Democratic People's Republic of Algeria
Ministry of Higher Education and Scientific Research
University of May 8, 1945 - Guelma -
Faculty of Mathematics, Computer Science, and Material Sciences
Department of Computer Science

Master's Thesis

Field: Computer Science

Option: "Sciences and Technologies of Information and Communication"

Theme :

A Recommendation System Based On Sentiment Analysis

Presented by : *Merabti Chayma*

Members of the jury :

- **President** : Dr. Seridi Ali
- **Supervisor** : Dr. Mehenaoui Zohra
- **Examiner** : Dr. Soussi Hakim

June 2024

Acknowledgments

First and foremost, I would like to express my profound gratitude to Allah Almighty for granting me the strength and patience to complete this project.

I also wish to extend my heartfelt thanks to my supervisor, "**Mehenaoui Zohra**" , for her unwavering support, dedication of time and energy, and for providing me with the necessary resources to make this work possible. Her invaluable advice, shared expertise, and constant encouragement to strive for excellence have been instrumental in this endeavor. May Allah reward you abundantly, Madame.

Additionally, I would like to thank the members of the supervisory committee for their constructive criticism, insightful feedback, and their valuable time.

My deepest appreciation goes to my dear family and friends for their encouragement and support, enabling me to give my best.

I wish to thank everyone who contributed to the completion of this thesis, whether directly or indirectly. I thank you all from all my heart.

Last but not least, I want to thank myself, Merabti Chayma, for believing in my abilities and for persevering through the challenges.

Dedication

*To my beloved father, **Hocine**, and mother, **Zohra**, whose unwavering support and endless sacrifices have been the foundation of my journey. Your love and encouragement have been my guiding light. May Allah protect you and keep you always in good health and happiness.*

*To my wonderful siblings, **Zayneb**, **Malak**, and **Wassim**, whose camaraderie and cheer have been a constant source of strength and inspiration. Your constant presence and shared moments of joy and challenge have made my path easier and more meaningful.*

*To my best friend, **Salmi Randa**, whose steadfast friendship and unwavering belief in me have been a pillar of support through thick and thin. You have filled my journey with good memories. May Allah bless our priceless friendship and make it last forever.*

*To my precious nephew, **Taim**, whose innocent smiles and boundless energy bring joy to my life.*

*To my dear friends, **Manar**, **Sara**, and my other beloved friends, whose kindness, encouragement, and laughter have made this journey not only possible but also enjoyable.*

This thesis is dedicated with all my love and gratitude.

Abstract

The recommendation system is an essential component of multiple domains, such as e-commerce, health care and social media, where it predicts user preferences by providing relevant suggestions. In this work, to improve the performance of the recommender systems, we propose a new approach to recommendation that we have applied on the Yelp Business dataset. This approach is based on sentiment analysis using three Machine Learning (ML) methods: Support Vector Machines (SVM), Naive Bayes (NB), Logistic Regression (LR) and Bidirectional Encoder Representations from Transformers DistilBERT. The proposed approach aims to improve recommendations through sentiment analysis of their user's reviews. We employ the method of Singular Value Decomposition (SVD) to help treat sparsity in user-item interaction data, ensuring minimum variance results with limited data inputs as well. The results using Neural Collaborative Filtering (NCF) as a recommendation model showed a good performance of the proposed approach, demonstrating its efficiency and accuracy in recommendation tasks.

Keywords: Recommendation System, Sentiment Analysis, Support Vector Machine, Naive Bayes, Logistic Regression, DistilBERT, SVD, NCF.

Résumé

Le système de recommandation est un composant essentiel dans de nombreux domaines tels que le commerce électronique, les soins de santé et les médias sociaux, où il prédit les préférences de l'utilisateur en fournissant des suggestions pertinentes. Dans ce travail, afin d'améliorer les performances des systèmes de recommandation, nous proposons une nouvelle approche de recommandation que nous avons appliquée sur le dataset Yelp Business. Cette approche est basée sur l'analyse des sentiments en utilisant trois méthodes du Machine Learning (ML): les machines à vecteurs de support (SVM), Naive Bayes (NB), Logistic Regression (LR) et une méthode de Deep Learning: Bidirectional Encoder Representations from Transformers DistilBERT. L'approche proposée vise à améliorer les recommandations en prenant en considération les avis des utilisateurs en analysant leurs sentiments à travers le traitement automatique de leurs commentaires. Pour traiter la rareté des données d'interaction entre l'utilisateur et l'item, nous employons la méthode de décomposition en valeurs singulières (SVD), ce qui nous a permis d'obtenir des résultats des de variance minimale avec des données restreintes. Les résultats obtenus en utilisant le filtrage collaboratif neuronal (NCF) comme modèle de recommandation ont montré une très bonne performance de l'approche proposée, démontrant son efficacité et sa précision dans les tâches de recommandation.

Mots clés: Système de recommendation, Analyse des sentiments, Support Vector Machine, Naive Bayes, Logistic Regression, DistilBERT, SVD, NCF.

الملخص

يعد نظام التوصية مكونًا أساسيًا في مجالات متعددة، مثل التجارة الإلكترونية والرعاية الصحية ووسائل التواصل الاجتماعي، حيث يتنبأ بتفضيلات المستخدم من خلال تقديم اقتراحات ذات صلة. في هذا العمل، لتحسين أداء أنظمة التوصية، نقترح نهجًا جديدًا للتوصية قمنا بتطبيقه على مجموعة بيانات Yelp Business .

يعتمد هذا النهج على تحليل المشاعر باستخدام ثلاث طرق للتعلم الآلي (ML) : آلات دعم المتجهات (SVM) ، تصنيف نايف بايز (NB) ، والانحدار اللوجستي (LR) ، وتمثيلات التشفير ثنائية الاتجاه من المحولات DistilBERT .

يهدف النهج المقترح إلى تحسين التوصيات من خلال تحليل المشاعر لمراجعات المستخدمين. نحن نستخدم طريقة تحليل القيمة المفردة (SVD) للمساعدة في معالجة التباعد في بيانات التفاعل بين المستخدم والعنصر، مما يضمن الحد الأدنى من نتائج التباين مع مدخلات بيانات محدودة أيضًا.

أظهرت النتائج باستخدام التصفية التعاونية العصبية (NCF) كنموذج توصية أداءً جيدًا للنهج المقترح، مما يدل على كفاءته ودقته في مهام التوصية.

الكلمات المفتاحية: نظام التوصية، تحليل المشاعر، آلة المتجهات الداعمة، نايف بايز، الانحدار اللوجستي، ديستيل بيرت، تحليل القيم الفردية، الشبكات العصبية التعاونية.

Table of contents

List of Tables

List of Figures

General Introduction

Recommender systems help users find relevant items by suggesting items that match their interests and preferences. They are important in domains like e-commerce, healthcare, and social media, where users need help discovering new products, movies, books, and more.

However, traditional recommendation methods often struggle with several challenges that impact their accuracy and relevance. One major issue is the sparsity of interaction data between users and items, which means that many users have only interacted with a small subset of items. This lack of sufficient interaction data can make it difficult for the system to accurately infer user preferences. Furthermore, traditional methods may not effectively capture the nuanced preferences of users, because they primarily rely on ratings and overlook the rich qualitative data found in user reviews and comments. These limitations can result in recommendations that do not fully reflect what users like or want, leading to less personalized and less effective suggestions.

User-generated feedback, such as reviews and comments, contains valuable information about products and users' opinions. As people share more reviews and comments online, this data grows rapidly. These reviews include users' thoughts, sentiments, and emotions about things they've bought, books they've read, music they've listened to, and more.

Sentiment analysis, a branch of natural language processing (NLP), can improve recommender systems by analyzing these textual reviews. It helps extract insights about users' preferences and feelings towards specific items or services. This allows recommender systems to understand users better and provide more personalized suggestions.

This thesis explores how to integrate sentiment analysis into recommender systems to improve their performance. The goal is to use sentiment analysis to make recommendations more accurate and relevant. By analyzing user reviews, the system can make better suggestions based on both past interactions and the sentiments expressed in reviews.

The study explains the methodology for adding sentiment analysis to recommender systems. It covers analysing textual data, using sentiment analysis models like Support Vector Machines (SVM), Naive Bayes (NB), Logistic Regression, and DistilBERT, and incorporating these insights into the recommendation algorithm.

To test this approach, we used a subset of Yelp business data that includes 11,344 reviews of hospitals in the United States. By showing the benefits of adding sentiment analysis to recommender systems, this thesis helps advance our understanding of how NLP techniques can improve recommendation algorithms. It highlights the importance of using user-generated data to make more personalized and effective recommendations in various fields, including e-commerce, social media and healthcare for instance.

The thesis is organized as follows:

- The first chapter provides an overview of recommendation systems and sentiment analysis, and how they can be integrated.
- The second chapter describes the design of our system, detailing the proposed approach and the techniques used.
- The last chapter discusses the implementation of our system, the working environment, the libraries used, and the training and testing details. It also presents and discusses the results.
- The thesis concludes with a general conclusion and future perspectives.

Chapter 1

Recommendation systems and sentiment analysis

1.1 Introduction

Sentiment analysis is task of natural language processing that involves extracting information from text and automatically identifying the polarity (positive, negative, or neutral) of the sentiment within the text, it is widely used in different application domains such as recommendation systems which are systems that suggest relevant or interesting items to users by filtering valuable information to find the most relevant items.

This chapter provides an overview of recommendation systems and sentiment analysis, and their integration for personalized recommendations. It begins by defining recommendation systems, their objectives, and the terminology used. It then discusses the different approaches to recommendation, including content-based filtering, collaborative filtering, and hybrid filtering.

The chapter then introduces the concept of sentiment analysis, it also highlights the difference between sentiment analysis and emotion analysis. It explores the different levels of sentiment analysis, including aspect-level, sentence-level, and document-level analysis. It also discusses the various applications of sentiment analysis, such as in e-commerce, healthcare, recommendation systems, and other domains.

Finally, the chapter presents related works on the integration of sentiment analysis into recommendation systems across various application domains.

1.2 Recommendation systems

1.2.1 Definition

[Burke, 2002] defined recommendation systems as: *"any system that produces individualized recommendations as output or has the effect of guiding the user in a personalized way to interesting or useful objects in a large space of possible options"*.

In other words, recommendation systems suggest relevant or interesting items to a user by filtering the valuable information to find the most relevant items [Vaidya et Khachane., 2017].
These systems are widely used in various fields such as e-commerce [Karthik et Ganapathy, 2021], streaming services [Ramadhan et Setiawan, 2022], social media [Mouzhi Ge et D'Auria, 2023], e-learning [Mehenaoui, 2018], etc.

1.2.2 Objective

The general objective of recommendation systems is to generate meaningful and personalized recommendations of items such as products, services, etc., for a set of users. Some specific objectives of recommendation systems include:

- Improving the quality of user experience.
- Optimizing performance by targeting important indicators, such as reducing the time required for navigation and searching.
- Implementing efficient management of a considerable volume of data that exceeds the capacities of manual processing.
- Automation of the data filtering task, thus enabling more efficient information management [Hacene et Tekia, 2021].
- Guiding the user by facilitating the discovery of new items through providing tailored suggestions.

1.2.3 Adopted Terminology

A. Item

An "item" can represent a product, an article, a service, or any other item that the system searches to recommend to a user [Souilah, 2019].

B. Profile

a "profile" of an object is a set of characteristics allowing it to be represented, it can represent a user or an item where:

- User "profile" represents a set of information about a user's preferences or past interactions with items. These profiles are often used to personalize recommendations based on the specific tastes and interests of each user.
- Item "profile" represents a description of the content of a resource which can be a document or an image, etc., through a set of keywords, generally weighted [Mehenaoui, 2018].

C. Vote

The term "vote" refers to an explicit action taken by a user to express their evaluation like rating or opinion on a particular item. Votes can take different forms, depending on the platform or recommendation system, such as numerical values (like from 1 to 5 stars), binary (like and dislike), etc [Souilah, 2019].

D. Similarity

The concept of "similarity" refers to the measure of resemblance or proximity between items, such as users, articles, products, or other entities. This measure of similarity is used to assess how much two elements are alike to each other.

1.2.4 Different approaches to recommendation

Recommendation systems are classified based on their filtering algorithms. There are several classifications of these algorithms, including the classical classification proposed by [Adomavicius et Tuzhilin, 2005], the classification by [Su et Khoshgoftaar, 2009], and the classification by [Rao, 2008]. These classifications are shown in the figure 1.1.

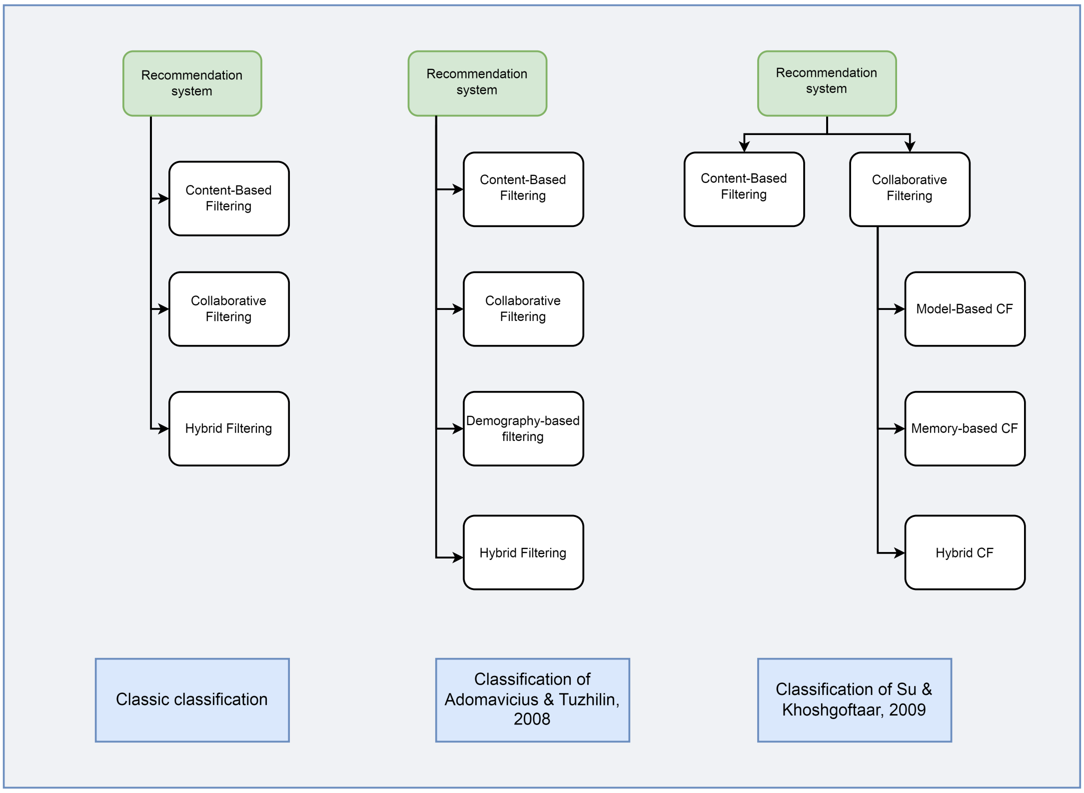


Figure 1.1: *Different classifications of recommendation systems adapted from [Souilah, 2019]*

