National University of Singapore

BT4222 Mining Web Data For Business Insights

Project : Aspect Based sentiment analysis on google restaurant reviews

Name	Matriculation Number
Lin Yaohua	A0206212Y
Teo Ri Han Henry	A0199956Y
William Chng	
Lim Chang Yu	A0201946H

Content Page

1. Problem Description

Online reviews play a huge role in our day-to-day decisions on where to eat. In fact, a 2018 survey conducted by Tripadvisor revealed that 94% of users in the US look at reviews before deciding on a restaurant to visit (Guta, 2018). Reviews allow consumers to make an informed decision before choosing to visit a restaurant. Google Maps, a common platform to search for reviews, provides overall ratings for restaurants in the form of a score out of five stars. However, as consumers have different preferences for restaurants, choosing a restaurant based on an overall review may not be the most effective as restaurants may perform better in one area (eg. food quality) better than another (eg. service). Furthermore, sifting through the plethora of online reviews on Google Maps for each restaurant makes the decision-making process a much more laborious one. Presently, consumers choose to perform the latter. However, doing this regularly can take up too much time. One third of users in the US aged 18 to 34 spend at least 25 minutes reading reviews before making a decision and 12% spend at least 45 (Kats, 2020). As such, it is evident that the current process of utilising reviews on Google Maps to choose a restaurant is not the most optimal one.

1.1 Aspect-based Sentiment Analysis

However, ordinary reviews contain various opinions on different aspects of the dining experience, which may not be useful when one is more concerned about particular aspects.

To combat this problem, we proposed utilising aspect-based sentiment analysis which allows for a more granular analysis of review sentiments. This consists of identifying common aspects that consumers look out for when choosing a restaurant to dine at, then generating an overall sentiment for each of these aspects. Doing so allows consumers to skip the process of sifting through reviews as they are able to find out how a restaurant is performing in whichever aspect is important to them, hence making the decision-making process much more efficient.

The goal of the project is to experiment with different ways of conducting aspect based sentiment analysis, specifically ways to extract and map aspect relevant parts of text to be used for sentiment classification as well as coming up with a process to assign aspect sentiment scores for restaurants to be used.

2. Facts about the dataset

2.1 Aspects Chosen & dataset

Before obtaining our dataset, we first decided which aspects to consider in our project. A study conducted by Eat, a leading restaurant reservation management company, found that the top three things that consumers expect are **Service, Food Quality** and **Ambience** (Andrews, 2022), implying that these aspects are the most highly considered by consumers when choosing a restaurant to dine at. Thus, we included them in our dataset as shown in Table 1.

Table 1. Dataset Information

Column	Description	Dtype
caption	Reviews on the restaurant	str
rating	Overall score a reviewer gives to the restaurant (from 1 to 5 stars)	Int [1,5]

We obtained the data through web scraping reviews on Google Maps using Selenium (Gasparini, 2019). The dataset contains 3897 reviews of 7 different restaurants.

Table 2. True sentiment distribution of classes post labelling

Aspect	Positive	neutral	negative
Food	2564	981	347
Service	778	2724	390
Ambience	1199	2561	132

The proportion of sentiments for each aspect shown in Table 2.

2.2 Methodology of labelling sentiment scores for supervised learning

Sentiment scores of 1,0,-1 representing "positive", "neutral" and "negative" sentiment for each of the 3 identified aspects we hand labelled by our team. To reduce personal bias, each datapoint is labelled by every member of the team and the final score is determined by majority vote. In the event of tie scores, the datapoint is then reviewed together by the team before a sentiment score is allocated. A neutral score is allocated (for that particular aspect) when the review does not contain that particular aspect.

We also made the assumption that a non-mention of that particular aspect in a review represents a neutral sentiment for that particular aspect, i.e there is nothing bad or good about that aspect that warrants a comment by the customer and therefore should be assigned a neutral label.

3. Aspect term identification & extraction

A single review can discuss up to all three aspects. To find the sentiment of each aspect in a review, we extracted the portion of each review pertaining to each aspect.

