULL)
  #region=basename(directory)

  #we need to read in the file, then split up the title and text. After we have the title and text,
  #then we can start extracting the necessary features.

  title= str_squish(raw_text[1])
  #start=which(str_detect(raw_text,"QR Code Link to This Post")) + 1
  #text = str_squish(paste(raw_text[start:length(raw_text)],collapse="\n"))
  #text = str_squish(raw_text[2:length(raw_text)])
  text= str_squish(paste(raw_text[2:length(raw_text)],collapse="\n"))

  #begin extracting info from post

  #make numbers by removing the $ and ,.  numeric (in the read post)
  price_title= str_extract(title, "\\$[0-9,..]+")
  price_title = str_remove_all(price_title, "[^0-9.]")
  price_title= as.numeric(price_title)

  #price_user = "Price: $1,600.00"
  price_user= str_extract(text, "Price: \\$[0-9,.]+")
  price_user= str_remove_all(price_user, "[^0-9.]")
  price_user= as.numeric(price_user)

  #deposit = "$34,222.12.. deposit only in cash."
  deposit_text = str_extract(tolower(text),"(deposit(:| )\\s\\$[0-9,.]+)|(\\$[0-9,.]+[^ ] deposit)|
                       (deposit is \\$[0-9,.]+)|(security deposit(:| )\\s\\$[0-9,.]+)|security deposit starting at
\\$[0-9,.]+")
  deposit_text = str_remove_all(deposit_text,"[^0-9.]")
  deposit_text = str_remove_all(deposit_text, "[.]+$")
  deposit_text = as.numeric(deposit_text)
```

```
  #pet_deposit="$53,222.23 pet deposit and then "
  pet_deposit = str_extract(tolower(text), "((pet[s] deposit fee max[:] \\$[0-9,.]+)|(pet(s| )deposit(:| )\\$[0-
9,.]+)
                        |(pet(s| )security deposit(:| )\\$[0-9,.]+)|(pet[s] security deposit[:] \\$[0-9,.]+)
                        |(deposit for pet(s| )\\$[0-9,.]+)|(deposit for pet(s| ): \\$[0-9,.]+)|(\\$[0-
9,.]+\\spet\\sdeposit))")
  pet_deposit= str_remove_all(pet_deposit, "[^0-9.]")
  pet_deposit=as.numeric(pet_deposit)

  num_bedrooms = str_extract(text, "Bedrooms: [0-9]+")
  num_bedrooms= str_remove_all(num_bedrooms, "[^0-9.]")
  num_bedrooms= as.numeric(num_bedrooms)

  num_bath = str_extract(text,"Bathrooms: [0-9]+")
  num_bath = str_remove_all(num_bath, "[^0-9.]")
  num_bath = as.numeric(num_bath)

  sqft= str_extract(text, "Sqft: [0-9,.]+")
  sqft= str_remove_all(sqft, "[^0-9.]")
  sqft=as.numeric(sqft)

  longit= str_extract(text, "Longitude: -[0-9,.]+")
  longit = str_remove_all(longit,"Longitude: ")
  longit = as.numeric(longit)

  latitude = str_extract(text, "Latitude: [0-9,.]+")
  latitude = str_remove_all(latitude, "Latitude: ")
  latitude = as.numeric(latitude)

  date_posted = str_extract(text,"(Date\\sPosted(:| ) (January|February|March|April|May|June|July|
                        August|September|October|November|December)\\s(\\d\\d?).+?(\\d\\d\\d\\d))")
  date_posted= str_remove_all(date_posted, "Date Posted(:| )\\s")

  pets=NA

  terms_other = "(hamster)|(guinea pig)|rabbit|turtle|chickens|lizards|amphibians|reptiles|([^ ]fish[^ ])"
  if(str_detect(tolower(text), terms_other)) pets="other"

  terms_dogs= "(dog-friendly)|(dogs allowed)|(dogs only)|(dog-only)|(dog friendly)|(dogs are
welcome)|(dogs welcome)|(dog park)|(dog lover)|(dogs ok)"
  terms_no_cats=("(no cats)|(no felines)|(cats not allowed)|(allergic to cats)|(cat allergies)|(cats aren't
allowed)|(cats are not allowed)")
  search_dogs = str_extract(tolower(text), terms_dogs)
  search_no_cats= str_extract(tolower(text), terms_no_cats)
```

