Final Report

November 29, 2016

Duc Thien Bui

1 Overview

In this project I finished 6 task:

- Task 1: visualize the reviews and its topics, compare negative and positive reviews in term of words.
- Task 2: Base on the reviews of categories visualize and make similarity table of different categories.
- Task 3: Ran program to find dishes name from reviews.
- Task 4: Visualize and compare dishes that found in task 3.
- Task 5: Find a way to recommend best restaurant for a chosen dish.
- Task 6: Base on the data provided about restaurant, make a classifer to find a restaurant pass health inspection test or not.

---

2 Highlight:

2.1 Task 1:

In task 1, I ran LDA topic mining to assign reviews into topics and fingure out what are the most reviews talk about. For example 4 first 4 topics are about food restaurants, burger, pizza, and music show.

```
In [4]: print_model(topic_model_alias)

Topic 0:
```

Topic 1:

Topic 2:

Topic 3:

After that, I attrached positive(5 stars reviews) and negative reviews (less than 3 stars reviews) and ran LDA on them. Because I'm aware of the unblanced data, I resample the reviews so that number of positive and negative reviews that I ran clustering on roundly equal.

```
In [17]: reviews.groupby('stars',{'count':gl.aggregate.COUNT()}).topk('stars',5)

Out[17]: Columns:
                stars        int
                count        int
```

```
Rows: 5

Data:
+-------+--------+
| stars | count  |
+-------+--------+
|   5   | 406045 |
|   4   | 342143 |
|   3   | 163761 |
|   2   | 102737 |
|   1   | 110772 |
+-------+--------+
[5 rows x 2 columns]
```

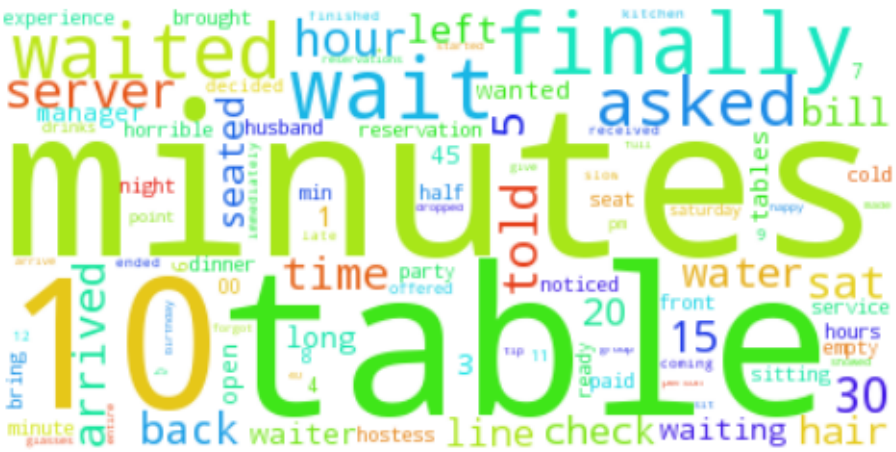
```
In [8]: print_model(positive_model_alias)
```

Topic 0:

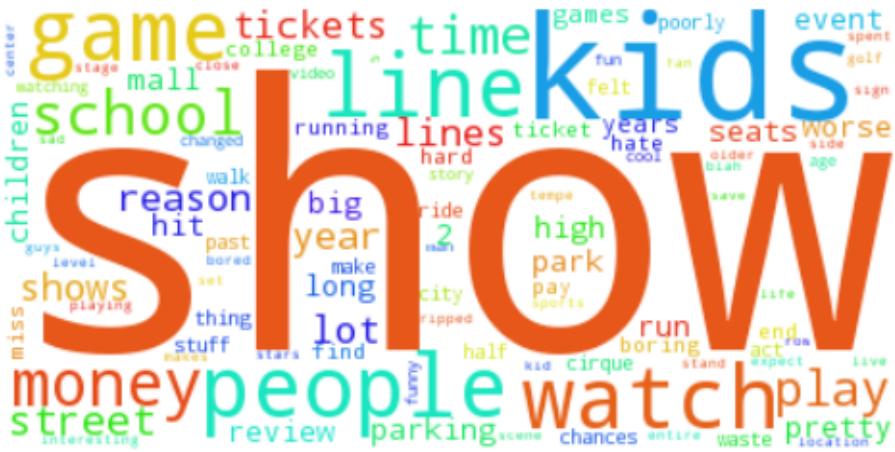
Topic 1:

In [9]: print_model(negative_model_alias)

Topic 0:



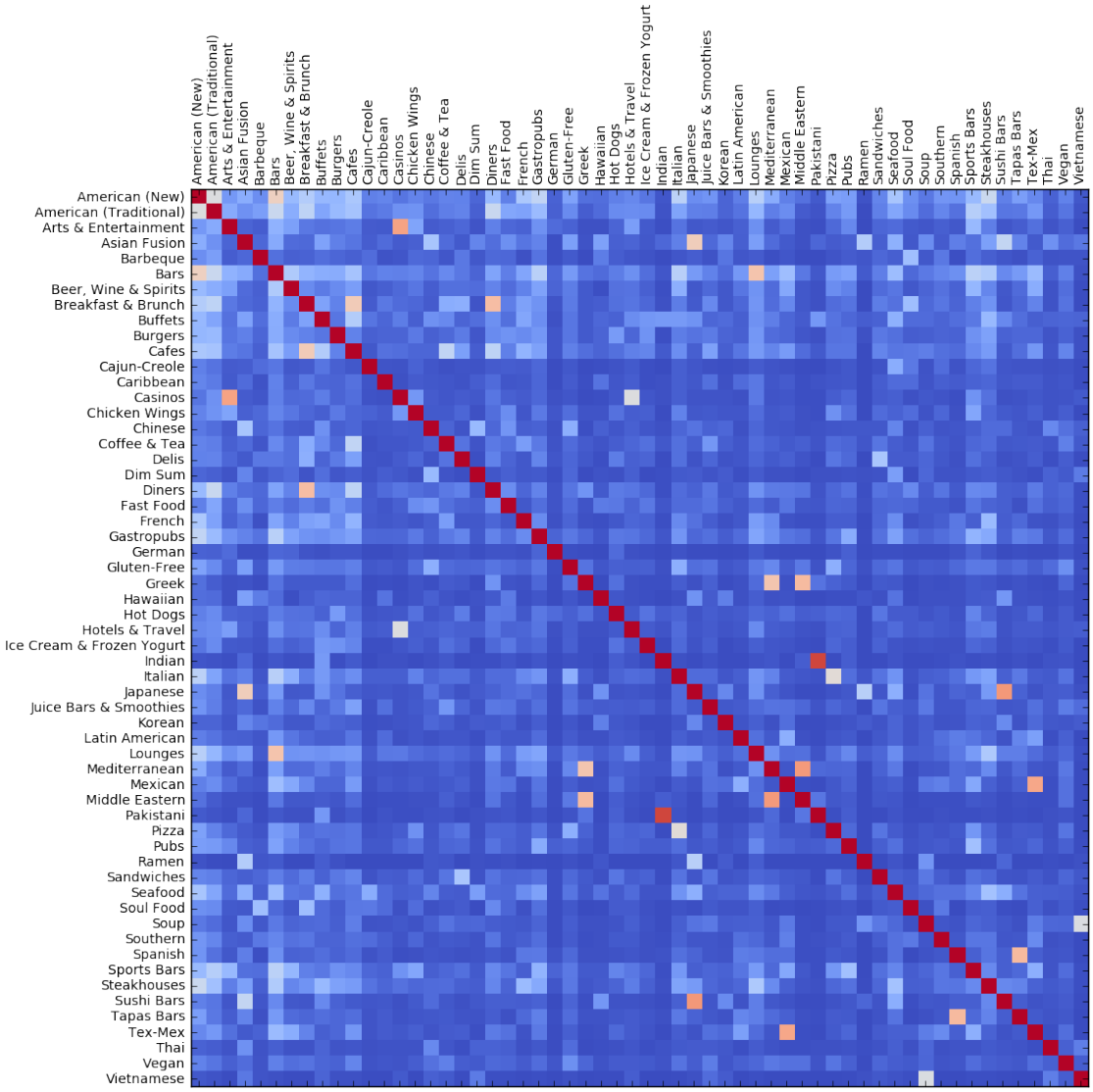
Topic 1:



2.2 Task 2

In this task, I compared the similarities of each pair in 58 categories in term of reviews. After this task, I found some categories are similar and some are very different. At first, I used TF_IDF vector and use cosine distance to calculate the different between each pair of categories. However this method clearly only show the similar between 2 categories that are too close like American traditional restaurants and American new restaurants. So after that, I used LDA modeling and as clustering and assign categories into some topics. Then I set vector represent a category is the probabily of that category belong to that topics. And As can be seen, I worked much better due to the fact the categories the be assigned to a topic are often close to other categories that also belong to that topic.

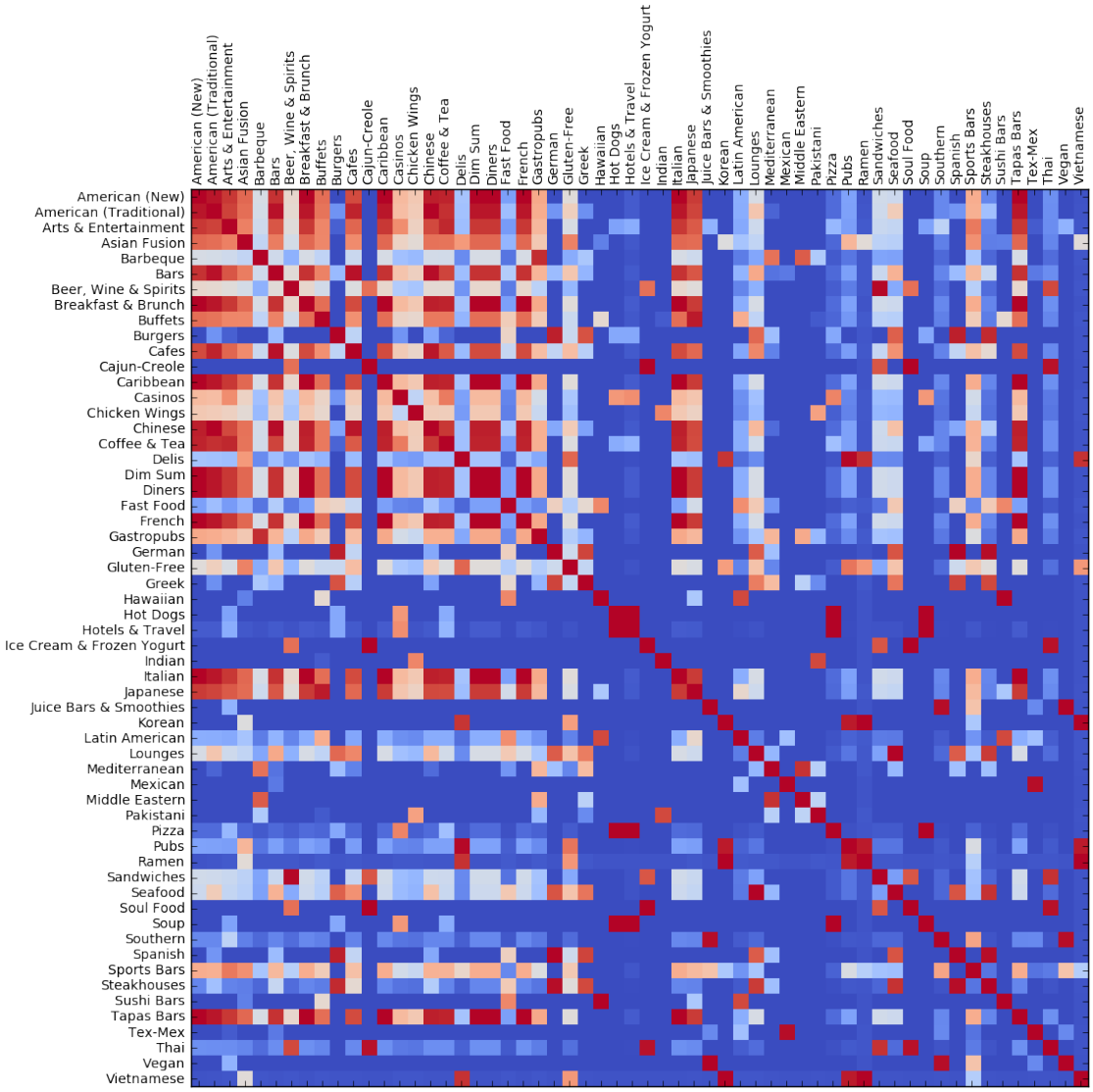
```
In [19]: draw(similarity_ifidf)
```



In [20]: draw_ntopic(10)