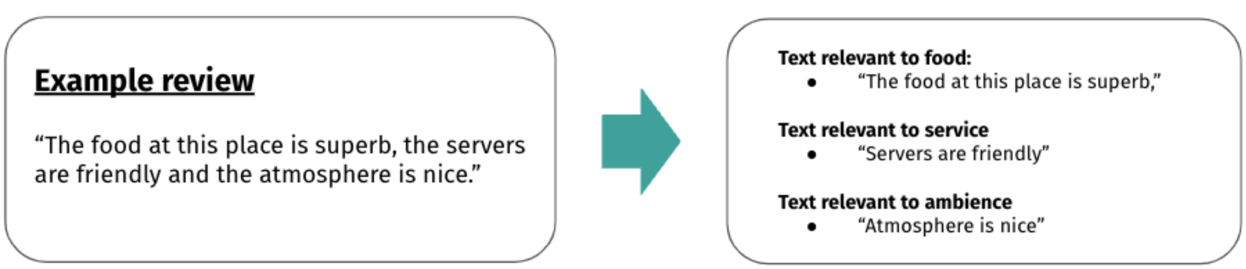


Figure 1. Example of Aspect Extraction

Figure 1 shows an example of aspect extraction on a review, where chunks of text pertaining to each aspect are extracted and returned separately.

After aspect extraction, we performed encoding separately using Bag-of-Words (BoW) or Term Frequency-Inverse Document Frequency (TF-IDF) and train their own separate classifiers.

3.1 Aspect identification

To help us identify the different aspects in each review, we performed lexicon matching utilising a corpus from Yelp which contained lexicons for different restaurant aspects. Of which we have chosen to focus on 3 aspects namely “Food” , “Service” and “Ambience”.

3.2 Aspect Extraction Methods

3.2.1 Aspect Extraction Method 1

We want to have a benchmark method for aspect extraction. To do this we made use of open ai’s gpt davinci variant model, which is able to extract chunks of texts from the original review with surprising accuracy, as long as the prompt is well crafted. This is the prompt we decided after trial and testing.

f”extract all parts of text associated with food quality or service quality or ambience quality (can be NA). reply in the form [food: all parts of text, svc: all parts of text, amb: all parts of text] \n\n {text}”

The above prompt will return a response of the form “[food : xxx, svc: xxx, amb: xxx]”, we then use regex to separate the food, service and ambience part and store them in a tuple.

```
Example review:
 After their renovation, the food quality has dropped significantly
 The calamari was cold and greasy, without salt
 The music was boring, being played on a poor quality music system in a corner
------------------
('After their renovation, the food quality has dropped significantly. The calamari was cold and greasy, without salt. ',
 'NA ',
 'The music was boring, being played on a poor quality music system in a corner.')
```

Fig 2. Aspect Extraction Example for Method 1

Figure 2 shows an example of extracted aspects using this method. The first component of the tuple corresponds to extracted text relevant to “Food”, the second component contains text relevant to “Service”. And the last component correspond to text relevant to “Ambience”

Post extraction, the extracted texts are then individually further processed by removing stopwords, converting emojis into their text representation, removing punctuations and lemmatization. The following table summarises the number of chunks extracted for each aspect.

Table 3. Aspect Extraction Result for Method 1

Aspect	Number of chunks	% of total reviews
Food Quality	3385	87
Service	2283	59
Ambience	1870	48

3.2.2 Extraction Method 2

For extraction methods **2 and 3**, we identified **additional lexicons** for each aspect, summarised in the table below.

Table 4. Additional Lexicons Identified

Aspect	Food	Service	Ambience
Lexicons	Food, Dish	Staff, Waiter, Waitress, Manager, Person, Lady, Man, Cashier, Bartender, Chef, Owner, Barista	Environment, Lighting, Noise, Decor, View, Music, Place

The difficulty with aspect extraction is the fact that the same meanings can be expressed with different words and sentence structure. Given the complexity of the English language, it is difficult to abstract the meaning behind the text.