```
  if(!is.na(search_dogs) & !is.na(search_no_cats)) pets="dogs"

  terms_cats= "(cat-friendly)|(cats allowed)|(feline friendly)|(cats only)|(cats-only)|(cat friendly)| (cats are welcome)|(cats welcome)|(cat lover)|(cats ok)"
  terms_no_dogs=("(no dogs)|(no canines)|(dogs not allowed)|(allergic to dogs)|(dog allergies)|(dogs are not allowed)| (dogs aren't allowed)")
  search_cats = str_extract(tolower(text), terms_cats)
  search_no_dogs= str_extract(tolower(text), terms_no_dogs)
  if(!is.na(search_cats) & !is.na(search_no_dogs)) pets="cats"

  terms_both= "(pets welcome)| (cats and dogs welcome)|(dogs and cats welcome)|(pets allowed)|(pets are allowed)|(pets are welcome)|(pet deposit)|(pet
  security deposit)|(animals( | are )allowed)|(animals( | are )welcome)|(pet-friendly)|(pet friendly)|(small pets ok(ay| ))"
  if(str_detect(tolower(text), terms_both)) pets="both"

  terms_none= "(no pets)|(pets not allowed)|(not pet-friendly)|(not pet friendly)|(no animals)|(pets aren't allowed)|(pets are not allowed)"
  if(!is.na(str_extract(tolower(text), terms_none)))  pets="none"

  heat = NA
  terms_fireplace="(fireplace)|(fireplaces)|(wood (|- )burning stove)"
  if(str_detect(tolower(text), terms_fireplace)) heat="fireplace"

  terms_heater ="(heater)|(heating unit)|(central heating)|(central air heating)|heating"
  if(str_detect(tolower(text), terms_heater)) heat="heater"

  if(str_detect(tolower(text), terms_fireplace)&str_detect(tolower(text), terms_heater)) heat="both"

  terms_no_heat="no (fireplace|(fireplaces)|(heater)|heating|(central heat))"
  if(str_detect(tolower(text), terms_no_heat)) heat="none"

  ac=NA
  terms_ac= "(air conditioning)|(air-conditioning)|(central air)|$(ac)|a/c|cooling"
  if(str_detect(tolower(text), terms_ac)) ac="air conditioning"

  terms_no_ac=" no (heating (\\&|and|or)|((air conditioning)|(air-conditioning)|(central air)|ac|a/c))"
  if( str_detect(tolower(text), terms_no_ac)) ac="none"

  #phone_number= str_extract(tolower(text), "\\(?\\d{3}\\)?[.-]? *\\d{3}[.-]? *[.-]?\\d{4}")
  phone_number= str_extract(tolower(text), "\\(?\\d{3}\\)?[.-]? *\\d{3}[.-]? *[\\.-]?\\d{4}")

  email= str_extract(tolower(text),"^[[:alnum:].-_]+@[[:alnum:].-]+$")
```

```
  #now that we have all the features we need to return all these. B/c return is used jsut with one,
  #I will create a variable features of all the variables that I want to return.
  features=list(Title=title,Text=text,Latitude=latitude,Longitude=
longit,Date_Posted=date_posted,Price=price_title,
            User_Price=price_user,Deposit=deposit_text,Sqft= sqft,
Bedrooms=num_bedrooms,Bathrooms=num_bath,
            Pets=pets,Pet_Deposit=pet_deposit,Heat=heat,
AC=ac,Email=email,Phone_Number=phone_number)

  return(features)
}
read_post(file)

read_all_posts=function(directory){

  #here we have a function that reads all posts from a directory
  files=list.files(directory, full.names = T)
  #set each column as a character or numeric
  n=length(files)
  Title=character(n)
  Text=character(n)
  Price = numeric(n)
  User_Price= numeric(n)
  Deposit= numeric(n)
  Pets= character(n)
  Pet_Deposit= numeric(n)
  Heat= character(n)
  AC = character(n)
  Bedrooms = numeric(n)
  Bathrooms= numeric(n)
  Sqft = numeric(n)
  Longitude=numeric(n)
  Latitude=numeric(n)
  Email=character(n)
  Phone_Number= character(n)
  Date_Posted = character(n)
  Region = character(n)

