ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ
ATHENS UNIVERSITY OF ECONOMICS AND BUSINESS

BUSINESS ANALYTICS
Master of Science

Machine Learning and Content Analytics

Professor: **H.Papageorgiou**

Assistant Professor: **G.Perakis**

Project:

“Your Next Best Restaurant’’

Building a Recommendation Engine for YELP

Team Members

Kotsomytis Vasileios p2821908
Romanidis Efstratios p2821922
Spanou Athina p2821924

Table of Contents

1. Introduction

The aim of this project is to create a Restaurant Recommendation System by finding similar reviews of users. We are a group of three people that belong in the Business Intelligence Department of Yelp which consists of Data Scientists and Business Analysts. Yelp is an American public company headquartered in San Francisco, California. The company develops, hosts, and markets the Yelp.com website and the Yelp mobile app, which publish crowd-sourced reviews about businesses. It also operates an online reservation service called Yelp Reservations. Our team is located in the city of Las Vegas, thus, our project is run with data about the state of Nevada. This project is a significant one for Yelp as it will enhance its relationship with the customers. By providing targeted recommendations, the aim is to increase the user satisfaction and as a result the loyalty of the customers. When a user gets a personalized restaurant recommendation based on his reviews he will feel more connected to the service provided. As, a result the customer will tend to use more the app of Yelp, which, at the end of the day, will result in revenue increase. What is more, by getting the user to spend more and more on the app, we can increase his or her familiarity with the Yelp brand, thus, increasing their probability of preferring our services in the future.

Our Recommendation System is based on the logic that users with similar reviews and review history will prefer similar restaurants in the future. To make such a system work, obviously, we need a large number of historical reviews as well as detailed data on the user's behavior. The aforementioned needs are satisfied by the Yelp dataset, which is a subset of its businesses, reviews and user data for use in various projects and purposes. Specifically, it contains the below information:

- 1,320,761 tips by 1,968,703 users
- Over 1.4 million business attributes like hours, parking, availability, and ambience
- Aggregated check-ins over time for each of the 209,393 businesses from 2008 to 2019

2. Mission

At this moment, Yelp only publishes reviews and does not provide any further recommendations. What happens is that if a user wants to pick a restaurant he or she has to apply multiple filters and read multiple reviews to determine a preferable restaurant. With a recommendation engine, the users will be motivated to review and rate more in return for a better user experience.

To build our recommendation model we will use Natural Language Processing (NLP) techniques in order to build an engine that will process and utilize the reviews from users. As we are looking into reviews from users, every review is characterized by a star that the user has given to the restaurant. The engine that has been developed is comprised of two main components. The first is a neural network that faces the multi-class classification problem of the stars of every review. Each review belongs to one out of five classes which are created from the star the user has given to the restaurant from 1 to 5. This project, therefore, is looking into the development, training, optimization and evaluation of four neural network models for multi-class classification. The four models are compared based on their ability to classify a review on the appropriate class it belongs to. When the best model is chosen, we move on to the second component of our project, which is the creation of an interactive recommendation system the recommends restaurants in real-time based on the preferences of the user such as the cuisine he or she prefers.

3. Data Preprocessing

Data Import

First, the datasets of Yelp are provided in json format and as a result we are required to handle 3 json files of different shapes.

The first json file includes the businesses, which are in Yelp.com and concern several states of USA. There are 209393 unique businesses with some attributes such as name, address, hours that businesses are open or not, the average rating in stars and how many reviews refer to of each one. Moreover, each business has a list with binary attributes (True or False), for instance "Good for kids":True.

The second json file is about reviews, with more than 8 million rows (8.021.122) and 9 columns. Each review has a review id, user id, business id, rating (stars from 1 to 5), the date in which the review was submitted and the review text which accompanied it.

The last json file refers to users, with approximately 2 million unique user ids and 22 columns. Apart from the user id, there are 21 other attributes, such as name, the total number of reviews of a user and the average stars, the first registration in platform of yelp and many other indexes of satisfaction and compliments.

Both reviews and user datasets are big enough, therefore we created a function which read each file in chunks of size of 400k and created as many pickles files as needed in order to load easier the data.

Data Cleaning

To begin with, we keep the columns we want from the reviews and the user dataset in order to merge them based on the business id. The aim is to create a data-frame in which we have the information about the business the user has made the review for. Then, we merge the user, business and reviews datasets. Consequently, we keep from the merged dataset the records that contain the word "restaurant" in its categories, because we want to analyze only businesses which are restaurants. We also remove records in which the user has done more than one review in the same business and we keep only the most recent one. Now, the dataset contains 4.078.659 rows and 23 columns. In the context of this project, we focus on the state of Nevada, so we keep only Nevada state which has 1.362.885 unique reviews.

After keeping the state of Nevada, we provide a first insight about cities and observe that except for the main city of Nevada Las Vegas, we have other 17 different city names. The 14 city names have on average 5 businesses and less, because of wrong tagging. Therefore, we rename the cities, keeping only 4 names (Las Vegas, Henderson, North Las Vegas and Boulder City). The table of final dataset with the statistics per city is on the next chapter of Exploratory Data Analysis.

NLP Text Preprocessing

For our analysis, we used “text” and “categories” as input variables, so, before we proceed to the classification, we applied thorough text preprocessing. For “**text**”, we follow the processes below:

- ✓ ***Lower Text:*** We convert the text to lowercase
- ✓ ***Remove non-English words and symbols:*** We discovered that our dataset also contained non-English characters, like Chinese characters, so we decided to keep only English words.
- ✓ ***Replace contractions:*** We replace the apostrophe in short words in our text, like *won't* with phrase *will not* and *'m* with *am*.
- ✓ ***Remove punctuations:*** We remove all punctuations
- ✓ ***Convert accented characters:*** We convert accented characters from text like café
- ✓ ***Remove Whitespace:*** We remove whitespace
- ✓ ***Tokenization:*** The next step is to tokenize the reviews text. Basically, we split the text into word tokens while we also remove any words that are only one character long. Finally, we remove numbers, but not words that contain numbers.
- ✓ ***Remove Stopwords:*** Another step in the text cleaning process is the removal of stopwords. Basically, stopwords are a set of commonly used words like "a", "the", "is" and etc. The purpose behind the removal of stopwords is that by removing low information words from a text, we can genuinely focus on important words instead.
- ✓ ***Lemmatization:*** Lemmatization is another normalization technique and basically is the conversion of each word to its base form, the lemma that we encounter in dictionaries.

When it comes to “**categories”**, the preprocessing is simpler than for the text. First, we removed all punctuation and any extra space. After getting a view on the categories, we decided to keep the 23 most common cuisines. We kept only the 23 categories removing all the other words or characteristics or keywords. These are: *'American', 'Mexican', 'Japanese', 'Italian', 'Asian', 'Chinese', 'Korean', 'French', 'Mediterranean', 'Hawaiian', 'Vietnamese', 'Greek', 'Ethnic', 'Spanish', 'Brazilian', 'Taiwanese', 'African', 'British', 'Caribbean', 'Pakistani', 'Thai', 'Indian', 'Irish'*. What is more, we removed any duplicate cuisine categories (more than once in a business) and remove whitespace. We have to point out that one restaurant may have more than one cuisine category.

Finally, we merged the clean text of reviews and the categories into one column in order to enhance the information of our dataset. Then, we drop the meaningless columns for our analysis. Thus, our final dataset’s shape is **1013794** rows and **21** columns, namely 1013794 unique reviews and its attributes.

Exploratory Data Analysis

Exploration of Rating Stars

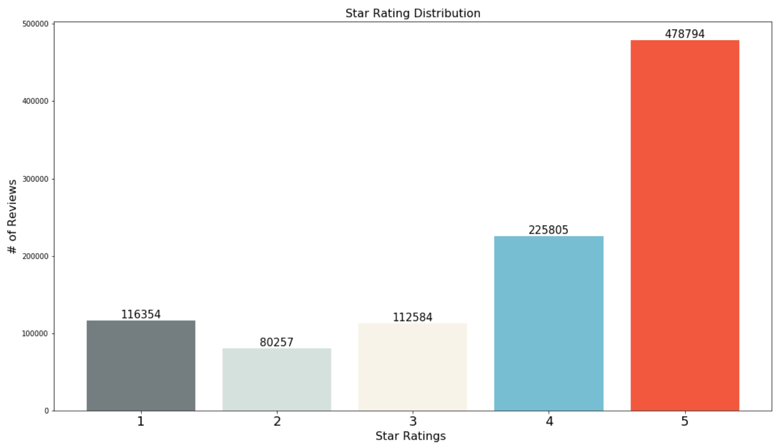


Figure 1 Star Rating Distribution

As we can see at the figure 1, approximately the 47.2% of reviews rated the restaurants with 5 stars (478794 reviews). Moreover, we can notice that the restaurants are rated with 4 stars are 225805, namely the 22.3% of the total reviews. On the other hand, just the 30.5% constitute the “negative” reviews, from 1 to 3 stars. To be more specific, the just the 11.5% of reviews have 1 star, the 7.9% have 2 stars and the 11.1% have 3 stars.

