DS5500 Capstone: Final Report
Business Review & Search Copilot

Spring 2024

4/1/2024

Name and IDs:
Feiran Zhang
002742415

Table of Contents

Abstract

The

1. Introduction

For my capstone project, I implemented a business review & search copilot which mainly aims to give recommendations for the users when/after they write the reviews and feedback for the restaurants. Different from the traditional business recommendation systems which only recommend based on the categories, my search copilot did a content-based classification on the user's reviews first. It improved the correlation between the recommended businesses and reviews by users. Also, my copilot expected to display my recommendations on the map, which helps users to make better choices. For the review classification part, we trained the LSTM model by the original dataset from Yelp. For the recommendation algorithm, I calculated the similarity scores based on the business categories given by the Yelp dataset and the classification on the reviews. My recommendation listed the top 10 highest similarity scores for the business which were reviewed by the users.

For my restaurant search copilot system, I used a dataset that is a subset of Yelp's businesses, reviews. It was originally put together for the Yelp Dataset which is a chance for students to conduct research or analysis on Yelp's data and share their discoveries. In the dataset, you'll find information about businesses across 11 metropolitan areas in four countries. According to the Yelp dataset's term of use, I will cite the open source at the end of the report. (*Yelp Dataset*, n.d.)

2. Methods

I. Data Preprocessing and EDA

My project is mainly based on the two datasets from the Yelp dataset. I mainly collected the restaurant names, their geographical information (latitudes and longitudes), and their business categories as the key factors for the recommendations from the yelp_business dataset. I also collected the reviews for the paired restaurants from the yelp_reviews dataset. Figure 1 is the overview of our review and business dataset.

business_id	stars	date	text
AEx2SYEUJmTxVVB18LICwA	5	2016-05-28	Super simple place but amazing nonetheless. It...
VR6GpWIda3SfvPC-lg9H3w	5	2016-05-28	Small unassuming place that changes their menu...
CKC0-MOWMqoeWf6s-szl8g	5	2016-05-28	Lester's is located in a beautiful neighborhoo...
ACFtxLv8pGrrxMm6EgjreA	4	2016-05-28	Love coming here. Yes the place always needs t...
s2I_Ni76bjJNK9yG60iD-Q	4	2016-05-28	Had their chocolate almond croissant and it wa...

Figure 1.1. Head rows of the review datasets for training

name	stars	longitude	postal_code	business_id	latitude	review_count	categories
Arizona Biltmore Golf Clu	3.0	-112.018481	85016	b'1SWheh84yJXfytovILXOAQ'	33.522143	5	Golf| Active Life
Emerald Chinese Restaurant	2.5	-79.652289	L5R 3E7	b'QXAEGFB4oINsVuTFxEYKFQ'	43.605499	128	Specialty Food| Restaurants| Dim Sum| Imported...
Musashi Japanese Restaurant	4.0	-80.859132	28210	b'gnKjwL_1w79qoiV3IC_xQQ'	35.092564	170	Sushi Bars| Restaurants| Japanese
Farmers Insurance - Paul Lorenz	5.0	-112.395596	85338	b'xvX2CttrVhyG2z1dFg_0xw'	33.455613	3	Insurance| Financial Services
Queen City Plumbing	4.0	-80.887223	28217	b'HhyxOkGAM07SRYtlQ4wMFQ'	35.190012	4	Plumbing| Shopping| Local Services| Home Servi...

Figure 1.2. Head rows of the business datasets for the recommendations and locating

For the review datasets, the total indexes are 5261668. Due to the server's memory and time cost, I randomly collected 50,000 rows of reviews for the model training. I split the dataset into 50000 for training and 10000 for validation.

