REPUBLIC OF TURKEY
YILDIZ TECHNICAL UNIVERSITY
DEPARTMENT OF COMPUTER ENGINEERING

ANALYSIS OF FACILITY REVIEWS

16011705 — YUNUS EMRE DEMIR
16011128 — SEMIH DURMAZ

SENIOR PROJECT

Advisor
Prof.Dr. Banu DİRİ

July, 2021

ACKNOWLEDGEMENTS

We wish to express our sincere appreciation to our supervisor, Professor Banu Diri and Teaching Asistant Ahmet Elbir for their consistent support and guidance during the running of this project. Without their persistent help, this project would not have been possible.

YUNUS EMRE DEMIR
SEMIH DURMAZ

TABLE OF CONTENTS

LIST OF ABBREVIATIONS

GB	Gigabyte
MB	Megabyte
TF-IDF	Term Frequency-Inverse Document Frequency
VADER	Valence Aware Dictionary and Sentiment Reasoner
W2V	Word2Vec

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

ANALYSIS OF FACILITY REVIEWS

YUNUS EMRE DEMIR
SEMIH DURMAZ

Department of Computer Engineering
Senior Project

Advisor: Prof.Dr. Banu DİRİ

Our project aims to accomplish sentiment analysis on the reviews made for a facility on social media. Our aim is not only get information about the positive and negative reviews for the facility but also what the positive and negative reviews are about. The hotel reviews in the dataset that will be used do not have any tags. Therefore, as first step, the positive and negative reviews were determined by VADER, one of the lexicon based sentiment analysis methods. Then eleven attributes have been determined that can represent the hotel or can be used when defining the hotel. These are staff, location, room, hotel, breakfast, bed, service, bathroom, restaurant, view and food.

Later, other similar words to identify these hotel attributes used by users were found with the help of W2V. Fasttext was used to find the comments by referring to a hotel attribute but with some typing mistakes in the word. With the help of W2V and Fasttext, the eleven hotel attributes determined previously have been expanded into a list. Then the percentage of positive and negative reviews for the hotel, the percentage of positive and negative reviews about the attributes describing the hotel and what these reviews are presented to the user with charts. Thus, for the hotel reviews expressed in thousands, the information about which attributes of the hotel are good, which attributes are bad, which features are liked by the users and which ones are not particularly liked by the users presented to the hotel owner with reports, graphs and percentages. As a result, time is saved and more rational & practical solutions were offered.

Keywords: social media, sentiment analysis, VADER, fasttext, word2vec

ÖZET

Sosyal Medya Yorumları Analizi

YUNUS EMRE DEMIR
SEMIH DURMAZ

Bilgisayar Mühendisliği Bölümü
Bitirme Projesi

Danışman: Prof.Dr. Banu DİRİ

Projede hedeflenen sosyal medya üzerindeki turistik mekanlar hakkında yapılan yorumlar üzerinden duygu analizi yapmaktır. Amacımız, hakkında yorum yapılan mekanlar için sadece olumlu ve olumsuz bilgisinin yanı sıra yapılan olumlu veya olumsuz görüşlerin ne hakkında olduğunu bilgisini de çıkarmaktır. Projede kullanacağımız datasette bulunan otel yorumları herhangi bir etikete sahip değildir. Bu yüzden ilk önce yorumlara ait olumlu, olumsuz veya nötr olduğu bilgisi sözcük tabanlı duygu analizi metodlarından Vader ile tespit edilmiştir. Oteli temsil edebilecek (oteli tanımlarken kullanılabilecek) 11 adet özellik belirlenmiştir.

Daha sonra, kullanıcıların bu hotel özelliklerini tanımlamak için kullandığı benzer kelimeler Word2Vec yardımıyla bulunmuştur. Kullanıcıların bir otel özelliğini kastederek yazdığı ama yazım hataları yaptığı yorumları bulmak için ise fasttext kullanılmıştır. W2V ve Fasttext yardımıyla, daha önce belirlenen 11 otel özelliği genişletilerek birer dizi haline getirildi. Ardından seçilen otel hakkında (genel) ve oteli tanımlayan özellikler hakkındaki olumlu/olumsuz yorumların yüzdesi ve bu yorumların neler olduğu bilgisi kullanıcıya grafiklerle sunulmuştur. Böylelikle sayılara binlerle ifade edilen otel yorumları hakkında otel sahiplerinin otelinin hangi özelliklerinin iyi olduğu, hangi özellik kötü olduğu, hangi özelliklerinin kullanıcılar tarafından beğenilip, hangi özellikle kullanıcılar tarafından beğenilmediği bilgisi rapor, grafik ve yüzdelerle otel sahibine sunulmuştur. Böylelikle hem zamandan tasarruf edilmiş olup hem daha akılcı ve pratik çözümler sunulmuştur.

Anahtar Kelimeler: duygu analizi, sosyal medya, Word2Vec, Vader

1 Introduction

Technology is constantly evolving and developing. Our habits are also changing along with developing technology. We make restaurant and hotel reservations, shopping and even entertainment on internet. For instance, we prefer spending our time playing computer games instead of going out and playing outdoor games. We no longer make reservations by phone calls but computer, using online reservation systems. Same goes for hotel reservations as well. We have a couple of habits that have changed.

In the past, business establishments and museums used to have visitor record books in which people who were visiting were asked to write their name, address, and anything they would like to say about their visits. In this way the owner of the establishment could read the book to check positive and negative reviews about the place. The developing technology made us forget that habit. In fact, it carried our habit into another platform. Now hotels and restaurants have their own websites as well as there are large-scale web platforms that contain thousands of hotels and restaurants for online reservation. Customers make their reservations on these websites. Visitor record books are supplanted by online review systems. The customers write their good or bad thoughts of the hotel&restaurant to the related hotel&restaurant on the website. While these reviews are important in terms of guiding the next possible customer, they also help the hotel&restaurant owner with seeing their pearls and pitfalls.

However, when the number of these reviews are issued as thousands or tens of thousands, it becomes quite difficult for hotel&restaurant owner to read all and analyze the most loved and hated properties of the place. We have started this project to eliminate this problem. Our project aims to accomplish sentiment analysis based on the reviews made on touristic places in social media. The reviews are mixed, not divided into groups such as positive or negative. Therefore, the first job to be done is to divide reviews into two groups as positive reviews and negative reviews using lexicon-based approach. Afterwards, the information of the reviews' categories will be retrieved from reviews by using Word2Vec algorithm. Processed data will be reported to the hotel owner with the help of graphics and charts.

2
Preliminary Survey

In this chapter, some methods previously used in sentiment analysis are examined. Necessary changes will be made in order for these methods to work with higher efficiency in the project. As well as, you will be informed about the dataset to be used in the project.

Sentiment analysis is the interpretation and classification of emotions (positive, negative and neutral) within text data using text analysis techniques. Sentiment analysis allows businesses to identify customer sentiment toward products, brands or services in online conversations and feedback [1].

2.1 Sentiment Analysis Approaches

In this section, two different approaches to sentiment analysis are examined.

2.1.1 Machine Learning Method

This approach, employes a machine-learning technique and diverse features to construct a classifier that can identify text that expresses sentiment. In this approach training and testing datasets are required. A training dataset is used to learn the documents and test dataset is used to validate the performance. Nowadays, deep-learning methods are popular because they fit on data learning representations.

2.1.2 Lexicon-Based Methods

This method uses a variety of words annotated by polarity score, to decide the general assessment score of a given content. The strongest asset of this technique is that it does not require any training data, while its weakest point is that a large number of words and expressions are not included in sentiment lexicons [2]. Two methods of the Lexicon-based approach will be discussed below.

2.1.2.1 Corpus-Based Approach

The corpus-based approach has the objective of providing dictionaries related to a specific domain. These dictionaries are generated from a set of seed opinion terms that grows through the search of related words by means of the use of either statistical or semantic techniques [3].

2.1.2.2 Dictionary-Based Approach

Dictionary based approach suggests to implement judging of sentiment based on presence of signalling sentiment words (and perhaps some shorter context, like negations in front of them and emojis) + some sort of counting mechanism to arrive at sentiment prediction. This approach will be used during the development of the project.

2.2 Word2Vec Method

Word2Vec is a set of algorithms for finding the distance between words in a vector [4]. With the tools written on the vector structure, it is possible to list the words near or far from a word. The frequent use of certain words in the same sentence increases the closeness of these two words in this words vector. Using this method, the category of the reviews can be found.

2.3 Fasttext

FastText is a library for text classification and representation. It transforms text into continuous vectors that can later be used on any language related task.

It uses a hashtable for either word or character ngrams. One of the key features of fastText word representation is its ability to produce vectors for any words, even made-up ones. Indeed, fastText word vectors are built from vectors of substrings of characters contained in it. This allows to build vectors even for misspelled words. This is why FastText will be used along with Word2Vec model. While Word2Vec finding the the words with closest meaning, fastText will look for possible misspelled words.

