VADER Sentiment Analysis of Restaurant Reviews

Christian Acosta - 6369251
Malka Aktepe - 6617131
Bart Breekveldt - 2904144
Jeroen Dekker - 5981042

INFOMCDMMC
Dr. Dennis Nguyen
Utrecht University

Table of Contents

1 Introduction **3**

2 Literature Review **3**

3 Methods **4**

3.1 Data Collection 4

3.2 Data sampling and preprocessing variables 4

3.3 Methods for data analysis 6

4 Results **8**

5 Discussion and Conclusion **13**

References **15**

Appendix **17**

Contribution Form 17

1 Introduction

The field of natural language processing (NLP) has seen significant advancements in recent years, and one of the most widely used techniques is sentiment analysis (Birjali et al., 2021; Khurana et al., 2022). Sentiment analysis, also known as opinion mining, is the process of determining the emotional tone of a piece of text. It is a rapidly growing area of research and has a wide range of applications, such as social media analysis, customer feedback analysis, and product reviews. In this paper, we will be using the VADER sentiment analysis tool, which is a part of the Natural Language Toolkit (nltk), to analyze restaurant reviews.

The goal of this study is to predict the star rating of a restaurant based on customer reviews using VADER. We believe that this is an important task as it can help restaurant owners and managers understand the perceptions of their customers and make improvements accordingly. Additionally, it can also assist customers in making informed decisions when choosing a restaurant. Now people don't consider a restaurant with less than four stars while their review isn't always that bad. This makes for a more fair competition and especially for restaurants that just started.

VADER, which stands for Valence Aware Dictionary and Sentiment Reasoner, is a lexicon-based sentiment analysis tool. It uses a combination of lexical and grammatical heuristics to determine the sentiment of a piece of text. VADER is particularly useful for social media text, which is often short and informal, as it does not require any training data. It has been shown to be highly accurate for sentiment analysis of social media texts, making it an ideal tool for our analysis of restaurant reviews(Hutto, 2014). Which leads to the main research question:

How can sentiment analysis be used to predict the star rating of restaurant reviews?

2 Literature Review

There has been much research performed on restaurant reviews. Multiple papers have shown that there are four main aspects someone can judge their dining experience on (Jeong & Jang, 2011). These four aspects are: food quality, service, ambiance, and price fairness (Nakayama & Wan, 2019). Pantelidis (2010) conducted a topic analysis on reviews of restaurants in London from 2005-2007 and 2008-2009. His results revealed that the four topics mentioned above are stable across time.

Jeong and Jang (2011) also examined that a positive experience in a restaurant motivates the customer to leave a positive review. This can be a reason why there are more positive reviews on these review platforms than negative reviews.

Previous research has stated that there is minimal effort from the hospitality and tourism sector to process and analyze big restaurant data through AI (Lee, Kwon & Back, 2021). Lee's study aimed to compare models that predict review helpfulness. Helpfulness in this study is defined as the option to give a thumbs up/down to the review. Their results

suggested that attributes regarding a reviewer's credibility were more important than the sentiment of the text or ratings.

3 Methods

3.1 Data Collection

For retrieving restaurant reviews at first Google Maps is used as a data source. Retrieving reviews from the internet is executed by webscraping, which is facilitated by the Google Places API. A Puppeteer scraper hosted on the Apify platform is used to perform the retrieval of reviews and relevant data. Puppeteer is a browser automation library that allows you to control a browser using JavaScript. The Google Places API returns information about places using HTTP requests. A Puppeteer scraper hosted on the Apify platform is used to perform the retrieval of reviews and relevant data. 122 restaurants are selected in the London area, using the "restaurants near London" search term .A maximum of 250 reviews are scraped from each restaurant. Due to some restaurants having less than 250 reviews, we are left with a dataset totalling 29.046 reviews. This averages out to about 238 reviews per restaurant.

To avoid a bias towards one source of information, in this case returning relevancy as defined by Google, a second source is used for retrieving restaurant reviews. For this a self-built algorithm is made to scrape Tripadvisor reviews. Tripadvisor is chosen along Google Maps due to its high abundance of information including location, price range and cuisine of a restaurant. For Google Maps, the 122 most relevant restaurants are scraped due to sorting limitations in the scraping API. Meanwhile for Tripadvisor a representative set of restaurants in London (519) are scraped, retrieving one from every page of 30 restaurants sorted best to worst. 15 restaurants did not contain any reviews. For the remaining 504 restaurants a maximum of 300 reviews are scraped by restaurants to avoid the influence of restaurants with a lot of reviews. Reviews count 46.303 in total, with 92 reviews by restaurant on average. The algorithm is executed by a locally hosted Chrome driver. With building the algorithm used Python modules contain Selenium, BeautifulSoup and Requests.

3.2 Data sampling and preprocessing variables

The data consists of 27741 reviews from Google Maps and 46303 reviews from TripAdvisor, which sums up to a dataset of 74025 after removing 19 empty reviews. In the figure below the occurrences of each star-rating within the dataset can be seen. This consists of 58% of 5-star reviews, which is overrepresented. At Google Maps-reviews this number is 69% and for Tripadvisor 52%. Distribution of star-ratings is very similar between both sources as can be seen below.