In this method, we aim to split the reviews into chunks containing target aspects (e.g. food) and their sentiment descriptions (e.g. yummy). We will be doing this via spacy's POS (Part-Of-Speech) tags.

We attempt to first identify patterns in the reviews. We were able to identify common patterns such as Adjective-Noun combinations, Noun-Adjective combinations. In addition, adverbs which can shed insights about the intensity of the sentiment descriptive words should be included. Optional patterns such as Adverb(optional)-Adjective-Noun combinations are included as well.

```
import spacy
from spacy.matcher import Matcher
def match_pattern(text):
    matcher = Matcher(nlp.vocab)
    patterns = [
            [{"POS": "ADV", "OP": "{,1}"},{"POS": "ADJ"},{"POS" : "NOUN"}], #for descriptive word -> noun (w adv)
            [{"POS" : "NOUN"}, {"POS": "ADV", "OP" : "{,1}"},{"POS": "ADJ"}],
            [{"POS": "ADV", "OP": "{,1}"},{"POS": "ADJ"},{"POS" : "PROPN"}],
            [{"POS" : "PROPN"}, {"POS": "ADV", "OP" : "{,1}"},{"POS": "ADJ"}],
            [{"POS" : "ADV", "OP" : "{,1}"},{"POS": "ADJ"}],
            [{"POS" : "NOUN"}, {"POS" : "VERB"}, {"POS": "ADV", "OP" : "{,1}"},{"POS": "ADJ"}],
            [{"POS" : "NOUN"}, {"POS": "AUX", "OP" : "{,1}"},{"POS" : "VERB", "OP" :"{,1}"},{"POS": "ADJ"}]
```

Fig 3. POS Patterns Identified

Fig 3 shows the different POS patterns identified. Additionally, refinements are done in the preprocessing to improve performance. For instance, a customised stop word list that excludes adverbs, adjectives and conjuncts to prevent essential words being removed when preprocessing shown in appendix B.

These combinations of patterns are passed through the spacy Matcher function that extracts the parts of text that matches the pattern.

Subsequently, the chunks are compared against the lexicons associated with the different aspects and categorised accordingly. For instance, food chunks containing words in food_lexicons such as "food" will be grouped under food_aspect. The following table summarises the number of chunks extracted for each aspect.

Table 5. Aspect Extraction Result for Method 2

Aspect	Number of chunks	% of total reviews
Food Quality	2493	63
Service	857	18
Ambience	1015	21

The following figure shows an example of extracted aspects using this method.

```
Example review:
 Friendly staff, superb service, food was serve hot and fresh, not so pricey, pretty service staff
 Place is a bit outdated might needs some upgrading
------------------
food_aspect_extraction_2                                          [food serve hot]
service_aspect_extraction_2                          [friendly staff, pretty service]
ambience_aspect_extraction_2                                                   NaN
Name: 2, dtype: object
```

Fig 4. Aspect Extraction Example for Method 2

3.2.3 Extraction Method 3 - NLTK

For our last extraction method, we used NLTK to perform aspect extraction of the reviews. Before performing the extraction, we tokenized and lemmatized the lowercase of each review. For each review in a particular aspect, we identified a word from the aspect lexicon list that is present in the review, called the aspect keyword. To allow for a sentiment to be obtained, NLTK was used to return a chunk of words in the review containing the aspect keyword and an adjective or adverb. Adjectives and adverbs can appear soon before, or after a noun (Wikibooks, 2009).

Hence, we searched up to 5 words before and 10 words after the aspect keyword and returned the chunk of words containing the keyword and its first associated adjective or adverb. Additionally, if an adverb identified comes after the keyword, the chunk returned will contain words from the keyword to the adverb and one additional word after it, to account for a potential adjective that may come after the adverb.