2.4 Vader

VADER is a lexicon and rule-based sentiment analysis tool that is specifically attuned to sentiments expressed in social media[5]. It uses a combination of a sentiment lexicon is a list of lexical features (e.g., words) which are generally labelled according to their semantic orientation as either positive or negative. VADER has been found to be quite successful when dealing with social media texts, movie reviews, and product reviews. This is because VADER not only tells about the Positivity and Negativity score but also tells us about how positive or negative a sentiment is. That is the main reason why VADER is selected for use during the development of the project.

VADER has a lot of advantages over traditional methods of Sentiment Analysis, including:

- It works exceedingly well on social media type text, yet readily generalizes to multiple domains
- It doesn't require any training data but is constructed from a generalizable, valence-based, human-curated gold standard sentiment lexicon
- It is fast enough to be used online with streaming data, and
- It does not severely suffer from a speed-performance tradeoff.

Tests of sentiment using VADER will be examined in Chapter 7.

2.5 Dataset to be Used During the Project

The dataset to be used contains the reviews of the top 10 most expensive and least expensive London-based hotels. It's provided by `kaggle.com` That dataset is considered as sufficient since it serves the purpose of the project.

Although the dataset is said to have 27329 reviews, after some examinations made, it was observed that 431 of these reviews are made up of invalid characters such as "" and 3350 reviews are in different languages than English. Since these numbers are too large to be ignored considering the size of available data, the irrelevant comments mentioned above were first detected through languageDetection.py. Later, letters and words that were repeated in these reviews and did not exist in English were identified. These letters/word groups (på, ich, wir, des, ò, ci, Bij, piu, Che dire, Ci, è, un, ed, ó, á, ä, å, di, ç, ğ, ş, ö , ü) removed from dataset.

3 Feasibility

Feasibility analysis related to the project are examined in this section.

3.1 Legal Feasibility

The database to be used in the project is shared by kaggle. It is not expected to cause any legal problems since the data it contains is derived from booking.com and the data is already public. The language, environments and libraries to be used during the development of the software are all open-source and free therefore they are not expected to cause any legal problems either.

3.2 Technical Feasibility

Technical Feasibility is examined in two sections as Software Feasibility and Hardware Feasibility.

3.2.1 Hardware Feasibility

The hardware features that will be required in the development of the system are given in the table below.

Table 3.1 Minimum Hardware Requirements

Properties	System Requirements
RAM	1GB
Processor Speed	1.8 GHz
Disk size	1GB
Required Software	PyCharm

3.2.2 Software Feasibility

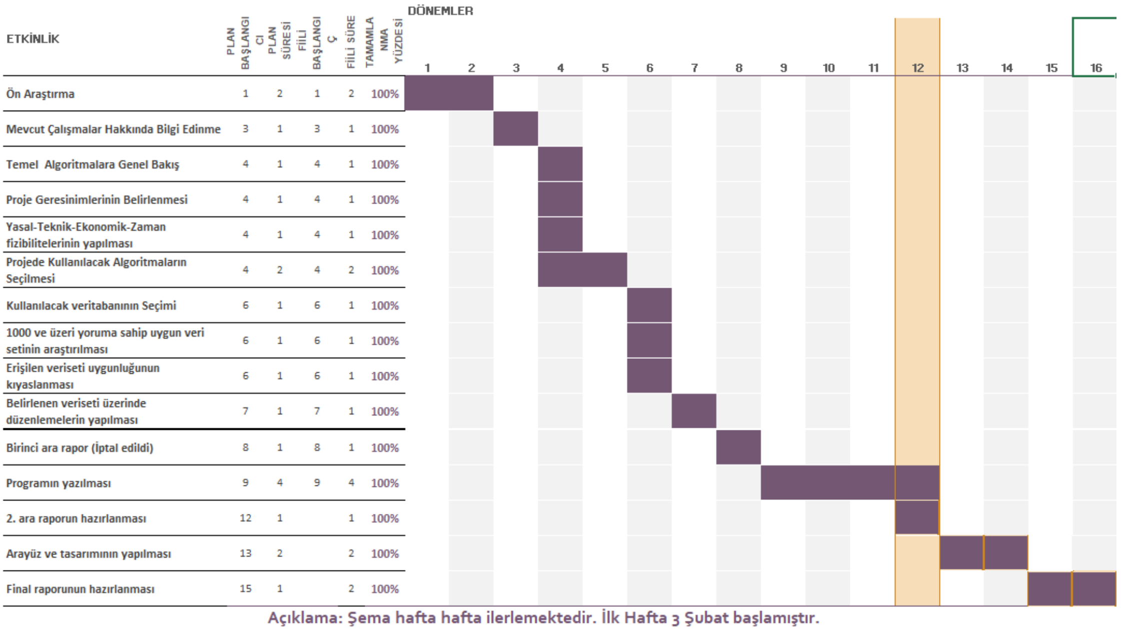
As a programming language, Python will be used in in the development of the software. The reason why Python is preferred is because it is supported by most operating systems, and the ease and possibilities it provides in the development of the system.

3.3 Economic Feasibility

The programs, databases and libraries that will be used in the development of the project are not expected to generate any costs since they are all open source. It is expected that the cost of the two engineers, who will work for 4 months during this period, will be 48.000 Turkish Liras.

3.4 Workforce and Time Schedule

Gantt chart (below) shows the plans of the works to be done according to the weeks during the semester.



ETKİNLİK	PLAN BAŞLANGICI	PLAN SÜRESİ	FİİLİ BAŞLANGIÇ	FİİLİ SÜRE	TAMAMLANMA YÜZDESİ	DÖNEMLER 1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Ön Araştırma	1	2	1	2	100%	■	■														
Mevcut Çalışmalar Hakkında Bilgi Edinme	3	1	3	1	100%			■													
Temel Algoritmalara Genel Bakış	4	1	4	1	100%				■												
Proje Geresinimlerinin Belirlenmesi	4	1	4	1	100%				■												
Yasal-Teknik-Ekonomik-Zaman fizibilitelerinin yapılması	4	1	4	1	100%				■												
Projede Kullanılacak Algoritmaların Seçilmesi	4	2	4	2	100%				■	■											
Kullanılacak veritabanının Seçimi	6	1	6	1	100%						■										
1000 ve üzeri yoruma sahip uygun veri setinin araştırılması	6	1	6	1	100%						■										
Erişilen veriseti uygunluğunun kıyaslanması	6	1	6	1	100%						■										
Belirlenen veriseti üzerinde düzenlemelerin yapılması	7	1	7	1	100%							■									
Birinci ara rapor (İptal edildi)	8	1	8	1	100%								■								
Programın yazılması	9	4	9	4	100%									■	■	■	■				
2. ara raporun hazırlanması	12	1		1	100%												■				
Arayüz ve tasarımının yapılması	13	2		2	100%													■	■		
Final raporunun hazırlanması	15	1		2	100%															■	■

Açıklama: Şema hafta hafta ilerlemektedir. İlk Hafta 3 Şubat başlamıştır.

Figure 3.1 Gantt Chart

4
System Analysis

4.1 Use-Case Scenario

Analysis of Facility Reviews

Primary Actors: Admin, System, Client

Requirements: Customer to have a set of reviews about the product/service

1. Client (preferably a hotel/restaurant owner) allows admin to access the database that has the reviews he wants to be analyzed.
2. Admin, after having connected to the database, copies the data needed for the process to another database not to cause any corruption on the original data.
3. Admin connects to this database through the system and the data is transferred to the Python.
4. After the necessary arrangements, the data is made ready to enter the algorithm.
5. Processed data is analyzed by the algorithm and the result is obtained as a report
6. Admin reports the obtained results to the client.

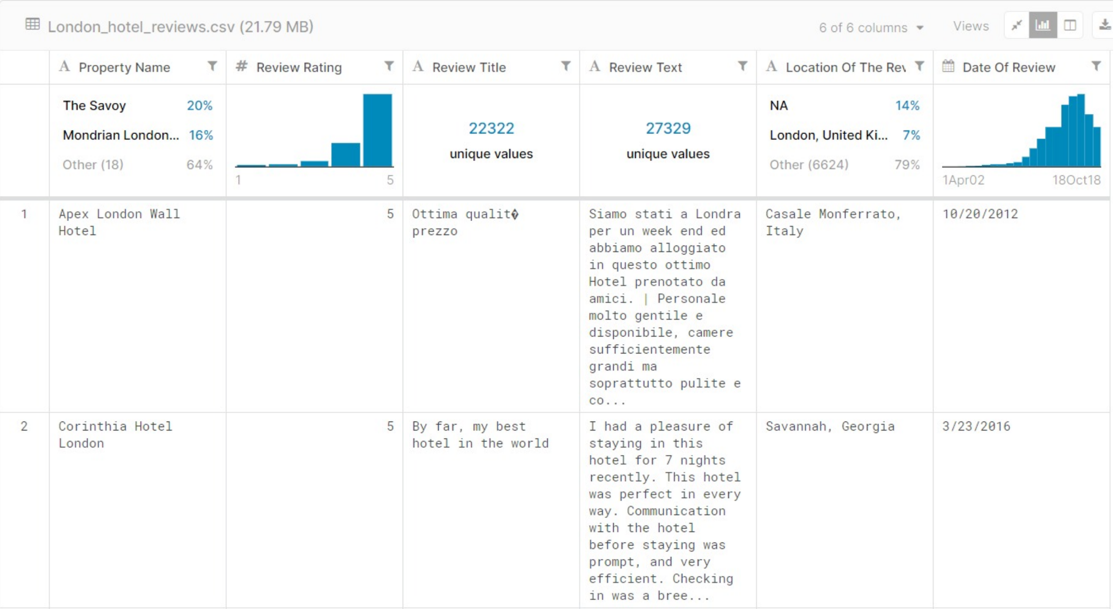
5
System Design

5.1 Software Design

The system is implemented with Python as the back-end programming language, Django as the web framework, a csv as the dataset, and the built-in server of the Django framework. The front layer of the application is built using HTML,CSS and various JavaScript libraries.