Exploration of Cuisine

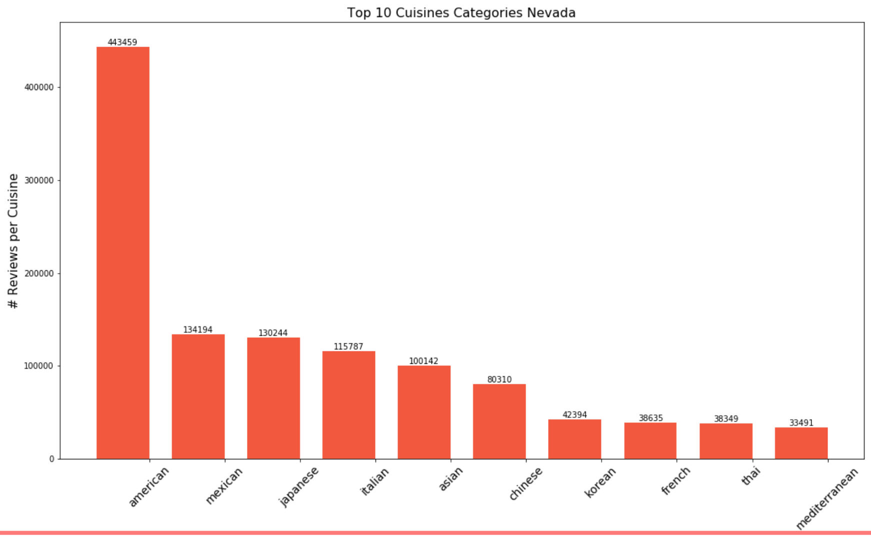
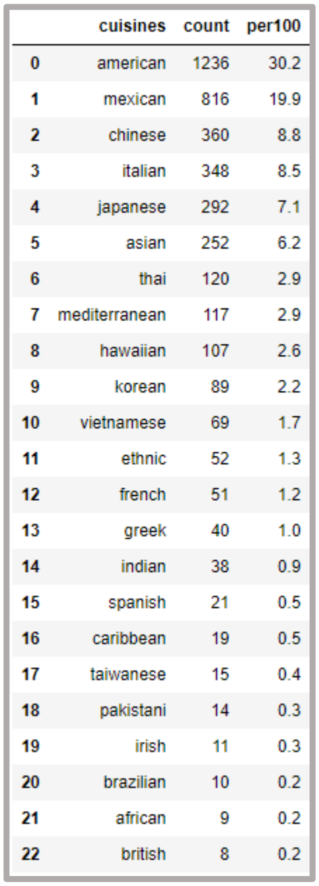


Figure 2 Number of reviews per cuisine Top-10 categories

At the figure above, we can see the top10 cuisines according to the number of reviews. Each restaurant may have more than one cuisine category. So, the most famous cuisine in Nevada is the American with 443459 reviews. Mexican and Japanese cuisines follow with approximately 130k reviews.



	cuisines	count	per100
0	american	1236	30.2
1	mexican	816	19.9
2	chinese	360	8.8
3	italian	348	8.5
4	japanese	292	7.1
5	asian	252	6.2
6	thai	120	2.9
7	mediterranean	117	2.9
8	hawaiian	107	2.6
9	korean	89	2.2
10	vietnamese	69	1.7
11	ethnic	52	1.3
12	french	51	1.2
13	greek	40	1.0
14	indian	38	0.9
15	spanish	21	0.5
16	caribbean	19	0.5
17	taiwanese	15	0.4
18	pakistani	14	0.3
19	irish	11	0.3
20	brazilian	10	0.2
21	african	9	0.2
22	british	8	0.2

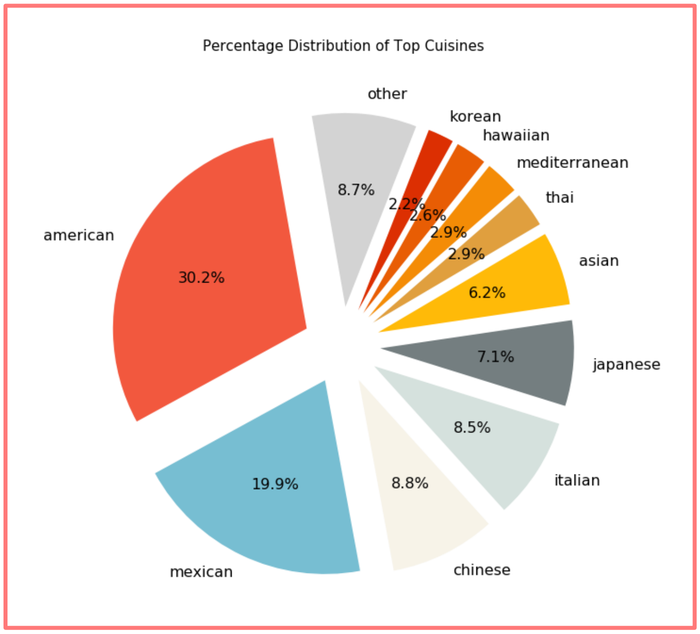
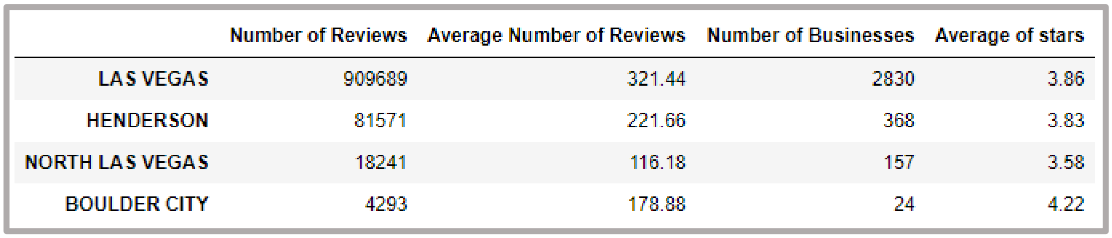


Figure 3 Frequency table & pie chart for unique businesses per cuisine

As expected, the most common cuisine in Nevada is the "american", counting only the categories from the 3379 unique restaurants in Nevada. We have to point out that one restaurant may have more than one cuisine category. To be more specific, the max cuisine categories of a restaurant is 3. Therefore, the American cuisine constitutes the 30.2% of the restaurant cuisines. Furthermore, Mexican with 816 businesses constitutes almost the 20 of restaurant categories. On the other hand, we have 10 cuisines with only 8.7% of the restaurant cuisine distribution.

Exploration of Cities



	Number of Reviews	Average Number of Reviews	Number of Businesses	Average of stars
LAS VEGAS	909689	321.44	2830	3.86
HENDERSON	81571	221.66	368	3.83
NORTH LAS VEGAS	18241	116.18	157	3.58
BOULDER CITY	4293	178.88	24	4.22

Figure 4 Statistics of Reviews per State

As we can see at the figure 4, the most reviews (909689) refer to restaurants which are located to the main area of Las Vegas. In Las Vegas there are 2830 restaurants with average 321 reviews per each business and 3.86 stars per review. The second city of Nevada in number of reviews is Henderson with 81571 reviews. There are 368 restaurants with 221.66 reviews per business and 3.83 stars per review. However, Boulder City which near to Las Vegas has just 24 restaurants with very good rating 4.22 stars per review, but with 178 reviews per business.

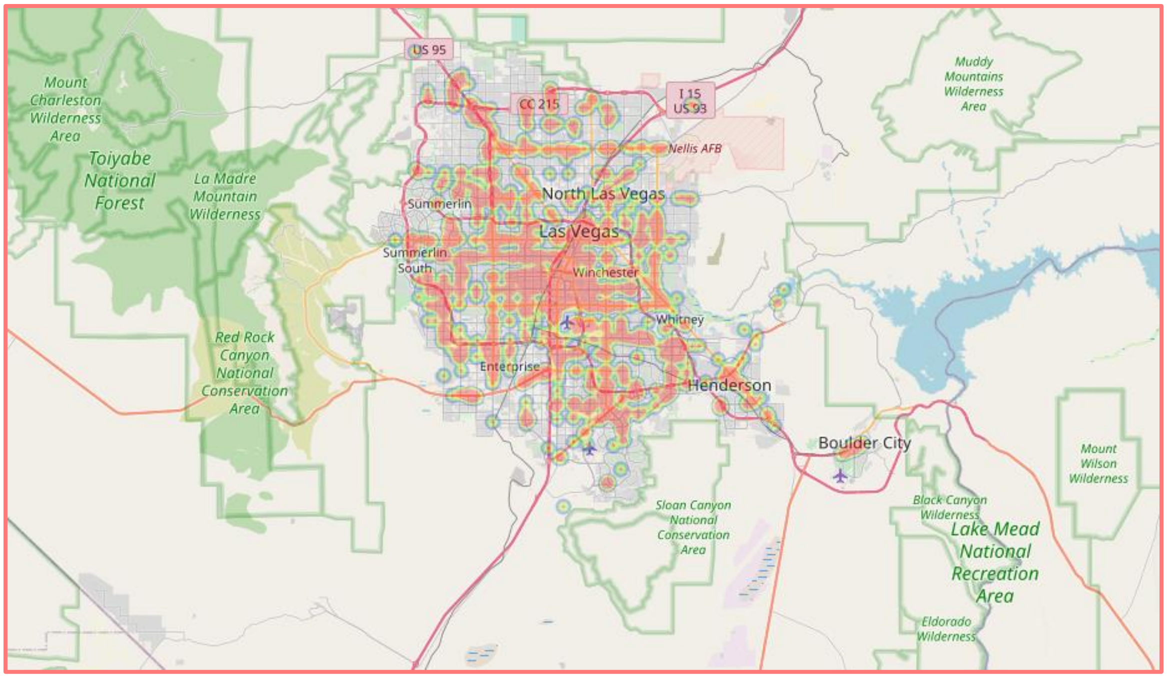


Figure 5 Heatmap of 5 stars restaurants of Nevada

Above, we can see a heatmap with the 5-stars reviewed restaurants. We can observe the 4 different areas of Nevada which we analyzed in the previous table. Once again, we notice that the city of Las Vegas is the one with the most business, thus, the most businesses rated with 5-stars.