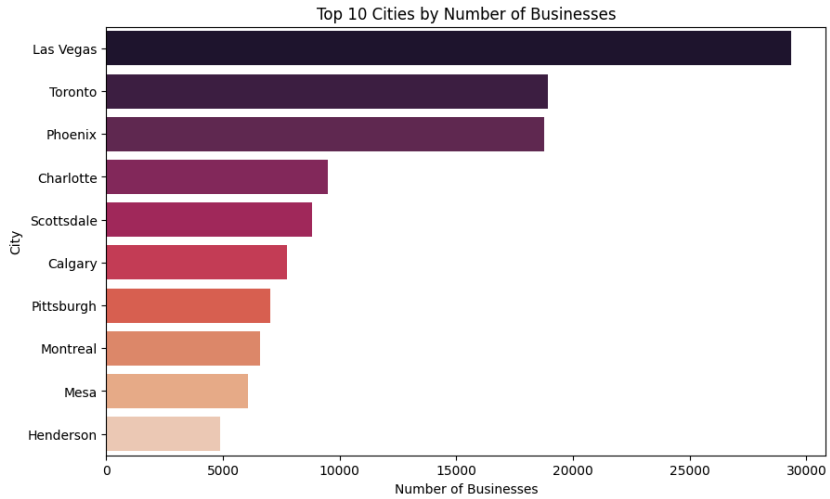


Figure 2. The business distribution in Yelp_business dataset

Based on the EDA on the business dataset on Figure 2, I found that Las Vegas contains most business in the dataset. I decided to make a result in Las Vegas to evaluate the performance of my recommendation. I believe that a larger sample is beneficial for the result evaluation.

token	world	W_a	W_about	W_after	W_all	W_also	...	...	W_zest	W_zucchini	W_zulu	W_zumba	W_zurich
0	a	1	0	0	0	0	...	...	0	0	0	0	0
1	about	0	1	0	0	0	...	...	0	0	0	0	0
2	after	0	0	1	0	0	...	...	0	0	0	0	0
3	all	0	0	0	1	0	...	...	0	0	0	0	0
4	also	0	0	0	0	1	...	...	0	0	0	0	0
...	...	...	...	...	...	...	...	...	...	...	...	...	...
...	...	...	...	...	...	...	...	...	...	...	...	...	...
49996	zest	0	0	0	0	0	...	...	1	0	0	0	0
49997	zucchini	0	0	0	0	0	...	...	0	1	0	0	0
49998	zulu	0	0	0	0	0	...	...	0	0	1	0	0
49999	zumba	0	0	0	0	0	...	...	0	0	0	1	0
50000	zurich	0	0	0	0	0	...	...	0	0	0	0	1

Image by the author: one-hot encoding vector example

Figure 3. The one-hot encoding vector example from Demirci's article

For the data preprocessing, I did the word tokenization based on the one-hot encoding technique. For the one-hot encoding and embedding for the text, it can tokenize each token as a binary vector. That means only one single element in the vector is 1 ("hot") and the rest is 0 ("cold"). (Demirci, n.d.) I believe that one-hot encoding is a word tokenization technique that is suitable for the review sentiment classification. It makes it much easier for us to accurately detect the 'important' words from the reviews and make a classification.

II. LSTM model (Long Short-Term Memory model)

The purpose of using an LSTM (Long Short-Term Memory) model to classify the reviews is that LSTM models can make them particularly suitable for handling the challenges associated with text classification tasks, especially when dealing with sequential data. According to Khan's paper, the aspect category detection (ACD) and long short-term memory (LSTM) in the recommendation system can presents the state-of-the-art studies related to aspect based category detection in which we proposed the methodology to predict if the sequence of the next word based on semantic aspect and contextual information using attention based modified LSTM model. (Khan et al., 2023, #)

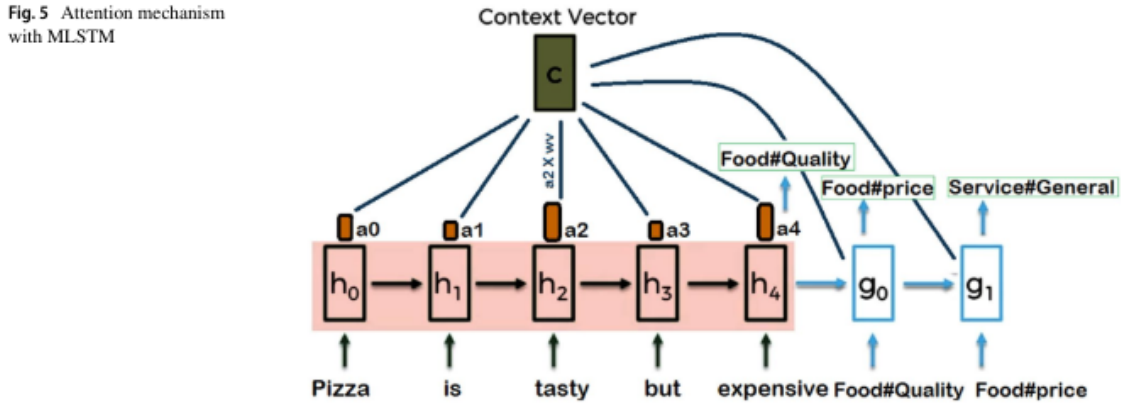
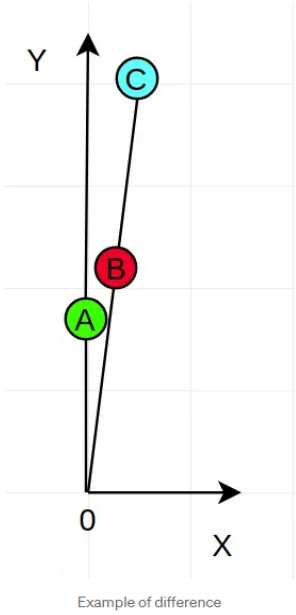