```
C:\Anaconda3\envs\gl-env\lib\site-packages\gensim\models\ldamodel.py:527: RuntimeWa
  (perwordbound, numpy.exp2(-perwordbound), len(chunk), corpus_words))
```

Number of topic: 10



2.3 Task 3

In this task, I tried to extract chinese dishes name from reviews about chinese food. To achieve that, I use a heuristic that I only consider phrases that have at least 2 words as I do not want words that are too broad like rice or noodle. I used Phrases and word2vec from genism libraries.

Top 10 phrases using Phases:

```
In [23]: printtop()

chinese food
fried rice
chinese restaurant
orange chicken
```

```
food good
sweet sour
egg rolls
hot sour
pretty good
panda express
```

Top 10 phrases using Word2Vec:

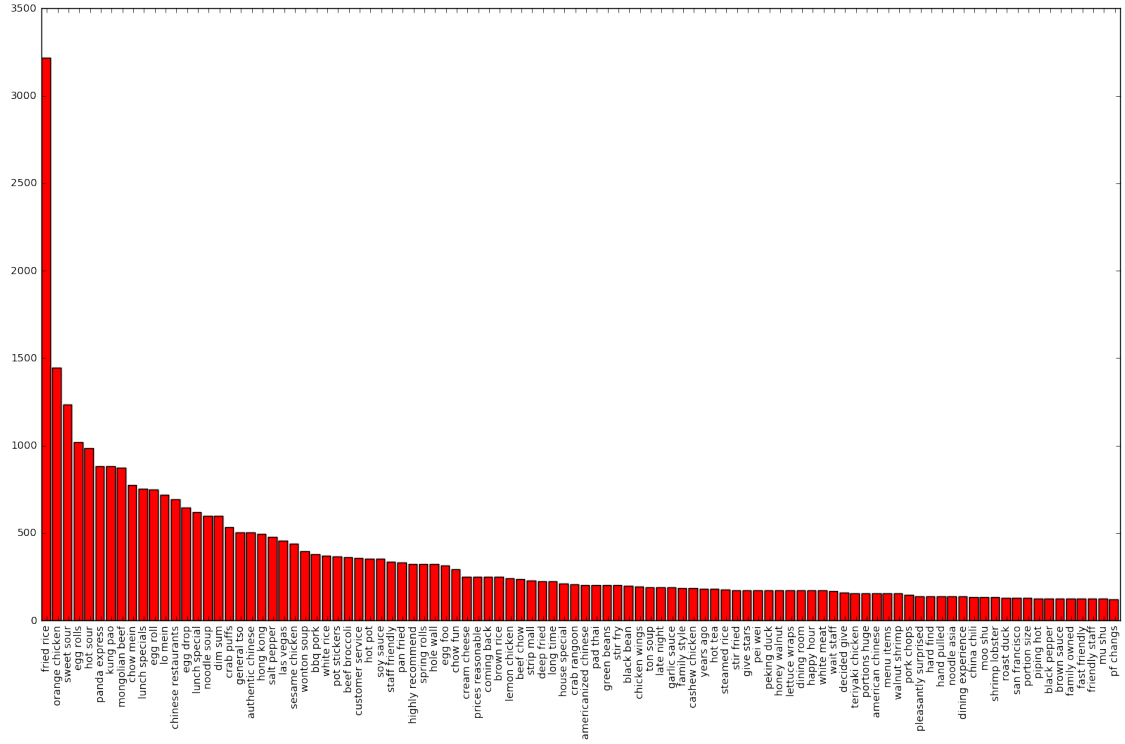
```
In [26]: printtop()

fried rice
orange chicken
sweet sour
egg rolls
hot sour
panda express
kung pao
mongolian beef
chow mein
lunch specials
```

