ISTANBUL TECHNICAL UNIVERSITY
FACULTY OF COMPUTER AND INFORMATICS

Deep Hotel Recommender System Using Aspect-based Sentiment Analysis of Users' Reviews

Graduation Project Final Report

Ayşe Betül Çetin
150180730
Bilgehan Emiral
010160050

Department: Computer Engineering
Division: Computer Engineering

Advisor: Assoc. Prof. Dr. Yusuf Yaslan

January 2022

Statement of Authenticity

We hereby declare that in this study

1. all the content influenced from external references are cited clearly and in detail,
2. and all the remaining sections, especially the theoretical studies and implemented software/hardware that constitute the fundamental essence of this study is originated by my/our individual authenticity.

İstanbul, Ocak 2022

Ayşe Betül Çetin

Bilgehan Emiral

Deep Hotel Recommender System Using Aspect-based Sentiment Analysis of Users' Reviews

(SUMMARY)

Data science is a crucial growing field that is used for numerous areas and has applications such as pattern recognition, image recognition, recommendation engines, and personalization systems. The value of search engines has increased with the digitalization of industries such as e-commerce and entertainment and data science is making its presence felt in search engine optimization. There are numerous areas where recommendations are used and many different methods and technologies have been used to develop recommender systems. Deep learning is a type of machine learning that uses multiple layers of processing to extract progressively higher-level characteristics from input. It is based on artificial neural networks. Deep learning-based methods are most effective in recommendations as they improve the system's overall performance by obtaining state-of-the-art predictive accuracy. However, collaborative filtering is the most famous technique used in the recommneder systems that rely on the user-item interaction.

Besides, Aspect-Based Sentiment Analysis (ABSA) is a text analysis method that divides opinions into aspects and determines the sentiment associated with each. Aspect is a category, feature, or topic that is being talked about and sentiments are positive or negative opinions about a particular aspect. The goal of ABSA is to improve recommendations by analyzing user feedback.

Unlike previous works, we combine ABSA with multi-criteria collaborative filtering using deep learning. In this project, the Tripadvisor dataset was used. This dataset includes user evaluations of hotels. These evaluations consist of sub-ratings (value, service, location, rooms, sleep, cleanliness), review text, and overall rating. We use polarities obtained by aspect-based sentiment analysis from user reviews combined with the overall rating and sub-ratings in a deep neural network model to predict user-item interactions.

The model is composed of three parts: The first part includes Aspect-based Sentiment Analysis (ABSA) which is applied to the reviews, and polarities are extracted, then the final dataset is generated. In the second part, the user and item embeddings are concatenated and given as input to the Polarities and Sub-ratings Deep Neural Network (DNN) to predict the sub-ratings and polarities. Finally, in the third part, we give obtained criteria array as input to the Overall Score DNN to predict the overall score. Based on polarities the proposed approach leverages aspect-based user's opinion.

We compared our method with other known methods and analyzed the results. Since we started the project, we have tried algorithms on many datasets such as Amazon, MovieLens, but as a result, we decided to use the Tripadvisor dataset due to the rich user reviews it contains and we developed our model by experimenting on this dataset. The obtained results indicate that proposed approach performs significantly better on the TripAdvisor dataset.

Kullanıcı İncelemelerinin En Boy Tabanlı Duyarlılık Analizini Kullanan Deep Hotel Tavsiye Sistemi

(ÖZET)

Veri bilimi, çok sayıda alan için kullanılan ve Örüntü Tanıma, Görüntü Tanıma, Öneri motorları ve kişiselleştirme sistemleri gibi uygulamalara sahip, büyüyen ve çok önemli bir alandır. E-ticaret ve eğlence gibi sektörlerin dijitalleşmesi ile arama motorlarının önemi oldukça arttı. Veri bilimi, arama motoru optimizasyonunda varlığını hissettiriyor. Birçok farklı yöntem ve teknolojinin kullanılarak geliştirilen öneri sistemlerinin kullanıldığı çok sayıda alan vardır. Derin Öğrenme, girdiden aşamalı olarak daha yüksek seviyeli özellikleri çıkarmak için birden çok işleme katmanı kullanan bir tür makine öğrenimidir. Yapay sinir ağlarına dayanmaktadır. Derin öğrenmeye dayalı yöntemler, son teknoloji tahmine dayalı doğruluk elde ederek sistemin genel performansını iyileştirdikleri için en etkilidir. Bununla birlikte, işbirlikçi filtreleme, kullanıcı-öğe etkileşimine dayanan tavsiye edilen sistemlerde kullanılan en ünlü methoddur.

Bunların yanı sıra, Aspect-Based Sentiment Analysis (ABSA), görüşleri yönlere ayıran ve her biriyle ilişkili duyguyu belirleyen bir metin analizi yöntemidir. Görünüm, hakkında konuşulan bir kategori, özellik veya konudur ve duygular, belirli bir yön hakkında olumlu veya olumsuz görüşlerdir. ABSA'nın amacı, kullanıcı geri bildirimlerini analiz ederek önerileri iyileştirmektir.

Önceki çalışmalardan farklı olarak, derin öğrenmeyi kullanarak ABSA'yı Çok Kriterli İşbirlikçi Filtreleme ile birleştiriyoruz. Bu projede Tripadvisor veri seti kullandık. Bu veri seti, otellerin kullanıcı değerlendirmelerini içerir. Bu değerlendirmeler, alt derecelendirmelerden (değer, hizmet, konum, odalar, uyku, temizlik), inceleme metni ve genel derecelendirmeden oluşur. Kullanıcı-öğe etkileşimlerini tahmin etmek için derin bir sinir ağı modelinde genel derecelendirme ve alt derecelendirmelerle birleştirilmiş kullanıcı incelemelerinden görünüşe dayalı duygu analiziyle elde edilen kutupları kullanırız.

Model üç bölümden oluşmaktadır: İlk bölüm, incelemelere uygulanan Aspect-based Sentiment Analysis'i (ABSA) içerir ve polariteler çıkarılır, ardından nihai veri seti oluşturulur. İkinci bölümde, kullanıcı ve öğe yerleştirmeleri birleştirilir ve alt derecelendirmeleri ve polariteleri tahmin etmek için Kutuplar ve Alt Derecelendirmeler Derin Sinir Ağı'na (DNN) girdi olarak verilir. Son olarak, üçüncü bölümde, toplam puanı tahmin etmek için Genel Puan DNN'sine girdi olarak elde edilen ölçüt dizisini veriyoruz. Kutuplara dayalı olarak, önerilen yaklaşım, görünüşe dayalı kullanıcı görüşünden yararlanır.

Yöntemimizi bilinen diğer yöntemlerle karşılaştırdık ve sonuçları analiz ettik. Projeye başladığımızdan beri Amazon, MovieLens gibi pek çok veri setinde algoritmalar denedik ancak sonuç olarak içerdiği zengin kullanıcı yorumlarından dolayı Tripadvisor veri setini kullanmaya karar verdik ve modelimizi bu veri seti üzerinde deneyler yaparak geliştirdik. Elde edilen sonuçlar, önerilen yaklaşımın TripAdvisor veri setinde önemli ölçüde daha iyi performans gösterdiğini göstermektedir.

Contents

1 Introduction and Project Summary

The importance of search engines has increased tremendously with the digitalization of countless industries such as e-commerce and entertainment. Consequently, there are numerous areas where the recommendation is widely used. The variety of data that can feed recommendation systems is constantly increasing. In addition to product descriptions and information, user profile features, ratings, and reviews also contains important and useful data that can guide a recommendation system. Such data is used to identify similar users or similar products and recommend products based on these similarities. Recommendation systems are categorized as Collaborative, Content-based, Knowledge-based and hybrid in the literature. In the Content-based recommendation systems, items recommended by the system are determined according to similarity with the ones that the user liked in the past [1]. On the other hand, Collaborative filtering depends on similarities between users and makes predictions according to those similarities. Collaborative filtering approaches do not require any prior knowledge of users or items to produce recommendations since they rely on interactions between them [2]. Besides, the knowledge-based method pursues a knowledge-based approach to generating a suggestion, reasoning about which items satisfy the user's criteria using knowledge about users and products [3]. Combining these approaches is also possible and is known as hybrid systems. Deep learning has been reviving the recommendation architectures significantly and it is bringing more opportunities to improve the performance of recommendation models. The main factor of this success comes from capturing the non-linear and non-trivial user-item connections[4].