Among these three classifications, the classical classification is the most common and widely adopted. Filtering algorithms can be categorized according to the classical classification to content-based filtering, collaborative filtering, and hybrid filtering, as illustrated in the following figure:

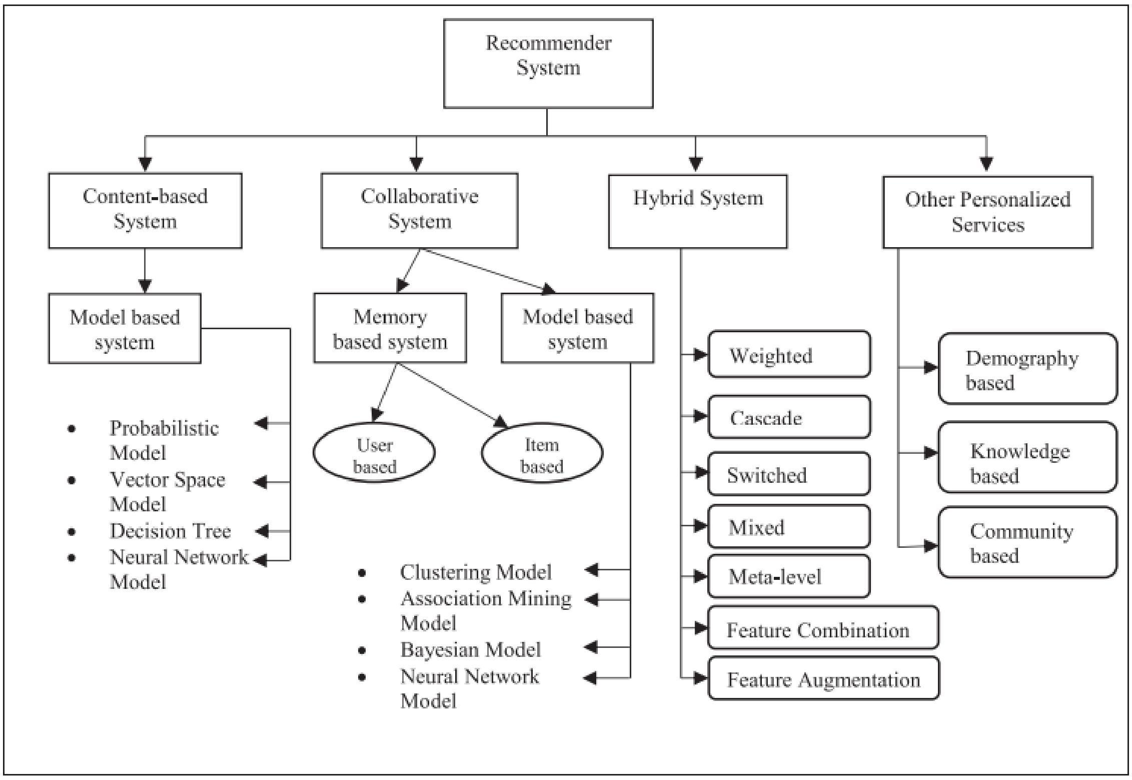


Figure 1.2: *Recommendation approaches [Sinha et Dhanalakshmi, 2022]*

content-based filtering

Content-based recommendation systems work by using item descriptions and a user interest profile to recommend the most relevant items to a specific user. Although the details of these types of recommendation systems vary from one application to another, the common general idea is to provide recommendations to a user by describing all items that can be recommended and comparing them with a user-created profile that describes his interests [Pazzani et Billsus, 2007].

This type of recommendation system relies on the user's past behavior, such as ratings. It works by constructing a user profile that encapsulates these preferences and analyzing the content of this profile [Hacene et Tekia, 2021].

Content-based recommendation systems compare the user profile, represented by the items that the user has already rated, with unrated items by calculating their similarity. This type of recommendation system requires both the user profile and the item profile to generate recommendations based on the correlation between these two entities [Majda, 2016].

Collaborative Filtering

Unlike content-based filtering, the collaborative filtering takes into consideration interactions or ratings from similar users regarding a target item. This approach is based on the premise that users with common interests in certain items generally have similar preferences for other items. They are attempting to predict the utility denoted as utility(U, i) of an item i for a user U based on the utilities (Us, i) assigned to the item i previously evaluated by the set of users Us who exhibit a similarity of interests with the user U [Mehenaoui, 2018, Majda, 2016].

Collaborative filtering can be classified into two categories of algorithms: memory-based algorithms and model-based algorithms [Breese *et al.*, 1998].

1. **Memory-based collaborative filtering:**

   In these algorithms, also known as neighborhood-based [Majda, 2016], based on a sample of other users from the database, a vote for a specific user can be predicted [Breese *et al.*, 1998]. As shown in the figure 1.2, memory-based algorithms can be further divided into item-based algorithms and user-based algorithms.

   - **Item-based algorithms:** In item-based collaborative filtering, recommendations are made based on the similarity between items. That is, if item **i** and item **j** are frequently liked by the same users and a user likes item **i**, then item **j** might be recommended because it is similar to item **i**.
   - **User-based algorithms :** In user-based collaborative filtering, recommendations are made based on the preferences of users similar to the target user. For example, if user **U** and user **V** have similar tastes and preferences, the items liked by user **V** but not yet seen by user **U** may be recommended to user **U**.

2. **Model-based collaborative filtering:**

   From a probabilistic standpoint, this method predicts a vote that has not yet been evaluated by using the user database to learn and estimate a model [Breese *et al.*, 1998].

   Model-based techniques analyze the utility matrix containing the ratings assigned to items by users to identify relationships within the dataset. This process involves the use of machine learning and data mining techniques, such as regression and clustering, to compare the list of the top-k recommendations [Sinha et Dhanalakshmi, 2022, F.O. Isinkaye et c, 2015].

Hybrid filtering

This type of filtering involves merging the two previous approaches (content-based filtering and collaborative filtering), making use of the advantages of each while mitigating the drawbacks associated with their separate use to achieve improved performance [Majda, 2016].

Other personalized recommendation systems:

As illustrated in the preceding figure, labeled as Figure 1.2, various approaches to recommendation systems are observed, including demography-based filtering, knowledge-based filtering, and community-based filtering [Sinha et Dhanalakshmi, 2022].

1. **Demography-based filtering**:

   Recommendations are generated by utilizing users' demographic data, such as age and gender, based on categorizing information according to these data [Majda, 2016].

2. **Knowledge-based filtering**:

   The principle of this type of filtering is to suggest items based on inferences regarding users' needs and preferences. Knowledge-based approaches possess functional knowledge, meaning they have an understanding of how a particular item meets the specific needs of a user, allowing them to reason about the relationship between needs and possible recommendations [Burke, 2002].

3. **Community-based filtering:**

   Communities characterized by common interests form, user-item interaction is used within these communities, and items are recommended based on the combined decision obtained from the community [Sinha et Dhanalakshmi, 2022].

1.2.5 Process

The process of creating a recommendation system relies on considering several factors, including the type of data available in the database, encompassing ratings and characteristics of items that can be classified. Additionally, the chosen filtering approach, the type of model used—whether memory-based or model-based and the use of specific techniques such as probabilistic approaches and bio-inspired algorithms are essential. Furthermore, the performance metrics of the system, considering both time and memory consumption, play a crucial role. The pursued objectives, such as predictions and top N recommendations, along with the desired quality of results, encompassing factors like novelty, coverage, and accuracy, are also taken into account [Bobadilla *et al.*, 2013].

1.2.6 Advantages and disadvantages of Different Recommendation Approaches

Table 1.1 summarizes some of the advantages and disadvantages of various filtering techniques:

Filtering Approach	Advantages	Disadvantages
Content-Based Filtering	Recommendations are generated by analyzing the user's profile and the attributes of items. Therefore, this type of filtering does not require the calculation of similarity between users. New items can be recommended even when there is not enough information from other users. This approach offers the ability to quickly adjust recommendations if the preferences of the user change, while ensuring user privacy since they can receive recommendations without sharing their profiles.	It requires a significant amount of information, such as well-organized item descriptions and user profiles, to create a successful recommendation system. However, this type of filtering may encounter the issue of over-specialization by recommending items similar to those already defined in user's profile.
Collaborative Filtering	Collaborative Filtering (CF) algorithms perform well in large user spaces, providing a diverse personalized list. Additionally, it is not necessary to have domain information in the initial recommendation process. They offer the ability to recommend relevant items to a user even when the content is not in the user's profile.	Collaborative Filtering (CF) faces major challenges such as the cold-start problem, making it difficult to suggest items not encountered during the training phase. Complexity increases with large datasets, requiring the calculation of similarity indices for numerous users. Additionally, the sparsity of real-world data can impact the recommendation system.
Hybrid Filtering	Utilizing a combination of different filtering techniques can yield effective results and optimize the recommendation system. Moreover, it mitigates drawbacks compared to using a single approach.	As hybridization involves employing various filtering techniques, it leads to high complexity (in terms of time and space) and implementation costs.

Table 1.1: *Advantages and disadvantages of different filtering approaches*

1.2.7 Evaluation of Recommendation Systems

An evaluation is a collection of research techniques and related methodologies that are specifically used to assess the performance and effectiveness [Zangerle et Bauer, 2022]. An evaluation of a recommender system seeks to determine how well it serves users by offering relevant and customized recommendations; however, this process is challenging due to the possibility that algorithms will perform differently on various datasets, evaluation goals may vary, and selecting appropriate metrics to compare different approaches is a challenging task.
A recommender system can be evaluated in several ways. We are going to perform the most popular kind in the research community in this study, which is known as offline evaluation, which means the use of a pre-collected dataset containing users' explicit (like ratings) or implicit (like purchased or clicked items) feedback on items to simulate user behavior based on historical interactions [Zangerle et Bauer, 2022]. This kind of evaluation employs the following evaluation metrics:

- **Precision**: Precision measures the percentage of recommended items that are also relevant [Zangerle et Bauer, 2022]. Here is its equation:

$$\text{Precision} = \frac{|\text{Relevant Items} \cap \text{Recommended Items}|}{|\text{Recommended Items}|} \tag{1.1}$$

- **Recall**: Recall quantifies the percentage of relevant items that are actually recommended [Zangerle et Bauer, 2022]. Its equation is:

$$\text{Recall} = \frac{|\text{Relevant Items} \cap \text{Recommended Items}|}{|\text{Relevant Items}|} \tag{1.2}$$

- **Area under the ROC curve (AUC)**: This is Main tool for usage prediction,The True positive rate be plotted against False positive rate. In recommendation system context, it is probability of recommending a relevant item higher than recommending not relevant item. The curves can be combined into a single score by using such metric [Zangerle et Bauer, 2022].

- **Mean Absolute Error (MAE)**: It is the average absolute deviation of the actual rating of an item by a user and a predicted rating for that item [Herlocker *et al.*, 2004]. Its formula is given by:

$$\text{MAE} = \frac{1}{N}\sum_{i=1}^{N}|y_i - \hat{y}_i| \tag{1.3}$$

where : N is the total number of predictions,
y_i is the actual rating,
$\hat{y}_i$ is the predicted rating.

- **Root Mean Squared Error (RMSE)**: It is a statistic that is computed as the square root of the average of the squared differences and is used to quantify the average magnitude of errors between predicted and actual values [Herlocker *et al.*, 2004]. It can be calculated as follows:

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2} \tag{1.4}$$

1.3 Sentiment Analysis

1.3.1 Definition of Natural Language Processing

Natural Language Processing (NLP) is a branch of artificial intelligence and linguistics, its goal is to enable computers to comprehend words or sentences written in human languages. The basic ideas of NLP come from linguistics, the scientific study of language and its structure. NLP can be divided into two categories: Natural Language Generation (NLG), which advances the task of producing texts, and Natural Language Understanding (NLU), or linguistics, which develops written content understanding. It came into being to make user work easier and fulfill the desire to speak with computers in natural language [Khurana *et al.*, 2023].

1.3.2 Definition of Sentiment

Sentiment is the expression of a view or opinion towards something, such as products or services. It can be determined verbally or through writing. Sentiment reflects a person's positive or negative feelings [Tyagi et Sharma, 2018].

1.3.3 Definition of Sentiment Analysis

Sentiment analysis, sometimes mentioned as Opinion Mining [Yue *et al.*, 2019], is a task of Natural Language Processing (NLP) that involves extracting information presented in text and automatically identifying subjectivity: opinion or sentiment within that written text [ali *et al.*, 2021]. Moreover, emerging sentiment analysis methods are beginning to integrate data from textual sources and other forms of media, including audio and visual data [Ortis *et al.*, 2020, Poria *et al.*, 2016]. It can be categorized as a sub-field of effective computing, which is a developing research field with the objective of enabling intelligent systems to identify, experience, and interpret human emotions [Poria *et al.*, 2017].

The objective of sentiment analysis is to determine the contextual polarity (positive, negative, or neutral) of people's sentiments and discover their opinions toward products, services, etc., in order to get useful information.

1.3.4 Difference between Sentiment Analysis and Emotion Analysis

There is a lot of terminology used in the field of sentiment analysis due to its richness and wide usage within the research community. Thus, the terms *sentiment analysis* and *emotion analysis* have been used interchangeably sometimes. However, it's important to mention that they differ in some aspects: sentiment analysis, as mentioned in the definition, means extracting the polarity of the data, whether it's positive, negative, or neutral, while emotion detection is the identification of human emotion types such as happiness, sadness, love, etc. [Nandwani et Verma, 2021]. According to E. Cambria et al. [Ramzy et Ibrahim, 2024], emotion detection is a sub-task of sentiment analysis.

1.3.5 Levels of sentiment analysis

Sentiment analysis can be done at three levels: sentiments or opinions can be detected at the aspect level, sentence level, or document level:

Aspect-level sentiment analysis

This level involves determining sentiments and opinions regarding a specific aspect of an entity (such as service, product, etc.) [Nandwani et Verma, 2021]. It performs fine-grained analysis. For instance, considering the following review: "the camera of Samsung S24 Ultra is awesome". Here, the camera represents an aspect of the entity Samsung S24 Ultra, and the polarity of the review is positive. The benefit of this level is the ability to detect what users like and what they don't because it goes beyond discovering the general sentiment of a sentence or a paragraph to focus on the aspects of the entities [ali *et al.*, 2021]

Sentence-level sentiment analysis

The objective of this level is to identify the polarity of a sentence, i.e., whether it expresses a positive, negative, or neutral opinion. So, this level focuses on the sentence by splitting documents and paragraphs into sentences [Nandwani et Verma, 2021, ali *et al.*, 2021]. In this level, we deal with two main classifications: subjectivity classification and sentiment classification. Subjectivity classification involves determining whether a sentence presents objective or subjective information while sentiment classification categorizes subjective sentences into either positive or negative expressions [Liu *et al.*, 2010].