5.2 Database Design

Due to the structure of the project, a ready-to-use dataset containing tens of thousands of reviews must be used. Therefore, as explained before in the introduction chapter, the perfect dataset is found at kaggle. You can see the mentioned dataset's preview image below.



London_hotel_reviews.csv (21.79 MB)

	Property Name	Review Rating	Review Title	Review Text	Location Of The Rev	Date Of Review
	The Savoy 20% Mondrian London... 16% Other (18) 64%	1 5	22322 unique values	27329 unique values	NA 14% London, United Ki... 7% Other (6624) 79%	1Apr02 18Oct18
1	Apex London Wall Hotel	5	Ottima qualit� prezzo	Siamo stati a Londra per un week end ed abbiamo alloggiato in questo ottimo Hotel prenotato da amici. | Personale molto gentile e disponibile, camere sufficientemente grandi ma soprattutto pulite e co...	Casale Monferrato, Italy	10/20/2012
2	Corinthia Hotel London	5	By far, my best hotel in the world	I had a pleasure of staying in this hotel for 7 nights recently. This hotel was perfect in every way. Communication with the hotel before staying was prompt, and very efficient. Checking in was a bree...	Savannah, Georgia	3/23/2016

Figure 5.1 Dataset Preview

5.2.1 Information about the Dataset

The dataset to be used:

1. consists of twenty-seven thousand lines
2. has the size of 22 megabytes.
3. the reviews aren't divided into two groups such as positive-negative
4. the file is in .csv format

The counts of reviews based on review score given in the database are shown below.

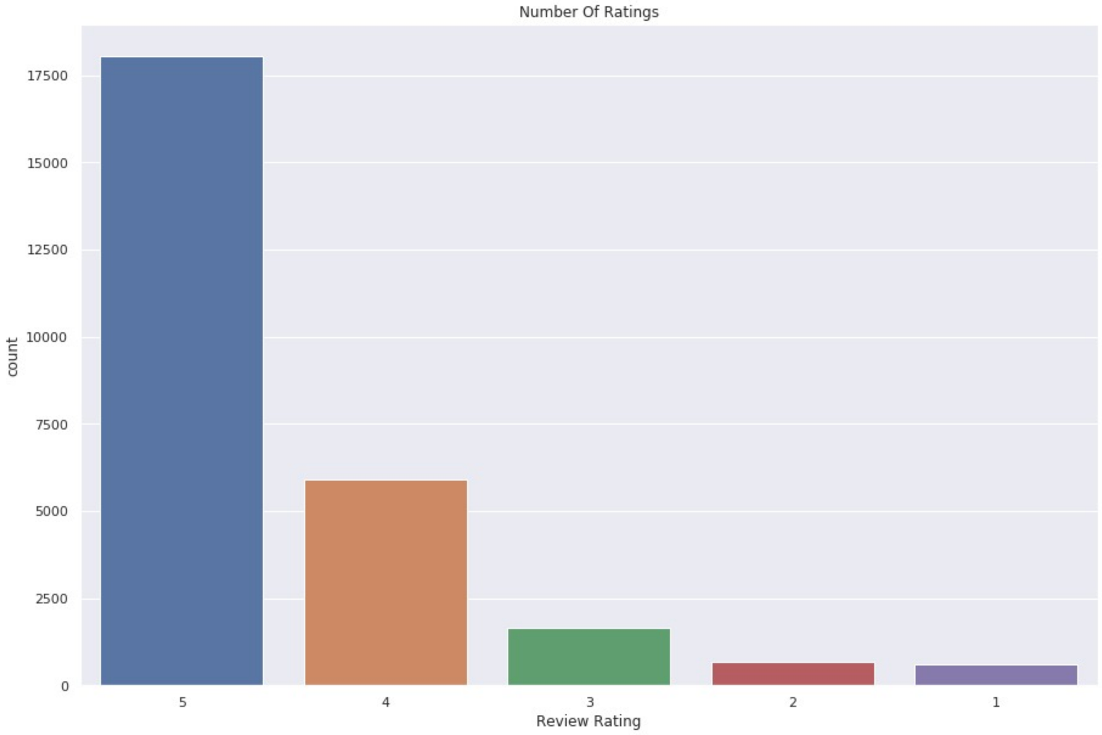


Figure 5.2 Distribution of Reviews by Score

There are 6 columns in the table. The names and descriptions of some of these columns are as follows.

- Property Name: Name of property
- Review Rating: Score the reviewer has given to the hotel, based on his/her experience
- Review Title: Title of the review that may give a hint about reviewer's experience whether it's positive or negative

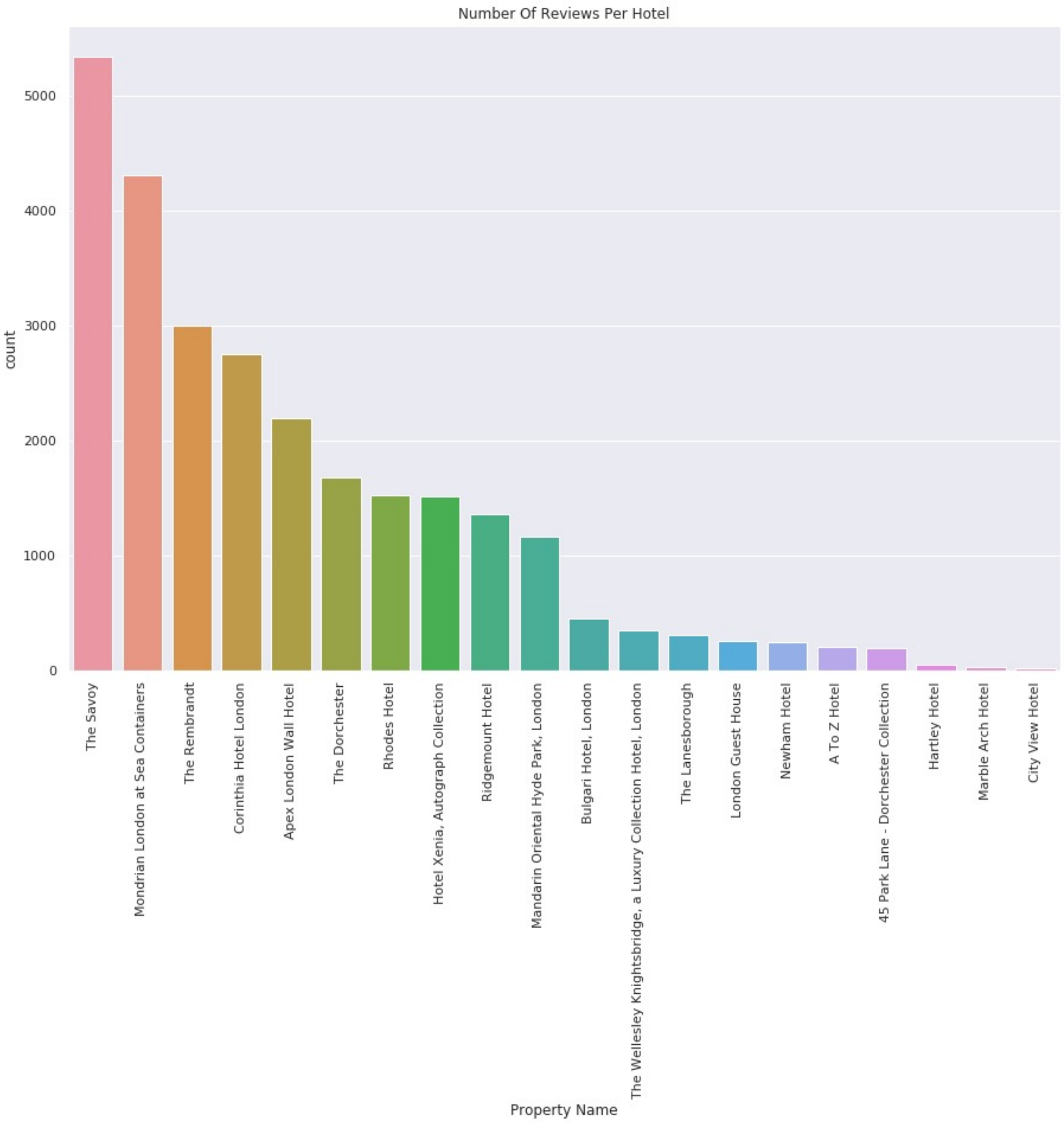


Figure 5.3 Distribution of Reviews by Hotel

- Review Text: The review that the reviewer gave to the hotel.
- Location Of The Reviewer: The location of the reviewer (not to be mistaken with location of the hotel)
- Date Of Review: The date when reviewer posted the corresponding review.