Even though top labels are dishes. These algorithms still have drawback as panda express are ranked even higher in word2vec comparing with phrases. Furthermore, These algorithms can only find well known dishes that are mentioned in many reviews like fried rice or rolls.

2.4 Task 4:

In this task, I ranked the dishes that I found in task 3 base on their popularities and then visualize it. I found that fried rice is the most common chinese food that are reviewed. And it appeared more than double the second most common food: orange chicken.

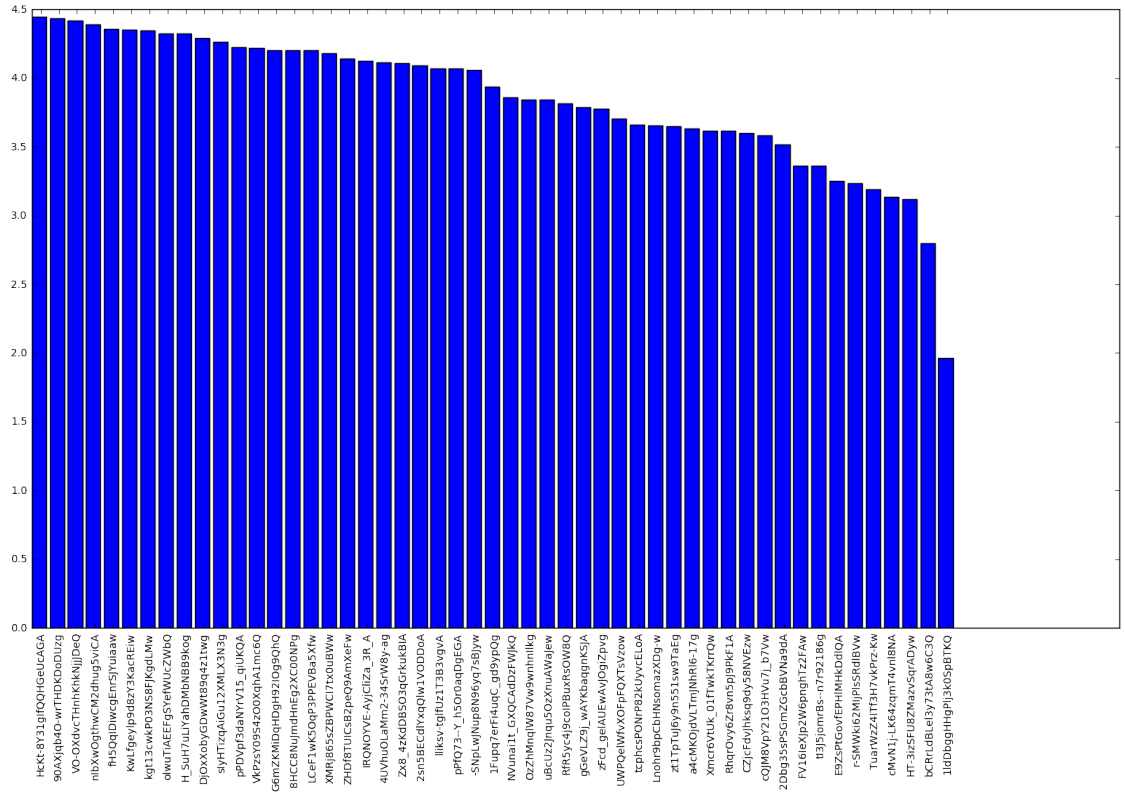
```
In [30]: draw(counter.most_common(100))
```