Document-level sentiment analysis

Sentiment analysis at the document level involves identifying the overall sentiment presenting in the entire document regarding a singular entity, such as a specific product. This task poses challenges as it necessitates consideration of the links between words and sentences, as well as the contextual semantic meaning embedded within the document and when the document evaluates or compares multiple entities, the complexity of sentiment analysis further increases [Nandwani et Verma, 2021].

1.3.6 Sentiment analysis techniques

In order to improve the performance of sentiment analysis and propose solutions for its challenges, researchers apply, compare, and hybridize different sentiment analysis techniques. These techniques are usually divided into three categories: lexicon-based approach, machine learning-based approach, and hybrid-based approach. The figure 1.3 summarizes different sentiment analysis techniques.

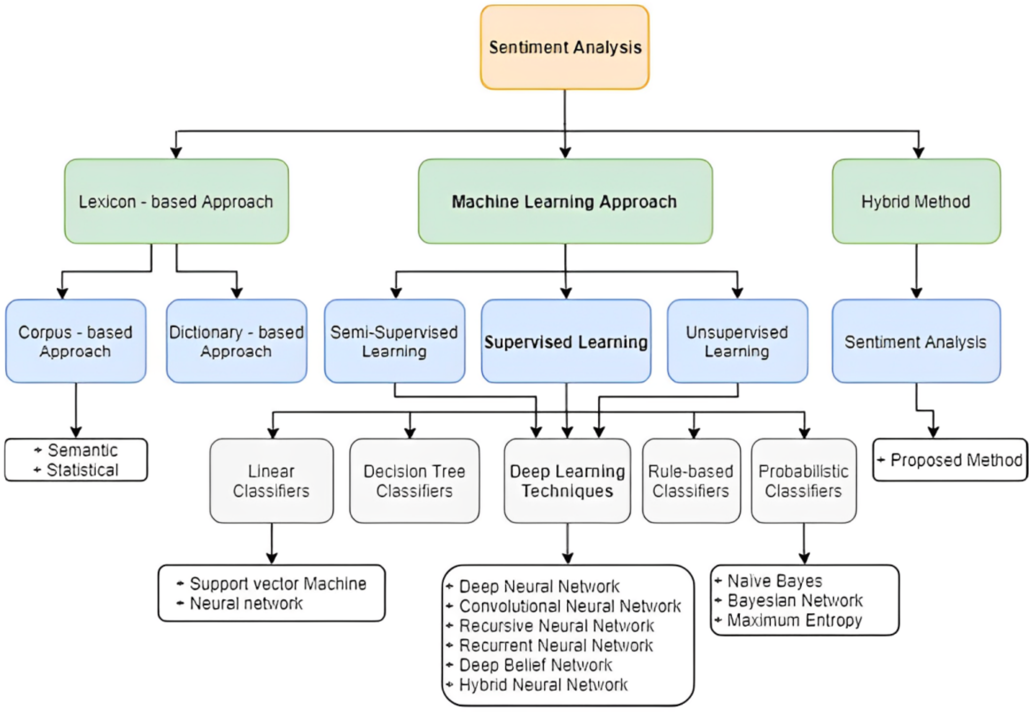


Figure 1.3: *Sentiment analysis techniques [Dang et al., 2021]*

Lexicon-based approach

The lexicon-based approach classifies textual data relying on a dictionary of labeled sentiment words associated with their sentiment scores. The overall sentiment of the entire sentence is determined by either the mean or the sum of all sentiment scores [Nandwani et Verma, 2021].
There are two types of lexicon-based approach: dictionary-based approach and corpus-based approach.

1. **Dictionary-based approach**: This approach utilizes a seed, which means a small set of sentiment words associated with their polarity labels collected manually, to create a dictionary which is maintained by looking for their synonyms and antonyms from online dictionaries such as WordNet [Liu, 2022].

2. **Corpus-based approach**: Because the dictionary-based approach is not domain-specific, this approach solves this limitation of by incorporating domain-specific sentiment words with their corresponding sentiment scores based on the context of the study [Nandwani et Verma, 2021].

Machine learning approaches

By relying on machine learning algorithms, machine learning approaches can be applied in sentiment analysis and text classification problems [Medhat *et al.*, 2014]. They involve classifying sentiment polarity (for example: positive, negative, and neutral) based on both training and test sets [ali *et al.*, 2021]. The training set is used to train the machine learning model by providing properties for different instances of the item, while the testing set is used to measure the performance of the model [Nandwani et Verma, 2021].

Figure 1.4 represents the process of sentiment polarity classification using machine learning approaches.

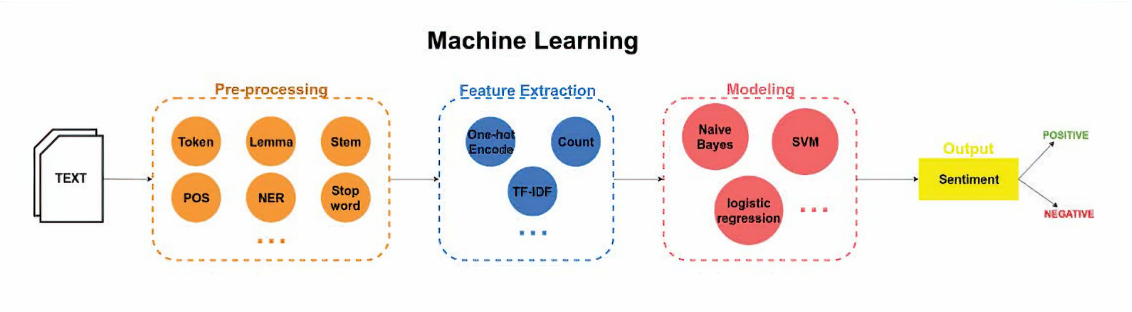


Figure 1.4: Process of sentiment polarity classification using machine learning approaches [Dang et al., 2020]

Machine learning approaches can be divided into supervised learning, unsupervised learning, semi-supervised learning and deep learning.

1. **Supervised learning**

   Supervised learning is the machine learning approach that requires datasets labeled with classes, typically referred to as labels (for example: positive and negative). Generally, algorithms utilized in sentiment analysis and sentiment classification fall under supervised learning approaches [Nandwani et Verma, 2021]. These algorithms may include Support Vector Machine (SVM), Naïve Bayes (NB), Logistic Regression (LR) and more.

   - **Support Vector machine**: The Support Vector Machine (SVM) is a supervised machine learning technique proposed by Vapnik in [Vapnik, 2013] that is typically used for regression and classification tasks. This method operates by identifying the optimal possible line, known as a hyperplane, within a high-dimensional space that called the feature space in order to separates data points into different classes. The margin, defined as the distance between the hyperplane and the nearest data points (also known as support vectors) from each class, is maximized during the hyperplane selection process. By maximizing the margin, the algorithm's resistance to overfitting is enhanced, consequently improving its capacity to generalize to new data [Cervantes *et al.*, 2020]. The figure 1.5 shows the separation of two classes using SVM where the ideal separating hyperplane is located in the middle of the margin, and the points that are on the borders are support vectors.

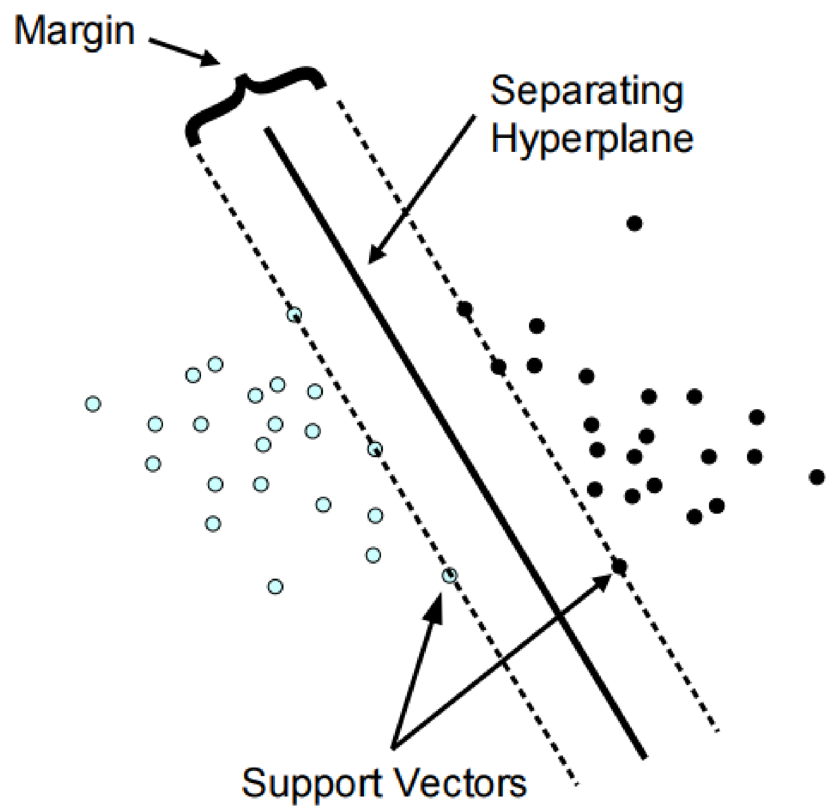


Figure 1.5: *Separation hyperplane of SVM in two classes classification case [Meyer et Wien, 2001]*

SVM has been utilized in many studies; [Rana et Singh, 2016] employed linear SVM and Naïve Bayes for analyzing movie reviews. According to their findings, the linear SVM approach had the highest accuracy of 75%. Thus, we select SVM as one of the classifiers to perform sentiment analysis.

- **Naive Bayes**: The Naïve Bayes classifier, a popular Bayesian network classifier, is a probabilistic classifier that falls under supervised machine learning that uses the Bayes theorem under the assumption of Naïve (Strong) independence. Bayesian networks are graphical models in which each node denotes a variable and each edge denotes a conditional dependency. They are used to illustrate the probabilistic relationships between a set of variables. As a conditional probability model, the Naïve Bayes model determines an event's likelihood based on previously known conditions that may be connected to it. The probability of an event happening in the presence of another event that has already happened is known as conditional probability. The Naïve Bayes classifier becomes a popular baseline technique for text classification, handling the problem of document classification only based on word frequencies. In text retrieval, pertinent data must be located across a vast text repository. To help in this process, the Naïve Bayes classifier is used to classify and retrieve documents according to their content.
The Naive Bayes classifier applies Bayes' rule to classify a document d into category c whenever it maximizes $P(c \mid d)$ [ali *et al.*, 2021]. Bayes' rule is given

by:

$$P(c \mid d) = \frac{P(d \mid c)P(c)}{P(d)}$$

where:

- $P(c \mid d)$ is the posterior probability of category c being assigned to document d,
- $P(d \mid c)$ is the likelihood of document d given category c,
- $P(c)$ is the prior probability of category c,
- $P(d)$ is the prior probability of document d.

In numerous studies, the Naïve Bayes classifier has demonstrated satisfactory performance even with its robust assumptions and simplicity [Dey *et al.*, 2016]. For instance, [Wongkar et Angdresey, 2019] used Naive Bayes sentiment analysis on tweets on the Republic of Indonesia's 2019 presidential candidates and contrasted the results with SVM and K-NN. Naive Bayes yielded the best results With an accuracy of 80.90%.

- **Logistic Regression**: A supervised machine learning technique used for predictive analysis and classification tasks. Linear regression, a basic machine learning algorithm that utilizes a linear equation to represent the connection between a dependent variable and one or more independent variables. the main applications for it are regression problems, where the objective is to predict a continuous output. The logistic regression algorithm applies the linear regression equation; however, in contrast to linear regression, its cost function is the sigmoid function. This function, often known as the logistic function, is an S-shaped curve. This algorithm's hypothesis is the limitation of the logistic function to a range of 0 and 1, facilitating the conversion of the outputs of linear regression into probabilities appropriate for classification tasks [Bewick *et al.*, 2005]. Figure 1.6 illustrates the Logistic Regression's working process.

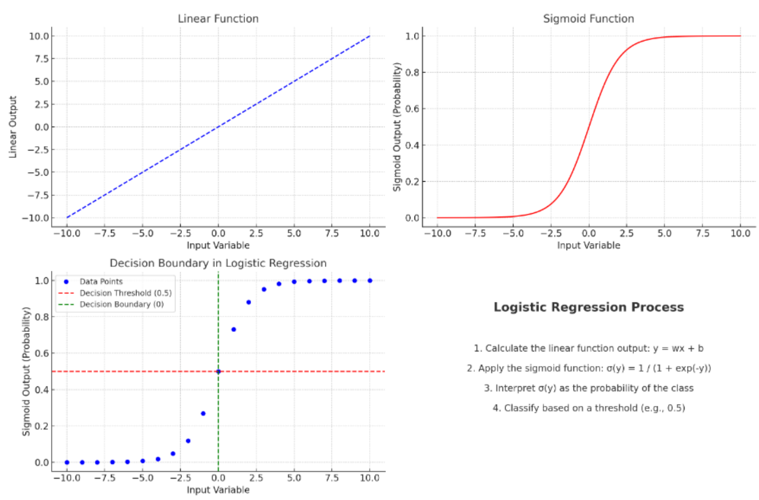


Figure 1.6: *Logistic Regression's working process*

Logistic regression has been widely employed in numerous studies on sentiment analysis, demonstrating its efficiency. For example, in the study by [Aliman *et al.*, 2022], it outperformed the Support Vector Classifier, Stochastic Gradient Descent, and Naive Bayes, achieving an accuracy of 81%, compared to 69 %, 71%, and 77%, respectively, for the others.

- **Artificial Neural Network**: It is a model inspired by the structure and function of the human brain. This network relies on the notion of taking input data and separating its features into linear combinations; the output is then modeled as a nonlinear function of these features.A typical neural network architecture consists of three layers: the *input layer*, which receives initial data for processing; *hidden layers*, which perform intermediate computations to transform input into meaningful representations; and the *output layer*, which produces the final predictions or outputs based on the processed information. Neurons in each layer are connected to neurons in the succeeding layer through links, each with an associated weight value estimated by minimizing a global error function using gradient descent training [ali *et al.*, 2021]

2. **Unsupervised learning**

   Even though most of sentiment analysis's utilized approaches employ supervised learning, sometimes collecting and creating datasets can be challenging, considering that textual data is unstructured most of the time. Therefore, some researchers have turned to using unsupervised learning, which relies on unlabeled datasets by dividing users into groups that have similar properties (clusters) [ali *et al.*, 2021].