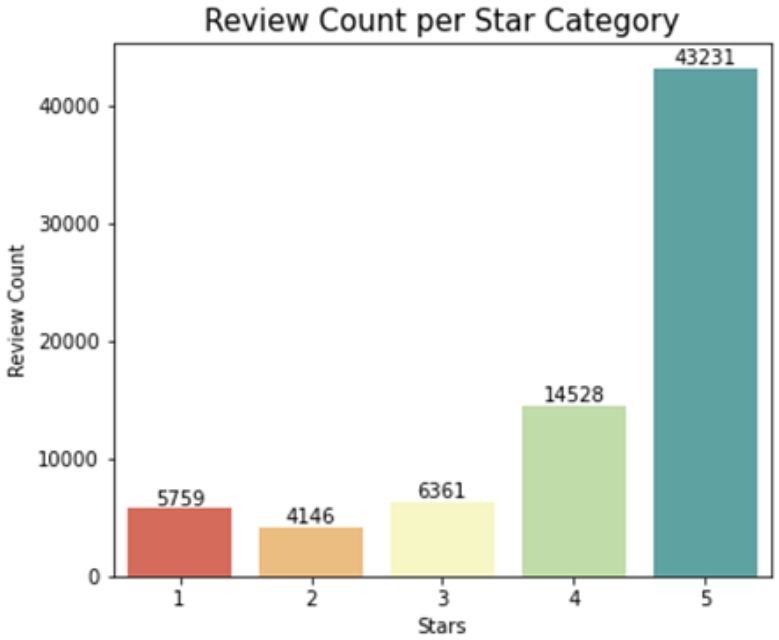

Figure 1: Review Count per Star

To solve the overrepresentation of the five star reviews in any model or configuration, a balanced sample was obtained. This sample size per star has the size of the lowest count of reviews of any star, in this case 4146. The total size of the sampled data is therefore 20.730 reviews which consists of 568 different restaurants in total with a mean overall rating of 3.79 (*sd* = 0.77). The sample contains 36 reviews on average (*sd* = 37) with range = 1-190.

In the table below are the first 5 rows presented of the sampled data. The column 'restaurant' stands for the name of the restaurant; 'review_data' for the dates the reviews were posted; 'total_score' is the mean overall rating of the restaurant; 'stars' are the amount of stars from 1 to 5 given in the review; 'text' is the text of the review; and 'word_count' are the amount of words in the review.

Row	Restaurant	Review Date	Total Score	Stars	Text	Word Count
1	The Victoria	2017-09-22	4.30	1	Never known a pub go downhill so fast	132
2	Thai Crystal	2018-04-27	3.74	1	I used to love this place	146
3	The Golden Pagoda	2015-08-27	2.45	1	Worst service in ChinaTown	37
4	No. 197 Chiswick Fire Station	2018-01-02	3.45	1	I came here with my family and we	78
5	The Lamb Tavern	2022-06-20	3.64	1	Decided to call in as it was a nice	124

Table 1: First 5 rows of the sampled data

Furthermore, most reviews from our sample come from 2019 and 2022, as seen in figure 2. A dip in the amount of reviews is seen in 2020-2021, most likely because of the covid-19 pandemic.

Figure 2: Reviews of scraped restaurants over time.

Next to that, there seems to be some kind of relation between word count and amount of stars given as 1-star reviews tend to have almost double the amount of words than in 5-star reviews. In table 2 these simple statistics are summarized.

Stars	**Count**	**Mean**	**S.D.**	**Min**	**25%**	**50%**	**75%**	**Max**
1	4146	80.517	76.899	1.0	31.0	57.0	105.0	757.0
2	4146	79.731	71.762	1.0	33.	58.0	104.0	986.0
3	4146	67.968	60.610	1.0	29.0	50.0	87.0	696.0
4	4146	53.169	45.225	1.0	26.0	41.0	66.0	713.0
5	4146	45.514	39.527	1.0	22.0	35.0	57.0	735.0

Table 2: Simple statistics of word count among stars

The pre-processing of reviews is done for better modeling results and removing unnecessary noise in the text. Methods of lowercasing, removing punctuation and hyperlinks are performed uniformly. No part of speech (POS) -tagging is done and no words are removed specifically in this way. The VADER-algorithm is already very sophisticated in processing real texts, including emojis, numbers, determiners and stopwords. In addition the other “modeling” algorithms each require their own further pre-processing. This is explained with these models.

3.3 Methods for data analysis

The pre-processed data is analyzed by NLTK VADER. VADER uses a lexicon based on 7k+ words that have a sentiment score ranging from -4 to 4 and a standard deviation (Hutto et al,

2014). The VADER-scores are determined by a representative set of human annotators. The average and standard deviations of these annotation results are stated as the sentiment mean and sentiment standard deviation. That can show how different words can have different sentiments with other people. With the VADER lexicon term frequency inverse document frequency (tf-idf) can be determined alongside the sentiment of the top tf-idf words and determine their differences. Furthermore VADER uses a number of hand-written pattern matching rules (e.g., negation, intensifiers). This way it can account for things like “The food was not good”, here it can see that “good” should not be counted as a positive sentiment but rather a negative. VADER can even recognize emojis and use them for sentiment scores. NLTK gives 4 values per review. A negative, neutral, positive and compound score. Compound is based on the three other different scores. Compounds range from -1(very negative) to +1 (very positive). Negative, positive and neutral scores are set on a scale 0-1, stating the share of words in a lexicon marked as positive, neutral or negative.