In a recommendation system, the use of features deduced from reviews and ratings strengthens the system. Such features are crucial to predict the rating of a user for a particular item. For hotel recommendation, the user's travel purpose is substantial, such as business, touristic and vacation, food preferences, room preferences, what he/she expects from a hotel, and whom he/she goes with (family, couple, single). Generally, multi criteria ratings are used to represent more complex representations, e.g., category rating of a item by users. Sub-ratings are ratings given to the item by the user for a certain category. In this study, the high null rate in the sub-ratings given by the users was a source of motivation for us to produce features from their reviews. Detailed data about the user's approach to the hotel is obtained from the reviews. For this reason, aspect-polarity pairs extracted from the reviews is used in the proposed model in order to enrich the collaborative filtering fed by the simple traditional rating. By aspects, we consider attributes or components of an entity (a food or staff, in this work). The polarity values represent the user's positive or negative interaction for that aspect.

In this project, we propose a novel solution for improving collaborative filtering model performance that combines aspect sentiment analysis and multi-criteria recommendation with deep learning. We use a deep neural network to discover the relationship between overall rating and sub-ratings with aspect polarities.

The main contributions of this project are summarized as follows:

- A novel deep multi-criteria collaborative filtering approach using aspect-based sentiment analysis of users' reviews was developed for hotel recommendation.

- The proposed model has demonstrated better results as the novel architecture is implemented that integrates aspect-based sentiment analysis and multi-criteria ratings.

The rest of this report is organized as follows: In Section 2, we analyze the related work. We introduce our Aspect-Based Sentiment Analysis (ABSA) method, recommendation model, and dataset in detail in Section 3. Section 4 and Section 5 presents the settings and experimental evaluations. Finally, in Section 6, we provide a conclusion and introduce our plans for future work.

2 Comparative Literature Survey

Many research papers have been published so far on recommendation systems. This section discusses relevant studies for multi-criteria recommendation and aspect-based sentiment analysis.

2.1 Recommendation Model

Neighborhood-based techniques employ user-item assessments to calculate the parallels among user-based [5] and item-based [6] to provide recommendations directly. In model-based techniques, user-item assessments are utilized to develop a prediction model, which is subsequently used to provide recommendations. Model-based techniques include Support Vector Machines [7], Bayesian Clustering [8], Latent Semantic Analysis [9] and Singular Value Decomposition [10]. Studies [11, 12, 13] consider the user's preference for certain product features to estimate more specific recommendations.

2.2 Multi-criteria Recommendation

Memory-based and model-based strategies are two main types of multi-criteria recommendation techniques based on their utility function. The similarity can be computed in advance methods and conservative methods using memory-based techniques. The advanced techniques utilize multidimensional distance metrics directly to calculate the distance between multi-criteria scores [14]. Conservative techniques utilize average [14], worst-case [14], and weighted sum of individual similarities [15] to combine traditional similarities values determined separately on each criterion into a single similarity.

The model based approach [14], predicts the unknown ratings with the assumption of item's overall rating is independent of other ratings. It generates a direct relationship between the overall rating and multi-criteria rating. Moreover, approaches like Probabilistic Support Vector Regression[16], Modeling[17], Multi-linear Singular Value Decomposition [18], and Genetic Algorithm [19] uses multidimensional distance metrics to find direct distance between multi-criteria scores [14].

Even though only few studies have been proposed on integrating deep learning into a collaborative filtering model. Deep learning for recommender systems has recently received much notice. Some researchers have developed deep learning-based recommendation models, that have shown capability to capture complex relationships [20], resulting in improved recommender system performance.

While the early literature on recommendation largely focused on raw explicit feedback [21][22], recent attention is increasingly shifting to data [23][24]. For instance, extracted data from user reviews, which is processed using aspect-based sentiment analysis techniques and produce polarity of users on specific aspects such as presented in this paper.

2.3 Aspect-based Sentiment Analysis

Akhtar et al. [25] analyze the hotel reviews to extract common aspects. The study categorizes the reviews into predetermined categories, which are the aspects that frequently recur in the reviews dataset. Our model is based on such analysis of aspect classification. Some publications explore Aspect-Based Sentiment Analysis for user reviews and using ABSA in recommendation systems [26, 27, 28]. Musto et al. [29] describe a method for justifying the recommendations given by a recommendation algorithm by automatically extracting relevant and distinguishable properties of the recommended item from user reviews. It claims that the users prefer review-based arguments over other explanation strategies. Bauman et al. [30] propose a method called the Sentiment Utility Logistic Model (SULM) that uses sentiment analysis of user reviews. This study demonstrates some experiments in which users who selected the items according to their recommendations for the most valuable aspects had better experiences, as measured by the rating. Liu et al. [31] propose a novel Multilingual Review-aware Deep Recommendation Model (MrRec) that mainly consists of a Multilingual aspect-based sentiment analysis module (MABSA) and a Multilingual recommendation module for rating prediction. Fukumoto et al. [32] present a collaborative filtering method for hotel recommendation incorporating user choices by using aspect-based sentiment analysis.

3 Developed Approach and System Model

3.1 Dataset

In this study, the model was evaluated with a multi-criteria rating dataset called HotelRec [33], containing 50 million reviews. This dataset is based on TripAdvisor and is a very large-scale, convenient data for a hotel recommendation system. Each review contains the user name and hotel URL, the date, the overall rating, the summary of the review, the whole review text, and the sub-ratings. Initial dataset is as shown in Figure 3.2.

Preprocessing Steps:

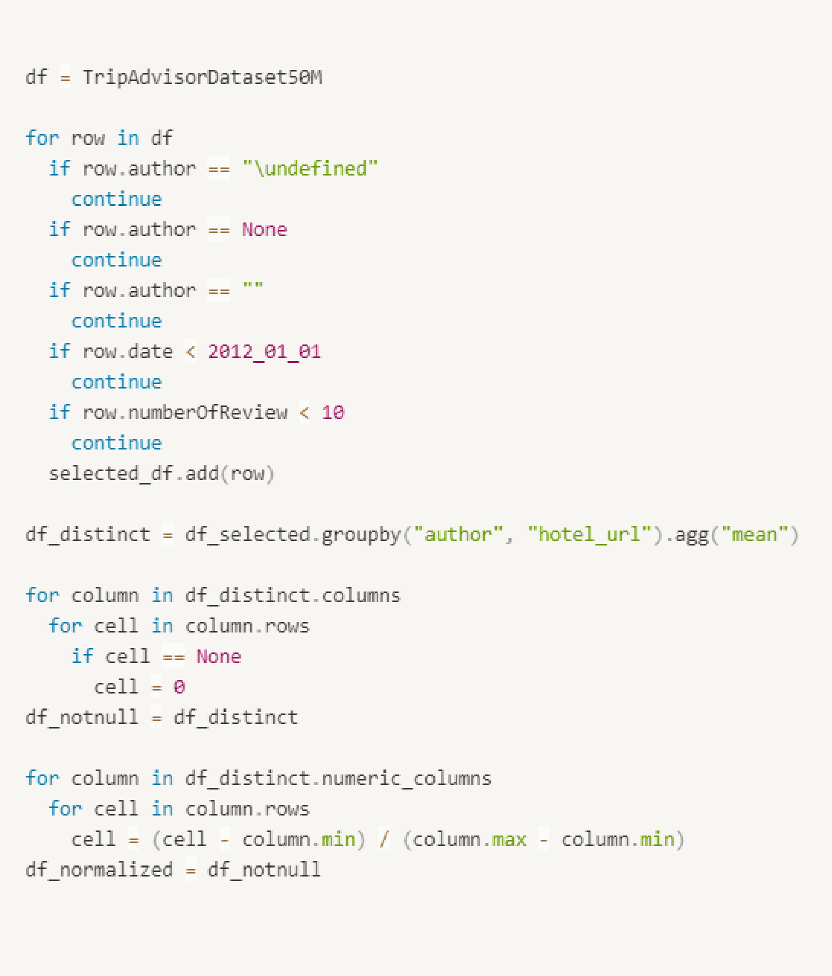
- Data cleaning (handle multiple reviews, untitled users)
- Filtering users and hotels
- Encoding the categorical data (author, hotel_url)
- Averaging multiple reviews (case of a user making more than one review on a hotel)
- Normalization of polarities and ratings
- Splitting the dataset (train - test)

The dataset of 50 million reviews was split into 11 pieces for analysis, and each part had about 5 million comments. For these pieces, we performed the following operations, respectively. First of all, rows with user columns undefined or none were removed. As a result, about 3 percent of data has been reduced. After that, reviews made in the last ten years were selected. Finally, users with ten or more reviews were selected for the model. As a result, approximately 1% of the original data is suitable for the model.