3. **Semi-supervised learning**

   Semi-supervised learning relies on datasets containing a limited set of labeled training data to guide the feature learning process. Consequently, it represents a fusion of supervised and unsupervised approaches [ali *et al.*, 2021].

4. **Deep learning**

   Deep learning, a subset of machine learning, utilizes artificial neural networks (ANN), which were discussed in the previous section. Unlike traditional machine learning approaches that require feature selection methods such as TF-IDF, deep learning employs multiple layers of perceptrons to automatically learn data representations and extract features [Dang *et al.*, 2020]. This ability to automatically define and extract features without manual intervention has led to an increasing use of deep learning in fields like sentiment analysis, where it surpasses traditional machine learning due to its capacity to detect sentiments from written language effectively, especially on large datasets [Nandwani et Verma, 2021]. The process of sentiment polarity classification using deep learning approaches is illustrated in Figure 1.7.

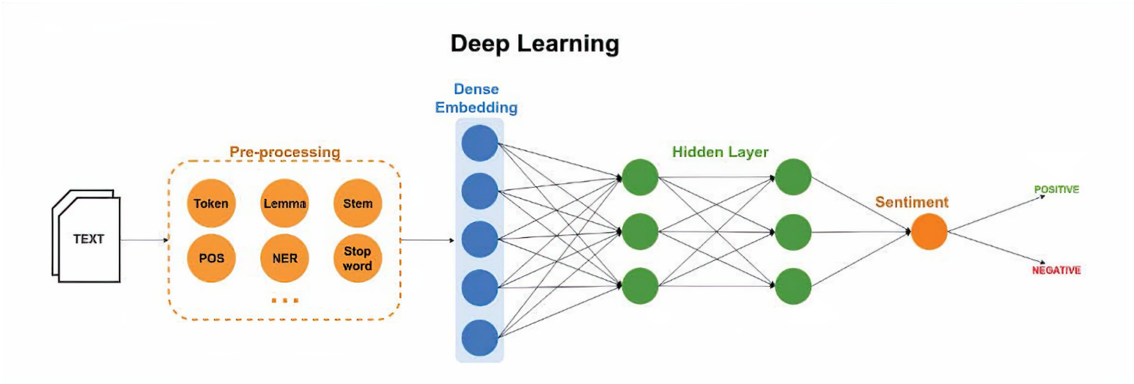


Figure 1.7: *Process of sentiment polarity classification using deep learning approaches [Dang et al., 2020]*

Transformers:

The Transformer model is a type of neural network architecture that does not rely on convolutions or recurrence, which are typical in sequence transduction models. Instead, it relies primarily on attention mechanisms. The encoder and decoder stacks, which are the main parts of the Transformer model, are composed of several layers with feed-forward networks and self-attention mechanisms. A multi-head self-attention mechanism and a fully connected feed-forward network are the two sub-layers found in each of the N identical layers which make the encoder in the Transformer model. A residual connection and layer normalization, which converts an input sequence of symbol representations into continuous representations, come after each sub-layer. Similar to the encoder, the decoder is composed of a stack of N identical layers with the same two sub-layers. It generates an output sequence of symbols by applying layer normalization and a residual connection to each sub-layer, which are based on the continuous representations the encoder produced. By enabling words in a sequence to focus on other words in the same sequence, the Transformer model's self-attention mechanism captures dependencies between distinct places and enables the model to take global dependencies into consideration without requiring sequential computation. The Transformer model utilizes multi-head attention to improve its capacity to capture complex relationships within the input sequences by allowing it to jointly attend to information from different representation subspaces at different positions. This improves the model's ability to learn dependencies between distant positions. The Transformer model makes use of layer normalization and residual connections to enhance training and network flow. Layer normalization ensures stable training by normalizing the inputs to each layer, while residual connections reduce the problem of vanishing gradients [Vaswani *et al.*, 2017].
The architecture of transformer model is illustrated in Figure 1.8.

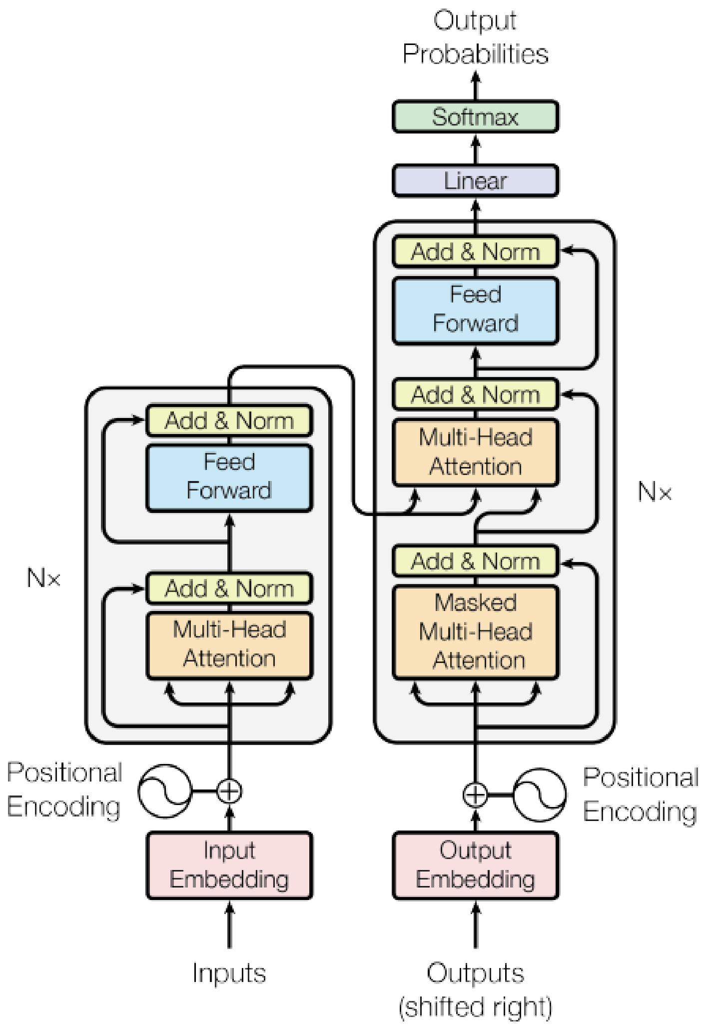


Figure 1.8: *The architecture of transformer model [Vaswani et al., 2017]*

BERT

BERT(Bidirectional Encoder Representations from Transformers) is a Pre-trained by a Transformer architecture neural network Yet BERT is bidirectional, making it capable of accounting for all words in a sentence when predicting missing words - not only those that come before or after a token. BERT uses a bidirectional multi-layer transformer model to create the contextual embeddings of words in a sentence. It also describes two additional training tasks, a masked language model and a next sentence prediction. Sentence Prediction helps the model to learn how does the position of words change in a sentence and between two sentence.
BERT was trained using a large corpus of unlabeled text (3.3 billion words of English from books or web pages) . The model was provided with partially masked input sequences and was trained to predict the randomly masked out words. Then came the second sentence prediction test. That is, giving two sentences to the model and asking, if the second sentence comes after the first sentence or not in the original text. By finishing this particular task, the model gets a better idea of the relationships between sentences. BERT was fine-tuned with many different focused NLP

tasks (from named entity recognition, sentiment analysis, to question answering) to provide new state-of-the-art results in multiple cases [Devlin *et al.*, 2018].

- **DistilBERT**: It is a transformer-based language model created by applying a model compression technique known as knowledge distillation to the BERT model. Knowledge distillation involves transferring knowledge from a large, complex model (referred to as the "teacher" model) to a smaller, more efficient model, known as the "student" model in machine learning. The objective of this process is to reduce the size of the model while minimizing the impact on its accuracy and performance. BERT is known for its extensive parameterization, which enables deep learning of linguistic structures. However, the large number of parameters may provide issues with processing overhead and memory usage. Knowledge distillation addresses this issue by teaching a smaller student model to mimic the behavior of the larger teacher model. DistilBERT, which was pre-trained through knowledge distillation, has 40% fewer trainable parameters than BERT but maintains good performance. This reduction in size makes DistilBERT well-suited for application in environments with limited computing resources [Akpatsa *et al.*, 2022].
  Table 1.2 illustrates the distinctions between models based on DistilBERT and those based on BERT.

Trainable parameters	BERT-base	DistilBERT-base
No. of layers (transformer blocks)	12	6
No. of hidden units	768	768
No. of self-attention heads	12	12
Total trainable parameters	110 M	66 M

Table 1.2: *Difference of Trainable Parameters between BERT-base and DistilBERT-base models [Akpatsa* et al.*, 2022].*

Hybrid approach

The hybrid method merges both lexicon and machine learning techniques, blending the efficiency of lexicon analysis with the adaptability of machine learning to handle ambiguity and incorporate sentiment context. The primary objective of this hybridization is to leverage the precision of machine learning while maintaining the stability of lexicon-based approaches. By integrating strategies from both methods, the hybrid approach aims to surpass their individual constraints and capitalize on their respective advantages [ali *et al.*, 2021].

1.3.7 Applications of sentiment analysis

With the increasing use of the World Wide Web by people who share their opinions towards products, services and more, sentiment analysis becomes widely used in a large number of application domains with different languages containing English, Arabic [Ramzy et Ibrahim, 2024], Urdu [Majeed *et al.*, 2020], Indonesian [Elfajr et Sarno, 2018],etc. Some of these domains are:

1. **E-commerce**
   The main objective of applying sentiment analysis in e-commerce is to improve product quality and marketing, and to enhance user satisfaction [Jim *et al.*, 2024]. It can be applied by studying customers' feedback, and by collecting customers' reviews of different products from English e-commerce websites like Amazon unpacked product reviews platform [Rasappan *et al.*, 2024] or non-English e-commerce websites , an example of such a study was conducted on a Turkish e-commerce website named thepsiburada.com [Demircan *et al.*, 2021]. Analysing products reviews can be useful also in predicting costumers' interests so that helps in company's decision making process [Savci et Das, 2023].

2. **Healthcare and medical domain**
   Sentiment analysis and emotion analysis are used in the medical domain by monitoring patients' reviews to obtain useful information regarding the mental state and mood of patients such as detecting patients' mental illness [Kim *et al.*, 2020], analyzing drugs reviews [Gräßer *et al.*, 2018], classifying medical centers: for example in [Al-Mashhadany *et al.*, 2022] used SA of Facebook comments for Iraqi beauty centers classifications as 'healthy' and 'unhealthy'.

3. **Recommendation system**
   In order to improve recommendations, sentiment analysis has been widely used in recommendation systems across various domains. For instance, movies websites can utilize sentiment reviews to make better recommendations [Saraswat *et al.*, 2020, Kumar *et al.*, 2020]. Additionally, sentiment analysis can be applied in e-commerce to offer relevant products based on users' preferences and emotions such as books [Luţan et Bădică, 2023].

4. **Other application domains**
   Sentiment analysis and emotion detection are widely applied by many researchers, some of them use the lexicon-based approach, such as the study conducted by [Bernal et Pedraz, 2024], who analyzed press reaction reports on financial stability by creating a Spanish dictionary specifically for financial stability using those reports.
   Creating domain-specific dictionaries helps improve sentiment analysis tasks. For instance, [Ojeda-Hernández *et al.*, 2023] created a customized dictionary for sentiment classification and compared it with traditional lexicons, demonstrating better performance in sentiment analysis using Twitter Tweets Sentiment Analysis.
   One of the drawbacks researchers found in creating dictionaries manually is that it consumes a lot of time and energy. As a consequence, researchers have turned to automatically classify sentiments and emotions using machine learning approaches or using both lexicon and machine learning techniques (hybridisation). For example, in finance, a study by [Cam *et al.*, 2024] aimed to assess public mood in Turkey to predict stock market behavior in BIST30 companies by performing sentiment classification using different machine learning approaches where the best performing approach was when using Multilayer Perceptron classifier and support vector machine. [Feng, 2023] applied sentiment analysis to employees' reviews published on indeed.com and classified them using two machine learning classifiers, he found

that management sentiment has a direct impact on firms' financial outcomes.
Sentiment analysis can also improve tourism. For example, according to [Leelawat *et al.*, 2022], predicting sentiments can help increase tourism. Additionally, sentiment analysis has gained attention in politics, as researchers apply it to study public opinion about political events, such as the Russia-Ukraine war: [Al Maruf *et al.*, 2022] detected emotion and racism and analyzed sentiment regarding the Ukraine-Russia war using machine learning techniques, comparing their performance in this task.

1.4 Related works

With the increasing use of online platforms, users' reviews and comments have become widely utilized to enhance and improve the performance of recommendation systems in several studies conducted by researchers who applied this approach in different application domains.

For instance, in a study by [Revathy *et al.*, 2023] a pre-trained deep learning model (BERT) was utilized to construct a model named LyEmoBert. This model aimed to predict four emotions: happy, sad, angry, and relaxed from song lyrics using the Music4All dataset. The proposed model achieved an accuracy and precision of 92% and an F1 score of 96%.

Another music recommendation system was developed by [Calderon Vilca *et al.*, 2023]. This system employs sentiment analysis using a Multilayer Perceptron to analyze song comments collected from Twitter. The recommendation mechanism is based on the Euclidean distance between the user's sentiment vector and the vector representing each song. The results demonstrated that this recommendation system achieved an accuracy and recall rate of 80%.

researchers have focused on recommending movies. For instance, [Saraswat *et al.*, 2020] proposed a top-N recommendation model that extracts emotions from user reviews in the MovieLens 100K dataset using a lexicon-based method (WordNet). The recommendation system achieved the highest precision (0.671) when recommending the top 25 movies.
In the paper by [Pavitha *et al.*, 2022], sentiment analysis was applied to TMDb movies reviews dataset using SVM and Naive Bayes, to build a movie recommendation system using cosine similarity. The highest precision and recall were achieved using SVM (both equal to 98%).
[Kumar *et al.*, 2020] proposed a hybrid recommendation system that uses both content-based and collaborative filtering approaches combined with lexicon-based sentiment analysis that was done using the VADER sentiment lexicon on the MovieTweetings dataset from 2014 to 2017. The recommendations were evaluated using Precision@5 and Precision@10 metrics. Compared to baseline pure hybrid and sentiment-only models, the proposed hybrid sentiment-based model achieved better precision scores of 2.54 for Top-5 and 4.97 for Top-10 recommendations.