Exploration of Reviews

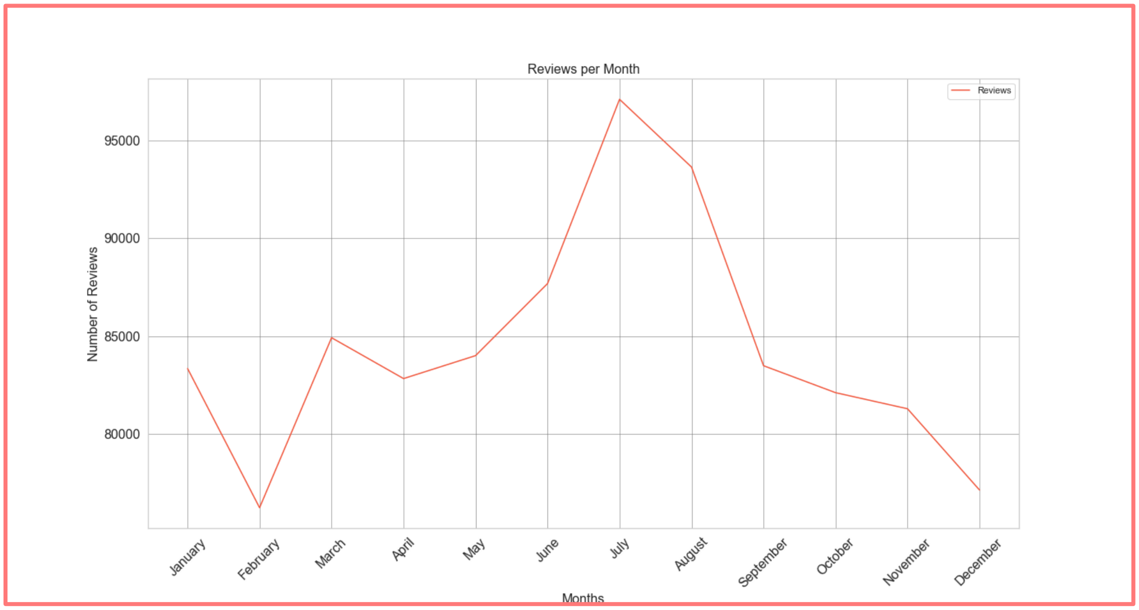


Figure 6 Number of Reviews per Month

The peak of the number of reviews in a month is observed in July. On the other hand, February is the worst month concerning the number of reviews. This is probably because July has more tourists than in other months, as July is the hottest month of the year and a holiday month while February is colder and far away from Christmas.

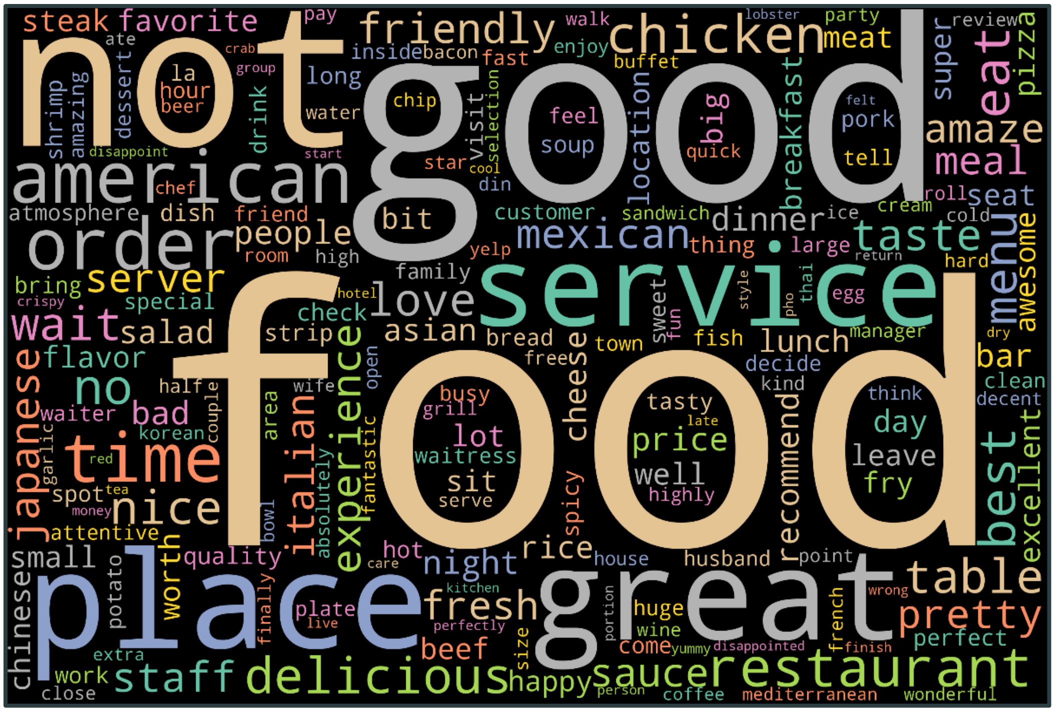
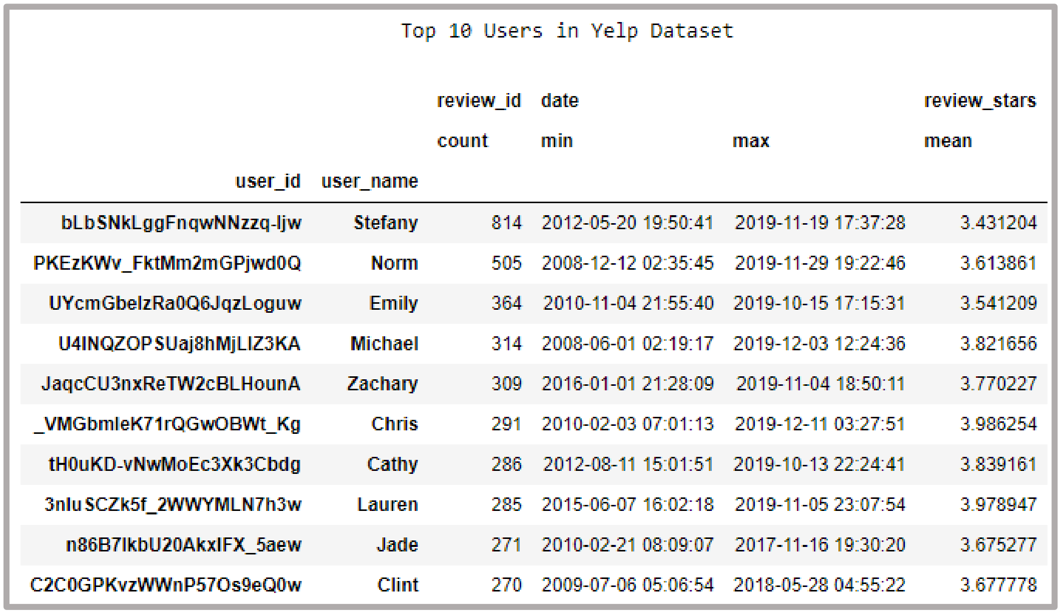


Figure 7 Word cloud of most common words of Input Text

From the figure 7 we can see the most common words contained in the input text. The most common words are bigger depending on their frequency in input text. So, "food" is the most frequent word as it is logical. Moreover, the words "good" and "not" are frequent.