The reason for searching up to 5 words before the noun is due to the fact that adjectives and adverbs , if before a noun, typically come soon before the noun (eg. **very good** food). Hence, searching too far back may include words from previous sentences, which can lead to inaccurate sentiments being derived. As for searching after the keyword, we utilised a range of 10 words as the adjective that comes after a noun does not need to come soon after the noun (eg. the chicken that we had yesterday evening was absolutely **delightful**). In the above example, the adjective ‘delightful’ came 8 words after the noun ‘chicken’. The following figure shows an example of extracting aspects using Method 3.

```
Example review:
 Took my son there for dinner and we had a good time
 We both ate a Philly cheese steak and it was really good
 Service was good, girls are very pleasant and pay attention to the service, prices are very reasonable
------------------
food_aspect_extraction_3               good time ate philly cheese steak
service_aspect_extraction_3     good girl pleasant pay attention service
ambience_aspect_extraction_3
```

Fig 5. Aspect Extraction Example for Method 3

This entire process was repeated two more times for each review to extract chunks for all three aspects. The following table summarises the number of chunks extracted for each aspect.

Table 6. Aspect Extraction Result for Method 3

Aspect	Number of chunks	% of total reviews
Food Quality	2493	64
Service	857	22
Ambience	1015	26

4. Classification models and results

4.1 Classification Model Pipeline

Post extracting and preprocessing the aspects using the above 3 methods, we then encode the text in each aspect using the Bag of Words unigram model as well as the Tf-idf model. We also decided to include the overall ratings as a feature.

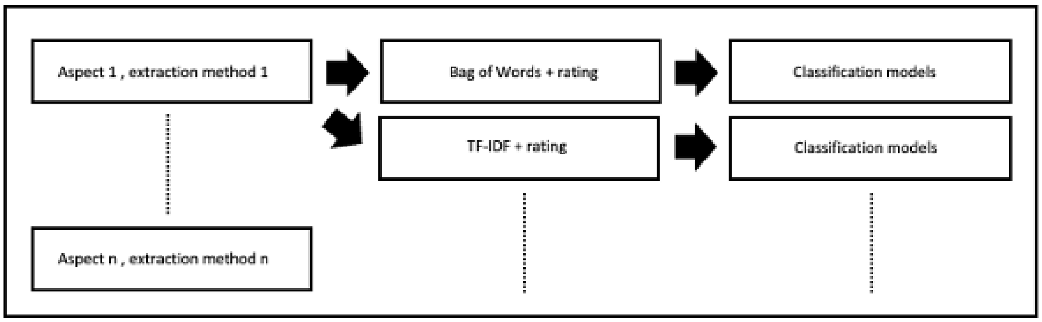


Fig 6. Modelling Pipeline

Fig 6 shows our modelling pipeline for each aspect.To cover a range of different model types, we have decided to try out 3 classification models, namely logistic regression, random forest classifier as well as XGB classifier. Below are the hyperparameter search space we have used for each of the models, tuned (Appendix A) using the hyperopt library by minimising negative weighted f1 score on the validation set.

In addition to training our own machine learning models, we performed estimations of the sentiment scores using two common models. We were able to use the performances of these models as a benchmark for our own machine learning models.

1. VADER: a rule based lexicon model that analyses written text to determine its sentiment, providing a score for positivity, negativity, and neutrality, while also taking into account the intensity of the sentiment. VADER uses a dictionary of words and phrases with associated sentiment scores and employs rules to handle sentiment modifiers (Mahreen, 2022).

2. BERT: We used a fine tuned model for BERT for reviews sentiment analysis. This model is intended for direct use as a review sentiment scorer for text in 6 languages (Hugging Face, n.d.).

4.2 Results

We first generated results at the macro level for each aspect extraction method and obtained the best-performing models for each aspect.

Table 7. Performance of best models from Extraction Method 1(benchmark)

Aspect	Model	Test Accuracy	Test Weighted F1
Food	BOW XGBoost	0.853	0.848
Service	TFIDF XGBoost	0.823	0.824
Ambience	TFIDF Random Forest	0.835	0.817

Table 8. Performance of best models from Extraction Method 2

Aspect	Model	Test Accuracy	Test Weighted F1
Food	BOW Random Forest	0.747	0.711
Service	TFIDF Logistic Regression	0.826	0.808
Ambience	TFIDF XGBoost	0.777	0.745

Table 9. Performance of best models from Extraction Method 3

Aspect	Model	Test Accuracy	Test Weighted F1
Food	BOW XGBoost	0.725	0.698
Service	BOW Random Forest	0.819	0.800
Ambience	TFIDF Random Forest	0.737	0.680

Overall, our extraction methods 2 and 3 performed relatively well, with test weighted f1 scores and accuracies ranging from 0.68 - 0.81 and 0.72 - 0.82 respectively. As observed in the results, extraction method 2 performed better.