5.3 Input-Output Design

The web application we will prepare within the development of this project takes the name of the hotel & restaurant -whose reviews need to be analyzed- as input. During the process, it's assumed that the reviews are transferred to our database properly. The process of analysis begins. As an output, the ratio of positive reviews to negative reviews is expected with graphical charts that makes it even easier to analyze for the owner.

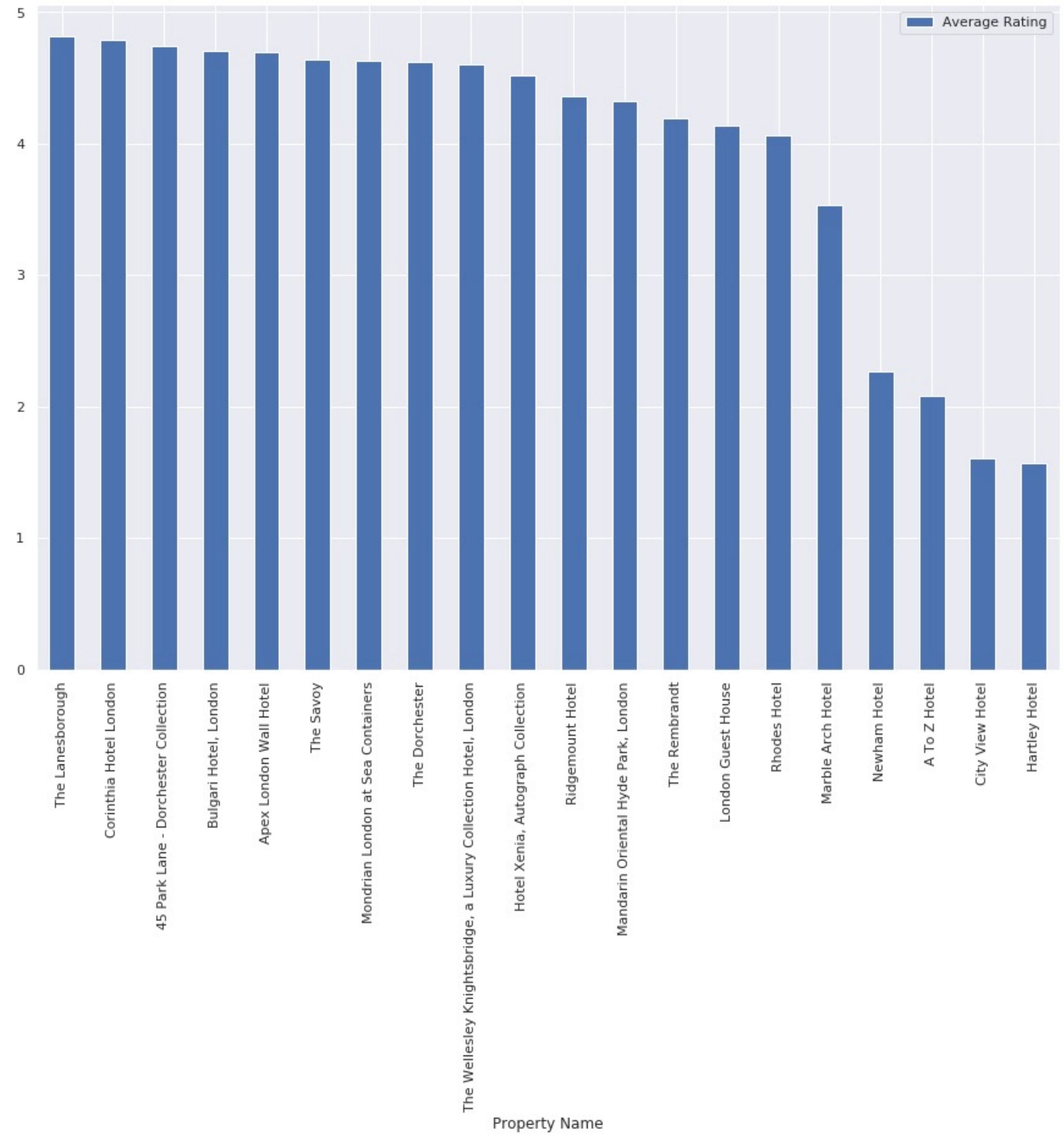
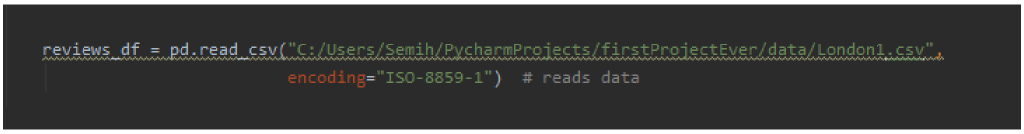


Figure 5.4 Average Scores of Reviews by Hotel

6
Implementation

In the application section, the dataset to be worked on should be in the .csv extension format. The location of the dataset on the computer should be specified in the code section as in figure 6.1 . Then, with the help of the web application designed, analysis can be started after selecting the hotel.

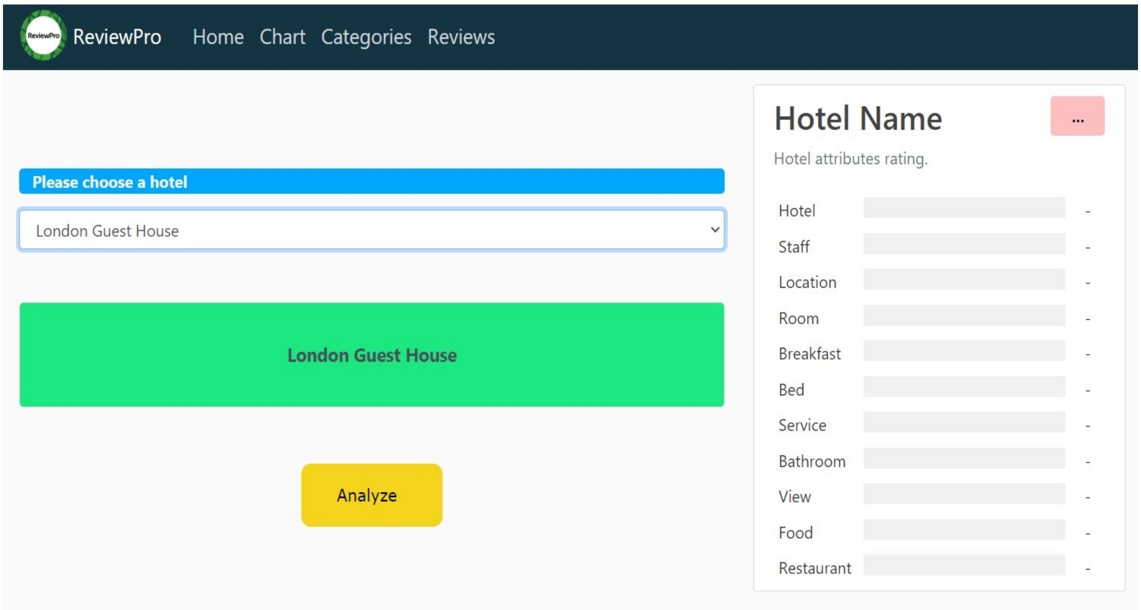


```
reviews_df = pd.read_csv("C:/Users/Semih/PycharmProjects/firstProjectEver/data/London1.csv",
                         encoding="ISO-8859-1")  # reads data
```

Figure 6.1 Setting dataset location

6.1 Application Screen

The list of hotels appears on the application screen. After the hotel selection, the reviews of the selected hotel are being processed.



Please choose a hotel

London Guest House

London Guest House

Hotel Name

Hotel attributes rating.