  #to keep track of loading, print the directory
  print(directory)

  #for length of each subfolder read each post and do the steps above from read posts function
  for(i in 1:n){
    file=files[i]  # set a specific file to a number from files
```

```
   file_info=read_post(file)
   #if(is.null(file_info)) next
   #region = basename(file[i])

   Title[i]= file_info$Title
   Text[i] = file_info$Text
   Price[i] = file_info$Price
   User_Price[i]= file_info$User_Price
   Deposit[i]=file_info$Deposit
   Pets[i]= (file_info$Pets)
   Pet_Deposit[i] =file_info$Pet_Deposit
   Heat[i] = file_info$Heat
   AC[i] = file_info$AC
   Sqft[i]= file_info$Sqft
   Bedrooms[i]= file_info$Bedrooms
   Bathrooms[i] =file_info$Bathrooms
   Longitude[i]= file_info$Longitude
   Latitude[i] = file_info$Latitude
   Email[i]= file_info$Email
   Phone_Number[i]= file_info$Phone_Number
   Date_Posted[i]= file_info$Date_Posted
   #Region[i] = file_info$Region
  }
  results =
data.frame(Title,Text,Region=basename(directory),Latitude,Longitude,Date_Posted,Price,User_Price,De
posit,Sqft, Bedrooms,Bathrooms,Pets = as.factor(Pets),
                    Pet_Deposit,Heat=as.factor(Heat), AC=
as.factor(AC),Email,Phone_Number,stringsAsFactors = F)
  return (results)
}
data=read_all_posts(directory)

dirs = list.files(directory, full.names = TRUE)
post_all= lapply(dirs, read_all_posts)
posts_all = do.call(rbind, post_all)
str(posts_all)
#check that all regions are there should have 9
unique(posts_all$Region)

#after data is loaded, save the dataframe
saveRDS(posts_all,file="~/Documents/UC Davis/Year 3/STA 141A/assignment 5/posts_all.RDS")
posts_all = readRDS("/Users/aboeriu/Documents/UC Davis/Year 3/STA 141A/assignment
5/posts_all.RDS")
##############################################################################################
```

```
#clean up whole dataframe created
#fix the prices / user price
range(posts_all$Price,na.rm = T)
posts_all$Price[34055] = 3742
posts_all$User_Price[34055]=3742

posts_all$Price[8984] = 1095
posts_all$User_Price[8984]=1095

#fix num beds, remove all beds>4 because they are homes
range(posts_all$Bedrooms,na.rm=T)
posts_all=posts_all[(is.na(posts_all$Bedrooms) | posts_all$Bedrooms<5),]

#fix num bathrooms
range(posts_all$Bathrooms,na.rm=T)
#posts_all[which(posts_all$Bathrooms==16),]
posts_all$Bathrooms[41801]=1
#posts_all[which(posts_all$Bathrooms==12),]
posts_all$Bathrooms[45665]=1
#which(posts_all$Bathrooms==6)
posts_all[15081,]
posts_all$Bathrooms[15081]=2
#which(posts_all$Bathrooms==5)
posts_all$Bathrooms[1978]=2.5
#remove all 5 bathrooms, b/c they advertise 1-3 beds and focus is 1 apartm/post
posts_all=posts_all[(is.na(posts_all$Bathrooms) | posts_all$Bathrooms<5),]

#are there posts where beds<baths
which(posts_all$Bedrooms > 0 & posts_all$Bedrooms < posts_all$Bathrooms)
#posts_all[556,]

#where bath > bedrooms
index = which(posts_all$Bedrooms > 0 & posts_all$Bedrooms < posts_all$Bathrooms)
posts_all$Bathrooms[index] = NA

#user price
range(posts_all$User_Price,na.rm = T)

#Deposit
range(posts_all$Deposit,na.rm=T)
#which(posts_all$Deposit==27950)
#posts_all[41661,]
posts_all$Deposit[41661]=2795
```