The review-texts and the distribution of stars may not give good predictions for the sentiment of a certain text. From this idea we divert to two alternatives: 1) to capture the most distinctive words for each star and 2) observe if a prediction from the review text to a continuous sentiment does a better job at predicting the sentiment than predicting the star itself. Thus question if given stars are consistent and if they should be used in general to capture people’s sentiment from a restaurant review. In addition, the most distinctive words can be compared to the VADER-sentiment lexicon and see if they differ. Standard preprocessing as mentioned is performed, but further preprocessing is not needed, because only words that occur in the VADER-lexicon contain any sentiment are subsetted for analysis. If the preprocessing goes too far, some terms occurring in the VADER-lexicon may be removed beforehand.

To capture the most distinctive words for each star, the term frequency inverse document frequency (tf-idf) can be used (Havrlant & Kreinovich, 2017). The idea behind the tf-idf is to measure both the occurrence of a term (tf) as the relative occurrence of a term (idf). The original tf-idf was applied to this case, but turned out to be working less well in the context of restaurant reviews. Only words that contain any sentiment were inputted in the tf-idf. The word “good” turned out to be present in three, four and five stars, while it just passed document uniqueness (idf), the sheer amount (tf) put it on top. Tf-idf with 25 documents a term should occur in and a max of 40% of documents at a maximum for a term to be in. The general idea of the tf-idf was used before for sentiment analysis (Mohd Nafis & Awang, 2021), but the tf-idf overvalues tf over idf in the case of restaurant review terms.
An alternative function of relative occurrence is used to capture both the term distinctiveness (similar to idf) as the term frequency depending on the star.

$$tfidf(t,x,D) = tf(t,x)\, idf(t,D) = \frac{f_{x,t}}{\sum_{t' \in d} f_{t',x}} \log \frac{N}{|\{d \in D : t \in t\}|} \quad (1)$$

$$relative\ occurrence\,(S,t,D) = \sum_{x \in S} f(t,d|x)\,(\frac{\sum_{x \in S} f(t,d|x)}{\sum f(D,t)} - 1) \quad (2)$$

Equation 1: The term frequency inverse document frequency (tf-idf) divides the term frequency (t) by all terms in one of the star-reviews (x). It then takes a logarithmic function of the number of documents (N) divided by the occurrence of term (t) in a one of the star-reviews (x) considering all documents (D)

Equation 2. The relative occurrence function sums all term frequencies (t) from documents (d) (review texts) depending on star (x ∈ S). The relative document frequency then takes this sub-equation and divides this by a sum of term frequencies (t) in all documents (D). These two are then multiplied.

Second to relative term occurrences, we want to observe if a prediction from the review text to a continuous sentiment does a better job at predicting the sentiment than predicting the star itself. We need to know (1) which of the three sentiment variables: compound, positive or negative can be predicted the best by the review text. (2) what kind of continuous model in which configuration gives the “best” result and how to measure this? And (3) can we compare the VADER-scores with any outcome of the model? The model requires a regression model instead of a classification model because the VADER-scores are given on a continuous scale in contrast to the stars which require classification models. Three models are compared: the linear regression, a polynomial regression and a support vector machine regression (SVM regression).

To measure which of the sentiment values to use the correlation coefficient, or R2 is used. The R2 measures the proportion of outcomes that can be predicted by the input variables and which can not on a scale from 0 to 1, with 1 being a perfect prediction. A R2 of 0.5 is equal to flipping a coin, thus a value at or near 0.5 is less relevant. The spread of the variables is measured by the root mean squared -deviation (RMSD) or -error (RMSE). The RMSE specifies the accuracy of a given prediction by specifying the adjusted mean deviation for observations to the regression line. To solve the problem of the RSME being variable-dependent, the RSME is made proportional to its mean for comparability.

4 Results

Figure 3 shows the distribution of stars and the compound score (number of reviews with that particular score) with the colors showing the difference in stars given.