Exploration of Users



Top 10 Users in Yelp Dataset

		review_id	date		review_stars
		count	min	max	mean
user_id	user_name				
bLbSNkLggFnqwNNzzq-Ijw	Stefany	814	2012-05-20 19:50:41	2019-11-19 17:37:28	3.431204
PKEzKWv_FktMm2mGPjwd0Q	Norm	505	2008-12-12 02:35:45	2019-11-29 19:22:46	3.613861
UYcmGbelzRa0Q6JqzLoguw	Emily	364	2010-11-04 21:55:40	2019-10-15 17:15:31	3.541209
U4INQZOPSUaj8hMjLIZ3KA	Michael	314	2008-06-01 02:19:17	2019-12-03 12:24:36	3.821656
JaqcCU3nxReTW2cBLHounA	Zachary	309	2016-01-01 21:28:09	2019-11-04 18:50:11	3.770227
_VMGbmIeK71rQGwOBWt_Kg	Chris	291	2010-02-03 07:01:13	2019-12-11 03:27:51	3.986254
tH0uKD-vNwMoEc3Xk3Cbdg	Cathy	286	2012-08-11 15:01:51	2019-10-13 22:24:41	3.839161
3nIuSCZk5f_2WWYMLN7h3w	Lauren	285	2015-06-07 16:02:18	2019-11-05 23:07:54	3.978947
n86B7IkbU20AkxIFX_5aew	Jade	271	2010-02-21 08:09:07	2017-11-16 19:30:20	3.675277
C2C0GPKvzWWnP57Os9eQ0w	Clint	270	2009-07-06 05:06:54	2018-05-28 04:55:22	3.677778

Figure 8 Top10 users according to most reviews

Above, we can see the top10 users according to most reviews for restaurants in Nevada. First in number of reviews is Stefany with 814 reviews and 3.43 stars as average rating. The first review of Stefany is recorded in May 2012 and the last one in November 2019.

4. Neural Network Models

Tokenizer-Input-Output

In order to continue with the creation of the models for our multi-class classification problem, we first need to split our data in train and test datasets and then apply a tokenization and padding technique. We split our dataset with a ratio for 80:20, which means that 80% of our initial dataset is used for training and validation and the rest, 20%, is used for evaluating each model. The aim for this is to prevent the situation of overfit. Overfit can occur when an algorithm continues to correctly classify data that has already learned but not unseen data. That is why, we will use the test dataset to evaluate our neural network.

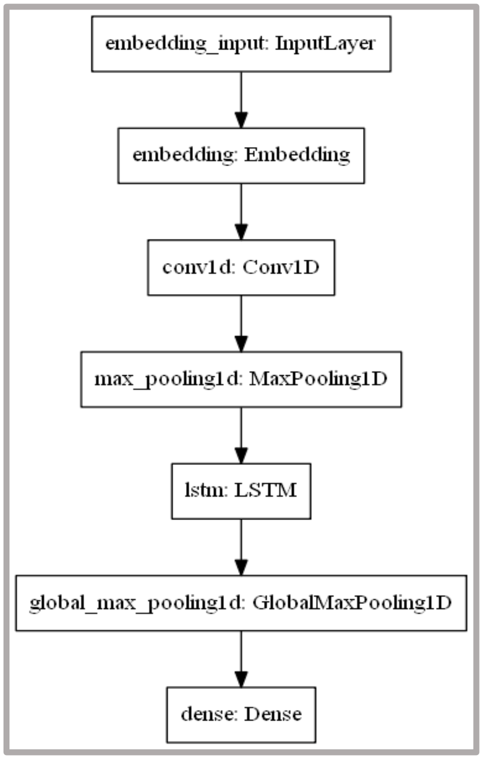
After splitting our dataset we continue by applying tokenization and padding in our dataset. Firstly, we apply the method “fit_on_texts”, a method that created the vocabulary index based on word frequency. As a result, every word gets a unique integer value. Apart from 0 that is reserves for the Out of Vocabulary words, lower integer indexes mean more frequent words. The next step, is to apply the method “texts_to_sequences”, which transforms each word in texts to a sequence of integers. More specifically, it takes each word in the text and replaces it with its corresponding integer value from the word_index dictionary created in the previous step.

It is worth mentioning that we fit once but convert to sequences many times. We fit on out training corpus and use the exact same word_index dictionary on the test dataset to convert actual text into sequences to feed them to the network.

What is more, in order to better address our classification problem we need to convert multi-class labels to binary labels. For this reason, we apply the LabelBinarizer method on our output variable which is the rating stars of a review.

To deal with the multi-class classification problem we have deployed four different neural networks which take as input the review of a restaurant merged with the cuisine it belongs to and provide as output the class of the rating star.

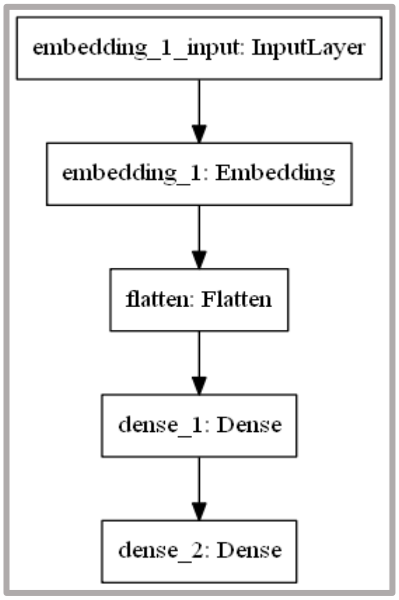
CNN using pre-trained Glove Embeddings


embedding_input: InputLayer
embedding: Embedding
conv1d: Conv1D
max_pooling1d: MaxPooling1D
lstm: LSTM
global_max_pooling1d: GlobalMaxPooling1D
dense: Dense

- We first add an Embedding layer in order to use our pretrained Glove embeddings. The Embedding layer is best understood as a dictionary that maps integer indices (which stand for specific words) to dense vectors. It takes integers as input, it looks up these integers in an internal dictionary, and it returns the associated vectors. It's effectively a dictionary lookup.
- Then, we add a Convolutional layer with 64 filters as the dimensionality of the output space,

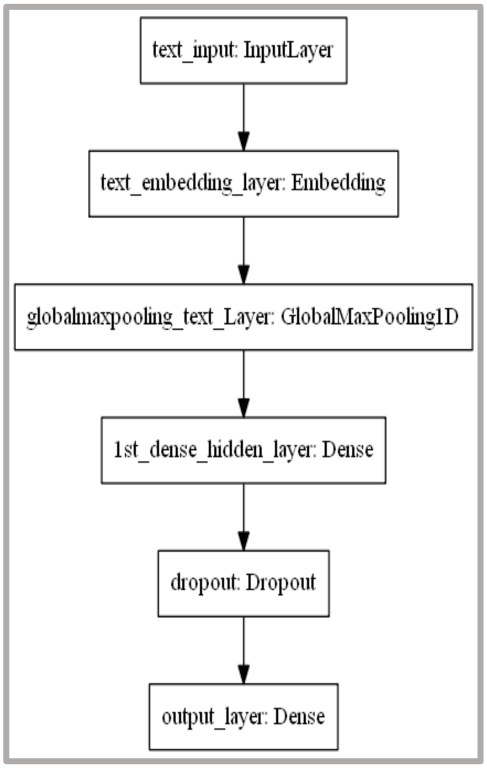
  and a 5-kernel size as the length of the 1D convolution window with a Relu activation function.
- In addition, we add a MaxPooling1D layer in order to downsample the input representation by taking the maximum value over the window (3).
- Next, we add an LSTM layer with 50 units and its default tanh activation function.
- Furthermore, we use a GlobalMaxPooling1D layer and finally since we have a multiclass classification scheme, we select 5 neurons with a softmax activation function.
- We, also, specify an early-stopping strategy. A strategy of ending the neural network training on an epoch earlier than the number of epochs specified while taking into consideration the improvement in the loss function.
- In the end, we compile the model with a categorical cross entropy, adam optimizer and the accuracy metric.

MLP using pre-trained Fast-Text Embeddings



- We first add an Embedding layer in order to use our pretrained FastText embeddings. The Embedding layer is best understood as a dictionary that maps integer indices (which stand for specific words) to dense vectors. It takes integers as input, it looks up these integers in an internal dictionary, and it returns the associated vectors. It's effectively a dictionary lookup.
- Afterwards, we add a flatten layer which flattens the input.
- After that, we add a Dense layer with 32 units and an Relu activation function.
- Moreover, since we have a multiclass classification scheme, we select 5 neurons with a softmax activation function.
- We, also, specify an early-stopping strategy. A strategy of ending the neural network training on an epoch earlier than the number of epochs specified while taking into consideration the improvement in the loss function.
- Finally, we compile the model with a categorical cross entropy, adam optimizer and the accuracy metric.

MLP Using Self-Trained Embeddings


text_input: InputLayer
text_embedding_layer: Embedding
globalmaxpooling_text_Layer: GlobalMaxPooling1D
1st_dense_hidden_layer: Dense
dropout: Dropout
output_layer: Dense

- First, we add an input layer to our model, the layer that receives the information from the independent variable. Its input dimension is equal to the vocabulary size of our text data while its input dimension is equal to the length of the input sequences.

- We then add an Embedding layer in order to train our own embedding. The Embedding layer is best understood as a dictionary that maps integer indices (which stand for specific words) to dense vectors. It takes integers as input, it looks up these integers in an internal dictionary, and it returns the associated vectors. It's effectively a dictionary lookup.

- Then, we use a GlobalMaxPooling1D layer which downsamples the input representation.

- Next, we add a Dense layer with 32 units, an L2 regularization and a Relu activation function.