593,000 rows were extracted from 50 million reviews. This filtered dataset includes 24,890 unique users, 178,096 unique hotels, and 31191 hotel locations.

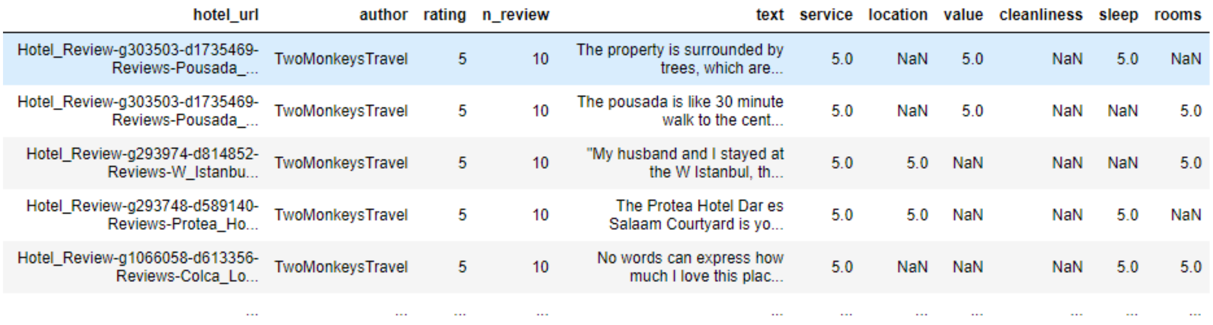
Tripadvisor dataset contains a date column for each review. We have done the research for the use of this column to contribute to the model by introducing new information as a season for the hotel in understanding which dates the user stayed. However, this information could not be obtained from this column, as the user can review it on any date within two years after staying at the hotel.

There are null values for sub-ratings (Service, Location, Value, Cleanliness, Sleep, Rooms, FrontDesk, Business) in the hotel dataset, and almost all of the ratings given for FrontDesk and Business are null. Therefore, the data for these two features (FrontDesk and Business) were ignored and not examined. In addition to these, the number of reviews made by users to the hotel was added as a feature since it could be useful for the model. Figure 3.3 shows the distribution of users according to the number of



```
df = TripAdvisorDataset50M

for row in df
  if row.author == "\undefined"
    continue
  if row.author == None
    continue
  if row.author == ""
    continue
  if row.date < 2012_01_01
    continue
  if row.numberOfReview < 10
    continue
  selected_df.add(row)

df_distinct = df_selected.groupby("author", "hotel_url").agg("mean")

for column in df_distinct.columns
  for cell in column.rows
    if cell == None
      cell = 0
df_notnull = df_distinct

for column in df_distinct.numeric_columns
  for cell in column.rows
    cell = (cell - column.min) / (column.max - column.min)
df_normalized = df_notnull
```

Figure 3.1: Dataset Pseudocode



hotel_url	author	rating	n_review	text	service	location	value	cleanliness	sleep	rooms
Hotel_Review-g303503-d1735469-Reviews-Pousada_...	TwoMonkeysTravel	5	10	The property is surrounded by trees, which are...	5.0	NaN	5.0	NaN	5.0	NaN
Hotel_Review-g303503-d1735469-Reviews-Pousada_...	TwoMonkeysTravel	5	10	The pousada is like 30 minute walk to the cent...	5.0	NaN	5.0	NaN	NaN	5.0
Hotel_Review-g293974-d814852-Reviews-W_Istanbu...	TwoMonkeysTravel	5	10	"My husband and I stayed at the W Istanbul, th...	5.0	5.0	NaN	NaN	NaN	5.0
Hotel_Review-g293748-d589140-Reviews-Protea_Ho...	TwoMonkeysTravel	5	10	The Protea Hotel Dar es Salaam Courtyard is yo...	5.0	5.0	NaN	NaN	5.0	NaN
Hotel_Review-g1066058-d613356-Reviews-Colca_Lo...	TwoMonkeysTravel	5	10	No words can express how much I love this plac...	5.0	NaN	NaN	NaN	5.0	5.0
...	...	...	...	...	...	...	...	...	...	...

Figure 3.2: Initial Dataset

reviews they make and the average of the overall ratings they give. This result shows us that as the total number of reviews made by the user increases, the average rating decreases slightly.

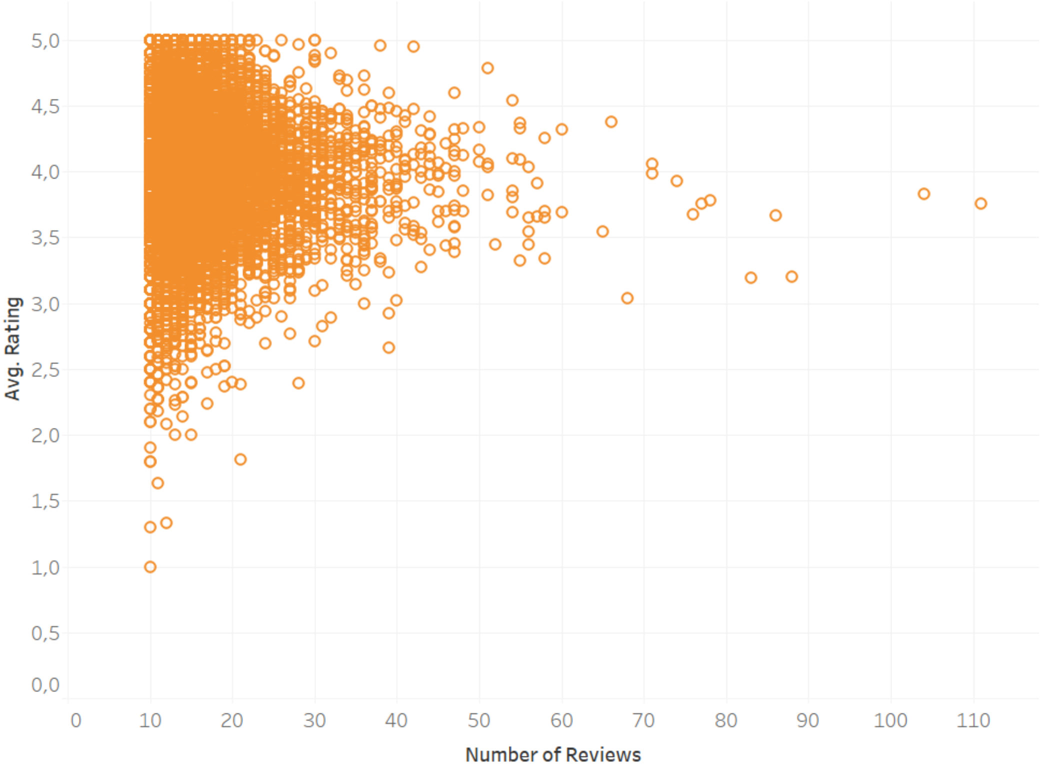


Figure 3.3: The Average Rating and Number of Reviews Given by the User

On the other hand, Table 3.1 shows the distribution of sub-ratings, Table 3.4 shows distribution of overall rating, and Table 3.2 shows null rates and mean values of ratings. As can be seen in these results, the null rate is relatively high in sub-ratings. In addition, the overall rating distribution is not very distinctive.

Table 3.1: Sub-Rating Distributions

	Service	Location	Value	Cleanliness	Sleep	Rooms
1	1.5%	0.3%	0.9%	0.6%	0.8%	0.7%
2	2.7%	1.0%	2.1%	1.3%	1.6%	2.0%
3	11.3%	6.2%	8.8%	5.6%	6.6%	9.2%
4	23.3%	14.5%	15.2%	14.1%	14.2%	15.0%
5	32.7%	20.5%	15.5%	20.8%	17.1%	14.6%

Finally, polarity feature columns were created according to aspects in the dataset as explained in Section 3.2. Total polarity is the general polarity that emerges from the whole review. If there is no sentence about an aspect in the review, a neutral (0) is given for that aspect. Mean values of extracted polarity columns are shown in Table 3.3. These values were extracted in interval [0,1].