Figure 4. The image shows the attention mechanism of LSTM to predict the sentiments based on the word sequence.

After I prepare the input data with word tokenization, the construction of the LSTM model incorporates an embedding layer, which assigns a high-dimensional vector to each word in our vocabulary, capturing the semantic and syntactic nuances of each word. To improve the model's understanding of context from both preceding and following sequences, a bidirectional LSTM layer is utilized. This dual-direction processing is vital for generating more accurate outcomes in the analysis of classification, as it ensures a comprehensive understanding of context. Additionally, the bidirectional approach is more effective in mitigating the issue of gradient disappearance compared to its unidirectional counterpart. To introduce non-linearity and enable the model to capture more intricate patterns in the relationship between input reviews and their corresponding rating categories, I integrated a dense layer equipped with 64 units and activated by the ReLU function, aiming to boost the model's accuracy.

III. Similarity metrics (KNN)

After the review classification, we need to make a recommendation in the given rating group we predicted from the user's review. In order to combine all the influential words from the business categories, I decided to use the similarity metric to calculate cosine similarity between different businesses or restaurants from the given rating groups.



$$\text{similarity} = \cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\|\|\mathbf{B}\|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2}\sqrt{\sum_{i=1}^{n} B_i^2}},$$

Figure 5. The difference example and formula of cosine similarity from Kirill's article for recommendation system

The basic logic for the cosine similarity metrics for recommendation is to find the closest vector from all different vectors generated by the different categories and features of businesses. According to Kirill's article, in Figure5, that AB < AC, and if we take huge threshold in distances we may loose data of C user. If we use cosine similarity, users B and C has the same similarity score with A, because they are in the one direction. In this case we can get a wider data explanation for future analysis and recommendations. (Bondarenko, 2019)

In my project, cosine similarity can be used to compare the similarity between vectors representing different businesses. These vectors could represent various features of the restaurants, such as types of cuisine, pricing levels, location preferences, user ratings, or textual reviews. In order to calculate the cosine similarity metrics, I did the TF-IDF vectorization for the business traits and categories.

3. Experimental Results (Evaluations)

I. Training epochs loss and accuracy curves

Based on the training epochs loss and accuracy curves, I believe that the training history for our LSTM model has an accuracy around 60%, but it is only the rough accuracy and loss history generated based on the training history. I can find that the model is lightly over-fitting on the training dataset.