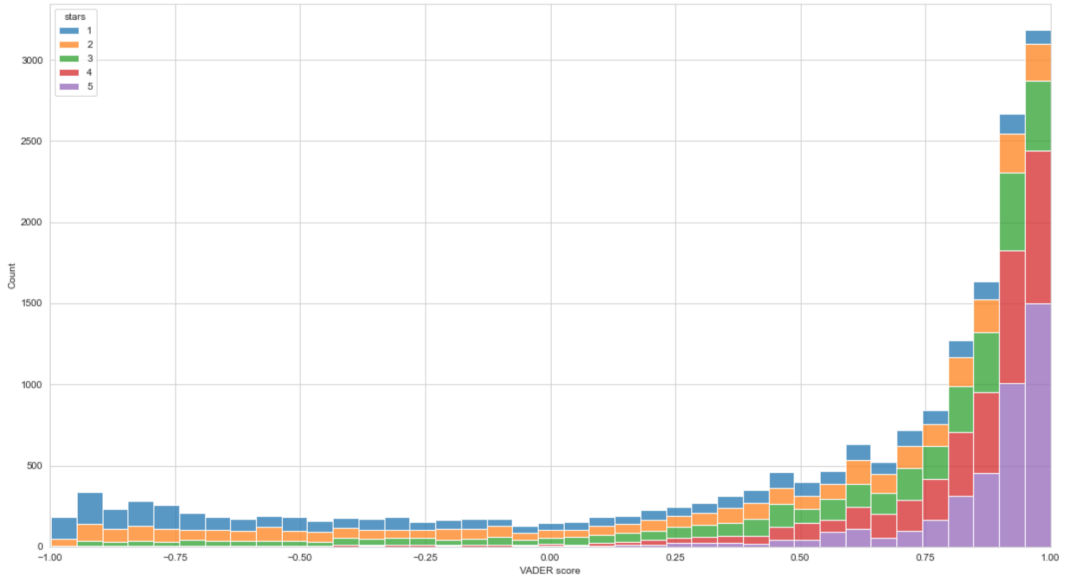

Figure 3: VADER scores on the x-axis is the VADER compound score given ranging from (-)1 to (+)1 on the y-axis are the amount (count) of reviews with the particular VADER compound score. The coloring displays the star rating given in the review.

From figure 3 as can be seen above the stars perform some tendency to their expected values, but the distribution per bin is more diverse than expected. There is also no difference in the relative sizes between one and two stars, and the same can be said between the distribution of three and four star reviews. Only the five star category performs a bit as expected. But when we look at the bin with the highest compound score, only about half is categorized as five stars, while the other part consists of four and less stars. These observations are confirmed by confidence intervals shown in figure 4 below. The distribution intervals also show some large overlap. Based on the distribution intervals, some distinction can be made between the low end (1-3 stars) and high end (4-5 stars) of the spectrum, but a difference between, for example, 1 and 2 stars is not distinguishable. A Logistic Regression for the five stars from stemmed text returns a F1-score of 0.542.

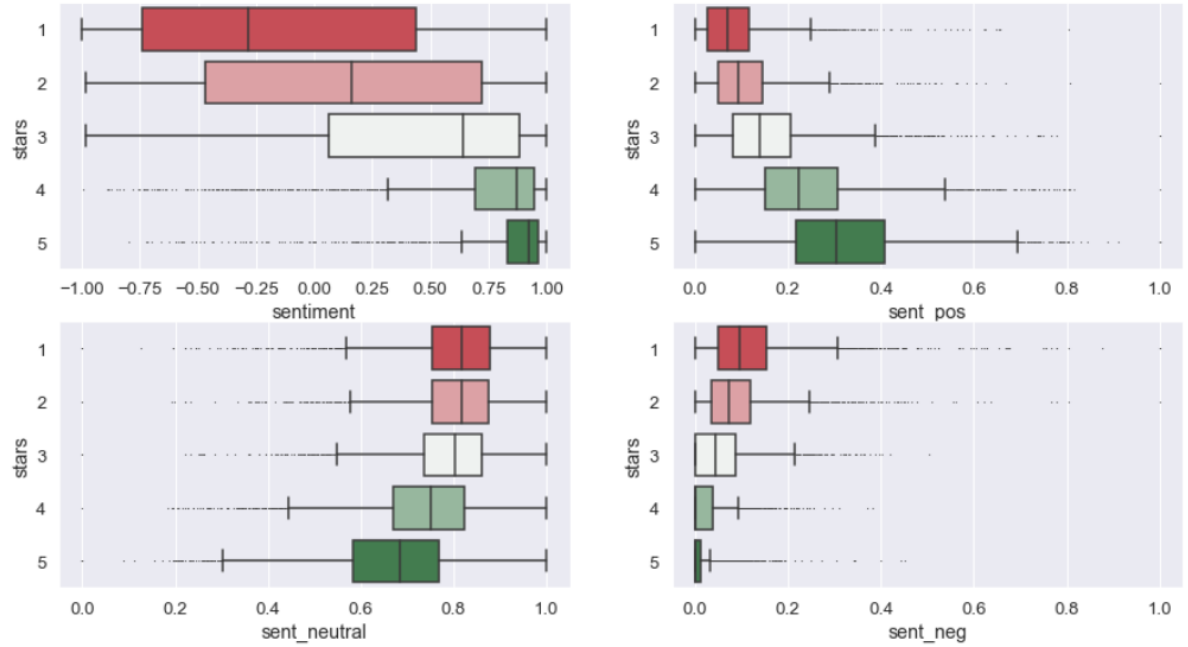

Figure 4: Boxplots on the VADER score. Boxplots for the outputs of the VADER-sentiment algorithm are shown: compound (ranging -1 to +1), positive sentiment (0-1), neutral sentiment (0-1) and negative sentiment (0-1). The boxplots show the distribution of star-occurrences by vader sentiment scores with the interquartile-range (IQR) being 50% of occurrences, and with the distribution interval widths maximized at 1.5x the IQR. Outside this, observations are seen as outliers and shown as points.