```
#which(posts_all$Deposit==21000)
#posts_all[27684,]
#remove because these are all homes,comment after removing
#posts_all=posts_all[-c(27682,27683,27684),]
which(posts_all$Deposit==17545)
posts_all[25521,]

#pet deposits
range(posts_all$Pet_Deposit,na.rm=T)
#which(posts_all$Pet_Deposit==2750)
#posts_all[1431,]
posts_all$Pet_Deposit[1431]=250

#which(posts_all$Pet_Deposit==2150)
#posts_all[729,]
posts_all$Pet_Deposit[729]=250

#which(posts_all$Pet_Deposit==1895)
#posts_all[38312,]
posts_all$Pet_Deposit[38312]=250

#which(posts_all$Pet_Deposit==1600)
#posts_all[13826,]
posts_all$Pet_Deposit[13826]=500

#which(posts_all$Pet_Deposit==1525)
#posts_all[14122,]
posts_all$Pet_Deposit[14122]=250

#which(posts_all$Pet_Deposit==1500)
#posts_all[11867,]
posts_all$Pet_Deposit[11867]=400

#which(posts_all$Pet_Deposit==1350)
#posts_all[13829,]
posts_all$Pet_Deposit[13829]=500


#check range for long and lat, longit range should be -180->+180 and lat 0-90
range(posts_all$Latitude,na.rm = T)
range(posts_all$Longitude,na.rm=T)

range(posts_all$Sqft,na.rm = T)
which(posts_all$Sqft==15000)
```

```
posts_all[17906,] #actually the building size
#18367 18373 21607 22143 25045 25082 25083

#############################################################################################
#check to see what causes NA coercion error after removing the necessary outliers
sum(is.na(posts_all$Longitude))
sum(is.na(posts_all$Date_Posted))
sum(is.na(posts_all$Price))
sum(is.na(posts_all$User_Price))
#############################################################################################
library(ggplot2)
library(ggrepel)
library(ggrepel)
library(viridis)
#############################################################################################
#user and rental price
length(posts_all$Price)
length(posts_all$User_Price)
sum(!is.na(posts_all$Price))
sum(!is.na(posts_all$User_Price))

table(posts_all$Price == posts_all$User_Price)
length(which(posts_all$Price != posts_all$User_Price))
unique(posts_all[which(posts_all$Price != posts_all$User_Price), c("Price", "User_Price")])

#find the avg difference of all posts that differ
differ=posts_all[which(posts_all$Price != posts_all$User_Price), c("Price", "User_Price")]
subt=differ[,1]-differ[,2]
mean(subt)


#############################################################################################
#rental price and deposit amounts
posts_all$Region = as.factor(posts_all$Region)
library(plyr)
#rename the regions to make it look nicer
posts_all$Region=revalue(posts_all$Region, c("losangeles"="Los
Angeles","sacramento"="Sacramento","sandiego"="San Diego",
"sfbay"="San Francisco Bay","sfbay_eby"="San Francisco East Bay","sfbay_nby"="San Francisco North
Bay",
"sfbay_pen"="San Francisco Bay Penninsula","sfbay_sby"="San Francisco South Bay","sfbay_sfc"="San
Francisco City"))

ggplot(posts_all[!is.na(posts_all$Deposit ) & !is.na(posts_all$Price), ])+
```

```
  geom_point(aes(x=Price,y=Deposit),color="darkslategray4",alpha=0.3)+
  facet_wrap(Region~.)+
  ylim(0,10000)+
  theme_minimal()+
  ylab("Deposit ($)")+
  xlab("Price ($)")+
  #xlim(0,10000)+
  theme(strip.background =element_rect(fill="floralwhite"))+
  theme(strip.text = element_text(colour = 'black'))+
  ggtitle("Comparing Rental Price and Deposit")+
  theme(panel.spacing = unit(0.5, "lines"))+
  theme(axis.text=element_text(size=10), axis.title=element_text(size=12,face="bold"))+
  ggsave("deposit vs price.png",height=5 ,width=10)

##############################################################################################
#pets
table(posts_all$Pets)