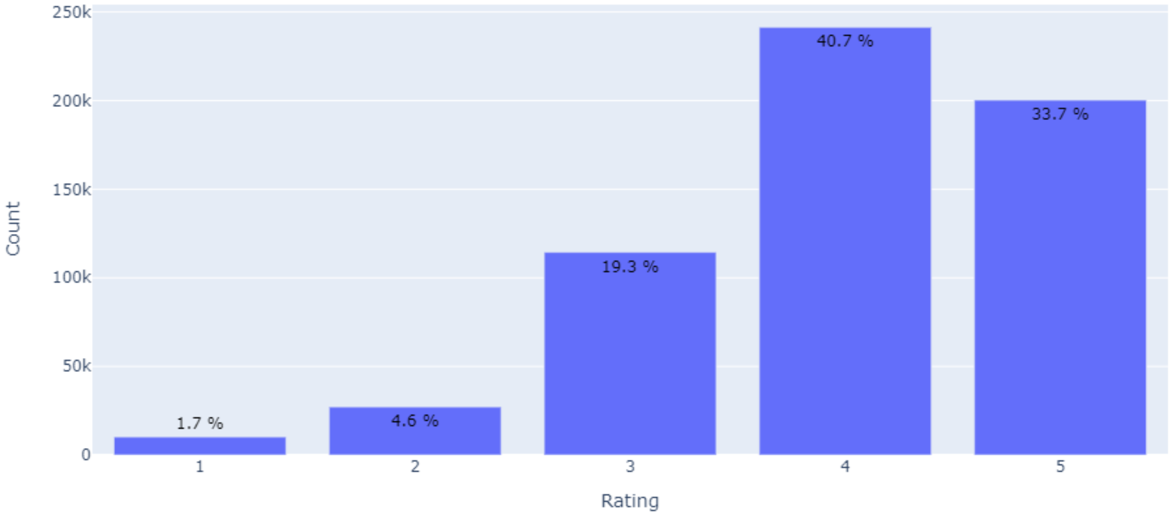


Figure 3.4: Total Rating Distribution

Table 3.2: Overall Rating and Sub-Rating Properties

	Overall	Service	Location	Value	Cleanliness	Sleep	Rooms
Null Rate:	0.0%	28,5%	57,5%	57,5%	42,5%	40,2%	58,4%
Mean:	4.0014	4.1617	4.2659	3.9971	4.2493	4.1198	3.9796
Normalized Mean:	0.7399	0.5946	0.3723	0.3479	0.3704	0.3406	0.3382

Users with maximum review counts are shown in Table 3.4.

3.2 Aspect-Based Sentiment Analysis (ABSA)

Aspect-based sentiment analysis operations were performed for each sentence of each review. It contains the following steps:

- Breaking the whole review into sentences by using the Natural Language Toolkit tokenization package and by using sent_tokenization method.
- Splitting the sentences into opinion units: Words in a review contains different importance with each aspect. Most of the times a user's reviews contains multiple aspect for a specific item. To explain it through an example sentence, for instance, we should perform the following operations for the sentence "I liked the room, but the food was horrible." First of all, it is necessary to find the sentence's predicate. In this sentence, "like" and "was" are the verbs we want to find. There is the verb "like" at the beginning, but there is a verb in the continuation of the sentence, which is "was". If there is more than one verb as in the example, the model looks at the rest of the sentence to find it. Thus, it detects more than one emotion in a sentence. By looking at the properties and positions of the tokens, it determines the domain of verbs in the sentence. In this way, sentences

Table 3.3: ABSA: Extracted Polarities

	Mean
Total Polarity	0.65327
Value Polarity	0.42399
Location Polarity	0.49672
Service Polarity	0.50180
Meal Polarity	0.47155
Facility Polarity	0.48178
Room Polarity	0.52255
Quality Polarity	0.46807
Staff Polarity	0.51647
Surrounding Polarity	0.48803

Table 3.4: Top Reviewers

User	**Rating Count**
AGoingGal	572
TravelManSouth	364
811bernardf	341
Riesst	332
mrsbrojoe	297
WeeLuv2Travel2	296
revjohannesm	287
759elib	284
WilliamJ244	282
drdroad	275

are divided into judgments according to different aspects. These judgments are called opinion units.

- Word Extraction stage: Operations include removing stop words (e.g., at, on, in), removing words containing less than three letters, removing numbers written in a typeface.

- Lemmatization stage: Process of distinguishing verbs and nouns by identifying a word's position in a sentence using NLTK.

- Detection of aspects phase: Aspects are found by identifying nouns in the sentence. The most frequently used nouns are designated as aspects. Word frequencies are as shown in Figure 3.5. We used aspects that [25] categorized. Categories and aspects are as shown in Table 3.5. For instance, if the aspect from the sentence is "Railway", "View", "Station", "Airport" or "Distance", the polarity of the "Location" category is obtained. To give another example, if the aspect of the sentence is "Drink", "Breakfast", or "Food", the polarity of the "Meal" category is obtained.

- The polarity was found using the SentimentIntensityAnalyzer method in the VaderSentiment library. For instance, this method helps classify polarity as positive if the word "great" is used for a particular aspect and negative if "bad" is used.

- Adding polarity phase, polarities created and extracted feature column according to aspects. If a user has made more than one review for a hotel and two of their reviews have two different polarities for an aspect, they are averaged. The information on how many times the user has reviewed the hotel has also been added as a feature since the number of visits to the hotel is an important indicator in the interaction between the user and the hotel.

- Polarity features are given between negative (-1) and positive (+1). Finally, each extracted polarity was normalized into the interval [0,1]. Examples of results are as shown in Table 3.6. Final dataset format is as shown in Figure 3.6.

Table 3.5: Aspect Categories

Value	Location	Service	Meal	Facility	Room	Quality	Staff	Surrounding
Price	Railway	Desk	Drink	Pool	Bed	Satisfactory	Good	Landmark
Amount	View	Check-in	Breakfast	Spa	Dirty	Ample	Polite	Monument
Rate	Station	Check-out	Spicy	Wi-fi	Clean	Hygienic	Helpful	Temple
Cheap	Airport	Reliable	Food	Gymnasium	Toilet	Proper	Friendly	Mosque
Worth	Distance	Fast	Tasty	Gym	Bathroom	Ambience	Reliable	Church
Low	Far	Convenient	Tea	Internet	Shower	Odour	Quick	Restaurant
Money	Close		Buffet	Ample	Dryer	Smell		Diner
Economical	Convenient		Bar	Parking	Fridge		Mall	
Reasonable	Train		Restaurant	Wireless	View			Market

Table 3.6: ABSA Examples

Sentence	Aspect	Polarity
Service is outstanding as well staff are all nice and very helpful.	Service	0.8999
Service was fast and efficient.	Service	0.6486
Service is nice and friendly.	Service	0.6908
Service is friendly.	Service	0.4939
Staff however were very friendly and accommodating.	Staff	0.5413
We were in Room 930 which is top floor, corner room.	Room	0.2023
Service and food at the beach was good though a bit pricey.	Service	0.4404
Rooms big but aircon noisy.	Room	-0.2617
Rooms always clean and great restaurant in the lobby.	Room	0.7845
Stayed in the G Deluxe Room (aka White Room).	Room	0.0
Location good.	Location	0.4404
Location great for tourist attractions, jonker st ect.	Location	0.7845
The Hotel is stunning The Location is perfect.	Location	0.743
Staff was always friendly and greeted me all the time.	Staff	0.6486
Service is excellence, and the staff truly cannot do enough for each and every guest, service is truly incredible!	Service	0.8805
Room was newly decorated, not amazing but nice with really comfortable bed and lovely toiletries.	Room	0.7938
Service is always impeccable from Genaro, Javier, Domingo, Luis, Mario, Benito, both Jorge's, (pool bar and Palazzo) both Ricardo's, (pool, Ventanas, Azia, and Palazzo) Eduardo, Erik, Miguel, Adan, Ilse, and of course Isis, I am very sure that there are many other wonderful RH staff members that we omitted, and we will be sure to give each of them the credit that is due in our next review.	Service	0.8805