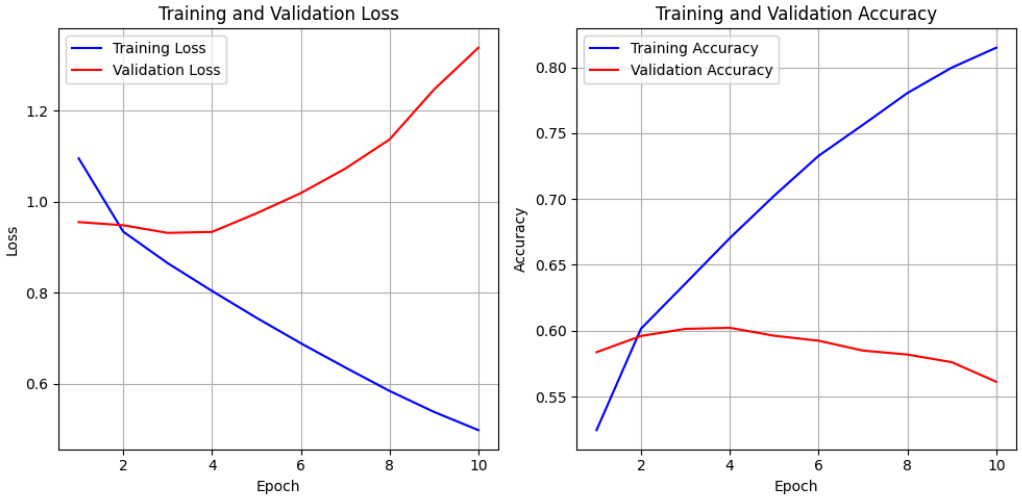


Figure 6. Training epochs loss and accuracy curves

II. K-fold Validation on the LSTM model

To get the actual prediction accuracy from my model, I implemented the K-fold validation to generate the average prediction accuracy for my model. For the K-fold validation evaluation: The average prediction accuracy for my LSTM model is around 78.34% on the validation dataset. Here I set the fold k = 5.

```
1563/1563 [==============================] - 49s 31ms/step - loss: 0.6015 - accuracy: 0.7849
accuracy: 78.49%
1563/1563 [==============================] - 49s 31ms/step - loss: 0.6054 - accuracy: 0.7842
accuracy: 78.42%
1563/1563 [==============================] - 52s 34ms/step - loss: 0.6051 - accuracy: 0.7824
accuracy: 78.24%
1563/1563 [==============================] - 48s 31ms/step - loss: 0.6076 - accuracy: 0.7827
accuracy: 78.27%
1563/1563 [==============================] - 48s 31ms/step - loss: 0.6060 - accuracy: 0.7830
accuracy: 78.30%
78.34% (+/- 0.09%)
```

Figure 7. K-fold validation evaluation epochs on the LSTM model

III. Result demonstration

For the recommendation result demonstration, I will explain the logic and the usage of my system. After the users finish feedback or reviews of a business of restaurants, my system will give top recommendations based on the reviews and business they are interested in. After they finish the reviews, we will give a rating classification for the reviews first. Then provides the better options or similar options in that ratings group.

```
1/1 [==============================] - 0s 35ms/step
The pretzel is good.A nice French restaurant in the Park Slope. -> 4.00
```

Figure 8. The Rating Classification for the reviews

Based on the Cosine similarity scores, I will give a recommendation for the users. But this interface will not be shown to the user in the future full-stacked system.

```
Recommendations for La Creperie from high to low:
                         name     score   longitude   latitude
49235           Pretzel Maker  1.000000 -115.197260  36.172534
54436         Wetzels Pretzels  1.000000 -115.185025  36.116382
108204  Auntie Annes Pretzels  1.000000 -115.136987  36.086168
171385             Pretzeland  1.000000 -115.174252  36.091267
180714        Wetzels Pretzels  1.000000 -115.203082  36.277652
16912   Auntie Annes Pretzels  0.875283 -115.172247  36.066778
180575        Wetzels Pretzels  0.875283 -115.177063  36.068267
93421         New York Pretzel  0.846407 -115.167443  36.102616
118940    German Bread Bakery  0.846407 -115.119204  36.021162
134872        New York Pretzel  0.846407 -115.169758  36.121558
```

Figure 9. The recommendations for a specific restaurants based on the ratings, locations, and business categories

The final demonstration is that the businesses / restaurants I recommended will display on the map as Figure 10. The users can learn the specific geographical information of restaurants which may help them to make a decision.