When compared to BERT and VADER models, both extraction methods outperformed the pre-train models significantly (Appendix B).

When compared to our benchmark model, extraction methods 2 and 3 give near par performance with the benchmark in the service aspect of reviews, all models scoring approximately 0.8 and 0.82 for the weighted average f1 score and accuracy on the test set. However, the models performance in both the food and ambience aspects are slightly weaker compared to the benchmark.

One reason for the difference in performance could be that the lexicons used for food and ambience are not contextualised to the words consumers use when discussing these aspects, as compared to service. For service, the terms used may tend to be more standardised (eg. service, waiter, waitress etc), while the terms used for food and ambience may have more variation across different cultures and can tend to be more specific as well. Chunks of text related to food and text may hence not have been identified, resulting in the assumption that the sentiment score is 0.

Additionally, in contrast to the service aspect, words related to food and ambience tend to be more generic in nature and may overlap with other topics unrelated to this study, resulting in inaccurate chunks being identified. This can decrease the prediction performance of sentiment scores.

5. Aggregating scores and Business Use Cases

5.1 Aggregating Scores

In this section, we have decided to use extraction method's 2 models, namely the BOW random forest model, the TF-IDF Logistic regression model and the TF-IDF gradient boosting classifier for estimating the sentiment score for food, service and ambience aspects respectively for each individual review, before aggregating the reviews to provide a single score across all 3 aspects for a single restaurant.

The reason for the above choice in models is because the models using extraction method 2 performs best overall in terms of test accuracy and weighted average f-1 outside of the benchmark model, and the benchmark models require extracting aspects using external tools (from open ai) that carries a usage cost.

Estimated sentiment score for the k^{th} review on a single aspect = $S_k \in \{-1, 0, 1\}$

$$Score_{i,j} = \frac{\sum_{k=1}^{n} S_{i,j,k}}{n} \quad i = aspect\ , j = restraunt$$

Fig 7. Formula for calculation of sentiment score

To aggregate sentiment scores across restaurants, we use the formula in Fig 7, which outputs a sentiment score per aspect for a restaurant in the range of (-1,1). An example of the aggregated sentiment scores can be seen in Appendix C. Integrating these aggregated scores into Google Maps can be done through categorising the scores into colour-coded bins, giving users an easy visual representation of a restaurant's performance for each aspect.

5.2 Business Use Cases

Integrating our aggregated scores into Google Maps can benefit two stakeholders: Consumers and Restaurant owners. It is found that consumers shift towards non-branded search (Detwiler, 2021), which means that they do not have a specific restaurant in mind, and are more willing to explore new options. Hence, Google can aim to display the colour-coded aspect-based sentiment scores for each restaurant on Google Maps (Appendix E), allowing consumers to make quick and informed decisions tailored to the aspects they prioritise. This improves the user experience for Google Maps, and an accurate representation of reviews is likely to increase the volume of users on the app, which will generate more revenue for Google.

The impact of online reviews on a restaurant's business can be significant (Graves, 2019). For restaurant owners, the aggregated sentiment scores can allow owners to make effective business decisions tailored towards the feedback of customers without having to scour through individual reviews. Such metrics can be provided to all restaurants (Appendix F1), and Google can offer the option to subscribe to business plans to view more detailed analytics for more strategic decision making.