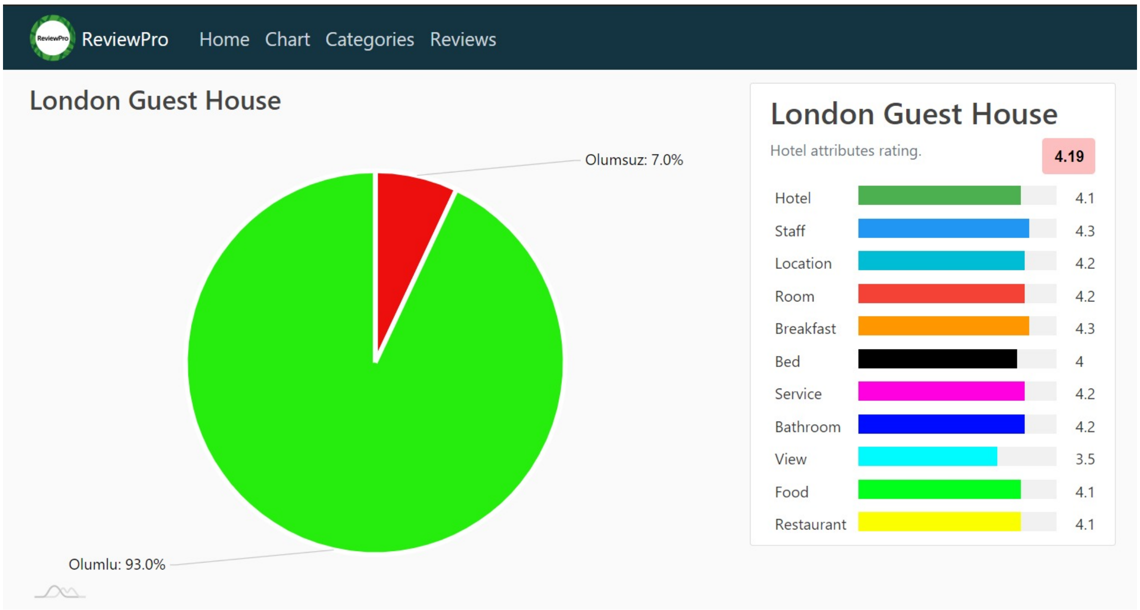
Hotel	-
Staff	-
Location	-
Room	-
Breakfast	-
Bed	-
Service	-
Bathroom	-
View	-
Food	-
Restaurant	-

Figure 6.2 Hotel Selection

6.2 Analysis Result Screen

After selecting the hotel, the process begins by pressing the "Analyze" button. Depending on the size of the dataset, it may take 1 to 5 minutes to run in the background. Following the end of the process, the analysis is displayed on the screen.

Percentages of positive reviews and negative reviews of the selected hotel are shown in the chart. The average score given for the attributes of the hotel are also shown in this screen.



London Guest House

Hotel attributes rating. 4.19

Hotel	4.1
Staff	4.3
Location	4.2
Room	4.2
Breakfast	4.3
Bed	4
Service	4.2
Bathroom	4.2
View	3.5
Food	4.1
Restaurant	4.1

Figure 6.3 Analysis Result

6.3 Analysis Result by Categories

For a better analysis of reviews per each attribute, the analysis result by categories are shown here in chart forms. By a quick look, the owner can detect the most liked or most disliked attributes of the hotel. Clicking on the chart of an attribute redirects you to the page where you can see the reviews only related with that attribute.

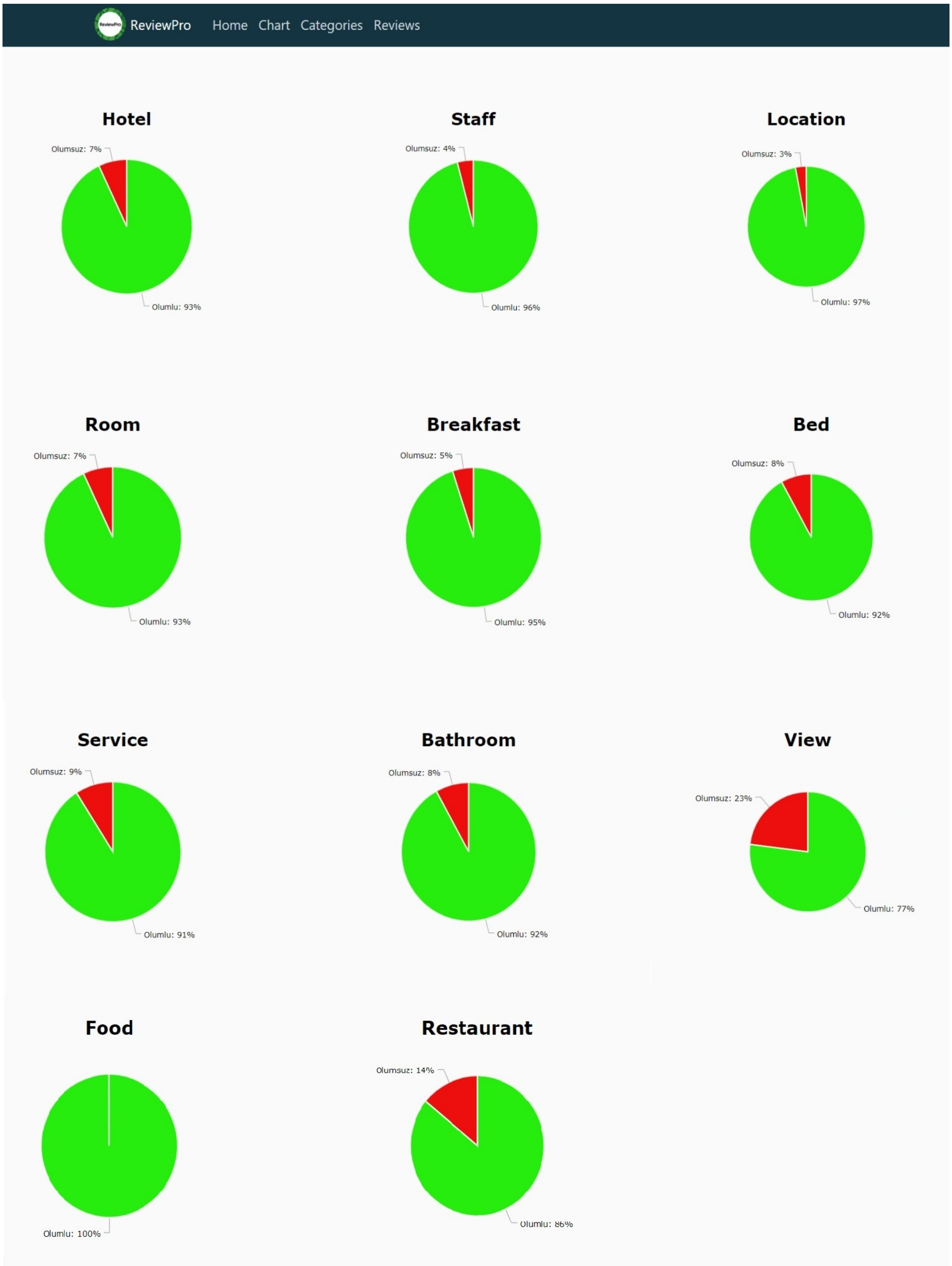


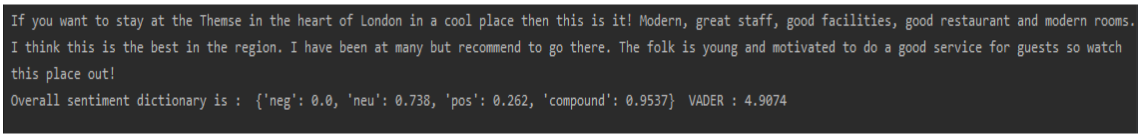
Figure 6.4 Analysis Result by Categories

7
Experimental Results

The tests for our application were conducted on the test dataset which contains reviews of the top 10 most expensive and least expensive London-based hotels. The dataset has 28 thousands positive-negative mixed reviews of nearly 20 London-based hotels.

7.1 How VADER Works

Let’s check how VADER performs on a review in figure 7.1



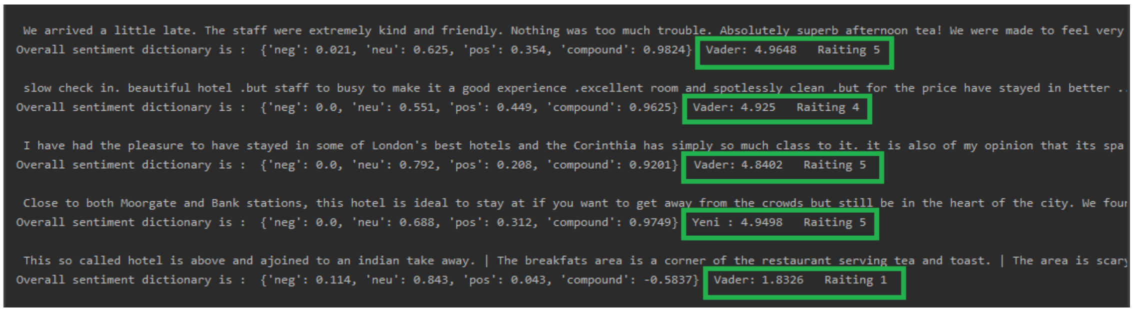
```
If you want to stay at the Themse in the heart of London in a cool place then this is it! Modern, great staff, good facilities, good restaurant and modern rooms.
I think this is the best in the region. I have been at many but recommend to go there. The folk is young and motivated to do a good service for guests so watch
this place out!
Overall sentiment dictionary is :  {'neg': 0.0, 'neu': 0.738, 'pos': 0.262, 'compound': 0.9537}  VADER : 4.9074
```

Figure 7.1 How VADER Scores

The positive, negative and neutral scores represent the proportion of text that falls in these categories. This means our sentence was rated as 26% positive, 73% neutral and 0% negative. Hence all these should add up to 1.

The compound score is a metric that calculates the sum of all the lexicon ratings which have been normalized between -1 (most negative) and +1 (most positive). In the case above, the compound score turns out to be 0.95 , denoting a very high positive sentiment.