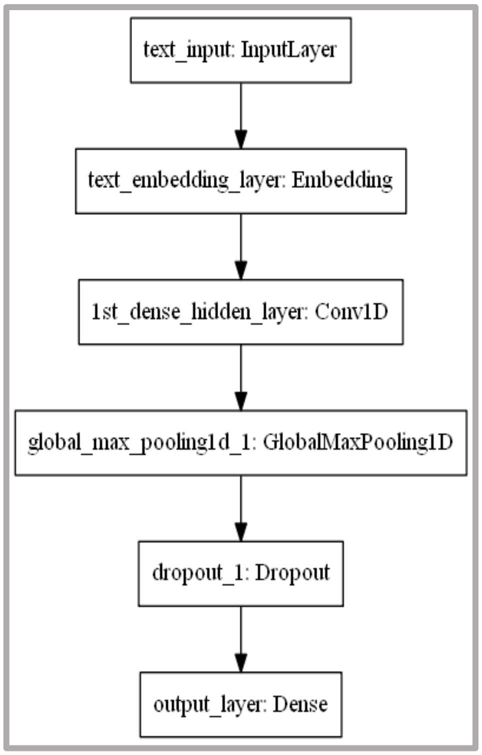
- Furthermore, we add a Dropout layer with a 10% fraction of input units to drop. This means that 10% of the neurons will receive a zero weight. This operation controls the regularization process and helps in preventing over-fitting.

- Afterwards, since we have a multiclass classification scheme, we select 5 neurons with a softmax activation function.

- We, also, specify an early-stopping strategy. A strategy of ending the neural network training on an epoch earlier than the number of epochs specified while taking into consideration the improvement in the loss function.

- Finally, we compile the model with a categorical cross entropy, adam optimizer and the accuracy metric.

CNN Using Self-Trained Embeddings


text_input: InputLayer
text_embedding_layer: Embedding
1st_dense_hidden_layer: Conv1D
global_max_pooling1d_1: GlobalMaxPooling1D
dropout_1: Dropout
output_layer: Dense

- First, we add an input layer to our model, the layer that receives the information from the independent variable. Its input dimension is equal to the vocabulary size of our text data while its input dimension is equal to the length of the input sequences.

- We then add an Embedding layer in order to train our own embedding. The Embedding layer is best understood as a dictionary that maps integer indices (which stand for specific words) to dense vectors. It takes integers as input, it looks up these integers in an internal dictionary, and it returns the associated vectors. It's effectively a dictionary lookup.

- Then, we add a Convolutional layer with 64 filters as the dimensionality of the output space, and a 3-kernel size as the length of the 1D convolution window with a Relu activation function.

- Next, we use a GlobalMaxPooling1D layer which downsamples the input representation.

- Moreover, we add a Dropout layer with a 10% fraction of input units to drop. This means that 10% of the neurons will receive a zero weight. This operation controls the regularization process and helps in preventing over-fitting.

- What is more, since we have a multiclass classification scheme, we select 5 neurons with a softmax activation function.

- We, also, specify an early-stopping strategy. A strategy of ending the neural network training on an epoch earlier than the number of epochs specified while taking into consideration the improvement in the loss function.

- In the end, we compile the model with a categorical cross entropy, adam optimizer and the accuracy metric.

Hyper-Parameters

For every one of the models a variety of hyper-parameter selections processes have taken place. The variables that have been optimized are the following:

- Learning Rate: Determines the step size at each iteration while moving toward a minimum of a loss function. Since it influences to what extent newly acquired information overrides old information, it metaphorically represents the speed at which a machine learning model "learns".
- Number of Epochs: The training procedure makes multiple passes through the entire set of training patterns, eventually reducing the learning rate step size, until the net performance stabilizes on the cross-validation set. Each pass is known as an epoch.
- Batch Size (Hidden Units): The number of sequences that are injected into the model, per epoch, per training.
- Regularization: Regularization penalizes the coefficients. In deep learning, it actually penalizes the weight matrices of the nodes. We need to optimize the value of regularization coefficient in order to obtain a well-fitted model. L1 and L2 are the most common types of regularization.
- Dropout Rate: Dropout is a regularization method that approximates training a large number of neural networks with different architectures in parallel. The default interpretation of the dropout hyper-parameter is the probability of training a given node in a layer, where 1.0 means no dropout, and 0.0 means no outputs from the layer.
- Optimizers: Algorithms or methods used to change the attributes of the neural network such as weights and learning rate to reduce the losses.

Training of the Models

Initially, we have to mention that all the model were run using python through Jupyter Notebook. One of the major **problems** we faced during the training of our models was the huge number of data that we had to process. To be more specific, we even had around 10 million trainable parameters in some of the models. As a result, the whole training process could take up to 4 hours for a model alone.

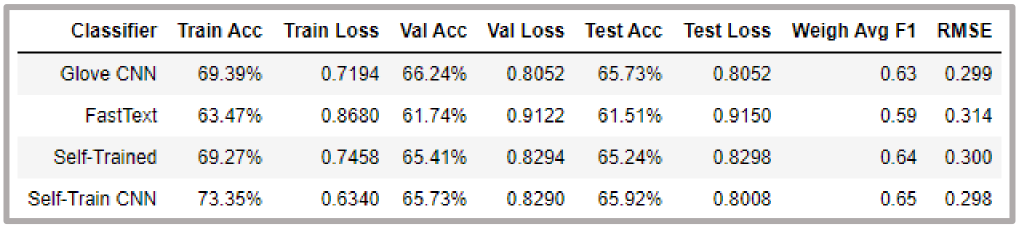
5. Evaluation-Results

The metrics that we have chosen in order to select the neural network that will stand out from the rest of the trained estimators are the following:

- **F1-score**: Select the model with the highest F1 score
- **Area Under the Curve (ROC-AUC) score**: Select the model with the highest ROCscore.
- **Confusion matrix & Classification report**: Select the best model that illustrates a complete capability of classifying all the five stars.
- **Training and validation learning curves**: Select the model that demonstrated the best progress of validation and training hamming loss & binary crossentropy through the training of epochs.

Metrics Report

The metrics for our classification models are summarized below:



Classifier	Train Acc	Train Loss	Val Acc	Val Loss	Test Acc	Test Loss	Weigh Avg F1	RMSE
Glove CNN	69.39%	0.7194	66.24%	0.8052	65.73%	0.8052	0.63	0.299
FastText	63.47%	0.8680	61.74%	0.9122	61.51%	0.9150	0.59	0.314
Self-Trained	69.27%	0.7458	65.41%	0.8294	65.24%	0.8298	0.64	0.300
Self-Train CNN	73.35%	0.6340	65.73%	0.8290	65.92%	0.8008	0.65	0.298

Figure 9 Table with metrics of 4 models

- Taking into consideration all the above metrics and plots, we choose the CNN with Self-Trained embeddings as the best model, because it achieves better train-validation and test accuracy and enough smaller loss. Apart from that, this model has the highest weighted average F1-score, as well as the lowest Root Mean Square Error.

- When it comes to ROC curves, the Glove CNN and the Self-Trained CNN models have similar behavior and metrics. So, we consider the accuracy, loss and F1 and RMSE scores in order to select the best model.

Confusion Matrix

We create the confusion matrix for our model:

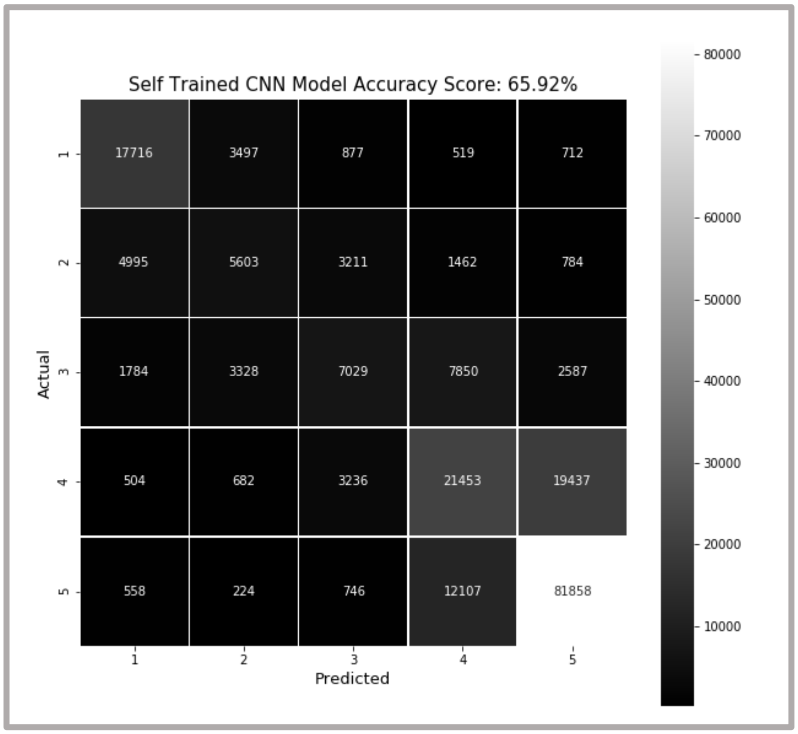
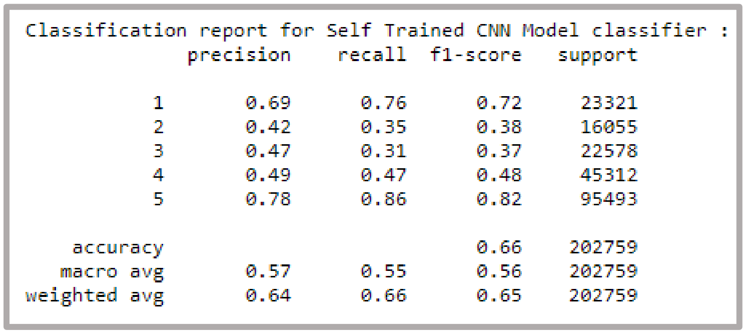


Figure 10 Confusion Matrix of CNN model with Self Trained Embeddings

Above, we have a good presentation for the confusion matrix. At the diagonal we can see the correctly predicted reviews' rating. Obviously, 5-stars rated reviews are predicted better than any other class with 81858 correctly 5-stars rated reviews. The least correctly predicted rated reviews concern the 2-stars with 5603, but it is about the smallest in size class with 16k reviews.