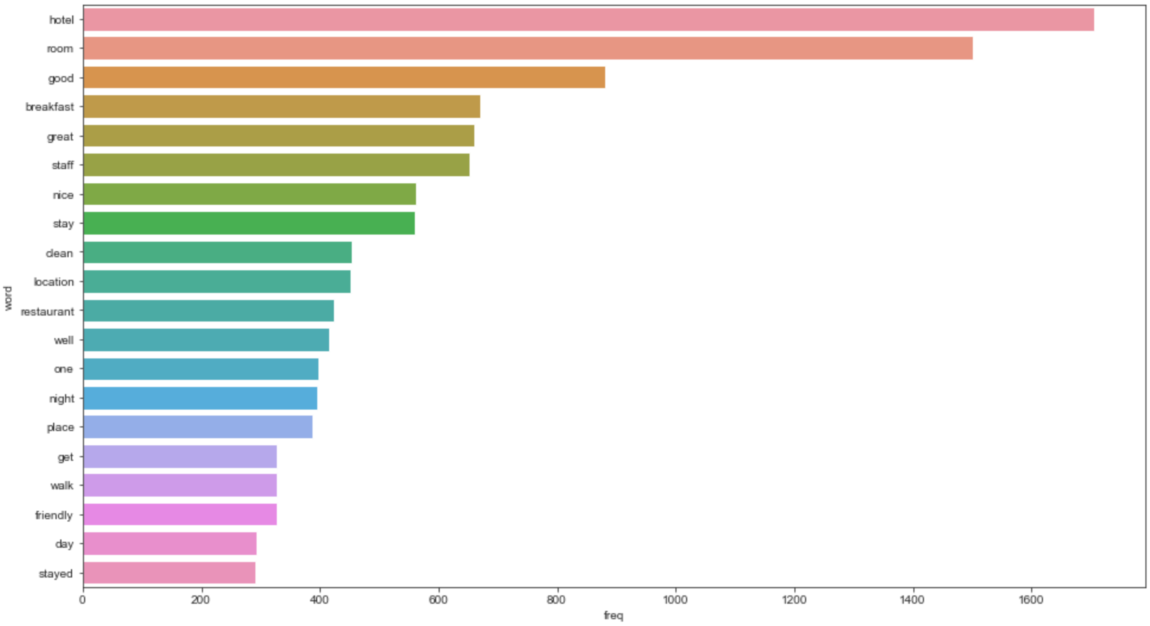
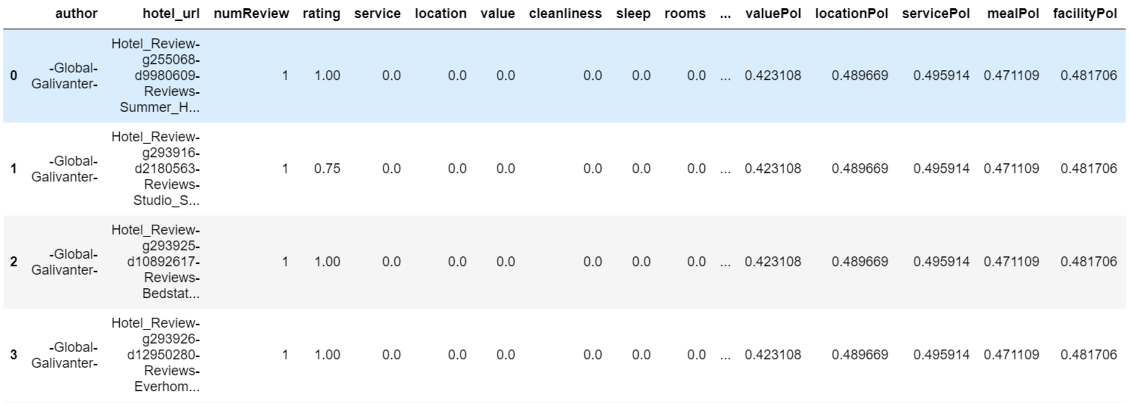
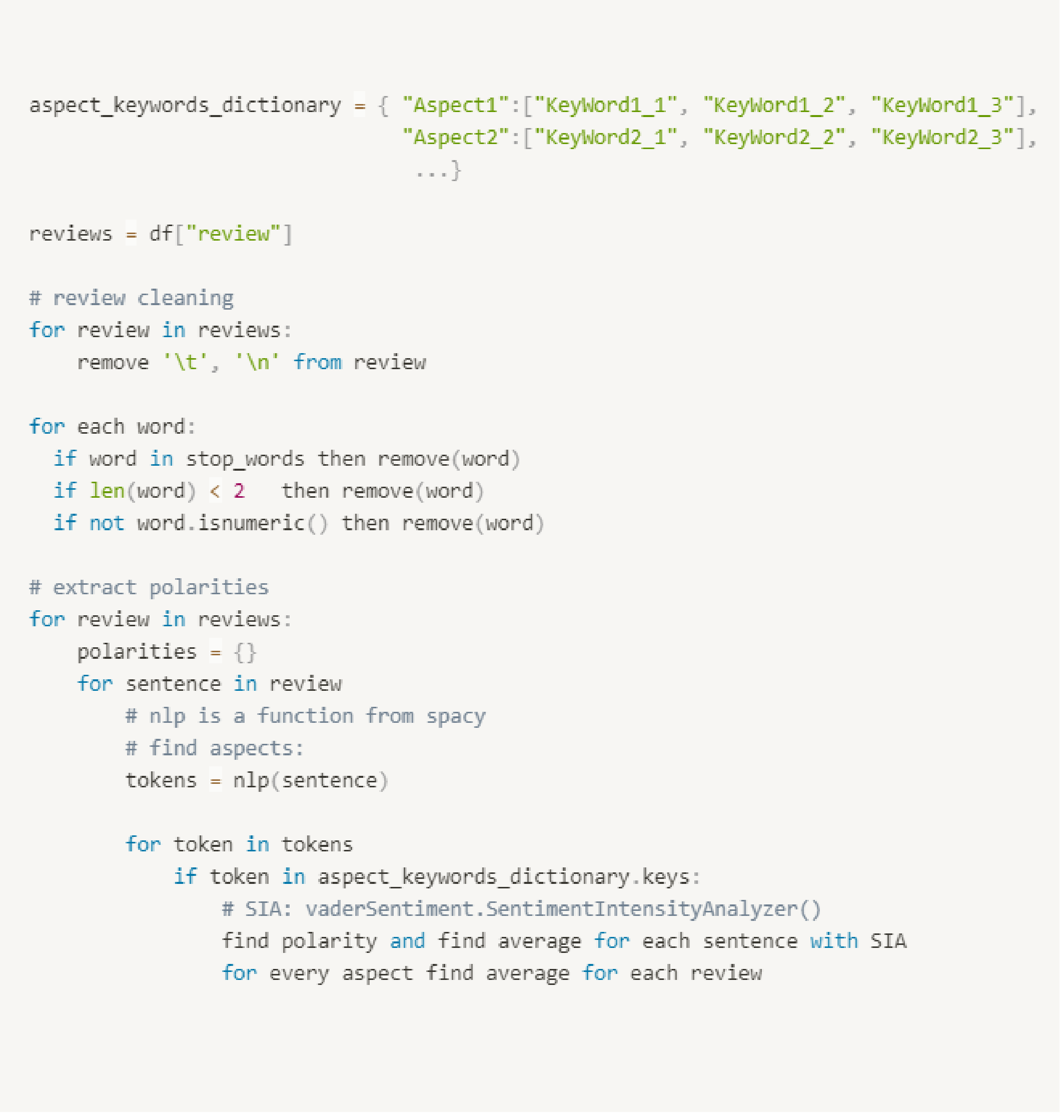


Figure 3.5: Word Frequencies



	author	hotel_url	numReview	rating	service	location	value	cleanliness	sleep	rooms	...	valuePol	locationPol	servicePol	mealPol	facilityPol
0	-Global-Galivanter-	Hotel_Review-g255068-d9980609-Reviews-Summer_H...	1	1.00	0.0	0.0	0.0	0.0	0.0	0.0	...	0.423108	0.489669	0.495914	0.471109	0.481706
1	-Global-Galivanter-	Hotel_Review-g293916-d2180563-Reviews-Studio_S...	1	0.75	0.0	0.0	0.0	0.0	0.0	0.0	...	0.423108	0.489669	0.495914	0.471109	0.481706
2	-Global-Galivanter-	Hotel_Review-g293925-d10892617-Reviews-Bedstat...	1	1.00	0.0	0.0	0.0	0.0	0.0	0.0	...	0.423108	0.489669	0.495914	0.471109	0.481706
3	-Global-Galivanter-	Hotel_Review-g293926-d12950280-Reviews-Everhom...	1	1.00	0.0	0.0	0.0	0.0	0.0	0.0	...	0.423108	0.489669	0.495914	0.471109	0.481706

