Analyzing Yelp Restaurant Reviews



By Cole Evans, Gustavo Grinsteins, Stephanie Stieber and Ruth Densamo



The Data Set



- Dataset: Yelp Dataset
 Challenge SQL Dataset
- Dataset consisted of businesses, users, reviews and other numerous tables
- Focus: restaurant reviews





Goal



Improve Yelp's reviewing process by analyzing the reviews using text mining techniques.



Start your review of Tapas on Main.



Data Extraction



Name:	optional	
	x • • • • • • • •	
Host:	Enter the MySQL username to connect with	
Username:	root	
Password:	••••	
Database:	optional	
Port:	3306	
	Connect using SSL	

Large Yelp database

 Used Sequel Pro to extract exact data needed

 Created a SQL database connection in R for specific query extraction



Data Extraction

```
select review.id, stars, review.text, review.date, review.useful,
review.funny, review.cool, business_id, review.user_id, fans
from review left outer join user on review.user_id = user.id
where business id in
                  (select distinct business_ID
                   from category
                   where business id in
                    (select business_id
                     from category
                     where category like "%estaurants")
                     and category like "Mexican")
        order by useful DESC
        limit 100
```

Focused on four popular types of restaurants

- Mexican
- ❖ Indian
- Chinese
- Korean

100 most useful reviews



Data Cleansing



- Created a corpus of the 100 most useful reviews - needs to be cleaned
- Cleaning revolved around text
- Took out punctuation, numbers, symbols and stopwords
- Normalized the text by putting it all in lowercase and stemming the words to prevent duplicate words



Data Handling



- Unsupervised approach in handling data
- Created a document term matrix to isolate individual words
- Performed a TF-IDF: term frequency and inverse document frequency analysis
- Created new data frame of the cross product values
- Normalized the values to reduce document length bias: divided each value by the square root of the total sum of each corresponding row of the new data frame

$$tf-idf_{(t,d)} = tf_{(t,d)} \times idf_{(t)}$$

t = term d = document



Word Cloud: TF-IDF

```
stopped wednesday
friend pepper probably offer excellent
serve decor combo
style add sush ia rea
priced mightmany guac
kind name
itemsanother tortillas small served everything spet
quick muchpork favorite
crunchy however delicious burrios
quick muchpork favorite
crunchy however delicious burrios
quick muchpork favorite
said way enchiladas aroundowner
choice put staff tasty first
ready next different side
offership of the pastor restaurant
price tacos is salas timeordered give street bite
gets drink
size actually eat bean in typical to getting
fries took
getting fries took
tamales soda stars ive
cantina lime
something
fries took
in an gas use guacamole
chorizo flour
topped flavors
now steak in
several cilantro
burrior
several cilantro
locked vegas
ininutes
parking definitely weit
authentic customers
anything sweet
parking definitely weit
authentic customers
anything sweet
parking definitely weit
salasting definitely weit
authentic customers
anything sweet
parking definitely weit
salasting definitely weit
parking definitely
perfect black
parking definitely weit
parking definitely weit
parking definitely
parking defi
                                                                                                                                                                                                                                                                                                                                                                                                             stopped wednesday
                                                                                                                                                                                                                                                                               rolled vegas minutes quesoplacesfront looked vegas large dish times problem specials quality tuesday looking
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  friends prices
```



Analysis

N = 1 : This is a sentence unigrams: this, is, a, sentence
N = 2 : This is a sentence bigrams: this is, is a, a sentence
N = 3 : This is a sentence trigrams: this is a, is a sentence

- Used the clean corpus to create a tokenizer and break up words into unigrams, bigrams and trigrams
- Rather than just ranking the n-grams based on frequency we weighed it using the useful value for each review
- The useful value is an integer value given to reviews
- Useful = measures usefulness that increases as people upvote it
 - Does not matter if review is positive

stars	text	date	useful	funny	cool
2	The tasty cheese crisp app was 4 or	2017-04-03 00:00:00	200	143	179
4	Chain eateries are good once in a w	2017-06-24 00:00:00	175	115	168
4	The service is amazing at the Border	2017-06-15 00:00:00	92	46	90