The results from the relative occurrence function, i.e. the most relevant terms by star, can be seen in table 3 below. The top twenty terms are attached to their VADER-score. The top terms of this function can be seen in table 3 below with their VADER-scores (-1 to +1). These terms give an indication which terms are used in reviews giving a certain star. The average VADER-scores by star are -0.2600 (1-star), -0.1205 (2-star), 0.1410 (3-star), 0.5015 (4-star) and 0.6275 (5-star). These indicate a large gap between three and four star scores. In addition the VADER-scores of terms are more biased positive than negative on average (+0.1779).

Pl.	1-star token	score	2-star token	score	3-star token	score	4-star token	score	5-star token	score
1	No	-0.30	No	-0.30	Good	0.48	Good	0.48	Great	0.78
2	Rude	-0.50	Like	0.38	Ok	0.40	Great	0.78	Amazing	0.70
3	Bad	-0.62	Ok	0.30	Ok	0.30	Friendly	0.55	Delicious	0.68
4	Worst	-0.78	Ok	0.40	Nice	0.45	Nice	0.45	Excellent	0.68
5	Like	0.38	Poor	-0.52	Better	0.48	Excellent	0.68	Best	0.80
6	Terrible	-0.52	Disappointing	-0.55	Pretty	0.55	Delicious	0.68	Friendly	0.55
7	Poor	-0.52	Disappointed	-0.52	Like	0.38	Well	0.28	Definitely	0.42
8	Awful	-0.50	Better	0.48	Pleasant	0.57	Enjoyed	0.57	Recommend	0.38
9	Avoid	-0.30	Unfortunately	-0.35	Disappointing	-0.55	Lovely	0.70	Fantastic	0.65
10	Leave	-0.05	Bad	-0.62	Okay	0.22	Value	0.35	Lovely	0.70
11	Pay	-0.10	Pretty	0.55	Sure	0.32	Fresh	0.32	Wonderful	0.68
12	Horrible	-0.62	Shame	-0.52	Special	0.42	Worth	0.22	Loved	0.72
13	Friend	0.55	Pay	-0.10	Fine	0.20	Interesting	0.42	Thank	0.38
14	Want	0.08	Fine	0.20	Shame	-0.52	Helpful	0.45	Perfect	0.68
15	Empty	-0.20	Sadly	-0.45	Limited	-0.22	Pleasant	0.57	Love	0.80
16	Refused	-0.30	Wrong	-0.52	Unfortunately	-0.35	Loved	0.72	Superb	0.78
17	Disgusting	-0.60	Disappointment	-0.57	Hard	-0.10	Efficient	0.45	Helpful	0.72
18	Wrong	-0.52	Want	0.08	Sadly	-0.45	Enjoyable	0.48	Super	0.45
19	Clearly	0.42	Empty	-0.20	Noisy	-0.18	Perfect	0.68	Well	0.28
20	Charged	-0.20	Party	0.42	Clean	0.42	Recommended	0.20	Beautiful	0.72
	Average	**-0.26**	**Average**	**-0.12**	**Average**	**0.14**	**Average**	**0.50**	**Average**	**0.63**

Table 3: Terms with the largest relative occurrence score by star accompanied by their VADER-score ("Vsc")

The same process is repeated with bi-grams. These are tuples of two terms seen as one entity. Bigrams are formed with a moving window with size two over the review text. One of the two bigram terms should contain a VADER-lexicon term. The average VADER-scores by star are now -0.3230 (1-star), -0.1150 (2-star), 0.2760 (3-star), 0.4865 (4-star) and 0.5135 (5-star). Their average is now +0.1676, a slight decrease in positive score bias. The 1-star score has decreased, the 3-star score increased and the 5-star decreased resulting in 4- and 5-star reviews having about the same average VADER-score, while the largest gap is now between 2-3 stars. The bigrams give a bit more context regarding interpretation i.e. "nothing special" and "special" contain two completely different sentiments. The top bigrams by star can be accurate star-descriptions in their own right. The VADER-scores overall seem about right, but "highly recommend[-ed]" has a lower score than "pretty good", while the first is more valuable in restaurant review terms, but this is probably the case because VADER is trained mainly on social media instead of reviews (Hutto & Gilbert, 2014).