For example, trends in aggregated sentiment scores can be plotted over time. Suppose an owner of a restaurant is interested in finding out the public reception of a change in menu, change in decorations or change in staff training, they can insert checkpoints on the plot to mark such business decisions and observe the changes in related aspect scores over time. This will give them insight on levels of success of their business plans (Appendix F2)

Limitations & Improvements

The supervised models are trained on relatively small amounts of data, due to the lack of manpower required for labelling the data as such might not generalise well especially on reviews that contain a lot of words the models are not trained on. This is especially true for the food and ambience aspects. Future improvements can be made by expanding the train dataset to include more variability in the reviews which can help to improve the robustness of the model.

As discussed in the results, the performance of extraction methods 2 and 3 depend heavily on the accuracy and completeness of aspect lexicon dictionaries. Hence, we can improve model performance by searching for nouns specific to the food and ambience aspects. In a similar fashion, the VADER model can be improved with an updated valence aware dictionary.

6. Conclusion

In conclusion, we believe that aspect based sentiment analysis is much more useful than normal sentiment analysis in the context of analysing restaurant reviews as it is able to target specific needs by the user. Our methods for conducting sentiment analysis show promising results and are a good starting point.

We believe that with the implementation of improvements mentioned, better performance could be achieved, allowing for actual commercial implementations, generating value for businesses.

References

Andrews, R. (2022, November 20). *10 Restaurant Customer Service Ideas That Increase Loyal Guests in 2023*. Eat App. Retrieved March 28, 2023, from https://restaurant.eatapp.co/blog/restaurant-customer-service

Detwiler, M. W. (2021, August 23). *Google still top restaurant search tool, false reviews remain challenging*. Fast Casual. Retrieved April 6, 2023, from https://www.fastcasual.com/articles/google-still-top-restaurant-search-tool-false-reviews-remain-challenging/

Gasparini, M. (2019, June 11). *gaspa93/googlemaps-scraper: Google Maps reviews scraping*. GitHub. Retrieved April 6, 2023, from https://github.com/gaspa93/googlemaps-scraper

Graves, A. (2019, February 28). *How Restaurant Ratings and Reviews Affect Your Business*. Bloom Intelligence. Retrieved April 6, 2023, from https://bloomintelligence.com/blog/how-restaurant-ratings-reviews-affect-business/

Guta, M. (2018, June 13). *94% of Diners Will Choose Your Restaurant Based on Online Reviews*. Small Business Trends. Retrieved March 28, 2023, from https://smallbiztrends.com/2018/06/how-diners-choose-restaurants.html

Hugging Face. (n.d.). *nlptown/bert-base-multilingual-uncased-sentiment · Hugging Face*. Hugging Face. Retrieved April 6, 2023, from https://huggingface.co/nlptown/bert-base-multilingual-uncased-sentiment

Kats, R. (2020, January 8). *How Are Consumers Spending Some of Their Time? Reading Reviews. Lots of Reviews.* Insider Intelligence. Retrieved March 28, 2023, from https://www.insiderintelligence.com/content/how-are-consumers-spending-some-of-their-time-reading-reviews-lots-of-reviews

Lutkevich, B. (2020, January 27). *What is BERT (Language Model) and How Does It Work?* TechTarget. Retrieved April 2, 2023, from https://www.techtarget.com/searchenterpriseai/definition/BERT-language-model

Mahreen, J. (2022, October 2). *Sentiment Analysis Using VADER*. Analytics Vidhya. Retrieved April 3, 2023, from https://www.analyticsvidhya.com/blog/2022/10/sentiment-analysis-using-vader/

Thomas, A. (2018, February 26). *The Secret Ratio That Proves Why Customer Reviews Are So Important*. Inc. Magazine. Retrieved April 3, 2023, from https://www.inc.com/andrew-thomas/the-hidden-ratio-that-could-make-or-break-your-company.html

Wikibooks. (2009, May 2). *English Grammar/Basic Parts of Speech/Adjectives*. Wikibooks. Retrieved March 28, 2023, from https://en.wikibooks.org/wiki/English_Grammar/Basic_Parts_of_Speech/Adjectives

ChatGPT Prompts

python replace emojis with text

To replace emojis with text in Python, you can use the emoji library. Here's an example code snippet that shows how to do it:

Here, we define a function called replace_emojis_with_text that takes a text parameter as input. The function then uses the emojize function from the emoji library to convert any emojis in the text to their corresponding Unicode codes. It then checks if each Unicode code is present in the UNICODE_EMOJI list from the emoji library. If it is, it uses the emojize function to convert the Unicode code back to its corresponding text description, and then replaces the emoji in the text with this text description.