#to get region of 18 other pet posts
table(posts_all$Region[which(posts_all$Pets=="other")])

ggplot(posts_all[which(posts_all$Pets=="other"),])+
  geom_bar(aes(x=Region),fill="cadetblue3")+
  geom_text(stat='count', aes(x=Region,label=..count..), vjust=-1)+
  theme(axis.text.x=element_text(angle=30,hjust=1))+
  theme(axis.text=element_text(size=12), axis.title=element_text(size=14,face="bold"))+
   ylab("Number of Apartments with Other Pets")+
  scale_y_continuous(breaks = seq(0, 14, by =2))+
  ggsave("num other pets by region.png",height=7,width=9)

which(posts_all$Region=="San Francisco East Bay" & posts_all$Pets=="other")


# table(posts_all$Pets!="none")
# length(which(posts_all$Pets != "none"))
# length(unique(posts_all[which(posts_all$Pets!="none"),]))

ggplot(posts_all[which(posts_all$Pets!="none"), ])+
 geom_boxplot(aes(x=Pets,y=Pet_Deposit))+
  xlab("Type of Pets")+
  ylab("Pet Deposit")+
  ggtitle("Exploring Pets and Pet Deposit ($)")+
  theme_minimal()+
  ylim(0,1000)+
```

```
  theme(axis.text=element_text(size=12), axis.title=element_text(size=14,face="bold"))+
  ggsave("pet$ vs pets.png",height=5,width=7)

which(posts_all$Pets=='both' & posts_all$Pet_Deposit==1000)

aggregate(Pet_Deposit~Pets,posts_all,mean)
aggregate(Pet_Deposit~Pets,posts_all,median)
##############################################################################################
#7 HVAC
table(posts_all$Heat)
table(posts_all$AC=="none")

#to see if there are more AC than heat, need to exclude the both apartments
#which posts have some sort of heat and no AC
table(posts_all$Region[which(posts_all$Heat!='none' & posts_all$AC=='none')])

#some kind of AC and no Heat
length(which(posts_all$Heat=='none' & posts_all$AC!='none'))

which(posts_all$Heat=='none' & posts_all$AC!='air conditioning')

#which have both heat and AC
d=length(which(posts_all$Heat=='both' & posts_all$AC=='air conditioning'))
e=length(which(posts_all$Heat=='heater' & posts_all$AC=='air conditioning'))
f=length(which(posts_all$Heat=='fireplace' & posts_all$AC=='air conditioning'))
sum(d,e,f)
sum(d,e,f)/45757*100 # to find %

ggplot(posts_all[!is.na(posts_all$Heat) &!is.na(posts_all$AC),])+
  geom_bar(aes(x=Heat, fill=AC))+
  facet_wrap(Region~.)+
  scale_fill_viridis_d(option="plasma")+
  ggtitle(" Heat vs AC")+
  ylab("Number of Apartments")+
  ylim(0,800)+
  theme_minimal()+
  geom_text(stat='count', aes(x=Heat,label=..count..),vjust=-.5)+
  theme(axis.text.x = element_text(angle = 30, hjust = 1))+
  theme(axis.text=element_text(size=12),
        axis.title=element_text(size=14,face="bold"))+
  theme(strip.background =element_rect(fill="snow"))+
  theme(strip.text = element_text(colour = 'black'))+
  ggsave("heat and AC.png",height=6,width=9)
```

```
ggplot(posts_all[!is.na(posts_all$Heat) &!is.na(posts_all$AC),])+
  geom_bar(aes(x=AC, fill=Heat),position = 'dodge')+
  facet_wrap(Region~.)+
  scale_fill_viridis_d(option="plasma")+
  ggtitle("AC vs Heat")+
  ylab("Number of Apartments")+
  ylim(0,800)+
  theme_minimal()+
  theme(axis.text.x = element_text(angle = 30, hjust = 1))+
  theme(axis.text=element_text(size=12),
        axis.title=element_text(size=14,face="bold"))+
  theme(strip.background =element_rect(fill="snow"))+
  theme(strip.text = element_text(colour = 'black'))+
  ggsave("ac and heat.png",height=6,width=9)


##########################################################################################
###############
#8 email and phone numbers
length(posts_all$Email)
sum(!is.na(posts_all$Email))

length(posts_all$Phone_Number)
sum(!is.na(posts_all$Phone_Number))
table(posts_all$Phone_Number)
which(posts_all$Phone_Number=="0107740054")
```