Classification Report



```
Classification report for Self Trained CNN Model classifier :
              precision    recall  f1-score   support

           1       0.69      0.76      0.72     23321
           2       0.42      0.35      0.38     16055
           3       0.47      0.31      0.37     22578
           4       0.49      0.47      0.48     45312
           5       0.78      0.86      0.82     95493

    accuracy                           0.66    202759
   macro avg       0.57      0.55      0.56    202759
weighted avg       0.64      0.66      0.65    202759
```

Figure 11 Classification report of CNN model with Self Trained Embeddings

We can notice that rating of 5 stars has the better recall, while classifier predict correctly the 81858 out of 95493 reviews which belong to 5th class. This was expected, because 5 stars rating is the class with the most observations. However, we see that 3 stars with only 7029 correctly predicted out of 22578 reviews in testing data return a very low recall 31%.

Area Under the Curve score (ROC-AUC)

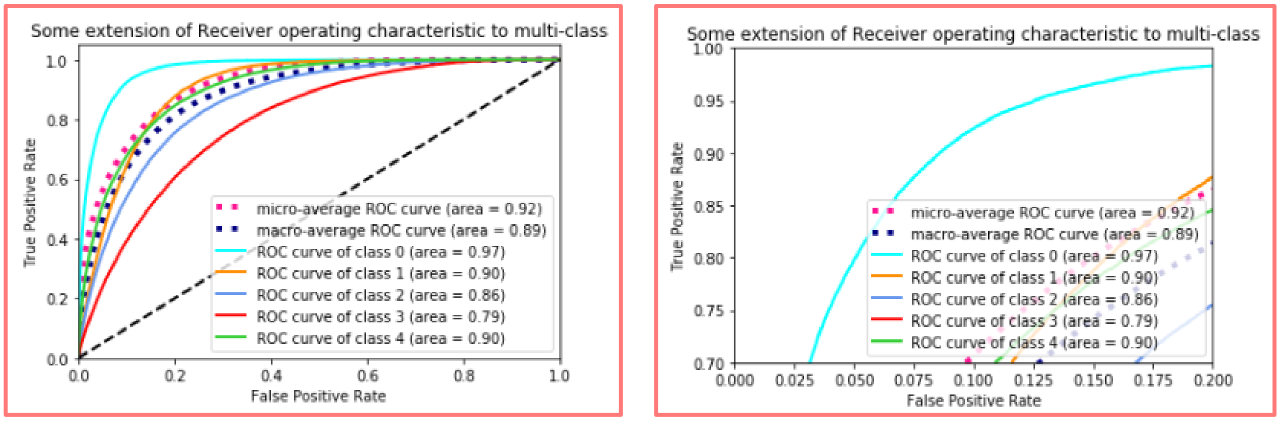


Figure 12 ROC Curves to multi-class

Above, we can see the Receiver Operating Characteristic curves for our model in general and for each class separately. It is a graph showing the performance of our classification model at all classification thresholds. It seems to fit good enough. Our classifier gives curves closer to the top-left corner, so this indicates a good performance.

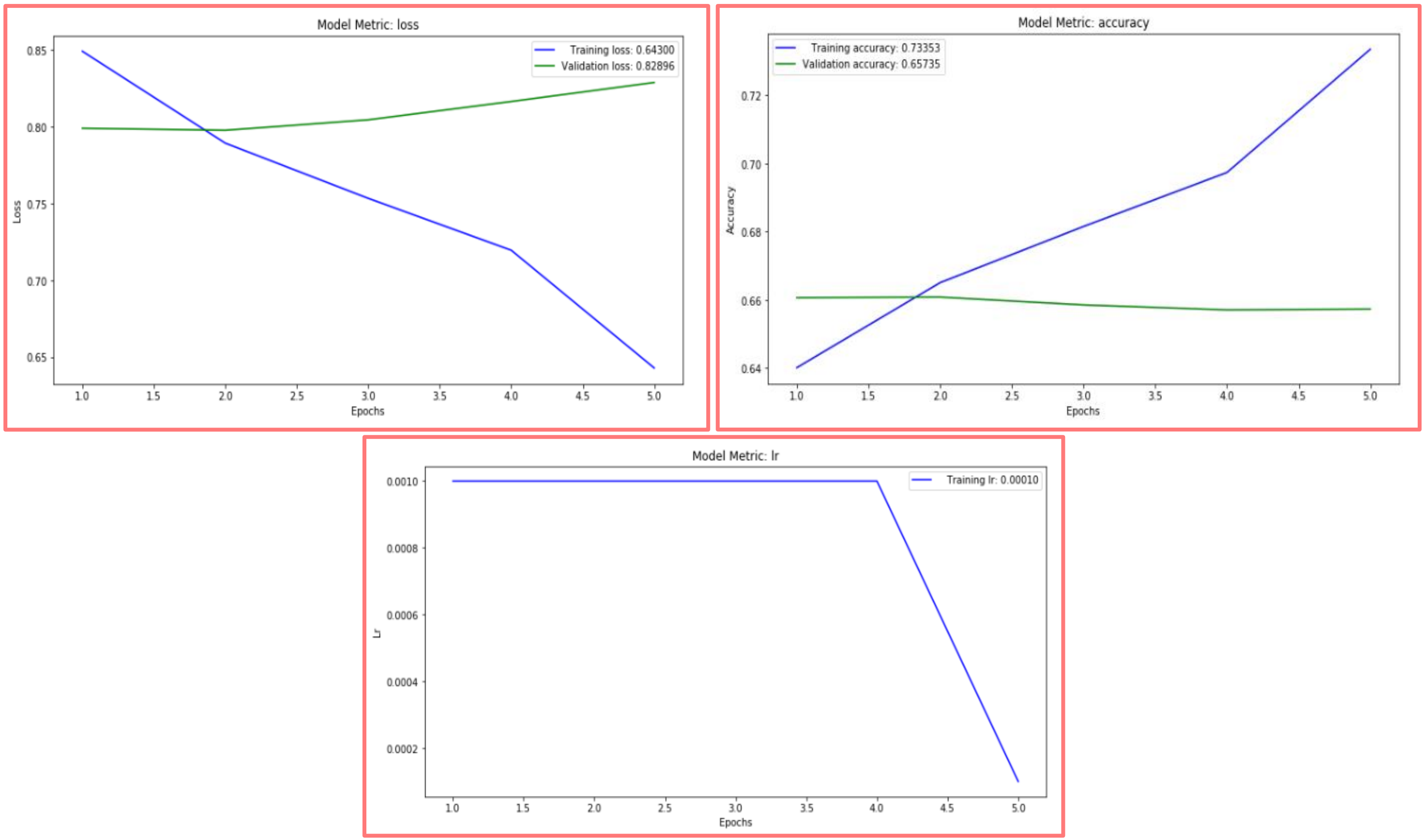


Figure 13 Plot the keras-history of the model for 3 metrics

At the plots above, we can see the history of our model for 3 different metrics. When it comes to loss and accuracy, we can compare the development of these metrics for both training and validation data. At the last plot, we can observe the history of training lr.

6. Recommendation System

In this section the recommendation engine and its structure are analyzed. The object of the recommender is to give the ability to the user to have real-time restaurant recommendations based on his/her past reviews and his/her current preferences. The recommendation agent takes as input the most recent review of the user as well as the one with the highest star in order to be more precise about his/her tastes. After we extract the embeddings of the best classifier, as described in the previous section, we normalize them so that the dot product between two embeddings becomes the cosine similarity. The recommendation engine, then, takes in a user review, a set of embeddings, and returns the n most similar items to the review. It does this by computing the dot product between the review and embeddings. Because we have normalized the embeddings, the dot product represents the cosine similarity between two vectors. Once we have the dot products, we can sort the results to find the closest entities in the embedding space. With cosine similarity, higher numbers indicate entities that are closer together, with -1 the furthest apart and +1 closest together. Afterwards, the user is asked about the cuisine he/she prefers to try and the engine returns the restaurant that satisfies the preferences of the user. It is worth mentioning that we have made sure the recommendation engine will exclude from the similar restaurants the past reviews of the same user, thus every restaurant that is recommended is one that the user has not made a review about yet also one with similar reviews and rating stars to his/her most recent and positive reviews.