Pl.	1-star bigram	score	2-star bigram	score	3-star bigram	score	4-star bigram	Score	5-star bigram	score
1	The worst	-0.62	Was ok	0.30	Was good	0.44	Very good	0.49	The best	0.64
2	Rude and	-0.46	Good but	0.24	Good but	0.24	A good	0.44	Friendly and	0.49
3	To pay	-0.10	Ok but	0.15	Was ok	0.30	A great	0.62	Highly recommend	0.42
4	To leave	-0.05	Not worth	-0.17	Was nice	0.42	Good and	0.44	A great	0.62
5	My friend	0.49	Very disappointed	-0.53	Nice but	0.23	Good food	0.44	Will definitely	0.40
6	Very rude	-0.51	Very disappointing	-0.54	Good and	0.44	Friendly and	0.49	Thank you	0.36
7	Not recommend	-0.28	Like a	0.36	Nothing special	-0.31	Very nice	0.48	Was amazing	0.59
8	No one	-0.30	Nice but	0.23	Is good	0.44	Good service	0.44	Great food	0.62
9	Want to	0.08	Very poor	-0.53	Ok but	0.15	Enjoyed the	0.51	Great service	0.62
10	Refused to	-0.30	A shame	-0.48	A good	0.44	A nice	0.42	Delicious and	0.57
11	Was no	-0.30	Poor service	-0.48	Not sure	-0.24	Friendly staff	0.49	Very friendly	0.54
12	Had no	-0.30	Not good	-0.34	Good the	0.44	Was good	0.44	Delicious food	0.57
13	Very poor	-0.53	Nothing special	-0.31	Not bad	0.43	Really good	0.49	Definitely be	0.40
14	Very bad	-0.58	At best	0.64	Not great	-0.51	And good	0.44	Highly recommended	0.27
15	Very disappointed	-0.53	Not recommend	-0.28	Were good	0.44	Was delicious	0.57	And friendly	0.49
16	With no	-0.30	Much better	0.44	Nice and	0.42	Great place	0.62	Amazing food	0.59
17	Avoid this	-0.30	To pay	-0.10	Was pretty	0.49	Worth a	0.23	Would definitely	0.40
18	Extremely rude	-0.51	Was no	-0.30	Pretty good	0.73	Very friendly	0.54	And great	0.62
19	So bad	-0.58	And no	-0.30	Was okay	0.23	Good value	0.65	Friendly staff	0.49
20	Was terrible	-0.48	With no	-0.30	Just ok	0.30	And friendly	0.49	Was delicious	0.57
	Average	**-0.32**	**Average**	**-0.12**	**Average**	**0.28**	**Average**	**0.49**	**Average**	**0.51**

Table 4: Bigrams with the largest relative occurrence score by star accompanied by their VADER-score ("Vsc")

The positive sentiment variabele can be best predicted by the review text with a R2 of 0.8 (pRSME = 0.18), while the negative sentiment is at 0.645 (pRSME = 0.79) and the compound at 0.6 (pRSME = 0.99). The positive sentiment is the best variable that can be predicted by the input text from both metrics, and is used further. Apart from the standard preprocessing, further preprocessing is done by stemming the text which reduces words to their base variant. This is done to densify the eventual tf-idf matrix. The tf-idf matrix is built with all unique terms and all observations (review texts) to serve as input for the various regression models. To prevent this matrix from being too sparse and therefore affect the predictability, in addition to stemming, a minimum number of documents in which a term occurs is specified. Optima for metrics are determined in test size (0.1), tf-idf minimum term occurrence in documents (25) and containing 1-size ngrams. Multiple iterations are

performed to avoid the outcomes being influenced by a single random train-test split. The (second degree) polynomial regression performs the best with a R2 of 0.835, better than the linear regression of 0.8, acknowledging that the relation is not linear. The three SVM variants perform worse than the two linear/polynomial regression configurations: Linear (R2 0.748), Radial Basis Function (R2 0.760) and Polynomial (R2 0.688). The same optima are used as in the linear/polynomial regression.

For interpretability reasons the average coefficients (of model iterations) of linear regression are taken to compare. The coefficients give the effect of a certain term to the positive sentiment with an intercept of zero. The combined positive score of a text averages all the coefficients. To compare the top terms the effect is multiplied by the term frequency to get the most influential terms. Words are retranslated to the most occurring non-stemmed word to enable comparison to the VADER-lexicon word scores. Table 4 below shows the top 25 words in terms of relevancy, combining term frequency and effect (coefficients). Words shown are only those which occur in the VADER-lexicon, the coefficients might be somewhat overstated with the distributions over a whole text. The difference in coefficients is somewhat steeper than VADER-lexicon score difference of certain terms.

Place	Token	Individual effect (coefficient)	VADER sentiment (scale -1 to 1)
1	Good	*0.511*	*0.48*
2	Great	*0.585*	*0.78*
3	Nice	*0.407*	*0.45*
4	Like	*0.275*	*0.38*
5	No	*-0.026*	*-0.30*
6	Lovely	*0.451*	*0.70*
7	Backed	*-0.020*	*0.02*
8	Friendly	*0.355*	*0.55*
9	Disappointed	*-0.070*	*-0.52*
10	Excellent	*0.172*	*0.38*

Table 5: Top terms by raw occurrence, shown with their positive sentiment coefficients. These coefficients are averaged when integrated in text. Only words existing in VADER-lexicon, its sentiment is also shown.