Figure 3.6: Final Dataset Fed with Polarities and Normalized Ratings



```
aspect_keywords_dictionary = { "Aspect1":["KeyWord1_1", "KeyWord1_2", "KeyWord1_3"],
                               "Aspect2":["KeyWord2_1", "KeyWord2_2", "KeyWord2_3"],
                                ...}

reviews = df["review"]

# review cleaning
for review in reviews:
    remove '\t', '\n' from review

for each word:
  if word in stop_words then remove(word)
  if len(word) < 2   then remove(word)
  if not word.isnumeric() then remove(word)

# extract polarities
for review in reviews:
    polarities = {}
    for sentence in review
        # nlp is a function from spacy
        # find aspects:
        tokens = nlp(sentence)

        for token in tokens
            if token in aspect_keywords_dictionary.keys:
                # SIA: vaderSentiment.SentimentIntensityAnalyzer()
                find polarity and find average for each sentence with SIA
                for every aspect find average for each review
```

Figure 3.7: ABSA Pseudocode

3.3 Model

The proposed model is based on the architecture implemented in [2]. As shown in Fig. 3.8, the model is composed of 3 parts: The first part includes aspect-based sentiment analysis. ABSA is applied to the reviews, then polarities are extracted, and the final dataset, fed by polarity features, is obtained.

In the second part, we get the concatenated author and hotel embeddings and feed them as input to the Polarities and Sub-ratings DNN. Since only IDs of users and items are used as a feature, this model uses absolute collaborative filtering. The embedding vectors of users and items are first set to random values, and then the values are changed during model training to minimize the loss function. A deep neural network is used to predict a user's criteria ratings and polarities on an item in this step.

Equation 1 shows the output of a hidden layer l. While W represents the weight matrix, b stands for bias vector.

$$h_i = ReLU(W_i h_{l-1} + b_l) \tag{1}$$

The output layer formulated as Equation 2. The output of Polarities and Sub-ratings DNN is prediction of criteria as represented in Equation 3. In the output layer, we predict the user criteria array using Eqs. 2,3 where L is the number of layers.

$$y_c = ReLU(W_L h_{L-1} + b_L) \tag{2}$$

$$y_c = [c_1, c_2, c_3, .., c_k]^T \tag{3}$$

Finally, in the third part, we normalize the continuous features $(c1, c2, c3, .., ck)$ with the normalization formula in Equation 4.

$$x' = (x - x_{min})/(x_{max} - x_{min}) \tag{4}$$

Then, we give obtained criteria array $(c1, c2, c3, .., ck)$ as input to Overall Score DNN to predict the overall score y that represents the probability of the user choosing the hotel shown in Equation 5.

$$y = f(c_1, c_2, c_3, .., c_k) \tag{5}$$

The criteria array contains the sub-ratings given by the user to the hotel and the polarity values obtained by ABSA. The relationship between the overall score and the criteria array is learned in this part. The common and obvious assumption in the aggregation-function-based method is that the overall rating could be assessed using an aggregate function of multi-criteria ratings.

The hidden layer sizes, learning rate, batch size, and embedding vector size used are determined for both DNNs according to the results of experiments in Section 4.1.

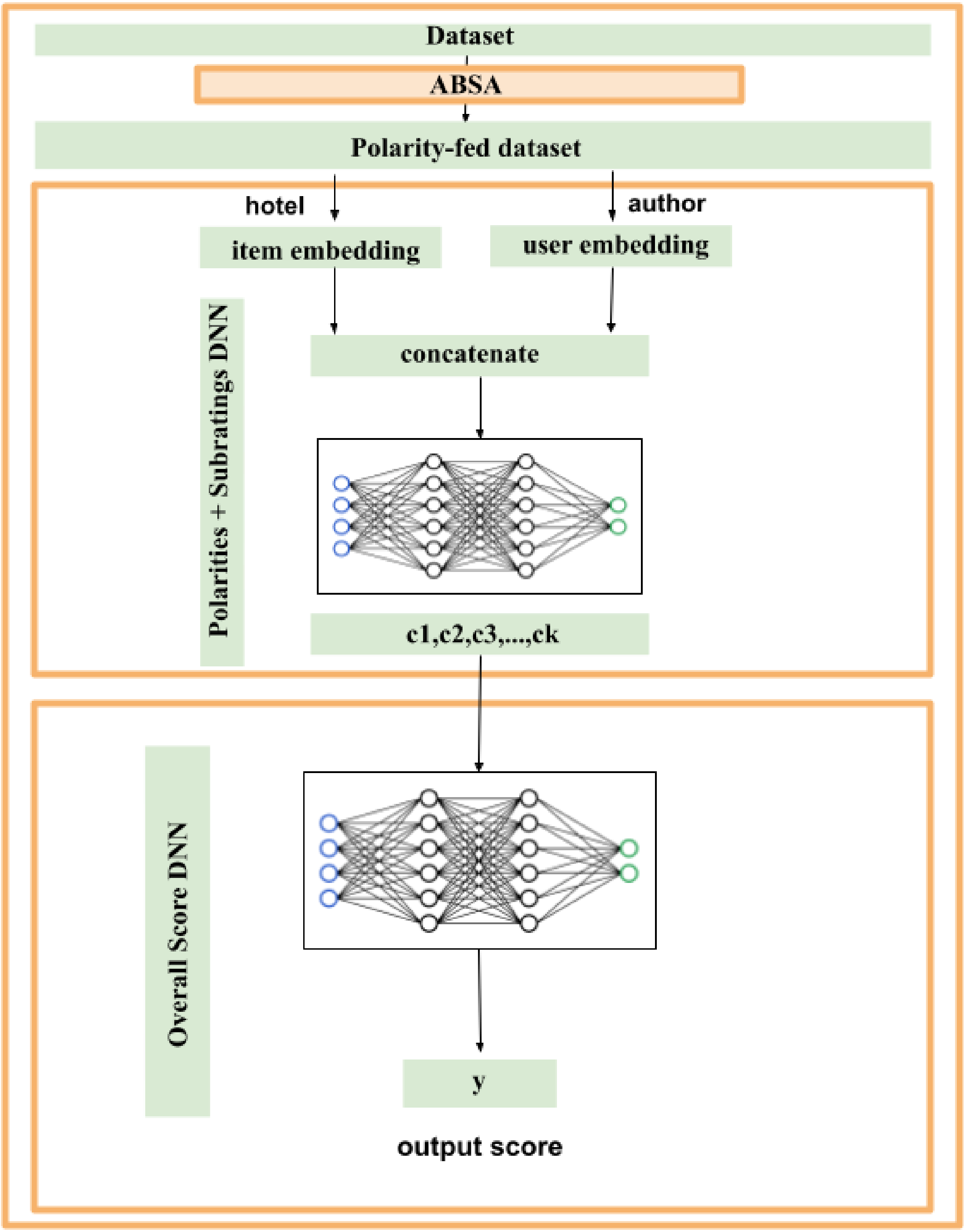


Figure 3.8: The architecture of the proposed model: Deep Neural Networks and ABSA

4 Experiments

4.1 Settings

We used Keras with TensorFlow for the neural network model implementation. On the other hand, we used Natural Language Toolkit (NLTK) for aspect-based sentiment analysis.

The learning rate is a hyperparameter that controls how much to change the model in response to the estimated error each time the model weights are updated. Learning rate experiment results are as shown in Table 4.1.

We use Rectified Linear Unit (ReLU) in model that is a single-point function that gives 0 when the input value is lesser than the set value and gives the linear multiple if the input is given is greater than the set value.

Table 4.1: Learning Rate

Learning Rate	**MAE**	**Elapsed Time (s)**
0.0001	0.100110	252.4514
0.0005	0.105321	255.2264
0.001	**0.09969**	**257.0208**
0.002	0.10106	277.6118
0.005	0.10996	267.9195

The batch size defines the number of samples that will be propagated through the network, and experiments results are as shown in Table 4.2. In addition, user and item embedding sizes experiment results and epoch experiment results are as shown in Table 4.4 and Table 4.3.