Analysis

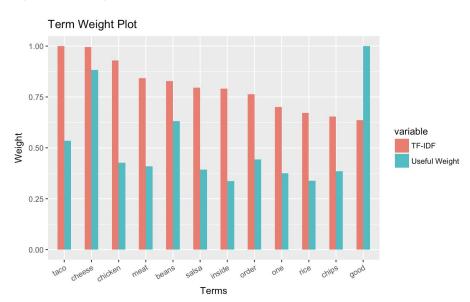


- Weight = useful value X frequency of n-grams
- Final weight is sum of the products across all documents in the term document matrix
- Ex.
 - Review useful value = 10
 - "Taco" in review twice
 - Taco useful value = 20 (for document)
 - Sum all documents in corpus
- Final data frame consisted of the n-gram with its associated total weight



Weighted Unigrams - Mexican Restaurants

Weighted Unigrams-Mexican

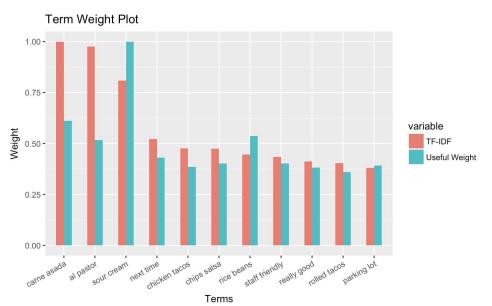


- Ranking terms from highest to lowest TF-IDF compared to usefulness value
- Trivial words such as "good" are weighted less using TF-IDF
- Words like taco, cheese, and chicken high TF-IDF value → indicative of Mexican restaurants
- Unigrams alone don't account for context → look at word sequences



Weighted Bigrams - Mexican Restaurants

Weighted Bigrams-Mexican

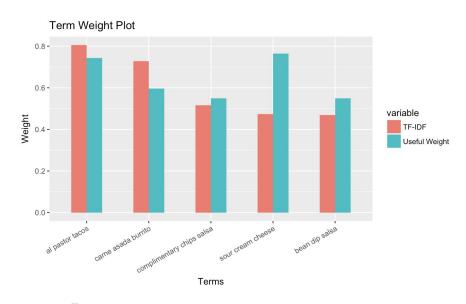


- Top three terms in this graph for TF-IDF:
 - Carne asada
 - Al pastor
 - Sour cream
- Same three values also ranked highly using "weighted" value



Weighted Trigrams - Mexican Restaurants

Weighted Bigrams-Mexican



- Top three terms in this graph are
 - Al Pastor Tacos
 - Sour Cream Cheese
 - Complimentary Chip Salsa
- May not always be best to use more words
- Ideally want to be more general to pertain to all restaurants in category
 - Ex. bean dip salsa too specific
 - Generic, salsa would be better



Results

Mexican:		
Taco	Rice and Beans	Chips and Salsa
Indian:		
Chicken Masala	Naan	Buffet
Chinese:		 -
Dim Sum	Fried Rice	Hot pot
Korean:	Į.	J.
Hot tofu soup	Pork Belly	Short Rib



Proposal

Taco:

Rice and Beans:

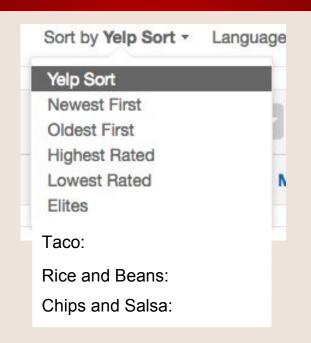
Chips and Salsa:

- Report back to Yelp:
- Implement "Quick Checks"
- Three textboxes per category of restaurant containing 3 most important word(s)
- Reviewers must leave comment for all 3
- Short comment: size of tweet
- Free text response optional





Proposal



- Implement new filter for sorting the reviews
- Add Taco, Rice and Beans, and Chips and Salsa as 3 new sorts (for Mexican restaurants)
- Now can filter reviews to only view comments on the chips and salsa



Benefits





- Leaves more helpful reviews → targeting what people want to know about
- Allows a quick glance at reviews without having to invest time
- Filter out specific item you want to read about (from Quick Check values)
 - Ex. want best chips and salsa around
- Suggest Yelp apply our text analysis to other restaurant categories/businesses as well to better understand what customers value



Questions?

By Cole Evans, Gustavo Grinsteins, Stephanie Stieber and Ruth Densamo