Within the e-commerce domain, a key objective for many companies is to maximize profitability through understanding customer behavior. This understanding allows for the development of personalized recommendation systems that cater to individual preferences. Researchers have actively pursued the development of such systems, integrating them into prominent e-commerce platforms like Amazon. For instance, [Choudhary *et al.*, 2023] constructed an emotion-based recommendation system for books. This system leveraged a deep neural network after applying lexicon-based sentiment analysis to both user ratings and reviews from the Amazon book dataset. Through experimentation with various hidden layer configurations, the authors achieved a maximum precision of 0.99, with the optimal architecture employing 16 hidden layers. In the study of [Asani *et al.*, 2021], They proposed a restaurant recommender system that uses sentiment analysis and semantic clustering to extract users' food preferences from their online comments, and then recommends nearby restaurants based on the similarity between the user's preferences and the restaurant's menu, while also considering context information such as location, opening hours, and previous customer reviews. The system achieved a precision of up to 92.8% in the top-5 recommendations.

1.5 Conclusion

In summary, the first chapter of this thesis provides a comprehensive introduction to recommendation systems and sentiment analysis, and their potential for integration to enhance personalized recommendations. The chapter highlights the importance of sentiment analysis in understanding user preferences and emotions, and its application in various domains, particularly in recommendation systems.

Overall, this chapter effectively establishes the context and motivation for the research. In the following chapter, we present our proposed model, which is a recommender system based on sentiment analysis. This model aims to leverage the insights gained from sentiment analysis to improve the accuracy and relevance of recommendations.

Chapter 2

Conception

2.1 Introduction

This chapter provides the conception of the proposed recommendation system. We begin with exploring the sentiment analysis task to the fullest - from preprocessing, through performing the classification to making the decision based on the classification. This includes multiple preprocessing techniques, techniques for feature extraction and the use of the chosen classification algorithms to successfully predict sentiment. Then we present the proposed methodology for the recommendation task and provide an in-depth explanation of its steps and methodologies. That involves the architecture of the recommendation algorithm and sentiment analysis results integration. In this chapter, we provide an extensive explanation of how the system was envisioned and the cases behind the design choices.

2.2 Objective

Traditional recommendation systems typically rely on user ratings generally to suggest items to consumers. However, in today's digital landscape, we have access to a wealth of unstructured data in the form of user-generated content, such as reviews, comments, and social media posts. These expressions encapsulate not just opinions, but also the sentiments, emotions, and nuanced experiences that users associate with products, movies, books, and more.

By leveraging sentiment analysis, we can extract valuable insights from this textual data and gain a deeper understanding of the underlying sentiments and attitudes that users have towards various items.

The primary objective of this research is to develop and deploy a recommendation system that integrates sentiment analysis techniques with artificial intelligence (AI) methodologies, such as machine learning and deep learning. By incorporating sentiment analysis into the recommendation process, we aim to enhance the accuracy and personalization of the recommendations provided to users. Through this study, we seek to evaluate the effectiveness of sentiment-aware recommendation systems compared to traditional approaches. By conducting comprehensive experiments and performance evaluations, we intend to determine whether sentiment-based recommendations offer superior outcomes compared to traditional methods.

2.3 System Architecture

The proposed system consists of two main components: the sentiment analysis phase and the recommendation phase.
The sentiment analysis phase involves classifying the sentiments of dataset reviews into three categories: satisfied, not satisfied, and neutral. This phase includes four steps: pre-processing, feature extraction, classification using three machine learning classifiers (SVM, NB, and LR) and a pretrained deep learning model (DistilBERT), and decision-making.
In the recommendation phase, a recommendation system is constructed using the sentiment scores extracted by the chosen classifier. After normalizing the scores, their weights are adjusted, and collaborative filtering is employed using Neural Collaborative Filtering (NCF). Figure 2.1 summarizes the overall architecture of the proposed system. In the following section, we detail the steps of each phase of our system.

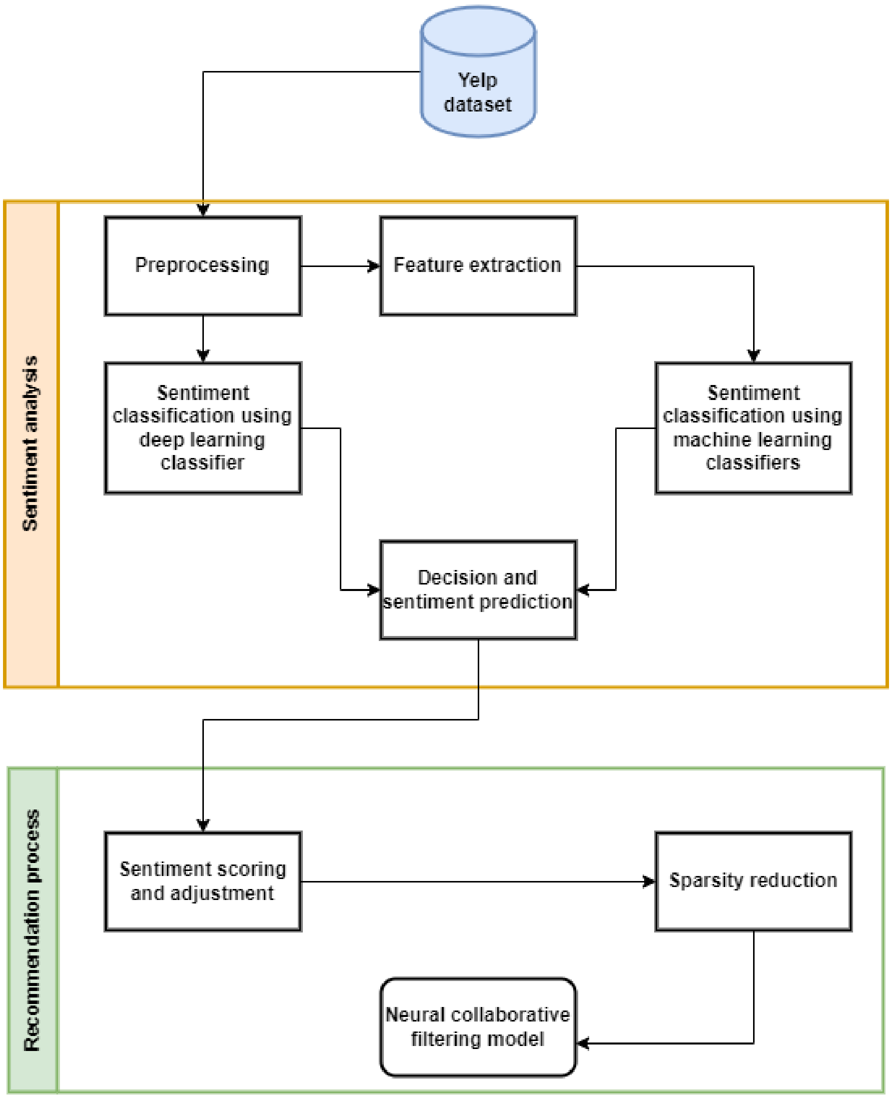


Figure 2.1: *Global architecture of the proposed system*

2.4 Approach Description

As outlined in the previous section, the proposed system goes through two main phases: sentiment analysis and recommendation system [Mehenaoui *et al.*, 2024].

2.4.1 Sentiment Analysis

In this part, we provide an approach for detecting sentiments from the analysis of Yelp user reviews in order to identify the sentimental state of the reviews for businesses that fall under the "Hospitals" category. Four steps constitute the suggested approach: preprocessing, feature extraction, classification, and decision.
Three classifiers from machine learning techniques were used to make the decision: Naive Bayes (NB), Support Vector Machines (SVM) and Logistic Regression (LR), along with a deep learning model named DistilBERT that extends Bidirectional Encoder Representations from Transformers (BERT), a type of transformer.

Pre-processing

Pre-processing involves transforming raw data into a format that is easier to understand, particularly emphasizing keywords that provide context to sentences and paragraphs. Essentially, it's about converting text into a structured format that can be easily interpreted, anticipated, and analyzed by machines using different machine learning algorithms. In our case, we used different pre-processing techniques that are used for text classification [Tabassum et Patil, 2020]. The pre-processing steps differ between machine learning models and deep learning transformer-based models. Here is an explanation of each step in the pre-processing task for machine learning classifiers:

- **Removing URL**: In order to apply sentiment classification to the chosen dataset, we manually labeled 1096 reviews. During this process, we observed that some users included URLs. Therefore, we implemented a step to remove URLs from the reviews.
- **Lower-casing**: Words are typically composed of uppercase letters. Although this stage of text preparation is frequently skipped, it is one of the easiest and most successful ones. This is especially true when the dataset is significantly sparse; it has been seen that capitalization differences ('Hospital' vs. 'hospital') affect the outcome. This indicates that the computer interprets the same words that have one capital letter and one lowercase letter as two distinct words, and in the following stages of word embeddings, two distinct word vectors are created. Thus, the optimum text preprocessing strategy has been to make all words lowercase [Tabassum et Patil, 2020]. Figure 2.2 presents a sample review before and after the application of lowercasing.

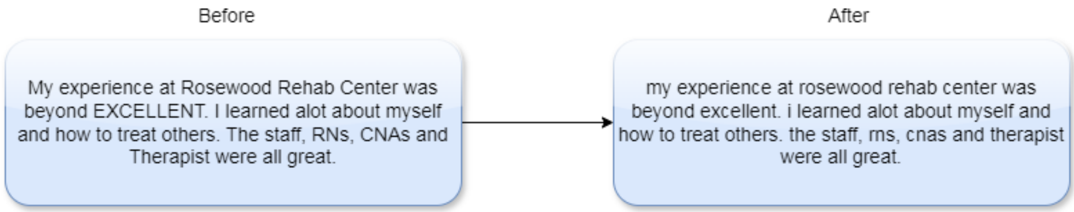


Figure 2.2: *Lowercasing*

- **Removing punctuation and non-alphanumeric tokens**: Because the machine is incapable of understanding punctuation, the text becomes noisy because of it. These types of punctuation are common in unstructured papers in particular, and include commas, apostrophes, exclamation marks, and more. Same thing about non-alphanumeric tokens that can be numbers, spaces, emojis, numbers, and other special characters [Tabassum et Patil, 2020]. They are excluded in this step as can be seen in the figure 2.3.

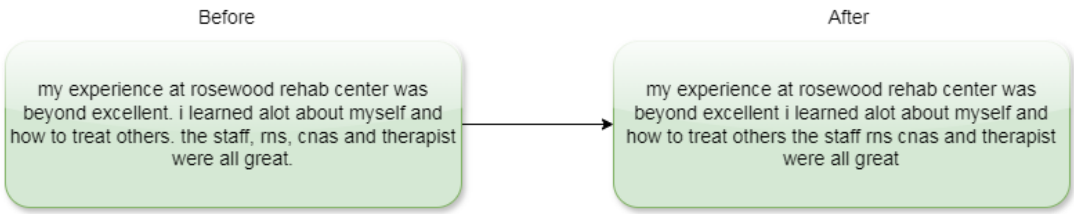


Figure 2.3: *Removing punctuation*

- **Changing 't to 'not'**: This additional step aims to preserve negation. Therefore, we replaced 't with 'not' due to its importance in reversing the meaning of a sentence.
- **Tokenization**: Tokenization is the process of dividing sentences into individual words, letters, and punctuation, which are referred to as tokens. The primary place for the splitting criterion to occur is when a punctuation or space appears. In later processing phases, this step aids in removing unnecessary terms [Tabassum et Patil, 2020] as demonstrated in the figure 2.4.

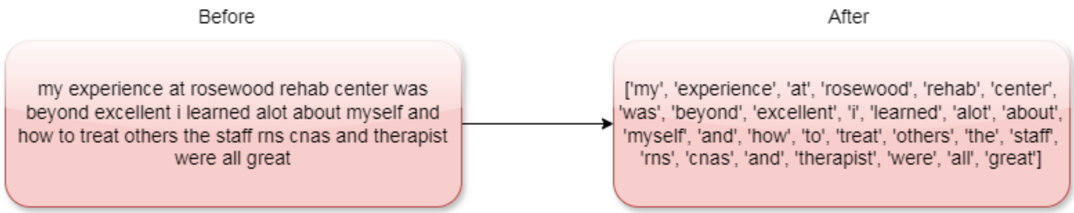


Figure 2.4: *Tokenization*

- **Removing stopwords except 'not'**: Natural language processing does not provide significance to words like "the," "are," "is," "and," and more, unless there are particular use cases. For instance, the use case for text or document classification does not provide any weight to these additional terms. Therefore, the better the results of categorization algorithms, the more these stopwords are found and cleared out. Although keywords that can make a difference are extracted, such as in sentiment classification, words like 'not' can change the overall sentiment of a sentence. That's why we excluded 'not' from the stopwords removal process [Tabassum et Patil, 2020]. Figure 2.5 illustrates a tokenized review before and after the application of stopwords removing.

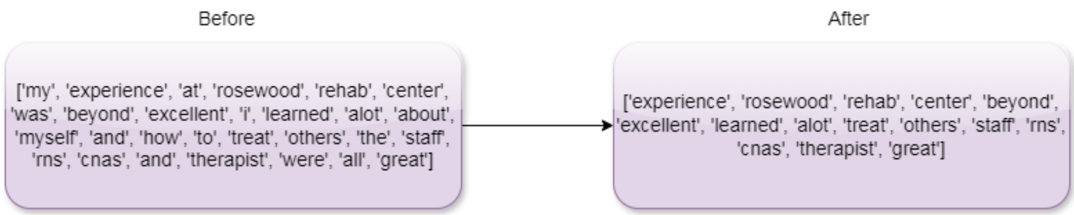


Figure 2.5: *Stopwords removing*

- **Lemmatization**: Lemmatization is the process of reducing a word to its basis, or lemma, by either eliminating it or substituting its suffix. Lemma, in opposition to stemmed words, is always meaningful. In natural language processing, lemmatization is a common text preprocessing procedure that produces excellent outcomes [Tabassum et Patil, 2020]. For example, the meaningful word "learn" is the lemma for the term "learned" as illustrated in the figure 2.6.

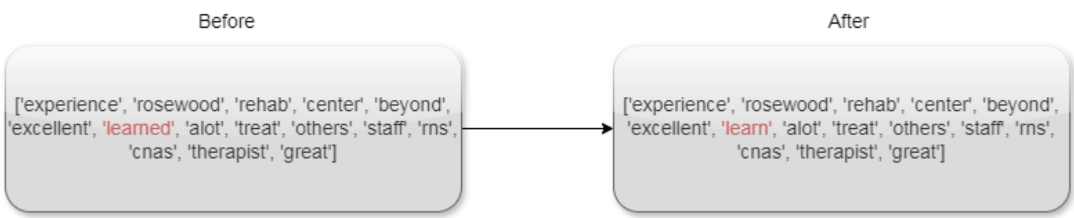


Figure 2.6: *Lemmatization*

In the other hand, DistilBERT is a deep learning model , which means we need to prepare and preprocess the text in a way that is suitable to be processed by such model. The major steps to preprocess text data when building a model with DistilBERT are:

1. **Tokenization**: The process of tokenization when using a DistilBERT model consists of:

    - *WordPiece Tokenization*: It further breaks down words into subwords or even characters. For example, tokenizing "helping" results in ["help", "ing"].