2.5 Task 5

In this task, I found recommendation for a dish. First, base on the target dish (as I set to be *"fried rice"*) I get all reviews that mentioned it and then sort the restaurants by these reviews avarage stars. However, because some restaurants did not receive sufficient number of reviews to be consider, I left out all restaurants that get less than 25 reviews about the target. And I found the best place to eat *fried rice* is **Rice Trax Teriyaki Grill**

```
In [53]: draw(best_res ,'b')
```



```
In [55]: best_choice[0]['name']
```

```
Out[55]: 'Rice Trax Teriyaki Grill'
```

```
In [58]: best_choice[0]['full_address']
```

```
Out[58]: '7780 S Jones Blvd\nSouthwest\nLas Vegas, NV 89139'
```

2.6 Task 6

2.6.1 Prepare

In this task, I used some algorithm to find the out if a restaurant pass health inspection test or not. The technique I used is very simple. First, I unpack each iteam category tag into features. For instance, if a restaurant belongs to categories *Chinese* and *Thai* then new freature *Cate.Chinese* and *Cate.Thai* of its item will be marked as 1 while other new columns marked (for example *Cate.Japanese*). After that, I run LDA clustering on each items' reviews with 10 topics. Base on the probability of each item into each of 10 topics, I added new 10 features to each item. Rather than LDA with unigram that I ran in task 1, I run LDA with bigram as I think the most common words would be more informative.

2.6.2 Run

After that, I filted out all the data that belong does not have label. And split remaining data into training set and testing set with ratio 90-10. Then, I ran some algorithm with these data to find the best algorithm. **Random forest** gave me 0.74 F1 score. **Boosted trees** gave me 0.68 F1 score. **Nearest Neighbors** gave me 0.70 F1 score. **Support Vector Machine** gave me 0.68 F1 score. And the best one is **Logistice Regrestion** which produced 0.77 F1 score.

---

3 Note

3.1 Usefulness of result

The result of each subtask is very helpful as we step by step learned how to deal with the data.

- The first task gave me some hint about the way I can do with review data.
- The second task showed me that some categories are very similar.
- The third task is crucial to run next 2 tasks and also let me know how to deal with phrases.
- The fourth task gave me the idea of the popular of each dish.
- The fifth task is good to learn some way to deal with data from different database
- The sixth task gave me opportunity using hints from task 1,2, and 3 to run Machine learning algorithms.

3.2 Novelty of exploration

3.2.1 Unpack category data

I think that many people also already use that way to deal with this kind of data. Since I think it is also one of the best way to use along with clustering to transform categories into smaller set of categories. I personally prefer this way if the number of categories is not too big.

3.2.2 Use probability of some algorithms as features for other algorihms to run

In this project, I used probabilities of each review being assigned into each of 10 topics to be their new features. The other way is use classifier algorthms that provide probability like *random forest* ore *neural network* as features added into data to use in other algorithm on top of that.

3.2.3 Use phrases or bigram instead of unigram

Words that stand alone often have too broad meaning. If you want to be specific, using phrases (like I did in task 3) or unigram (like I did in task 6) make more sense that using stand alone words or unigrams (like I did in task 1).

3.3 Contribution of new knowledge

After finishing this project, I found out some interesting things:

- Users are more likely to vote high stars like 4 or 5 than lower stars like 1 or 2. As a result average reviews' stars for each restaurant is often between 3.5 and 4.5
- **Fried rice** are the most common Chinese food in restaurants.
- Users' reviews and restaurants food are highly correlated. We can know many from a restaurant just by processing their customers' reviews.

---

4 Conclusion

Going thru all the tasks, I used many data mining techniques including, but not limited to, topic clustering, feature extracting, classifying. Even though, what I have done is not too good, I think I have learned a lot from this course.

"And as always, thanks for watching"