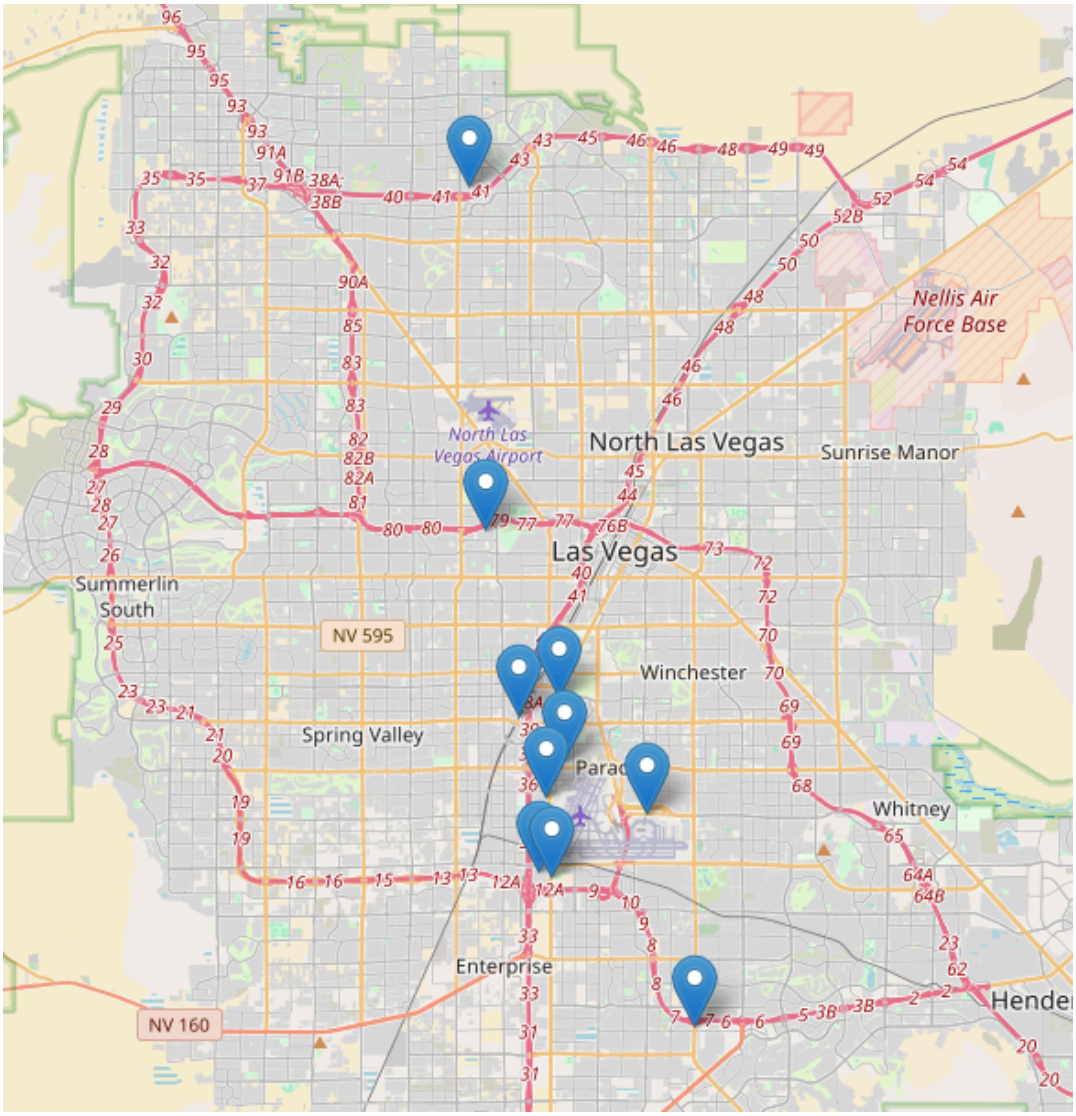


Figure 10. The recommendation map

4. Discussion (Analysis)

5. Conclusions

References

Bondarenko, K. (2019, February 27). *Similarity metrics in recommender systems | by Kirill Bondarenko | Medium*. Kirill Bondarenko. Retrieved April 2, 2024, from https://bond-kirill-alexandrovich.medium.com/similarity-metrics-in-recommender-systems-aed9d3b2315f

Demirci, F. (n.d.). *Inside GPT — I : Understanding the text generation*. towardsdatascience.com. https://towardsdatascience.com/inside-gpt-i-1e8840ca8093#:~:text=One%2Dhot%20encoding%20is%20the,0%20(%E2%80%9Ccold%E2%80%9D).&text=The%20tokens%20are%20represented%20with,total%20token%20in%20our%20corpus.

Khan, M. U., Javed, A. R., Ihsan, M., & Tariq, U. (2023). A novel category detection of social media reviews in the restaurant industry. 1825–1838. https://doi.org/10.1007/s00530-020-00704-2

Yelp dataset. (n.d.). Yelp Dataset. https://www.yelp.com/dataset/documentation/main