5 Discussion and Conclusion

The main question that needs to be answered: how can sentiment analysis be used to predict the star rating of restaurant reviews. From the exploratory data analysis, we can conclude that, however there is some tendency between the VADER-sentiment score and the given stars at restaurant reviews there is no significant distinction between the (compound) sentiment scores. The stars therefore can not be predicted by the VADER-compound score at a reasonable error rate. A possible explanation can be that the VADER algorithm is trained and modeled around social media sentiment and not reviews sentiment, which texts are often structured differently (Hutto & Gilbert, 2014). This can be seen when returning terms and their VADER-scores. However, some specific terms and bigrams can be determined and when compared to their VADER-lexicon term scores, the star-average of top terms is still in the chronological order of the stars, but the 1-2 and 4-5 star scores are sometimes barely distinguishable. For further research, a reduction of categories to 2 or 3 may be more suitable.

Currently, the majority of NLP methods and models are suited to a specific context or use-case, and will perform significantly worse outside their intended use (Khurana et al., 2022). For our specific example, VADER is designed to be used with social media texts and performs the best in that scenario, which is ideal for this research case. VADER would be outperformed by certain classification models in, for example, a use-case with domain-specific language and jargon. Sarcasm, synonyms, and homonyms are a few other areas where NLP methods still need to improve (Khurana et al., 2022)

The study knows some other limitations. There might also be some cultural differences. Nakayama and Wan (2019) have done a study on the differences between Japanese and English reviews on Yelp. Their results showed that there is a significant difference between Japanese and English reviews. Japanese reviews have different sentiment distribution patterns in the four attributes mentioned in the introduction (food quality, service, ambiance, and price (fairness)). London is a very international city which also attracts a lot of tourists. Although all reviews used in our study are in English, there is a possibility that they are written by someone who does not speak English as their first language. Next to that, not only their English level might differ, but also the content of the review may vary based on someone's cultural background as shown by Nakayama and Wan (2019).

In the current study we took a random sample of restaurants with various price ranges and have not used this in our analysis since the price ranges of Google Maps and TripAdvisor had different ranges. Routledge and Smith (2014), who examined 900.000 restaurant reviews, found that there is a difference between positive and negative reviews within a price range. The study has reviews from a full range of restaurants, from fast food to luxury restaurants. Both positive and negative reviews of expensive restaurants tend to be longer and use bigger words to mirror the reviewer's linguistic capabilities compared to reviews of inexpensive restaurants. We have however not accounted for this in the current study.

There is also sampling bias as we took a sample of a sample to create a balanced data set, because the data was very skewed and consisted roughly of 70% of 5-star reviews.

However, this strong positive skew in the star ratings is consistent with previous work analyzing reviews of movies, hotels, restaurants, and consumer products (Potts, 2011).

Yet with all these drawbacks we still managed a decent model. The positive sentiment is the best explainable variable from a restaurant review. The R2 is 0.835, which is better explainable than the stars with an R2 of 0.542. This might be an indication that a continuous scale may be a better fit than stars to predict a certain sentiment. The text may be better explaining feelings about their experience than a star-rating, which may be difficult to weigh in all the different aspects of your review.

The underperforming of the SVM regression in all its variants has probably to do with the sparsity of the dataframe, to something called the curse of dimensionality. This occurs when the number of observations is not much bigger than the amount of variables. SVM regression needs more data points to obtain solid support vectors than linear or polynomial regression, which is only focused around a trend line. At SVM the curse of dimensionality therefore occurs earlier, more data points in relation to the amount of variables, and is probably the reason why the SVM regression performs worse. This may also explain why polynomial SVM regression performs even worse, as the SVM regression and polynomial features are compounding factors to create a higher dimensionality. Further densifying apart from stemming or lemmatization can potentially be done by finding frequently co-occurring words a review, sentence or ngrams and use the clusters as input for the several regression models. This can potentially be done by K-means clustering (Riaz, Fatima, Kamran et al., 2019).

This concludes our question that it is possible to use sentiment analysis to predict star reviews. But it is unfortunately very limited. Our suggestion would be to use more as an indication for a positive or negative and not use the star rating system. Furthermore research into more specific questions would be advisable, things like specific price points or cuisines.

References

Birjali, M., Kasri, M., & Beni-Hssane, A. (2021). A comprehensive survey on sentiment analysis: Approaches, challenges and Trends. Knowledge-Based Systems, 226, 107134. https://doi.org/10.1016/j.knosys.2021.107134

Gan, Q., Ferns, B. H., Yu, Y., & Jin, L. (2016). A Text Mining and Multidimensional Sentiment Analysis of Online Restaurant Reviews. *Journal of Quality Assurance in Hospitality & Tourism*, *18*(4), 465-492. https://doi.org/10.1080/1528008X.2016.1250243

Havrlant, L. & Kreinovich, V. (2017 March 14) A simple probabilistic explanation of term frequency-inverse document frequency (tf-idf) heuristic (and variations motivated by this explanation). *International Journal of General Systems.* *https://doi-org.proxy.library.uu.nl/10.1080/03081079.2017.1291635*

Hutto, C., & Gilbert, E. (2014, May). Vader: A parsimonious rule-based model for sentiment analysis of social media text. In Proceedings of the international AAAI conference on web and social media (Vol. 8, No. 1, pp. 216-225).