We will now see the results of the recommendation engine on two kind of users, one with many past reviews and one with only one review.

Demonstration: User with Many Past Reviews

For this demonstration we have picked the user with id "*iW6YSCu3YVI-SNPNi0I-xg*" who has made 11 reviews so far. His name is Glenn and his most recent review and the one with the highest rating star is this one: *"time boy best onion not bad american".* The restaurant the review is about is of an American cuisine and the user has rated the restaurant with a 5 star. The engine computes and returns the 20 most similar reviews with that of the user. We can see from the table below, that apart from the first review, the second best review has a similarity of almost 0.7 with our target review.

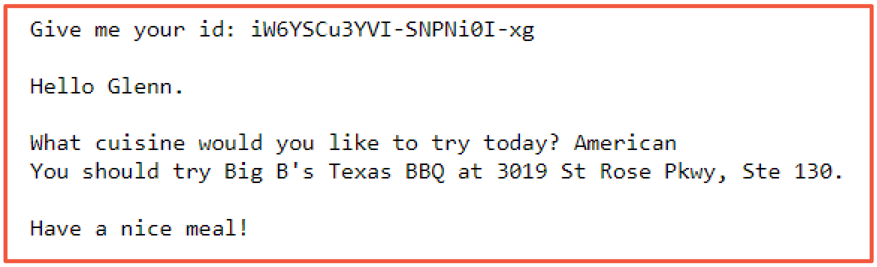
Number	Review	Similarity
0	time boy best onion not bad american	**1.0**
1	food absolutely perfect hit mark meat delicious side perfect american	**0.661**
2	eaten time great picky food notch service friendly quick finally decent food thai	**0.6**
3	great food affordable service good drag portion restaurant slow service occasion portion restaurant experience good food portion opposite mi japanese	**0.584**
4	problem place close live fish awesome chip good order thing time eat comment menu service spot enjoy game pretty good good favorite dislike repeat time mexican american	**0.574**

5	delicious favorite night love happy hour delicious highly recommend roll roll food atmosphere pleasant amaze come mode regret japanese	**0.574**
6	amaze piece advice pay extra honestly regret great food great price atmosphere comfort wait husband dinner night great friend meat delicious recommend place wait round japanese	**0.567**
7	large sign post restaurant bloody mary seat waitress late finally mary hour no apology no duty mexican	**0.563**
8	late lunch not busy surprise lot selection soft shell crab favorite price excellent japanese	**0.56**
9	best la beat meat town love place steak huge soft asian korean	**0.545**
10	staff come coming water sip constantly wait sip bam water place best soy sauce order entree asian chinese	**0.54**
11	good sandwich place sandwich shop man wrong customer service food great place clean table stylish walk friendly cashier time good experience sandwich caper lemon sauce bread order cashier sample soup bread dip sauce wait bread soft freshly bake beef soup nice flavor finally sandwich nice organiz...	**0.537**
12	ate week ago experience good excite seated lot open wait seat friendly helpful food good special omelet purple sweet potato toast cream pork belly bacon delicious worth favorite bone american	**0.537**
13	pretty good place people food good pretty clean staff pretty good courteous mention food good thai	**0.53**
14	hunt supposedly grand lux good thankfully open mind breakfast menu city no limit menu base time enjoy great meal grand lux minute wait place extremely popular seat menu huge literally taste tasty warm beef sour cream main safe chili cheeseburger nice fat patty cover delicious short rib chili goo...	**0.526**
15	restaurant ago watch eat feature spot tiger owner work magic grill time reservation visit time place clean roomy cozy food great nice quality food din experience japanese	**0.519**
16	great deal choice eat people alcohol price person good stuff limit person understandable reason star salmon poke plate food service great meal korean	**0.517**
17	order burrito yesterday bland rice grill chicken small dry no side pico no sauce tomatillo sauce not wet burrito not well option base price chip price spend burrito salty not good stomach ache stomach time eat ordered salad meat order no nice description not figure grill no no salad no meat not ...	**0.515**
18	fantastic lunch prepared perfection sauce sausage compliment sauce service efficient not overbearing time town italian	**0.514**
19	authentic street place good red super good mexican	**0.51**

The next step is for the engine to map the reviews to our initial dataset so as to extract the information about the restaurant, as well as the cuisine, the most similar reviews are about. In this step, the engine excludes from the recommendations reviews that map to the same user id as the one we have as input. Below, we can have a view on the list with the top recommended restaurants. We can see that the first restaurant is of an American cuisine, as the one of the user that we have used as an input. Also, given that the review we had as input had a 5-star rating, we can see that the recommendation also are not that far off from a 5-star.

Number	Name	Cuisine	Stars	User
0	**Big B's Texas BBQ**	american	**4.04**	QFwSaMCndJydXVxzErpEzw
1	**Thai Spoon Las Vegas**	thai	**3.67**	GmzwysGSE-gh0-RgG1_QNA
2	**Su Casa**	japanese	**4.1**	kLc-KHMaHBhPw6MVOibiOw
3	**Nacho Daddy**	mexican american	**3.97**	_5m99dbfiPa0qnf8tKWqRQ
4	**Kobe Sushi Bar**	japanese	**3.67**	O12IUTiitzYcvQfDYzhnww
5	**Cafe Sanuki**	japanese	**5.0**	3bgZCU_tHkn005u019frLg
6	**Carlos'n Charlie's**	mexican	**3.2**	EhVbz35G6vjdDwcoOgkxRg
7	**Hayashi Japanese Cuisine**	japanese	**4.0**	lFk5FBqxNuvDv6b90EJ1gw
8	**Captain6 Korean BBQ**	asian korean	**5.0**	RdyyZFs1bBdFG55Rtvuz5Q
9	**Rice & Company**	asian chinese	**5.0**	6iHdhJs6EAdkTW67CN2iHw
10	**Karved**	american	**4.6**	BhMXmlnRZL-D1cs7PkHZeg
11	**Truffles N Bacon Cafe**	american	**3.37**	AScKCwgY3O8TZiodqel6Sw
12	**Nittaya's Secret Kitchen**	thai	**4.43**	TzyZwy604WbCrZvUHrWrwQ
13	**Grand Lux Cafe**	american italian	**4.32**	GL-vCeVAYEEV9WZPUibUtw
14	**Musashi Japanese Steakhouse**	japanese	**3.64**	375K-TIRUQNEGsOHuVyW3w
15	**888 Korean BBQ**	korean	**3.93**	0QSDYYN0c4Sf05z05rlZKg
16	**Cafe Rio**	mexican	**3.21**	UsnVveOndIxCpxvHARI8AA
17	**Pasta Shop Ristorante**	italian	**5.0**	NiWvYEqkDiWKDBwqBmCzzA
18	**Tacos La Carreta**	mexican	**5.0**	HGSx8Vwsc5fbexFNcyBD2Q

Finally, the user is asked about the cuisine he would like to try and the engine filters the list of the best recommended restaurants in order to return the one that matches his preferences. Below, we can see the view of the recommendation engine from the side of the user.



```
Give me your id: iW6YSCu3YVI-SNPNi0I-xg

Hello Glenn.

What cuisine would you like to try today? American
You should try Big B's Texas BBQ at 3019 St Rose Pkwy, Ste 130.

Have a nice meal!
```

Demonstration: User with One Past Review

For this demonstration we have picked the user with id "*f38DGhAQOxQu07P-4JMpTw*" who has made 11 reviews so far. Her name is Shanna and her most recent review and the one with the highest rating star is this one: *"place awesome thought la order shack stack cheese man not disappoint mushroom mushroom act patty stuffed cheese fry wow cheese unbelievable husband shack stack cheese perfect husband good coffee shake chocolate mint shake winner winner chicken dinner eat lunch day not disappoint american".* The restaurant the review is about is of an American cuisine and the user has rated the restaurant with a 5 star. The engine computes and returns the 20 most similar reviews with that of the user. We can see from the table below, that apart from the first review, the second best review has a similarity of 0.62 with our target review.