By applying VADER on multiple views and comparing the VADER scores with user scores, we observed that VADER results are so close to scores given by users (shown in figure 7.2). That was a satisfying result that shows how correct VADER works.

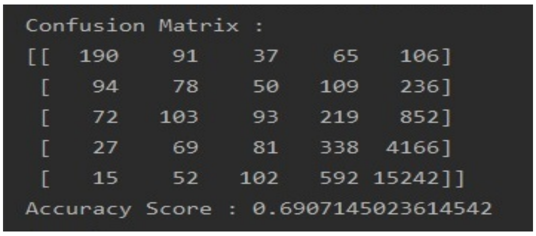


```
 We arrived a little late. The staff were extremely kind and friendly. Nothing was too much trouble. Absolutely superb afternoon tea! We were made to feel very
Overall sentiment dictionary is :  {'neg': 0.021, 'neu': 0.625, 'pos': 0.354, 'compound': 0.9824} Vader: 4.9648   Raiting 5

 slow check in. beautiful hotel .but staff to busy to make it a good experience .excellent room and spotlessly clean .but for the price have stayed in better ..
Overall sentiment dictionary is :  {'neg': 0.0, 'neu': 0.551, 'pos': 0.449, 'compound': 0.9625} Vader: 4.925   Raiting 4

 I have had the pleasure to have stayed in some of London's best hotels and the Corinthia has simply so much class to it. it is also of my opinion that its spa
Overall sentiment dictionary is :  {'neg': 0.0, 'neu': 0.792, 'pos': 0.208, 'compound': 0.9201} Vader: 4.8402   Raiting 5

 Close to both Moorgate and Bank stations, this hotel is ideal to stay at if you want to get away from the crowds but still be in the heart of the city. We four
Overall sentiment dictionary is :  {'neg': 0.0, 'neu': 0.688, 'pos': 0.312, 'compound': 0.9749} Yeni : 4.9498   Raiting 5

 This so called hotel is above and ajoined to an indian take away. | The breakfats area is a corner of the restaurant serving tea and toast. | The area is scary
Overall sentiment dictionary is :  {'neg': 0.114, 'neu': 0.843, 'pos': 0.043, 'compound': -0.5837} Vader: 1.8326   Raiting 1
```

Figure 7.2 Comparison of VADER Scores with User Scores

7.2 Success Test

There are many parameters we need to check when determining the accuracy of VADER such as accuracy score, f1-score, recall etc. For checking the accuracy score, we need to check on confusion matrix which shows us the comparison between the stars given by users and the stars we found using VADER. As it's very detailed, we can say accuracy rate in the confusion matrix is around 69%



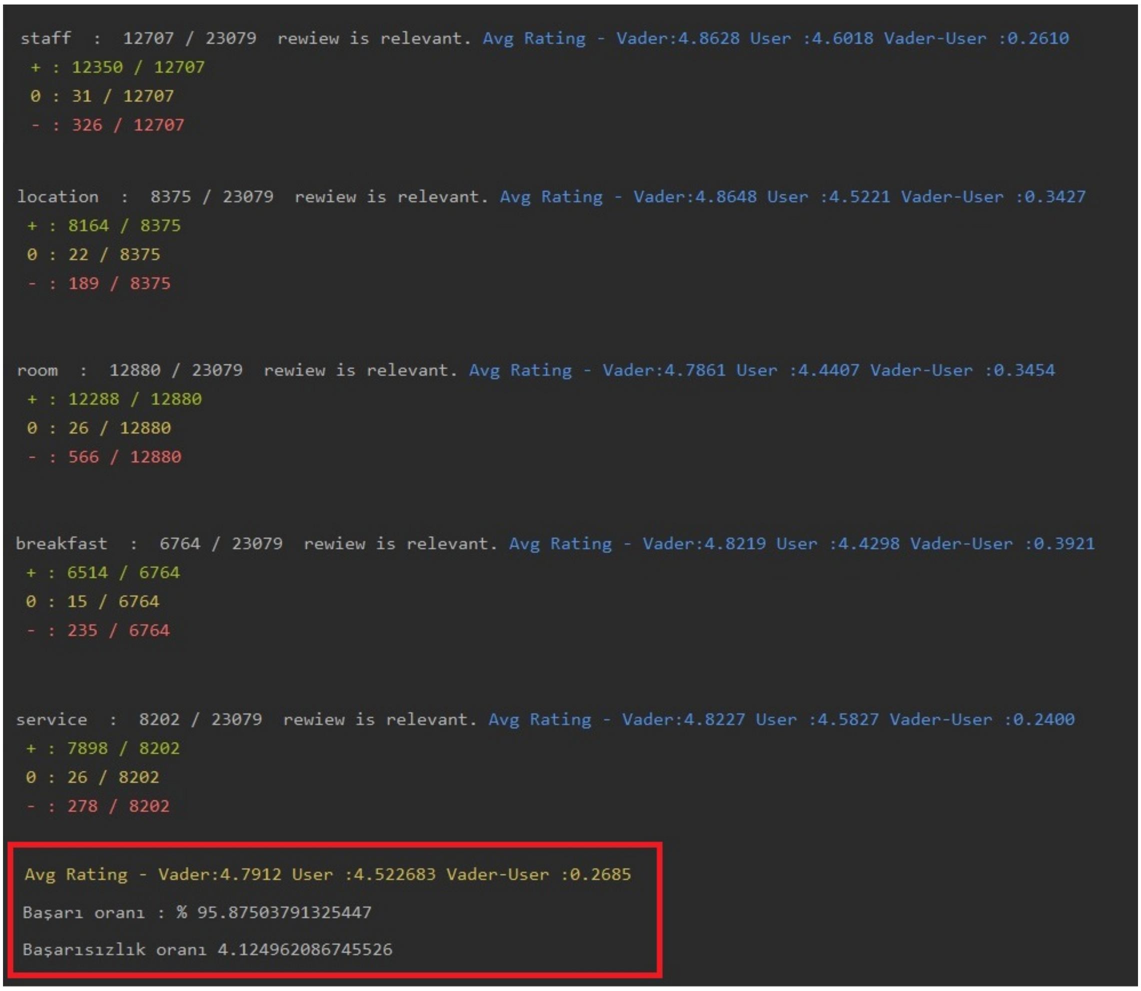
```
Confusion Matrix :
[[  190    91    37    65   106]
 [   94    78    50   109   236]
 [   72   103    93   219   852]
 [   27    69    81   338  4166]
 [   15    52   102   592 15242]]
Accuracy Score : 0.6907145023614542
```

Figure 7.3 Confusion Matrix

On the other hand, for most of the times, reviews in sentiment analysis consist of 3 main groups which are positive, negative or notr. For making such a grouping, we decided that;

- Stars between 0 - 2.7 equal to negative
- Stars between 2.7 - 4.0 equal to notr
- Stars between 4.0 - 5.0 equal to positive

As mentioned before, VADER not only provides positive and negative information of the reviews, but also the polarity - how positive or how negative the review is. Using that benefit with such group organizing mentioned above, it was observed that the accuracy rate is over 95%



```
staff  :  12707 / 23079  rewiew is relevant. Avg Rating - Vader:4.8628 User :4.6018 Vader-User :0.2610
 + : 12350 / 12707
 0 : 31 / 12707
 - : 326 / 12707


location  :  8375 / 23079  rewiew is relevant. Avg Rating - Vader:4.8648 User :4.5221 Vader-User :0.3427
 + : 8164 / 8375
 0 : 22 / 8375
 - : 189 / 8375


room  :  12880 / 23079  rewiew is relevant. Avg Rating - Vader:4.7861 User :4.4407 Vader-User :0.3454
 + : 12288 / 12880
 0 : 26 / 12880
 - : 566 / 12880


breakfast  :  6764 / 23079  rewiew is relevant. Avg Rating - Vader:4.8219 User :4.4298 Vader-User :0.3921
 + : 6514 / 6764
 0 : 15 / 6764
 - : 235 / 6764


service  :  8202 / 23079  rewiew is relevant. Avg Rating - Vader:4.8227 User :4.5827 Vader-User :0.2400
 + : 7898 / 8202
 0 : 26 / 8202
 - : 278 / 8202

 Avg Rating - Vader:4.7912 User :4.522683 Vader-User :0.2685

 Başarı oranı : % 95.87503791325447

 Başarısızlık oranı 4.124962086745526
```

Figure 7.4 Detailed Result of VADER

7.3 Performance Test

The test results given were created by combining the data obtained as a result of running the application on two different devices. Information for each device and the result of their measurements are given in the table 7.1 below. Operating time may change depending upon computer's current CPU usage, applications running at background etc.

Table 7.1 Information about devices and results

Criterias	**Values**	
Device	Device 1	Device 2
Operating System	Windows 10	Windows 10
Processor	Intel i5	Intel i7
Processor Speed	1.8 GHz	2.4 GHz
The number of reviews tested	5 thousand	5 thousand
Operating time for the test	1 min 54 seconds	1 min 14 seconds

8
Performance Analysis

In this chapter, experimental results in chapter 7 will be examined in more detail. We conducted our tests for different models, one of which was tested with twenty-seven thousand reviews. Test results will be examined by presenting the accuracy results.