You can call this function with any text that contains emojis to replace the emojis with their corresponding text descriptions.

Appendix

Appendix A - Hyperparameters search space for classification models

Logistic regression	{“C”: (0,0.2), “max_iter” , (100,300)}
Random Forest	{“n_estimators” : (100,300), “max_depth” : (8,200), “ccp_alpha” : (0.001,0.2)}
XGB Classifier	{“max_depth” : (1,8), “n_estimators” : (300,1000), “eta” : (0.01,0.2)

Appendix B - Customised stop word list

```
### custom stop words
#create new stop list that excludes adv and adv and CCONJ so the original sentence structure preserved
from nltk.corpus import stopwords
stop = stopwords.words('english')
for word in stop:
    for token in nlp(word):
        if token.pos_ == "ADV" or token.pos_ == "ADJ" or token.pos_ == "CCONJ":
            stop.remove(token.text)
```

Appendix C - BERT and VADER Results

Appendix C1 - Extraction Method (benchmark) BERT and VADER

Aspect	**Test Accuracy**	**Test Weighted F1**
BERT - Food	0.410	0.389
BERT - Service	0.509	0.525
BERT - Ambience	0.774	0.780
VADER - Food	0.522	0.539
VADER - Service	0.747	0.737
VADER - Ambience	0.760	0.741

Appendix C2 - Extraction Method 2 best models vs BERT and VADER

Aspect	**Test Accuracy**	**Test Weighted F1**
BERT - Food	0.345	0.226

BERT - Service	0.321	0.165
BERT - Ambience	0.316	0.165
VADER - Food	0.564	0.539
VADER - Service	0.228	0.111
VADER - Ambience	0.272	0.155

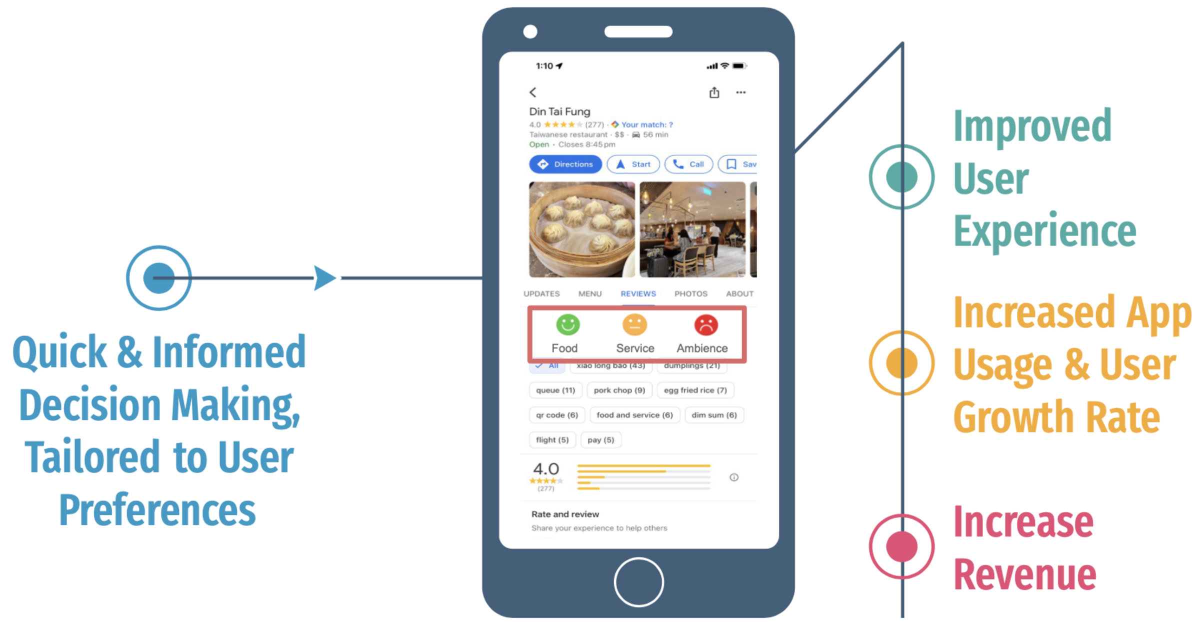
Appendix C3 - Extraction Method 3 best models vs BERT and VADER