- *Add Special Tokens*: Special tokens are added at the beginning and end of the sequence. To indicate the end of a sentence or pair of sentences, the [CLS] token is added at the beginning and the [SEP] token is added at the end.

2. **Truncation and Padding**: Fixed-length sequences are the input for DistilBERT models. The actions consist of:
   - *Truncation*: The input text is truncated, which means it gets cut off if it is longer than the model's maximum sequence length.
   - *Padding*: To preserve consistent sequence lengths, if the input text is shorter than the maximum length, it is padded with a unique [PAD] token.
3. **Creating Attention Masks**: Using binary masks, attention masks assist the model in differentiating between data tokens and padding tokens. Usually, the mask is a list of 1s and 0s, where tokens (including [CLS] and [SEP] tokens) are designated as 1 and padding tokens are set to 0.
4. **Token ID Conversion**: After tokenization, each token is converted to its corresponding token ID from the model's vocabulary. DistilBERT has a predefined vocabulary where each token is mapped to a unique integer ID.

Feature Extraction

Feature extraction (FE) or feature engineering is a crucial step in the sentiment analysis process when using machine learning because it directly affects its performance. The goal of this step is to extract meaningful data that characterizes significant aspects of the text, such as sentiment-expressing words [ali *et al.*, 2021]. This process represents features in vector form so that a machine can understand them. Eventually, each feature that these methods extract is represented as a vector, which is then sent to the classifier models [Tabassum et Patil, 2020]. We used two different features extraction techniques:

1. **Bag of Words (BoW)**: This technique classifies features according on how frequently a word appears in a document. The method ignores a word's placement within the text and only considers how often it appears. The basic premise is that documents with similar word frequency will probably have similar contexts [Tabassum et Patil, 2020].
2. **TF-IDF**: It is an extension of the BoW technique [ali *et al.*, 2021], the purpose of this TF-IDF is to penalize terms that are frequently used yet have little meaning inside the content [Tabassum et Patil, 2020]. It consists of two main components:
   Term Frequency (TF): Measures how frequently a word w appears in a document d. It is calculated as:

$$\mathrm{TF}(w,d) = \frac{\text{Number of times word } w \text{ appears in document } d}{\text{Total number of words in document } d} \tag{2.1}$$

   This provides a measure of the importance of the term within the specific document [Avinash et Sivasankar, 2019].
   Inverse Document Frequency (IDF): Measures how important a word is across the entire corpus. It is calculated as:

$$\mathrm{IDF}(w) = \log\left(\frac{\text{Total number of documents}}{\text{Number of documents containing the word } w}\right) \tag{2.2}$$

Terms that appear in many documents get a lower IDF score, while rare words get a higher IDF score [Avinash et Sivasankar, 2019].
TF-IDF Score: Combines TF and IDF to give a score for each word in each document [Avinash et Sivasankar, 2019]:

$$\text{TF-IDF}(w, d) = \text{TF}(w, d) \times \text{IDF}(w) \tag{2.3}$$

Classification

At this stage, the model classifies reviews into three categories:

- **Satisfied:** This category represents positive sentiment. We use the term "satisfied" because we are dealing with businesses that provide services, so a positive sentiment indicates that the user is satisfied with the service.
- **Not Satisfied:** This category represents negative sentiment, meaning that the user is not satisfied with the service.
- **Neutral:** This category applies when the user expresses no particular sentiment in their reviews.

The classification process involves training the model on a labeled dataset, where each review is already categorized into one of the three classes.

- **Training**
  After applying the previous steps to format the data suitably for the model, a training step is performed to enable the models to learn patterns and features in the text that correlate with each sentiment category. This training allows the model to predict the appropriate category for new, unseen reviews based on the learned patterns. Given that we worked with both machine learning and deep learning, the details of the training process differ for each approach.

  1. **Machine learning models**
     As mentioned in the previous section, we performed sentiment classification using SVM, NB, and LR. During the training phase, we employed the grid search hyperparameter optimization technique, which is used in machine learning to optimize hyperparameters by examining a predefined range of values. This technique involves creating a grid of values for the hyperparameters and evaluating the model's performance for every combination. By exploring the grid search space, we can identify the hyperparameter set that results in the best model performance [Belete et Huchaiah, 2022]. We will elaborate on the process of hyperparameter optimization in Section 3.4.1, Chapter 3.

  2. **Deep learning DistilBERT-based model**
     We utilised the DistilBERT-uncased model available from the Hugging Face website. This model shares the same features as described in Table 1.2. We cleaned and preprocessed the input text, then fed the tokens into the DistilBERT-based model for the multi-classification task.
     The architecture of the model is detailed as follows:

     – **DistilBERT Layer**: This layer uses a pre-trained DistilBERT model to preprocess the text data. Its inputs are `input_ids` and `attention_mask`.

    * `input_ids`: Tokenized input sequence
    * `attention_mask`: Allows the model to focus on the appropriate tokens

  It returns the hidden states of the DistilBERT model, which are then utilized in another part of the code.

- **Global Average Pooling Layer**: Reduces the dimensionality of hidden states that come from the DistilBERT model by computing an average value for each feature across all time steps. This is a standard practice for collapsing variable-length inputs down into a fixed-length representation [Gholamalinezhad et Khosravi, 2020].
- **Batch Normalization**: This assists in the normalization of the activations of the previous layer, a method that can help accelerate the training process and makes the finished model more robust and generalized.
- **Dense Layers**: The purpose of using these layers with ReLU activation is to help the model learn high-level features from the data. Since our model uses deep learning, adding non-linearity with an activation function like ReLU (Rectified Linear Unit) enables the model to better capture these features.
- **Dropout Layers**: This is a regularization technique to prevent overfitting by randomly setting a fraction of input units to zero at each update during training. By doing this, it helps to improve the generalization capability of the model [Raiaan *et al.*, 2024].
- **Classification Head**: The last layer predicts class probabilities. We have an output dense layer with a softmax activation function to generate the probability of the class. This layer has `l1_l2` regularization on the kernel to combat overfitting even more.

Decision

At this critical stage, the work involves selecting a suitable model from the set of classifiers that are used, which includes the state-of-the-art DistilBERT model, Support Vector Machines (SVM), Naive Bayes (NB), and Logistic Regression (LR). The goal is to predict the sentiment of the entire dataset's reviews and assign sentiment scores effectively. This decision-making procedure depends on the examination of the performance showed by each of the four models described before. The model that performs better in the sentiment prediction task will be determined by using an extensive evaluation that includes metrics like accuracy, precision, recall, and F1-score. This top-performing model will then be assigned to the sentiment prediction task.

2.4.2 Proposed recommendation system

In this step, we propose a recommendation system based on the previously performed sentiment analysis. This step involves sentiment scoring, where we use the chosen model to assign a probability to the predicted class for each review. We then applied min-max normalization to ensure that the normalized sentiments of each class fall between 0 and 1. Subsequently, an adjustment of the normalized sentiment scores is performed to differentiate them by class. Due to the high sparsity of the dataset, we applied Singular Value Decomposition (SVD) to predict the weights of a random subset of users and businesses. Finally, the recommendation system is constructed using Neural Collaborative Filtering (NCF) deep learning technique.

Sentiment scoring

One of the four sentiment classifiers used in the sentiment classification task is selected, and it assigns a sentiment score to the predicted class of each review. This score is a real number that reflects the strength of the predicted sentiment of the review. Beyond a basic categorical label, the sentiment score offers a more comprehensive understanding of the sentiment indicated in the review by quantifying the classifier's level of confidence in its prediction. The sentiment score acts as an indicator of how strongly a review expresses a particular sentiment. For example, a review with a high positive score suggests strong positive sentiment, while a low positive score indicates moderate positivity. This score can be helpful for the proposed recommender system because sentiment scores, as opposed to categorical sentiment labels (e.g., satisfied, not satisfied, or neutral), offer more detailed information about user preferences. In order to increase the likelihood that users will receive items they will like, items with higher positive sentiment scores can be given more weight in the recommendation algorithm. This helps the recommender system better comprehend the intensity of users' likes or dislikes.

Normalization

Normalization is the process of transforming values measured on a distinct range to a common range [Pandey et Jain, 2017]. It is utilized in this work to differentiate the strength of sentiment within each class, given the closeness of the probabilities. There are various normalization techniques; the one we are employing is called *min-max normalization*. This method scales a feature so that every value falls within the range [0, 1] [Pandey et Jain, 2017].
We used min-max normalization because we wanted the scores to fall within the same range, ensuring that the maximum score is 1 and the minimum score is 0 across all three classes (positive, negative and neutral). For instance, in the positive class, a score of 1 indicates a highly positive sentiment, while a score of 0 indicates otherwise.
We used the basic formula for min-max normalization which is:

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)}$$

where:

- x is the original value of a review's sentiment score.
- $\min(x)$ is the minimum value of sentiment scores in the class.
- $\max(x)$ is the maximum value of sentiment scores in the class.
- x' is the normalized value of a review's sentiment score.

Adjusting weights

This step involves adjusting the weights of the normalized sentiment scores. Given that all sentiment scores across the three classes are normalized and range within the

interval [0,1], there is no inherent indication of the class in the normalized values. Therefore, it is necessary to differentiate these values by class. To achieve this, we adjusted the weights based on the predicted class of the review by the sentiment classification model. The weight adjustment was performed using the following formulas:
Given that i is the correspond review:

For Neutral class:

$$\text{weight}_i = \text{sentiment score normalized}_i$$

Here, the corresponding weight is the same as the normalized sentiment score due to the lesser importance of neutral reviews in the recommendation task.

For satisfied class:

$$\text{weight}_i = \text{sentiment score normalized}_i \times \text{stars}_i$$

Where:
"stars" is the column in the dataset represents the rating of the review i. We use this column to increase the weight of the sentiment score, as a higher positive sentiment increases the likelihood of the item being recommended.

For not satisfied class:

$$\text{weight}_i = 1 - \text{sentiment score normalized}_i$$

Using this formula, we were able to decrease the likelihood of the item being recommended by reducing the importance of the sentiment score for the negative class.

Singular Value Decomposition (SVD)

Singular Value Decomposition (SVD) is a mathematical technique used to decompose a matrix into three smaller matrices, U, Σ, and V^T. This allows you to decompose a complex data structure into simpler parts. U contains the patterns of the original data (left singular vectors), Σ contains the singular values that indicate the importance of each pattern, and V^T contains the right singular vectors or patterns of the original data [Abdi, 2007].
In the context of recommender systems, SVD is used to analyze and predict the preferences of users by decomposing a user-item interaction matrix R (which contains the ratings given by users to movies, for example). SVD identifies latent factors that explain lots of user action to user item interaction relations by compressing the big matrix to simple most significant elemental matrices formed from the original matrix.
Singular Value Decomposition (SVD) decomposes the matrix R into three matrices:

$$R = U\Sigma V^T$$

- U (user matrix) represents the relationship between users and the latent factors.
- Σ (diagonal matrix) contains the singular values, which represent the strength of each latent factor.

- V^T (item matrix) represents the relationship between items and the latent factors.

Due to the high sparsity of the dataset we are working with, we applied Singular Value Decomposition (SVD) to a subset of the dataset, consisting of **1,000** users and **25** items (hospitals).

Neural Collaborative Filtering recommendation model

Matrix factorization (MF): The traditional collaborative filtering represents the interactions of users with items in a matrix. In this case, the whole behavior of the matrix is captured by decomposing this matrix into latent components. However, MF is quite efficient in recommending, with the history of interactions between the user and the item, but as the limitation,weights needs to be learned manually and it may not be able to capture more complex user preferences [He *et al.*, 2017].
Neural Collaborative Filtering (NCF): a generalized version of MF, is a state-of-the-art recommendation system technique that leverages neural networks to model and predict user-item preference. NCF brings the best by using deep learning algorithms and traditional collaborative filtering recommender systems to explore latent features that capture complex patterns of user-item interactions and bias terms. It gives a better way to model user behavior, than otherwise common method examples including matrix factorization [He *et al.*, 2017]. NFC is composed of two key components:

- **GMF (Generalized Matrix Factorization)** : or GMF in brief, is a component of NCF that works on user-item interactions by incorporating non-linear functions, the output of matrix factorization (dot product) only recommends items to a user and weighted matrix factorization is based on connected users as part of it. GMF can capture complex relationships between users and items, leading to a more precise recommendation thanks to its better adaptation [He *et al.*, 2017].
- **Multi-Layer Perceptron (MLP)**: is a form of neural network architecture made up of several layers of directly connected layers. NCF uses MLP to learn more complicated patterns and interactions between users and items. MLP makes it possible to capture these complex interactions in user-item data, which may not always be caught by simpler models like matrix factorization, as it utilizes multiple layers of processing [He *et al.*, 2017].

The model we built for recommendation uses NCF architecture. This architecture consists of input layers, embedding layers, GFM (which is a multiplication of users' and items' embeddings matrices generated from the embedding layers), and MLP (which is a general deep neural network composed of a set of hidden layers to capture complex patterns). Thus, the global architecture of our model consists of:

Input Layers: Separate input layers are defined for user IDs and business IDs.

Embedding Layers: Users and businesses are represented using embedding layers. These embeddings transform the high-dimensional, sparse input data into lower-dimensional, dense vectors. Each embedding is accompanied by a bias term specific to users and businesses, helping to account for systematic biases in weights.

Interaction Modeling: The core of the model captures interactions between users and businesses through the dot product of their respective embeddings. The dot product, along with the user and business biases, is concatenated into a single vector.

Dense Layers: This concatenated vector is processed through several fully connected (dense) layers with ReLU activations, which help to learn complex, non-linear relationships. Regularization techniques such as L1 and L2 penalties are applied to prevent overfitting.

Output Layer: The final layer outputs a single value, representing the predicted weight.

2.5 Conclusion

This thesis discusses our system in Chapter Two. After investigating the sentiment analysis procedures we employed, we carefully defined the necessary steps for our recommendation system approach after detailing the steps of sentiment analysis.

In the next chapter, we will delve into the implementation details and examine the results of both sentiment classification and our recommendation system. Consequently, we will explain these outcomes, interpret our findings, and propose potential directions for further research.

Chapter 3

Implementation

3.1 Introduction

In this chapter, we present the implementation process for the proposed recommendation system based on sentiment analysis. We begin by describing the experimental environment and the programming language and libraries utilized. Following this, we detail the dataset preparation steps and provide an overview of the sentiment analysis techniques employed.
The chapter proceeds with a thorough explanation of the training and testing phases, and an assessment of the performance of these techniques.
Finally, we discuss the implementation of the recommendation system itself, from dataset preparation to model training, testing, and performance evaluation.