8.1 Determination of Hotel Attributes

The first job was to determine the attributes to be paid attention about the hotel. While determining this, the following items were taken into consideration.

- What are the most mentioned words in the reviews?
- Are these determined attributes associated with a service or an object related with hotels?

Considering these two items, the following words were determined as hotel attributes. The number of times used per each attribute is found out by using WordCount module.

1. staff : 18214
2. location : 8806
3. room : 39174
4. hotel : 4392
5. breakfast : 12217
6. bed : 6015
7. service : 13370
8. bathroom : 5377

9. restaurant: 7014

10. view : 4586

11. food : 5368

8.2 Implementation of Word2Vec Model

These words were sent to the Word2Vec algorithm and the words with closest meaning are detected.

1. Most similar to ['staff'] [('personnel' 0.63), ('approachable' 0.56), ('everyone' 0.52), ('employee' 0.52), ('team' 0.50), ('informative' 0.49), ('incredibly' 0.46), ('professional' 0.45), ('genuinely' 0.44), ('unobtrusive' 0.43)]

2. Most similar to ['location'] [('position' 0.70), ('theatreland' 0.45), ('locate' 0.43), ('situate' 0.43), ('attraction' 0.43), ('center' 0.42), ('proximity' 0.39), ('multilingual' 0.38), ('buss' 0.38), ('museums' 0.38)]

3. Most similar to ['room'] [('bedroom' 0.52), ('double' 0.48), ('executive' 0.45), ('bed' 0.44), ('bathroom' 0.43), ('ensuite' 0.41), ('functional' 0.41), ('deluxe' 0.40), ('superior' 0.40), ('sufficient' 0.40)]

4. Most similar to ['hotel'] [('property' 0.67), ('place' 0.55), ('establishment' 0.50), ('reviens' 0.50), ('however' 0.49), ('accommodation' 0.49), ('consentono' 0.49), ('reason' 0.46), ('london' 0.46), ('building' 0.45)

5. Most similar to ['breakfast'] [('cereal' 0.51), ('freshly' 0.49), ('continental' 0.48), ('buffet' 0.48), ('alacarte' 0.45), ('variety' 0.45), ('plentiful' 0.45), ('eggs' 0.44), ('omelet' 0.44), ('healthy' 0.43)]

6. Most similar to ['bed'] [('mattress' 0.61), ('pillow' 0.58), ('bedding' 0.55), ('divinely' 0.52), ('cushion' 0.52), ('soundly' 0.51), ('roomy' 0.51), ('chair' 0.51), ('duvet' 0.50), ('squishy' 0.49)]

7. Most similar to ['service'] [('sevice' 0.47), ('presentation' 0.40), ('approachable' 0.39), ('consistently' 0.37), ('server' 0.36), ('focus' 0.35), ('ambiance' 0.34), ('skill' 0.34), ('attentiveness' 0.34), ('staff' 0.33)]

8. Most similar to ['bathroom'] [('bathrooms' 0.72), ('bath' 0.65), ('bathtub' 0.60), ('tub' 0.56), ('rainfall' 0.54), ('linen' 0.52), ('cubicle' 0.52), ('dressing' 0.52), ('furnishing' 0.51), ('loccitane' 0.51)]

9. Most similar to ['view'] [('overlook' 0.70), ('facing' 0.62), ('veiw' 0.59), ('veiws' 0.56), ('topiary' 0.50), ('vista' 0.49), ('glimpse' 0.48), ('face' 0.45), ('partial' 0.44), ('ferry' 0.43)]

10. Most similar to ['food'] [('dish' 0.57), ('meal' 0.57), ('cuisine' 0.55), ('risotto' 0.54), ('seafood' 0.53), ('burger' 0.50), ('menu' 0.50), ('massimo' 0.50), ('presentation' 0.49), ('sole' 0.49)]

11. Most similar to ['restaurant'] [('restaurants' 0.58),('boulud' 0.54) , ('eatery' 0.54), ('restuarant' 0.51), ('resturant' 0.50), ('resturants' 0.49), ('pierino' 0.47), ('massimo' 0.47), ('lebanese' 0.46), ('cafe' 0.45)]

8.3 Implementation of Fasttext Model

With the help of Fasttext model, linguistically most similar words to attributes are detected.

1. Most similar to ['hotel'] [('hotels' 0.92), ('otel' 0.87), ('whatahotel' 0.86), ('hotelrooms' 0.86), ('hotelier' 0.76), ('motel' 0.76), ('hotelroom' 0.73), ('rhodeshotel' 0.71), ('hote' 0.70), ('property' 0.69)]

2. Most similar to ['staff'] [('staffed' 0.91), ('staffer' 0.87), ('naff' 0.84), ('staf' 0.84), ('barstaff' 0.84), ('waitstaff' 0.80), ('stafford' 0.80), ('quaff' 0.76), ('doorstaff' 0.76), ('raff' 0.68)]

3. Most similar to ['location'] [('allocation' 0.91), ('localization' 0.81), ('position' 0.76), ('education' 0.75), ('occation' 0.75), ('cation' 0.74), ('locate' 0.73), ('located' 0.72), ('staycation' 0.72), ('disposition' 0.71)]

4. Most similar to ['room'] [('rooom' 0.95), ('roomy' 0.93), ('zoom' 0.90), ('inroom' 0.89), ('roomier' 0.88), ('roooms' 0.87), ('broom' 0.87), ('wetroom' 0.85), ('groom' 0.84), ('badroom' 0.81)]

5. Most similar to ['breakfast'] [('breakfats' 0.95), ('breakfat'0.94), ('brekfast' 0.93), ('breakfeast' 0.91), ('breafast' 0.89), ('breakast' 0.86), ('breakfest' 0.84), ('bfast' 0.78), ('breakdown' 0.77), ('breakout' 0.77)]

6. Most similar to ['bed'] [('bedded' 0.94), ('bedbugs' 0.93), ('beds' 0.92), ('bedbug' 0.91), ('robbed' 0.88), ('fobbed' 0.87), ('bedeck' 0.85), ('grabbed' 0.83), ('bedsheets' 0.83), ('bedskirt' 0.81)]

7. Most similar to ['service'] [('serviced' 0.95), ('servico' 0.95), ('serviceminded' 0.94), ('seervice' 0.93), ('disservice' 0.93), ('servicing' 0.91), ('roomservice' 0.88), ('serviceable' 0.86), ('serving' 0.83), ('setvice' 0.79)]

8. Most similar to ['bathroom'] [('bathrooom' 0.99), ('bathrooms' 0.98), ('bathrom' 0.95), ('bathroon' 0.93), ('batrooms' 0.89), ('bathrobe' 0.87), ('bathrobes' 0.87), ('baths' 0.86), ('bathe' 0.84), ('washroom' 0.84)]

9. Most similar to ['view'] [('views' 0.99), ('vieuw' 0.97), ('vie' 0.93), ('viewed' 0.92), ('viewing' 0.91), ('vienna' 0.86), ('overview' 0.84), ('viewpoint' 0.84), ('vi' 0.80), ('vii' 0.79)]

10. Most similar to ['food'] [('foodies' 0.95), ('foodie' 0.95), ('seafood' 0.84), ('fod' 0.84), ('foodhall' 0.79), ('meal' 0.70), ('hood' 0.65), ('menu' 0.64), ('mood' 0.63), ('oatmeal' 0.62)]

11. Most similar to ['restaurant'] [('restaurants' 0.99), ('restaurante' 0.98), ('restauraunt' 0.97), ('restauarant' 0.96), ('restauarants' 0.95), ('resaurant' 0.95), ('restraurants' 0.92), ('resturant' 0.91), ('restaurent' 0.90), ('reataurants' 0.90)]

8.4 Combination of Word2Vec and Fasttext Model

1. Most similar to ['hotel'] [('property' 0.59), ('accommodation' 0.50), ('place' 0.45), ('establishment' 0.45), ('hotels' 0.43), ('accomodation' 0.39), ('reason' 0.37), ('london' 0.35), ('building' 0.34), ('dame' 0.34)]

2. Most similar to ['staff'] [('personnel' 0.67), ('everyone' 0.52), ('employee' 0.51), ('informative' 0.50), ('team' 0.49), ('approachable' 0.49), ('chatty' 0.48), ('professional' 0.46), ('unobtrusive' 0.46), ('cheerful' 0.46)]

3. Most similar to ['location'] [('position' 0.72), ('locate' 0.44), ('situate' 0.40), ('attraction' 0.40), ('proximity' 0.38), ('spot' 0.38), ('center' 0.38), ('exceptionnal' 0.37), ('neighborhood' 0.37), ('theatreland' 0.36)]

4. Most similar to ['room'] [('bedroom' 0.49), ('double' 0.45), ('comfortably' 0.45), ('bathroom' 0.45), ('sufficiently' 0.43), ('functional' 0.43), ('roomy' 0.43), ('bed' 0.43), ('executive' 0.42), ('bedded' 0.42)]