Aspect	**Test Accuracy**	**Test Weighted F1**
BERT - Food	0.530	0.529
BERT - Service	0.612	0.546
BERT - Ambience	0.723	0.659
VADER - Food	0.563	0.533
VADER - Service	0.210	0.107
VADER - Ambience	0.292	0.145

Appendix D: Aggregated scores per restaurant

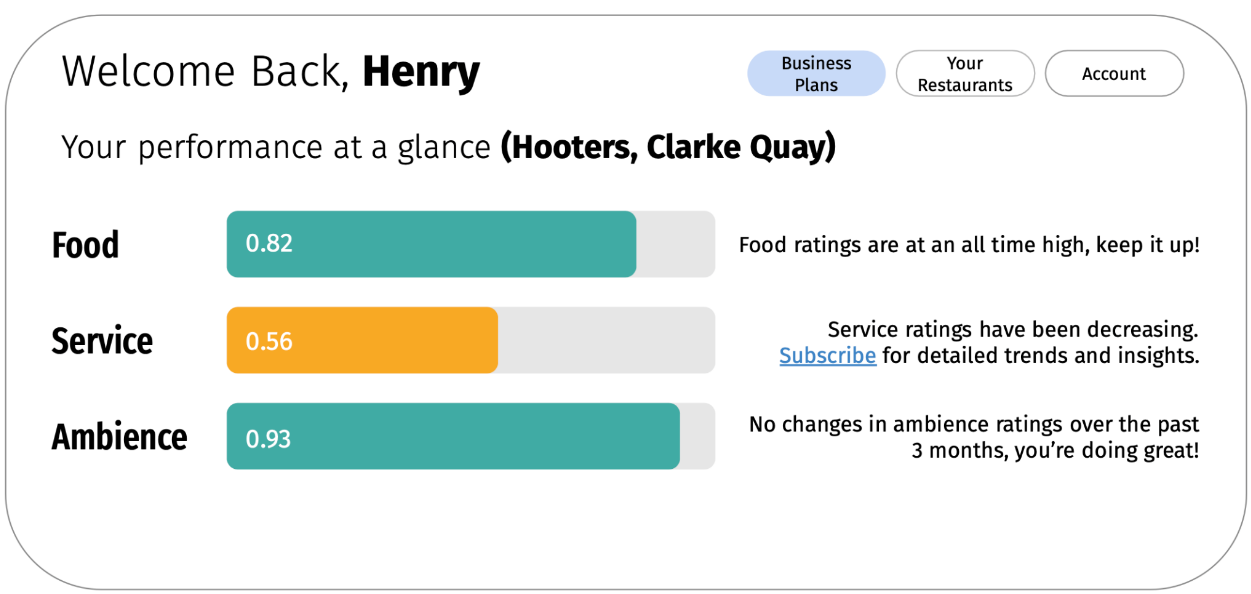
restraunt_name	food_sentiment	service_sentiment	ambience_sentiment
Chichenti	0.883562	0.102740	0.140411
Eggslut	0.678623	-0.008608	0.064562
Hooters	0.685864	0.054974	0.162304
KokSen	0.757895	-0.054386	0.036842
LolaCafe	0.863216	0.123845	0.316081
PennyUniversity	0.800718	0.034111	0.210054
WheelersYard	0.745098	-0.007130	0.320856

Appendix E: Google Map Users Use Case Example



Appendix F - Restaurant Owners Use Case Examples

Appendix F1 - Overview of Sentiment Scores



Appendix F2 - Aggregated Scores Over Time