Table 4.2: Batch Size

Batch Size	**MAE**	**Elapsed Time (s)**
8	0.10312	938.02425
16	0.10425	517.74439
32	**0.10107**	**157.65366**
64	0.14004	256.11178
128	0.10413	81.61894

In Sub-ratings and Polarities DNN, the output layer has 20 neurons that is equal to the number of the sub-ratings plus polarities and the number of reviews. In the Overall score DNN, the output layer has one neuron for the overall score representing the value between 0 and 1, showing the estimation of the probability of the user choosing the hotel. Experiments were conducted to determine the number of hidden layers. The results of architectures with different hidden layer sizes are shown in Table 4.5. Mean Absolute Error (MAE) was calculated by estimating the overall score. Selected

Table 4.3: Epoch

Epoch	MAE	Elapsed Time (s)
8	0.30479	46.1332
16	0.14823	76.2778
32	**0.10063**	**258.5229**
64	0.10246	136.4422
128	0.10175	525.5475

Table 4.4: Embedding Size

Embedding Size	MAE	Elapsed Time (s)
32	0.14229	277.6943
64	0.10668	277.6408
128	0.18006	277.6465
256	**0.10290**	**277.8883**
512	0.22208	266.2203

learning rate, optimizer, batch, epoch, and embedding size based on the results of experiments shown in Table 4.6. Rectified Linear Units (ReLU) was chosen as the activation function since it produces the most desired effect [34].

Table 4.5: Hidden Layer Sizes

Hidden Layers	MAE	Elapsed Time (s)
[16, 8]	0.11192	195.5212
[32, 16, 8]	0.10101	217.5914
[64, 32, 16, 8]	0.10161	217.5524
[128, 64, 32, 16, 8]	0.18150	277.5289
[256, 128, 64, 32, 16, 8]	0.26197	238.8719
[512, 256, 128, 64, 32, 16, 8]	0.17999	277.4658
[1024, 512, 256, 128, 64, 32, 16]	**0.09948**	**343.1469**
[2048, 1024, 512, 256, 128, 64, 32]	0.12090	337.6681

Table 4.6: Hyperparameter search space and best hyperparameters

Hyperparameter	Search Space	Selected Value
Learning Rate	[0.05-0.001]	0.001
Epoch	[8-128]	64
Optimizer	[Adam, Adamax, Rmsprop]	Adam
Embedding Size	[32-512]	256
Batch	[8-128]	32
Epoch	[8-128]	32

5 Comparative Evaluation and Discussion

Since we make predictions between 0-1, we can interpret 0.5 and above as the user prefers that hotel. However, since the recommendation system also requires a ranking, an ordered list can be suggested to the user based on this score. For this reason, we used MAE for the normalized rating prediction and the F-score evaluation based on the recall and precision values.

5.1 Accuracy Metrics

5.1.1 Mean Absolute Error (MAE)

$$MAE = \frac{1}{|N|} \sum_{r'_{ui} \epsilon N} |r_{ui} - r'_{ui}| \tag{6}$$

where r'_{ui} is predicted score of user u for item i, r_{ui} is the actual score, and $|N|$ is size of test set.

5.1.2 F-Score

F1 is calculated as Eq. 7. Precision is calculated as recommended and relevant items divided by recommended items. Recall is calculated as recommended and relevant items divided by relevant items. The definition of relevant item is based on its true rating r_{ui} which is greater than a specific threshold. If the estimated rating r'_{ui} is greater than the given threshold in its k-neighbours then that item is considered as a recommendation to the user. We set the threshold 0.5 since predictions in interval 0-1.

$$F1 = 2 \times \frac{(precision \times recall)}{(precision + recall)} \tag{7}$$

Table 5.1: Method Comparisons

Method	MAE	F1-Score
BaselineOnly	0.164395	0.728009
SVD++	0.168171	0.723819
KNNBaseline	0.171455	0.730498
KNNWithZScore	0.175365	0.753533
KNNWithMeans	0.175581	0.752269
KNNBasic	0.177417	0.819961
SVD	0.177629	0.723647
NormalPredictor	0.244793	0.639061
CoClustering	0.403692	0.759395
Single Rating DNN	0.154805	0.774335
Multi-criteria Ratings DNN	0.112469	0.794236
Proposed model	**0.099484**	**0.823616**

5.2 Comparisons with the state-of-the-art methods

The Surprise library is used [35] to compare singular recommendation methods such as SVD, SVD++, Baseline, and KNN. BasiclineOnly algorithm predicts the baseline estimate for a given user and item. On the other hand, SVD algorithm is equivalent to Probabilistic Matrix Factorization [36]. The SVD++ algorithm is an extension of SVD that takes into account implicit ratings. Besides, KNNBaseline is a basic collaborative filtering algorithm taking into account a baseline rating. KNNWithZScore is a basic collaborative filtering algorithm, taking into account the z-score normalization of each user. KNNWithMeans is another basic collaborative filtering algorithm, taking into account the mean ratings of each user. In addition, NormalPredictor algorithm predicts a random rating based on the distribution of the training set and it is one of the most basic algorithms that does not do complex work. Finally, Co-clustering is a collaborative filtering algorithm based on co-clustering.

In addition, to directly predict the score a singular neural network is used with only multi-criteria ratings to show the difference between the version with added polarities. To check the statistical authenticity of the achieved results for proposed method the Wilcoxon Signed Rank Test [37] is applied with the significance value of 0.05. The 10-cross validation results in terms of MAE and F1-Score indicated in that Table 5.1, illustrate that multi-criteria ratings and ABSA features are better than the baseline models. The results shown in the table indicate that our approach achieved 4% better results than the previous baseline and state of the art methods.

6 Conclusion and Future Work

In this project, we propose a deep recommender system using aspect-based sentiment analysis of users' reviews for hotel recommendations. To the best of our knowledge, this is the first study that merged reviews and extracted polarity for a deep multi-criteria collaborative filtering model. The proposed model discovers the relationship between overall rating and sub-ratings with aspect polarities. The model is composed of 3 parts: the first part includes aspect-based sentiment analysis which is applied to the reviews, then polarities are extracted, and polarity features feed the final dataset is obtained. In the second part, we get the author and hotel embeddings, concatenate them, and give them as input to a Polarities and Sub-ratings Deep Neural Network (DNN) model to predict the sub-ratings and polarities. Finally, in the third part, we give obtained criteria array as input to another DNN to predict the overall score. The proposed model was evaluated with a multi-criteria rating and review dataset called HotelRec, based on TripAdvisor, and the results are given above in detail.

Future work will improve the dataset with location and season information. Location information can be extracted and classified from the hotel URL. However, there may be problems in extracting the locations since the place names in the data are not very proper. In addition, users can be categorized as domestic - international travelers according to their activities. Traveler-type information such as single, couple, family, business, and friends could benefit. With such extra data, the system can be enriched, and the number of features can be increased. In future work, we also plan to improve the performance of the deep learning model by studying different deep learning methods such as Recurrent Neural Network (RNN), Generative Adversarial Networks (GAN), Convolutional Neural Network (CNN), and Autoencoder or other feature representation methods.

References

[1] F. Ricci, L. Rokach, and B. Shapira, "Introduction to recommender systems handbook," *Recommender Systems Handbook*, pp. 1–35, 2011. [Online]. Available: https://doi.org/10.1007/978-0-387-85820-3_1

[2] N. Nassar, A. Jafar, and Y. Rahhal, "A novel deep multi-criteria collaborative filtering model for recommendation system," *Knowledge-Based Systems*, vol. 187, 2020. [Online]. Available: https://doi.org/10.1016/J.KNOSYS.2019.06.019

[3] R. Burke, "Knowledge-based recommender systems," *Encyclopedia of library and information systems*, vol. 69, 05 2000.

[4] Z. Tao, Y. Wei, X. Wang, X. He, X. Huang, and T. S. Chua, "Mgat: Multimodal graph attention network for recommendation," *Information Processing & Management, 57*, vol. 5, 2020. [Online]. Available: https://doi.org/10.1016/J.IPM.2020.102277

[5] J. L. Herlocker, J. A. Konstan, A. Borchers, and J. Riedl, "An algorithmic framework for performing collaborative filtering," *ACM SIGIR Forum, 51*, vol. 2, pp. 227–234, 2017. [Online]. Available: https://doi.org/10.1145/3130348.3130372

[6] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, "Item-based collaborative filtering recommendation algorithms," *Proceedings of the 10th International Conference on World Wide Web WWW*, pp. 285–295, 2001. [Online]. Available: https://doi.org/10.1145/371920.372071

[7] M. Grčar, B. Fortuna, D. Mladenič, and M. Grobelnik, "knn versus svm in the collaborative filtering framework," in *Data Science and Classification*, V. Batagelj, H.-H. Bock, A. Ferligoj, and A. Žiberna, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 251–260.