5. Most similar to ['breakfast'] [('buffet' 0.50), ('cereal' 0.49), ('freshly' 0.49), ('eggs' 0.45), ('cooked' 0.45), ('plentiful' 0.45), ('continental' 0.44), ('croissant' 0.44), ('vegetable' 0.44), ('omelet' 0.43)]

6. Most similar to ['bed'] [('mattress' 0.60), ('pillow' 0.58), ('bedding' 0.56), ('couch' 0.50), ('soundly' 0.49), ('chair' 0.48), ('topper' 0.48), ('cushion' 0.47), ('roomy' 0.46), ('armchair' 0.46)]

7. Most similar to ['service'] [('sevice' 0.48), ('presentation' 0.42), ('consistently' 0.41), ('skill' 0.39), ('rajat' 0.39), ('approachable' 0.35), ('competent' 0.35), ('focus' 0.35), ('server' 0.34), ('ambiance' 0.34)]

8. Most similar to ['bathroom'] [('bathrooms' 0.71), ('bath' 0.62), ('bathtub' 0.60), ('tub' 0.54), ('washroom' 0.53), ('rainfall' 0.53), ('closet' 0.52), ('vanity' 0.52), ('shower' 0.51), ('furniture' 0.51)]

9. Most similar to ['view'] [('overlook' 0.72), ('facing' 0.63), ('veiw' 0.63), ('veiws' 0.59), ('glimpse' 0.50), ('clipper' 0.47), ('face' 0.47), ('vista' 0.46), ('partial' 0.45), ('overlooked' 0.44)]

10. Most similar to ['food'] [('meal' 0.59), ('cuisine' 0.56), ('dish' 0.55), ('risotto' 0.54), ('burger' 0.53), ('massimo' 0.52), ('sole' 0.52), ('steak' 0.52), ('seafood' 0.52), ('ingredient' 0.51)]

11. Most similar to ['restaurant'] [('boulud' 0.57), ('restaurants' 0.56), ('resturant' 0.53), ('eatery' 0.52), ('boloud' 0.52), ('lebanese' 0.50), ('restuarant' 0.50), ('massimo' 0.50), ('resturants' 0.50), ('blumenthal' 0.46)]

WV2 and Fasttex were used to learn other words that were used by users in the same sense similar to the hotel attributes we have previously determined, and to identify the words that users make misspelling but mean that attribute. W2V mostly focuses on the words that users use in a similar way. Fasttext, on the other hand, focuses more on spelling mistakes such as a user typing "brekfast" instead of "breakfast"

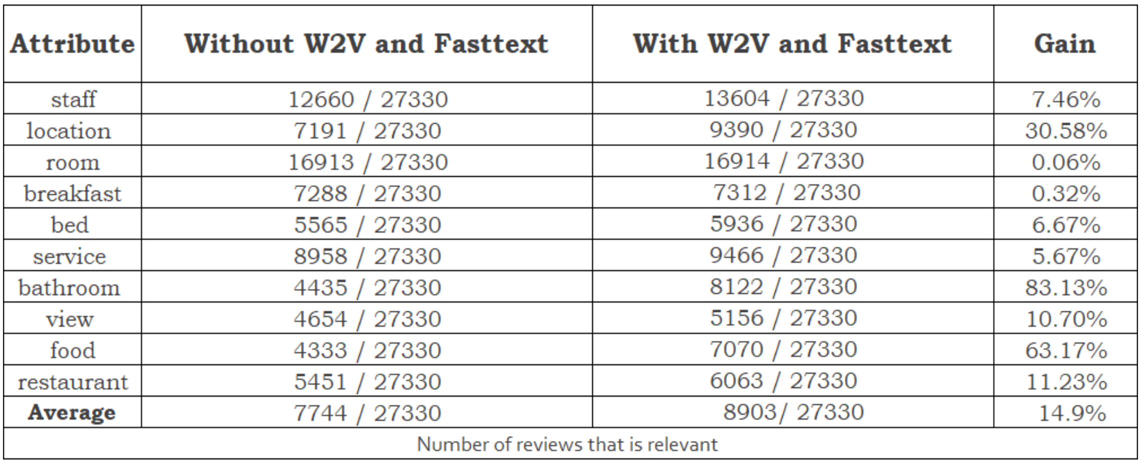
Our project, which consists of two separate programs, finds properties similar to hotel features with the help of W2V and FastText in the first program. These similar attributes are compared and written to a .txt file providing that the duplicate ones are removed. Then the 2nd program pulls the hotel attributes from the txt file to use for the next process. Content of txt file is shown below.

1. hotel, hotels, property, accommodation, place, accomodation, establishment
2. staff, staffer, staf, doorstaff, personnel, everyone, employee, team, professional
3. location, position, located, locate, attraction, spot, situate, proximity, center
4. room, rooom, roomy, inroom, roooms, bedroom

5. breakfast, breakfats, brekfast, breakfeast, breakfest, bfast, cereal, eggs, buffet, cooked, continental
6. bed, bedded, beds, robbed, bedsheets, mattress, pillow, bedding, duvet, chair, cushion
7. service, serviced, seervice, servicing, roomservice, sevice, presentation, approachable, competent
8. bathroom, bathrooom, bathrooms, bathrom, washroom, bath, bathtub, tub, rainfall, furniture, closet
9. view, views, viewed, viewpoint, overlook, facing, veiw, vista, veiws, ferry, partial, glimpse
10. food, seafood, meal, menu, dish, risotto, burger, cuisine, massimo, ingredient, sole
11. restaurant, restaurants, resturant, boulud, restuarant, eatery, pierino, boloud, massimo, resturants

8.5 The Benefits of Using Word2Vec and Fasttext Model

In performance tests conducted on twenty-seven thousand comments, it was determined that using W2V and Fasttext model, we can analyze more reviews with an average rate of 15% and have better results for our analysis.



Attribute	Without W2V and Fasttext	With W2V and Fasttext	Gain
staff	12660 / 27330	13604 / 27330	7.46%
location	7191 / 27330	9390 / 27330	30.58%
room	16913 / 27330	16914 / 27330	0.06%
breakfast	7288 / 27330	7312 / 27330	0.32%
bed	5565 / 27330	5936 / 27330	6.67%
service	8958 / 27330	9466 / 27330	5.67%
bathroom	4435 / 27330	8122 / 27330	83.13%
view	4654 / 27330	5156 / 27330	10.70%
food	4333 / 27330	7070 / 27330	63.17%
restaurant	5451 / 27330	6063 / 27330	11.23%
Average	7744 / 27330	8903/ 27330	14.9%

Number of reviews that is relevant

Figure 8.1 Gain Stats of Using W2V and Fasttext

9
Results

The results we obtained within the scope of the project are examined under the following headings

9.1 Subject Of the Project

The subject of our project is to achieve sentiment analysis based on the reviews made on touristic places in social media and split them into categories, determine liked-disliked ratio for each category and report these.

For this purpose, three different methods have been used. These methods are Word2Vec, fastText for determining which category a review belongs to and Vader to obtain the polarity scores

9.2 Achievement

For thousands of reviews given, we managed to find polarity for each review by detecting if the sentence is positive,negative or notr. Then we classified them into related group of reviews.

When we compared the results we found to real data, we found out that our results' success rate was over 90%.

References

[1] M. Taboada, “Sentiment analysis: An overview from linguistics,” *Annual Review of Linguistics*, vol. 2, Feb. 2016. DOI: `10.1146/annurev-linguistics-011415-040518`.

[2] R. Alguliev and R. Aliguliyev, “Evolutionary algorithm for extractive text summarization,” *Intelligent Information Management*, vol. 1, no. 02, p. 128, 2009.

[3] E. Hovy, C.-Y. Lin, *et al.*, “Automated text summarization in summarist,” *Advances in automatic text summarization*, vol. 14, 1999.

[4] W. Ling, C. Dyer, A. W. Black, and I. Trancoso, “Two/too simple adaptations of word2vec for syntax problems,” in *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2015, pp. 1299–1304.

[5] P. Pandey. (2018). Simplifying sentiment analysis using vader in python, [Online]. Available: `https://medium.com/analytics-vidhya/simplifying-social-media-sentiment-analysis-using-vader-in-python-f9e6ec6fc52f` (visited on 06/30/2020).

Curriculum Vitae

FIRST MEMBER

Name-Surname: YUNUS EMRE DEMIR
Birthdate and Place of Birth: 18.09.1995, Istanbul
E-mail: yunusemredemir002@gmail.com
Phone: 0545 429 26 03
Practical Training: MEMTEKS Company Software Department

SECOND MEMBER

Name-Surname: SEMIH DURMAZ
Birthdate and Place of Birth: 17.11.1996, Sakarya
E-mail: semihdurmaz54@gmail.com
Phone: 0539 382 14 12
Practical Training: EXEDRA Software Developer Intern (2 months)
ID3 Java Developer Intern (6 months)

Project System Informations

System and Software: Microsoft Windows, PyCharm
Required RAM: 1GB
Required Disk: 1GB