3.1.1 Google Colab

Google Colab is short for Google Colaboratory, a cloud-based development environment from Google. It provides a web-based interface that enables users to write and run Python code from a web browser without the need for Python installation, the interface is presented in figure 3.1. The main advantage of colab is that it provides free GPU and TPU resources, which can help to accelerate processes like machine-learning model training and data analysis. Google Colab is fully integrated with Google Drive and makes it easy for users to save notebooks and datasets, access them, and work on these notebooks collaboratively. It has been a highly preferred one among the researchers, educators, and data scientists for its ease, colab options, and availability. It offers a code playground for projects that need higher computational power on its flexible platform without the need for expensive hardware investments.
However, we face a problem using it when we tried to train our distilBERT-based model which is its restricted resources. The free tier in essence provides limited use of RAM , and the access to GPUs and TPUs (If available). Therefore, users might have to wait for longer or might not get access at all at busy times. In particular, GPUs (Graphics Processing Units) are necessary to improve training efficiency especially for deep neural network based models , so it's required to train our DistilBERT-based model. In response to this, we chose another platform that fulfills our requirement for the distilBERT-based model and it was Kaggle.

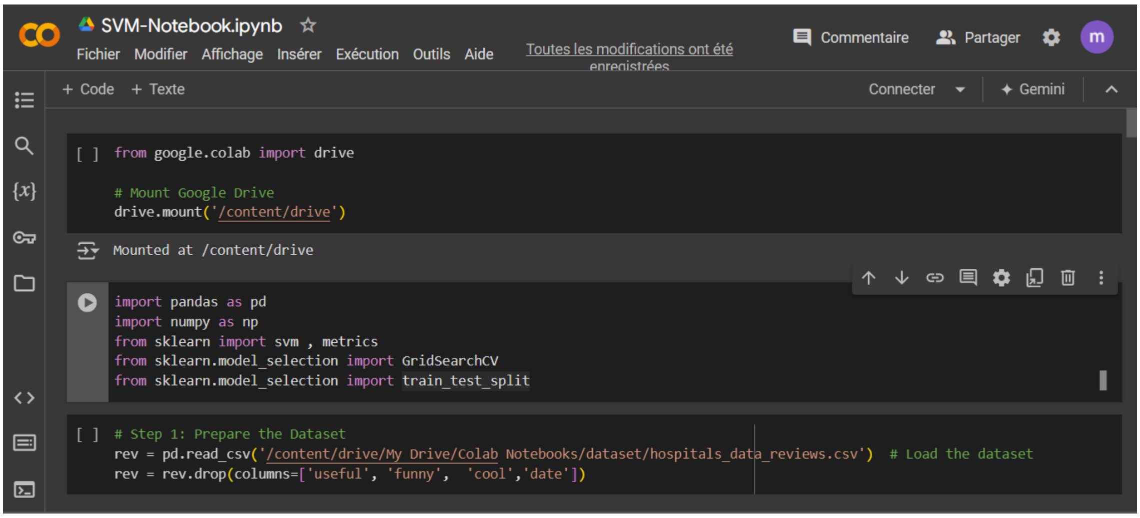


Figure 3.1: Google Colab interface

3.1.2 Kaggle

A Machine Learning and data science community, Kaggle is a place where users can come in and access a lot of datasets, participate in competitions, and work in data science projects in a collaborative environment. Kaggle provides high computational resources (up to 16Go of RAM) that include free GPUs (free 30 hours in a week) and TPUs that can assist a lot in training densely connected models without needing additional hardware. Kaggle offers notebooks too , which is a functionality through which users can share and run code using a web-based interface, which is great for working on projects together, or learning more efficiently. Figure 3.2 shows Kaggle interface.

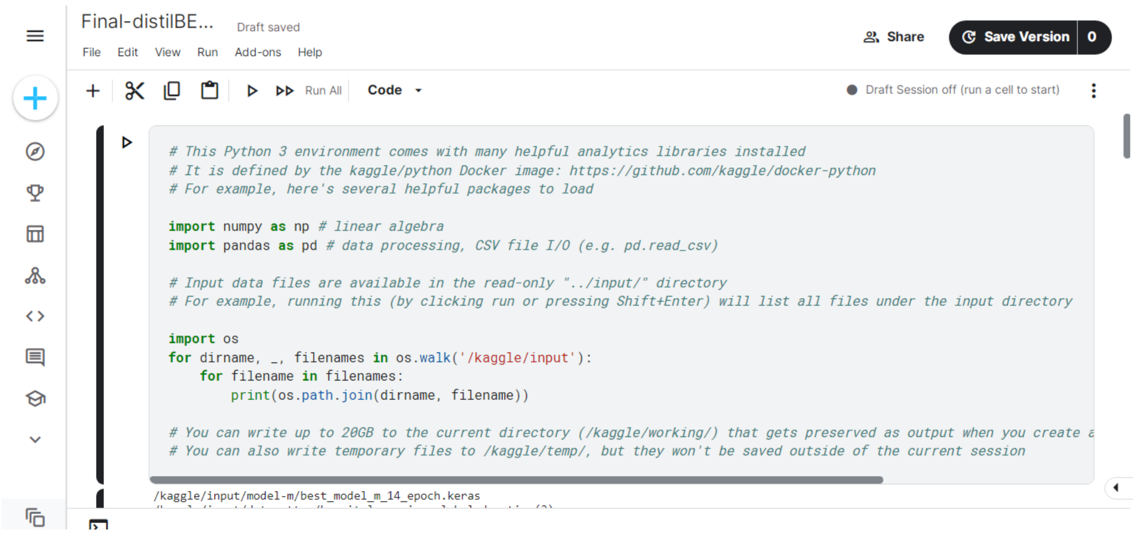


Figure 3.2: Kaggle interface

GPU configuration

Using a GPU in Kaggle requires choosing it from the accelerators list. To do that, simply click on the three dots (more settings) → Accelerator → GPU T4 x2 as illustrated in Figure 3.3.

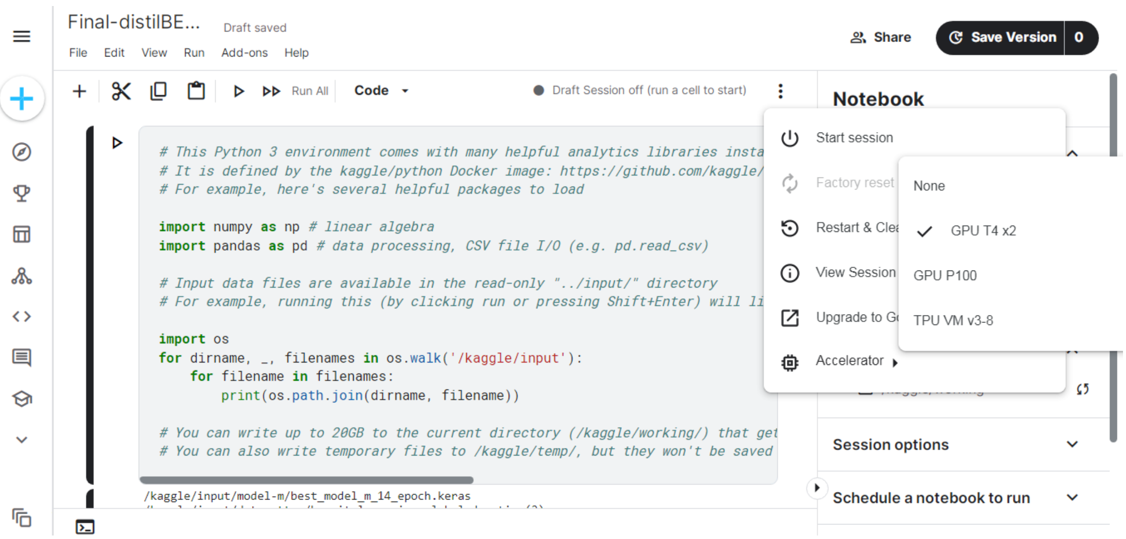


Figure 3.3: GPU Accelerator activation

3.2 Programming language and libraries used

3.2.1 Python

Python is a high-level, general-purpose programming language that is widely used in the scientific and computational communities. It is known for its simplicity, readability, and ease of use, making it an accessible language for both beginners and experienced programmers. Python is an interpreted language, which means it executes code line by line, providing rapid feedback and iterative development. Python has a large and active community, which has contributed to its extensive ecosystem of libraries and frameworks, making it a versatile tool for a wide range of applications, including data analysis, machine learning, web development, and scientific computing [Sun *et al.*, 2018].

3.2.2 Numpy

NumPy is the foundational package for scientific computing in Python. It offers a multidimensional array object combined with several derivative objects, such as matrices and masked arrays. In addition, NumPy provides many functions for quick array operations related to mathematical, logical, and shape manipulation along with functions for sorting, selection, input/output (I/O), discrete Fourier transforms, and other tasks. NumPy makes it easier to do effective numerical operations that are essential in the fields of science and engineering by supporting huge, multi-dimensional arrays and matrices. It is a fundamental component of many other Python libraries, including SciPy, Pandas, and Matplotlib, and is essential to projects involving data analysis, machine learning, and scientific computing.

3.2.3 Pandas

Pandas is a powerful open-source data analysis and manipulation library for Python. It provides data structures and data analysis tools for working with structured (tabular, multidimensional, potentially heterogeneous) and time series data. The primary data structures in Pandas are the DataFrame, which is a two-dimensional data structure with columns of potentially different types, and the Series, which is a one-dimensional array. Pandas is widely used in the data science and machine learning communities for tasks such as data cleaning, transformation, and analysis, making it an essential tool for working with complex datasets.

3.2.4 NLTK (Natural Language Toolkit)

NLTK is a Python library for working with human language data, commonly known as natural language processing (NLP). It provides a set of tools and resources for tasks such as text preprocessing, tokenization, stemming, tagging, parsing, and semantic reasoning. NLTK is designed to simplify the process of working with text data, allowing researchers and developers to quickly prototype and experiment with various NLP techniques. It includes pre-trained models and resources for a wide range of languages, making it a popular choice for NLP projects in academia and industry [Perkins, 2014].

3.2.5 Matplotlib

Matplotlib is a powerful and visually appealing plotting library for Python. Its ability to generate stunning graphs and plots makes it an invaluable tool for data visualization. Among its key features, Matplotlib can serve as a viable replacement for MATLAB, with the added advantage of being free and open-source. It supports numerous backends and output types, ensuring compatibility across various operating systems and desired output formats. Matplotlib is also recommended for its low memory consumption and efficient runtime behavior [Saabith *et al.*, 2021].

3.2.6 Tensorflow

TensorFlow is an open-source machine learning and deep learning framework developed by Google. It is primarily used for building and deploying large-scale neural network models and other machine learning algorithms. TensorFlow provides a flexible and efficient way to define, train, and deploy machine learning models, with support for both CPU and GPU acceleration. TensorFlow is widely used in a variety of applications, including image recognition, natural language processing, speech recognition, and predictive analytics.

3.2.7 Scikit-learn

Scikit-learn is a robust and versatile library for machine learning in Python, offering a comprehensive suite of efficient tools for classification, regression, clustering, and more.

3.3 Dataset

3.3.1 Description

To evaluate the proposed approach, we require a dataset. Specifically, we need a set of opinion-based textual data for both the sentiment analysis and recommendation phases. Given that our application domain is a recommendation system in the medical field, we selected the Yelp dataset from yelp.com, a popular online platform that allows users to

discover and review local businesses such as restaurants, shops, and service providers. We downloaded this dataset from the Kaggle platform. It is a subset of Yelp's business, review, and user data, stored in three JSON files, with the reviews written in English:
`yelp_academic_dataset_business.json`: This file contains 150,346 businesses, including restaurants, hotels, hospitals, and more.
`yelp_academic_dataset_review.json`: This file includes 6,990,280 reviews of all the businesses listed in the business dataset.
`yelp_academic_dataset_user.json`: This file consists of 1,987,897 unique users who have written reviews in the review dataset.

3.3.2 Data preparation

As mentioned in the previous sections, we aim to build a recommender system in the medical domain. Given that the Yelp dataset contains a large number of businesses, we needed to filter those related to the medical field. Specifically, we focused on recommending hospitals. We filtered the businesses dataset to retrieve entries that have the word 'Hospitals' in their categories, resulting in a subset containing **395** hospitals with their reviews. Figure 3.4 represents the five first samples of reviews dataset and hospitals dataset respectively.

	review_id	user_id	business_id	stars	text
0	7rNRbcMhxSnfOOaYiDgNBw	bbcO-pphyABr7uyGWCMzxQ	InbkcStjSbpQEK5JGsy8ZA	5.0	This hospital is by far the best for families ...
1	Sd-r9EeLIJ_r7rQ8rUkxKw	_7VEyrZ5gERe8sRRhqWGpw	4q4huGL_tQj1XpRB8P5F6A	5.0	Updating previous trip- I still would take my ...
2	BySlELnGkv8kQcdnsBRrrg	NZgP3GmqA0D_41f5jOY9eg	qAnKc-pentc9UUQvtUYNew	2.0	I was in so much pain from my accident, waited...
3	GQT_d1WWybpP6u1xpchW8A	pbm9st-olYQ3n5S4VcNIEQ	0WIlsDUUFuUKcDfJPaHRkA	5.0	My care at Temple has been so great that I am ...
4	J17y0gaHC_drK9O1qOBYQg	XKuyVKbfcYG-6WYXB2vUxA	0WIlsDUUFuUKcDfJPaHRkA	1.0	Please do not bring your loved one to this hos...

Figure 3.4: *First five samples of reviews dataset*

Labeling The dataset

The dataset we are working with is unlabeled, but the machine learning techniques we are using for sentiment analysis fall under the supervised learning, which requires a labeled dataset. To address this, we initially labeled **1,096** reviews manually: **500** were positive, **500** were negative, and **96** were neutral, reflecting the scarcity of neutral reviews, and saved them as `labled_1096_rev.csv`. We then explored various methods to label the remaining dataset, utilizing the ratings associated with the reviews and four well-known sentiment lexicons.

- Rating

  Given that the dataset we selected includes ratings associated with each review, we tested the utility of using these ratings to label the dataset. We labeled the dataset based on reviews ratings as follows:
  if the rating is greater than 3, the sentiment of the associated review is classified as positive;
  if the rating is equal to 3, the sentiment of the associated review is classified as neutral;
  if the rating is less than 3, the sentiment of the associated review is classified as negative.

- VADER

  VADER (Valence Aware Dictionary and sEntiment Reasoner) is a sentiment analysis tool specifically designed for interpreting sentiments expressed on social media. It excels in identifying the emotional intensity and polarity (positive, neutral, or negative) of text [Al-Shabi, 2020]. Analysis steps we performed using it are shown in table 3.1.