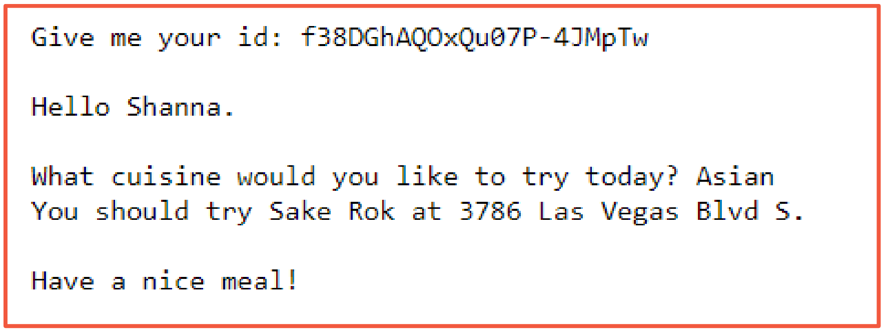
Number	Review	Similarity
0	place awesome thought la order shack stack cheese man not disappoint mushroom mushroom act patty stuffed cheese fry wow cheese unbelievable husband shack stack cheese perfect husband good coffee shake chocolate mint shake winner winner chicken dinner eat lunch day not disappoint american	**1.0**
1	good buffalo completely raw bite felt hard eat piece felt worse previous leave told waiter care serve bother taste discount response tell raw end not satisfied not american african	**0.627**
2	best friend love good love letter step best friend love childhood grow walk restaurant throwback corner store school grab bag chip soda candy suddenly transport din experience eat walk red vinyl club school hip hop era continue immerse create sit main star food flip menu family album continue di...	**0.603**
3	group choose place head town place simple menu cheap drive place middle plaza ton interested pepper lunch bakery order chicken dish rest group order majority table order beef noodle soup people order shrimp pork noodle soup pork shrimp dumpling noodle soup complaint noodle soup dumpling noodle s...	**0.594**
4	order service good red sauce nasty tho good opinion chicken good rice hard tasty happy hour bunch lime well strawberry strawberry sweet taste mexican american	**0.594**
5	small basically place small brim water dump half year wear water finally server water literally small bacon meat american	**0.584**
6	best amaze not disappointed casual bring family fun table enjoy customer service wonderful american	**0.572**

7	place golden nugget decide snack wait rest seat wait time min worker pass table waiter table shortly mins honestly dont table order food food good price guess casino waiter nice disappointed wait price soup japanese	**0.569**
8	love place great food entertainment fun place eat price jet average spot stand pork belly rock shrimp salt pepper shrimp asian japanese	**0.536**
9	search good answer eat presentation meat high restaurant enjoy tasty food feel restaurant needless price reasonable great birthday celebration asian korean	**0.526**
10	service favor late night salmon craving disappoint asian japanese	**0.521**
11	noodle hidden gem type review dine week amaze lunch special menu pm time catch lunch special literally rushing price amaze not moment check friendly attentive server completely adore family order salad appetizer literally time dine tomato basil dress delicious thing cold cold hot shrimp order sc...	**0.52**
12	tofu fry ice fish favorite japanese	**0.509**
13	awesome spot choose soup hot pot base add meat add belt concoct spice spice bar dig great concept delicious food confuse server helpful chinese japanese	**0.508**
14	rounded service good husband friend decide eat dinner seat nice dinner time even wait food shellfish platter good pretty big portion poke crisp good flavor dip sucker good not disappoint season not mac cheese taste good portion closer entree standard special lobster cream good outstanding la mar...	**0.507**
15	gentleman work window complete gentleman awesome customer service proper learn lot man sir exceptional service mexican	**0.503**
16	place awesome overly sweet super creamy super nice fast totally recommend pass area taiwanese	**0.501**
17	start review place amazing staff greet friendly food absolutely amazing extraordinary perfectly sear perfect blanket goodness generous cheap japanese asian	**0.497**
18	come best casual spot bring family dinner location reservation pm hostess accommodate work set table amazing restaurant floral original order menu family order dad live critical food kung traditional warmer dad personal favorite shrimp soup combine perfect appetizer salmon curry favorite food go...	**0.492**
19	surprisingly delicious pho eat pho life place quality special appetizer price order bowl load meat american vietnamese asian	**0.49**

The next step is for the engine to map the reviews to our initial dataset so as to extract the information about the restaurant, as well as the cuisine, the most similar reviews are about. In this step, the engine excludes from the recommendations reviews that map to the same user id as the one we have as input. Below, we can have a view on the list with the top recommended restaurants. We can see that the first restaurant is of an American cuisine, as the one of the user that we have used as an input. Also, given that the review we had as input had a 5-star rating, we can see that the recommendation also are not that far off from a 5-star.

Number	Name	Cuisine	Stars	User
0	**Guy Fieri's Vegas Kitchen**	american african	**3.5**	gK5Pb6hCPY-We5M7vcACCg
1	**Best Friend by Roy Choi**	korean mexican	**4.04**	YHXyjq3oklmFKKNPo2OAHg
2	**88 Noodle Papa**	chinese	**4.17**	Dl-30ayJYBpOXss4GCAP4g
3	**Nacho Daddy**	mexican american	**3.82**	FqdPOEEIkqKvmXKVZBRafQ
4	**PGA Tour Grill**	american	**3.81**	z8jdYKvF_vonC1djr9q-mg
5	**Shake Shack**	american	**3.73**	5Ggp9BBHigil9cjereEqEQ
6	**RED Asian Cuisine**	japanese	**4.08**	pi1gxpiKXVQo1HujbG94lw
7	**Sake Rok**	asian japanese	**4.04**	msIJhX2FRRgzqjQHUJaoGA
8	**Captain6 Korean BBQ**	asian korean	**5.0**	yxGKuflHkQl_Ob0x6btksA
9	**Sushi Way**	asian japanese	**3.95**	xo80PY8gX4GuhJb6_6yWVA
10	**Oodle Noodle**	japanese	**4.48**	JXaYWO4TGSWDb_8qas0e1Q
11	**Raku**	japanese	**5.0**	RYctIwOIktSdTNJpj00lkw
12	**Chubby Cattle**	chinese japanese	**4.02**	VgG_4NU41eZbpidLyfk3vw
13	**Searsucker**	american	**4.12**	Udvs6segqQEkmlC624ZUrQ
14	**Jack in the Box**	mexican	**2.6**	5nWkyQvcx0QF7iS3g_B8qA
15	**Icy Juicy**	taiwanese	**5.0**	RP65YGxnzgR_BI8lErW5Sg
16	**Fukumimi Ramen**	japanese asian	**4.25**	_Mov2mOtvd9-I3a_iHovVg
17	**Weera Thai Kitchen**	thai asian	**4.48**	IZmBU9pXJuiWCxCIH636ig
18	**Tasty Broth Pho Grill**	american vietnamese asian	**4.77**	iS4m_LE7f2oEzYl09HiIuw

Finally, the user is asked about the cuisine she would like to try and the engine filters the list of the best recommended restaurants in order to return the one that matches her preferences. Below, we can see the view of the recommendation engine from the side of the user.



```
Give me your id: f38DGhAQOxQu07P-4JMpTw

Hello Shanna.

What cuisine would you like to try today? Asian
You should try Sake Rok at 3786 Las Vegas Blvd S.

Have a nice meal!
```

7. Members/Roles

Vasileios Kotsomytis, *BI Analyst*

Vasileios is a graduate of the Department of Statistics, Athens University of Economics and Business. As a member of the BI Team in Yelp he has been involved in numerous projects regarding the processing and analysis of Data. During the project of the Restaurant Recommendation Engine he was involved in the Data Preparation, Preprocessing as well as the Exploratory Data Analysis. What is more, he created a neural network model while also being involved in the evaluation of the models.

Email: billkotsomitis@gmail.com
Linkedin Account

Efstratios Romanidis, *BI Analyst*

Stratos is a graduate of the Department of Informatics, Athens University of Economics and Business. As a member of the BI Team in Yelp he has been involved in numerous projects regarding the processing and analysis of Data. During the project of the Restaurant Recommendation Engine he was involved in the Data Preparation, Preprocessing as well as the Data Input Format for the Models. What is more, he created a neural network model while also being involved in the evaluation of the models.

Email: efstratiosromanidis@gmail.com

Athina Spanou, *BI Analyst*

Athina is a graduate of the Department of Management Science and Technology, Athens University of Economics and Business. As a member of the BI Team in Yelp she has been involved in projects regarding the analysis and modeling of Data. During the project of the Restaurant Recommendation Engine she was involved in the Data Preparation, Preprocessing as well as the Exploratory Data Analysis. What is more, she created two neural network models while also being involved in the creation of the Recommendation Engine.

Email: athispanou@gmail.com
Linkedin Account

Contact Person: *Vasileios Kotsomytis*

8. Time-Plan

	Week1	Week2	Week3	Week4	Week5	Week6
Research - Bibliography	■					
Business Analysis	■					
Data Cleaning-Preprocessing	■	■				
EDA-Visualization		■				■
Models-Evaluation			■	■	■	
Recommendation System				■	■	
Report						■

9. Bibliography

1. Yelp, Yelp Dataset Challenge, http://www.yelp.com/dataset_challenge
2. A Preference-Based Restaurant Recommendation System for Individuals and Groups https://www.cs.cornell.edu/~rahmtin/Files/YelpClassProject.pdf
3. A. Ihler et al., Recommender Systems Designed for Yelp.com http://www.math.uci.edu/icamp/summer/research/student_research/ recommender_systems_slides.pdf
4. Deep Learning with Python. F. Chollet. Manning, (November 2017)
5. Deep Learning Cookbook. by Douwe Osinga. (June 2018)
6. Geron, A., 2019. Hands-On Machine Learning With Scikit-Learn, Keras, And Tensorflow. O'Reilly Media, Incorporated.