[8] J. S. Breese, D. Heckerman, and C. Kadie, *Empirical Analysis of Predictive Algorithms for Collaborative Filtering*, 2013. [Online]. Available: https://arxiv.org/abs/1301.7363v1

[9] T. Hofmann, "Collaborative filtering via Gaussian probabilistic latent semantic analysis," *SIGIR Forum (ACM Special Interest Group on Information Retrieval), SPEC. ISS*, pp. 259–266, 2003. [Online]. Available: https://doi.org/10.1145/860435.860483

[10] R. Bell, Y. Koren, and C. Volinsky, "Modeling relationships at multiple scales to improve accuracy of large recommender systems," in *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2007, pp. 95–104.

[11] K. Zhang, K. Wang, X. Wang, C. Jin, and A. Zhou, "Hotel recommendation based on user preference analysis," in *Proceedings - International Conference on Data Engineering, 2015-June*, 2015, pp. 134–138.

[12] W. Zhou and W. Han, “Personalized recommendation via user preference matching,” *Information Processing & Management, 56*, vol. 3, pp. 955–968, 2019. [Online]. Available: https://doi.org/10.1016/J.IPM.2019.02.002

[13] J. Zhang, Q. Peng, S. Sun, and C. Liu, “Collaborative filtering recommendation algorithm based on user preference derived from item domain features.” *Physica A: Statistical Mechanics and Its Applications*, vol. 396, pp. 66–76, 2014. [Online]. Available: https://doi.org/10.1016/J.PHYSA.2013.11.013

[14] G. Adomavicius and Y. O. Kwon, “New recommendation techniques for multicriteria rating systems,” *IEEE Intelligent Systems, 22*, vol. 3, pp. 48–55, 2007. [Online]. Available: https://doi.org/10.1109/MIS.2007.58

[15] T. Y. Tang and G. Mccalla, “The pedagogical value of papers: A collaborative filtering based paper recommender,” *J. Digit. Inf. 10*, vol. 10, 2009.

[16] H. Drucker·, C. J. C. Burges, L. Kaufman, A. Smola··, and V. Vapoik, “Support vector regression machines,” *Adv. Neural Inf. Process. Syst.*, vol. 9, no. 1, pp. 155–161, 1997.

[17] N. Sahoo, R. Krishnan, G. Duncan, and J. Callan, “The halo effect in multicomponent ratings and its implications for recommender systems: The case of yahoo! movies,” *Information Systems Research, 23*, vol. 1, pp. 231–246, 2012. [Online]. Available: https://doi.org/10.1287/ISRE.1100.0336

[18] M. Hassan and M. Hamada, “A neural networks approach for improving the accuracy of multi-criteria recommender systems,” *Applied Sciences 2017, Vol. 7, Page 868, 7*, vol. 9, p. 868, 2017. [Online]. Available: https://doi.org/10.3390/APP7090868

[19] ——, “Genetic algorithm approaches for improving prediction accuracy of multi-criteria recommender systems,” *International Journal of Computational Intelligence Systems, 11*, vol. 1, pp. 146–162, 2018. [Online]. Available: https://doi.org/10.2991/IJCIS.11.1.12

[20] M. Fu, H. Qu, Z. Yi, L. Lu, and Y. Liu, “A novel deep learning-based collaborative filtering model for recommendation system,” *IEEE Transactions on Cybernetics, 49*, vol. 3, pp. 1084–1096, 2019. [Online]. Available: https://doi.org/10.1109/TCYB.2018.2795041

[21] R. Salakhutdinov, A. Mnih, and G. Hinton, “Restricted boltzmann machines for collaborative filtering,” *In ICDM*, pp. 791–798, 2007.

[22] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, “Item-based collaborative filtering recommendation algorithms,” in *Proceedings of the 10th International Conference on World Wide Web, WWW 2001*, 2001, pp. 285–295.

[23] D. Liang, L. Charlin, J. McInerney, and D. M. Blei, “Modeling user exposure in recommendation,” *25th International World Wide Web Conference, WWW 2016*, pp. 951–961, 2015. [Online]. Available: https://doi.org/10.1145/2872427.2883090

[24] X. He, H. Zhang, M. Y. Kan, and T. S. Chua, "Fast matrix factorization for online recommendation with implicit feedback," in *Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval.* SIGIR 2016, 2017, pp. 549–558.

[25] N. Akhtar, N. Zubair, A. Kumar, and T. Ahmad, "Aspect based sentiment oriented summarization of hotel reviews," *Procedia Computer Science, 115*, pp. 563–571, 2017. [Online]. Available: https://doi.org/10.1016/J.PROCS.2017.09.115

[26] Y. Zhang, J. Du, X. Ma *et al.*, "Aspect-based sentiment analysis for user reviews." *Cogn Comput 13*, pp. 1114–1127, 2021. [Online]. Available: https://doi.org/10.1007/s12559-021-09855-4

[27] P. Bai, Y. Xia, and Y. Xia, "Fusing knowledge and aspect sentiment for explainable recommendation," *IEEE Access, 8*, vol. 10, pp. 37 150–13 716, 2020. [Online]. Available: https://doi.org/10.1109/ACCESS.2020.3012347

[28] Y. Zhang, H. Lu, C. Jiang, X. Li, and X. Tian, "Aspect-based sentiment analysis of user reviews in 5g networks," *IEEE Network*, vol. 35, no. 4, pp. 228–233, 2021. [Online]. Available: https://doi.org/10.1109/MNET.011.2000400

[29] C. Musto, P. Lops, M. De Gemmis, and G. Semeraro, "Justifying recommendations through aspect-based sentiment analysis of users' reviews." *ACM UMAP 2019 - Proceedings of the 27th ACM Conference on User Modeling Adaptation and Personalization*, pp. 4–12, 2019. [Online]. Available: https://doi.org/10.1145/3320435.3320457

[30] K. Bauman, B. Liu, and A. Tuzhilin, "Aspect based recommendations: Recommending items with the most valuable aspects based on user reviews," *ACM Reference Format, 9*, 2017. [Online]. Available: https://doi.org/10.1145/3097983.3098170

[31] P. Liu, L. Zhang, and J. A. Gulla, "Multilingual review-aware deep recommender system via aspect-based sentiment analysis," *ACM Transactions on Information Systems (TOIS), 39(2)*, 2021. [Online]. Available: https://doi.org/10.1145/3432049

[32] F. Fukumoto, H. Sugiyama, Y. Suzuki, and S. Matsuyoshi, "Exploiting guest preferences with aspect-based sentiment analysis for hotel recommendation," *Communications in Computer and Information Science, 553*, pp. 34–49, 2014. [Online]. Available: https://doi.org/10.1007/978-3-319-25840-9_3

[33] D. Antognini and B. Faltings, "Hotelrec: a novel very large-scale hotel recommendation dataset," in *LRECth International Conference on Language Resources and Evaluation, Conference Proceedings.* 4917–4923., 2020, pp. 2020–12.

[34] O. Fuentes, J. Parra, E. Anthony, and V. Kreinovich, "Why rectified linear neurons are efficient: A possible theoretical explanation," *Studies in Computational Intelligence, 835*, pp. 603–613, 2020. [Online]. Available: https://doi.org/10.1007/978-3-030-31041-7_34

[35] N. Hug, “Surprise: A python library for recommender systems,” *Journal of Open Source Software*, vol. 5, p. 52, 2020. [Online]. Available: https://doi.org/10.21105/JOSS.02174

[36] R. Salakhutdinov and A. Mnih, “Probabilistic matrix factorization,” in *Proceedings of the 20th International Conference on Neural Information Processing Systems (NIPS’07)*, C. A. Inc., R. Hook, and U. S. A. Ny, Eds., 2007, pp. 1257–1264.

[37] E. Fix and J. L. H. Jr, “Significance Probabilities of the Wilcoxon Test,” *The Annals of Mathematical Statistics*, vol. 26, no. 2, pp. 301 – 312, 1955. [Online]. Available: https://doi.org/10.1214/aoms/1177728547