Jeong, E. H., & Jang, S. C. S. (2011). Restaurant experiences triggering positive electronic word-of-mouth (eWOM) motivations. *International Journal of Hospitality Management*, *30*(2), 356–366. https://doi.org/10.1016/J.IJHM.2010.08.005

Jurafsky, D., Chahuneau, V., Routledge, B. R., & Smith, N. A. (2014). Narrative framing of consumer sentiment in online restaurant reviews. *First Monday*.

Khurana, D., Koli, A., Khatter, K., & Singh, S. (2022). Natural language processing: State of the art, current trends and challenges. Multimedia Tools and Applications, 82(3), 3713–3744. https://doi.org/10.1007/s11042-022-13428-4

Lee, M., Kwon, W., & Back, K. J. (2021). Artificial intelligence for hospitality big data analytics: developing a prediction model of restaurant review helpfulness for customer decision-making. *International Journal of Contemporary Hospitality Management*, *33*(6), 2117-2136.

Mohd Nafis, N.S & Awang, S. (2021) An Enhanced Hybrid Feature Selection Technique Using Term-Frequency Inverse DOcument Frequency and Support Vector Machine-Recursive Feature Elimination for Sentiment Classification *IEEE Access vol.9 p.52177 - 52192. 10.1109/ACCESS.2021.3069001*

Nakayama, M., & Wan, Y. (2019). The cultural impact on social commerce: A sentiment analysis on Yelp ethnic restaurant reviews. *Information & Management*, *56*(2), 271–279. https://doi.org/10.1016/J.IM.2018.09.004

Pantelidis, I. S. (2010). Electronic meal experience: A content analysis of online restaurant comments. *Cornell Hospitality Quarterly*, *51*(4), 483-491.

Riaz, S., Fatima, M., Kamran, M. *et al.* Opinion mining on large scale data using sentiment analysis and k-means clustering. *Cluster Comput* **22** (Suppl 3), 7149–7164 (2019). https://doi.org/10.1007/s10586-017-1077-z

Zhang, Z., Ye, Q., Law, R., & Li, Y. (2010). The impact of e-word-of-mouth on the online popularity of restaurants: A comparison of consumer reviews and editor reviews. *International Journal of Hospitality Management*, *29*(4), 694–700. https://doi.org/10.1016/J.IJHM.2010.02.002

Appendix

Contribution Form

	Jeroen reflects on:	**Malka reflects on:**	**Bart reflects on:**	**Christian reflects on:**
Jeroen	Tried to keep the ocus on the end goal. Did most part of the /ADER analysis. Could have contributed more on echnical part at the nd	Jeroen made sure that ve did not lose focus of he main objective during meetings. He lso worked on the /ader analysis, ntroduction and liscussion.	Jeroen knows well which parts he can do est from his xperience. He often as the counterpart in liscussions if my houghts went too far ff track.	A good balance in the roup dynamic. Ensured ve stayed on track and ept our objectives ealistic. Did a good part f the VADER analysis, ntroduction, and liscussion sections.
Malka	Great teamplayer, did ast amounts of work n the EDA part. Good eam-member	Malka did much during he first part of the roject and was more of he creative mind of the eam. She worked ainly on the EDA, terature review, ampling and variables, nd the discussion.	Malka is a joyful group ember who brings ositive energy in a roup. Did good stuff r the EDA part, and new her code well.	A fun team member who eeps our spirits up. She ame up with good ideas r the research project. id a good part of the DA and literature review, nd helped out with the liscussion as well.
Bart	Did lots of things, very echnical. Lead most onversations but did ometimes lose focus n discussions but lways had great input	Bart's expertise was his oding peed and knowledge s he did most of the inal analysis. He also ontributed to the data nalysis part of the eport as well as the esult section. Although e sometimes gets lost eeply into minor details aking the meetings go n for a long time, his ibrant energy made im a very nice eammate to have.	This project required witching mindsets, as he initial three weeks o past fast. Often took he lead. Tried to keep he end goal in mind ontinuously, but lost it ccasionally when my nterest was aroused y something to learn, ke building the ripadvisor web craping algorithm.	Bart was the mad cientist of the group and ad many, very often mbitious, ideas. He has lot of energy and edication, which makes im a great person to ave in a team. Occasionally we had to udge him a little to not se the big picture in the etails, but this was not a eal problem for us. He as a powerhouse in erms of the most echnical part of this roject and worked on elivering detailed results.

Christian	The silent power of his group. Critical in he data collection. And overall contributed everywhere	Christian contributed to scraping the data and doing the presentation. He worked on almost every section of the eport as well. Although he was more on the quiet side during the meetings, he still made sure to do all his tasks horoughly.	Christian is a nice eammate to spar with bout technical issues nd opportunities. He vas a nice partner in resenting, I could trust im on that which contributed to making he presenting nice to lo. He may take a bit more initiative in ideas e thinks are worth elling, but overall a olid teammate to ave.	Worked on the data collection, presentation, nd a bit of everything in he report. The literature eview, discussion, and ata collection were the ain parts from my end. I vas a bit less talkative han my teammates, but vas always a keen bserver. Had a great xperience working with ll of them.