Analysis Step	**Description**
Word Presence Check	Each word in a sentence is examined to determine if it is present in the VADER lexicon.
Polarity Score Calculation	Uses `polarity_scores()` function to calculate polarity scores.
Sentiment Metrics	Returns metrics for positive, negative, neutral sentiments, and a compound score.
Compound Score Calculation	Sum of the normalized polarity indices, ranging from -1 to +1.
Sentiment Score Range	-1 to +1, where -1 is extreme negative sentiment, and +1 is extreme positive sentiment.
Sentiment Classification Thresholds	– Compound score $\geq$ 0.05: Positive sentiment – Compound score between -0.05 and 0.05: Neutral sentiment – Compound score $\leq$ -0.05: Negative sentiment

Table 3.1: *Summary of the analysis steps for sentiment analysis using VADER*

- TextBlob

  TextBlob is a Python library used for natural language processing tasks, it's simple and easy to use compared with other Python libraries. It includes a sentiment property that returns a tuple formatted as Sentiment (polarity, subjectivity). The polarity score ranges from -1.0 to 1.0, where -1 being highly negative statement and 1 being highly positive statement. Subjectivity, a number between 0.0 and 1.0, reflects the degree of objectivity (0.0 indicates highly objective and 1.0 signifies highly subjective) about a person's opinion or emotion on a specific topic [Pandey *et al.*, 2019]. With the same principle as VADER, TextBlob classifies the sentiment of a given text using a threshold. We used the same threshold as VADER's shown in Table 3.1 for the classification of sentiments.

- SentiWordNet

  SentiWordNet is a sentiment analysis tool built on WordNet, a dataset that groups words into synonym sets connected by semantic links. In SentiWordNet, each set is assigned sentiment scores indicating how positive, negative, or neutral the words are perceived to be, based on evaluations by judges. These scores are numerical values ranging from 0 to 1, where higher values suggest more positive sentiments and lower values indicate stronger negative sentiments [Al-Shabi, 2020]. Table 3.2

summarizes the classification and scoring principles used in sentiment classification with SentiWordNet.

Sentiment Classification	Description
Positive	Positive sentiment is classified when the positive score exceeds the negative score.
Negative	Negative sentiment is classified when the negative score exceeds the positive score.
Neutral	Neutral sentiment is classified when neither the positive nor the negative score dominates.
Positive Score	Represents the degree of positive sentiment assigned to the text.
Negative Score	Represents the degree of negative sentiment assigned to the text.
Objectivity Degree	Calculated as $1-(\text{positive score}+\text{negative score})$, measures the level of objectivity in the statement.

Table 3.2: *Classification and description of sentiment analysis metrics using SentiWordNet.*

- SpaCy

  SpaCy is a Python library, for Natural Language Processing (NLP) that's source and geared towards being fast, efficient and ready for practical use. It provides an array of features for handling and examining text data including dividing text into tokens identifying parts of speech recognizing named entities parsing dependencies and more [Pandey *et al.*, 2019].

Auto-labeling results

The results of applying sentiment labeling using ratings are illustrated in Figure 3.5.

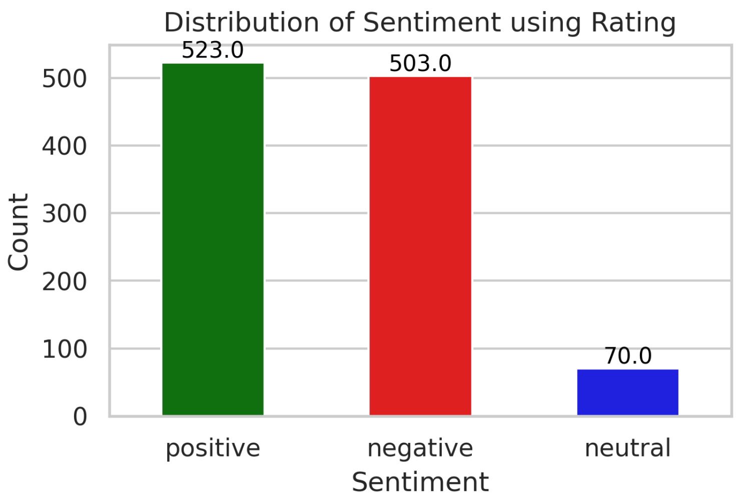


Figure 3.5: *Results of labeling sentiments using ratings*

Figures 3.6, 3.7, 3.8 and 3.9 illustrate the results of classifying the sample dataset using VADER, TextBlob, SentiWordNet and SpaCy respectively.

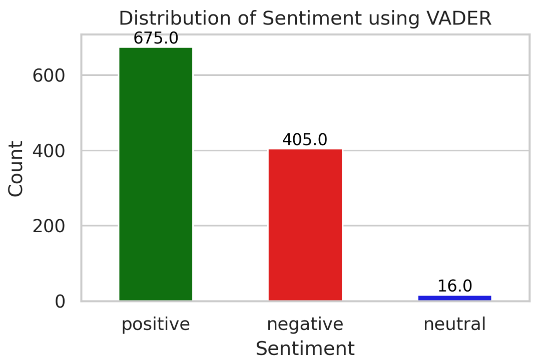


Figure 3.6: *Classification results using VADER*

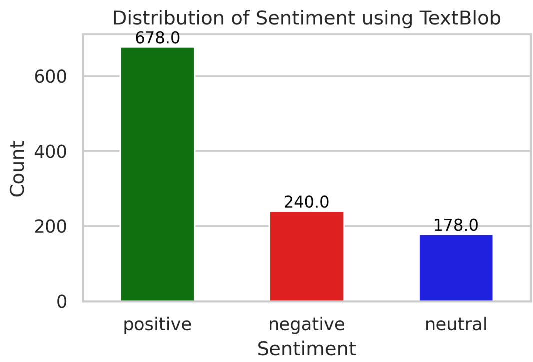


Figure 3.7: *Classification results using TextBlob*

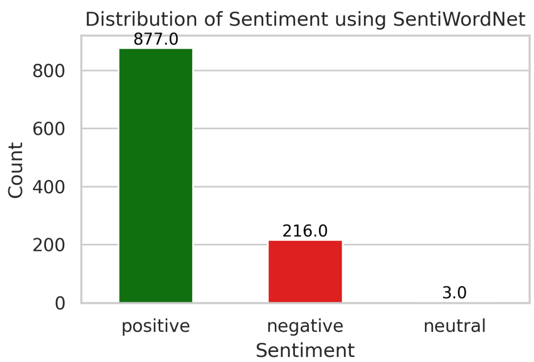


Figure 3.8: *Classification results using SentiWordNet*

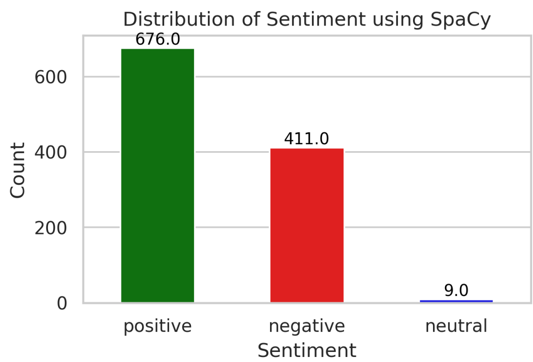


Figure 3.9: *Classification results using SpaCy*

After applying the aforementioned tools, the next step is to determine which one performed the best. To achieve this, various evaluation metrics can be used to assess their performance. These metrics include:

- **Accuracy**: it measures how effectively a classifier predicts [Noyunsan *et al.*, 2018]. In the context of sentiment classification, it's the ratio of correctly predicted sentiment labels (positive, negative, or neutral) to the total number of instances. It assesses how well the sentiment classifier correctly identifies the sentiment of text data. It's calculated as follows:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{3.1}$$

- **Precision**: Precision is the ratio of correctly predicted positive observations to the total predicted positives [Akinsola *et al.*, 2019].
  In sentiment classification, precision indicates the proportion of correctly identified sentiment instances among all instances classified as that sentiment.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{3.2}$$

- **Recall**: Also known as sensitivity, it is the ratio of correctly predicted positive observations to all actual positives in the dataset [Winkler *et al.*, 2019].
  In sentiment classification, recall signifies the proportion of correctly identified sentiment instances among all actual sentiment instances.

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{3.3}$$

- **F1-score**: It is the harmonic mean of the two previous metrics: precision and recall [Akinsola *et al.*, 2019].
  The F1-Score in sentiment classification provides insight into the classifier's overall performance in identifying sentiment.

$$\text{F1-score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{3.4}$$

Where:
TP (True Positive): when the model correctly predicts the Positive class.
TN (True Negative): when the model correctly predicts the Negative class.
FP (False Positive): when the model incorrectly predicts the Positive class when the actual class is Negative or Neutral.
FN (False Negative): when the model incorrectly predicts the Negative or Neutral class when the actual class is Positive.

As shown in previous figures, using ratings for labeling outperforms the four sentiment lexicons, achieving an accuracy of 97.62% and an F1-score of 93.92%. After that, SpaCy, VADER, and TextBlob achieved accuracies of 77.00%, 76.09%, and 65.23% respectively, while SentiWordNet marked the lowest measures with an accuracy of 59.48%. Therefore, we decided to label our dataset using ratings.

3.4 Sentiment analysis

3.4.1 Training and test

Training

To train our models, the dataset must be divided into a training set and a testing set. For the machine learning classifier, the division was 80% for training and the remaining 20% for testing. However, the division differs slightly when using DistilBERT. In this case, 80% of the data was used for training the model, and the remaining 20% was further divided into 10% for validation and 10% for testing. The purpose of adding a validation set is to monitor the model's performance during training and to fine-tune the model's parameters, ensuring it generalizes well to unseen data.

The implementation of machine learning models (SVM, Naive Bayes, and Logistic Regression) was done using scikit-learn library. During the training of these machine learning classifiers, parameter optimization was performed using the grid search technique. Each classifier has its own set of hyperparameters.

For the SVM classifier, the key hyperparameters are C, gamma, and kernel, where:

- C: When we train a SVM, we regularize C coefficient by adjusting them to control the trade-off between the maximization of the margin and minimization of the classification error.
- γ: Gamma is the parameter of the RBF kernel in SVM which controls the shape of the blob and the more significant gamma performs, the more complex decision boundaries become.
- **Kernel**: The kernel function calculates the similarity of the two data points in the feature space which makes SVM able to work on a higher-dimensional space without actually performing that transformation.

The hyperparameters used for the Logistic Regression classifier are:

- **Solver**: Solver is an algorithm to be used to optimize the coefficients.
- **C**: C is the penalty parameter of the error term, with smaller values indicating stronger regularization.
- **max_iter**: Maximum number of iterations the solver is allowed to take to converge.
- **multi_class**: This is the parameter which helps the application to decide how to treat the multiple classes.

Hyperparameters for Naive Bayes classifier are listed as follows:

- **tfidf__ngram_range / vect__ngram_range**: The n-grams to extract out of the text. Therefore, (1, 1) are unigrams, (1, 2) are unigrams and bigrams, and (2, 2) are bigrams.
- **tfidf__use_idf**: by using T/F it allows us to decide whether to use inverse document frequency (IDF) or not to down scale weight of less informative terms. True=use IDF weighting, False=do not use IDF weighting.
- **tfidf__norm**: Decides the normalization to be done to the tf-idf vectors. Two options are for the value of penalty are 'l1'and'l2' which are used for L1 and L2 normalisation.
- **vect__binary**: For BoW, indicates whether the term frequencies should be binary, with True meaning the presence or absence of a term is considered (binary) and False meaning the actual term frequency is used.
- **clf__alpha**: This parameter helps to handle zero counts in the data, with values such as 1, 0.1, and 0.01.

.

Given that DistilBERT is a deep learning model, its parameters differ. They are:

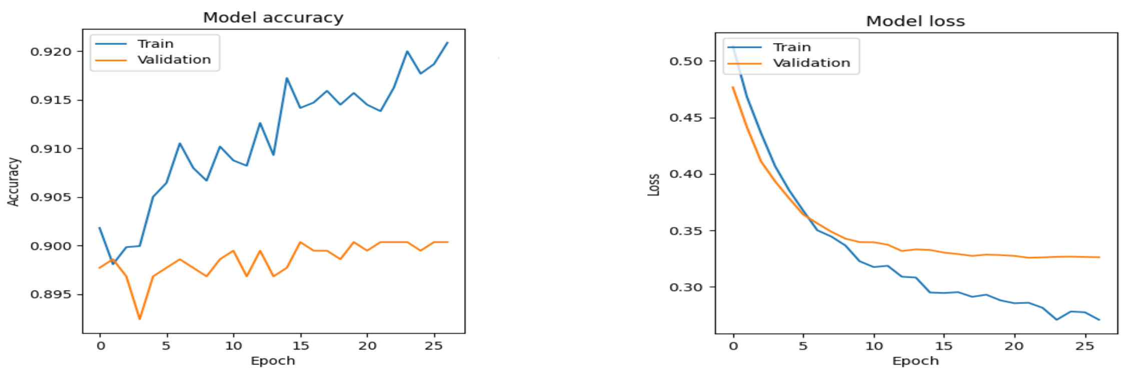
- **Learning rate**: how much to update the weight in response to finalize the model of the error estimate, with higher learning rates allowing the model to learn faster, At the risk of learning too quickly due to a high learning rate, and with lower learning rates potentially offering a constrained window for the model to effectively adjust its weights. [Raiaan *et al.*, 2024].
- **Epochs**: An epoch is one complete pass through the entire training dataset, with more epochs allowing the model to learn better but risking overfitting if too many [Raiaan *et al.*, 2024].
- **Batch Size**: The batch size is a number of training samples processed before the model is updated, smaller batch sizes provide noisier but more frequent updates, larger batch sizes provide more stable but less frequent updates and requires more memory [Raiaan *et al.*, 2024].

Table 3.3 shows the values fixed for each parameter.

Training Parameter	Value
Batch Size	32
Epochs	50
Learning Rate	Initial learning rate = 0.0001

Table 3.3: *Training Parameters for DistilBERT model*

By applying early stopping technique, our model was trained and stopped at **27th** epoch. Training and validation accuracies are shown in Figure 3.10a, while Figure 3.10b presents the training and validation loss.



(a) *Training and validation accuracy of DistilBERT model*

(b) *Training and validation loss of DistilBERT model*

Figure 3.10: *Training and validation plots of DistilBERT model*

Testing

For the testing stage, the primary purpose is to assess the functioning of sentiment analysis models when applied on the unseen data. The intention is to guarantee that the model used from the beginning is able to identify new data not previously known.

The test set consists of **2,269** labeled samples for machine learning classifiers and **1,135** samples for distilBERT-based model, these samples have not been seen by the system during training or for validation. It underwent the same kind of processing that input texts going into model training such as tokenization and turning into vectors for machine learning classifiers.

3.4.2 Results

To identify the best model for decision-making and further use in the recommendation task, their performance must be evaluated using the evaluation metrics (accuracy, precision, recall, and F1-score) mentioned in Section 3.3.2.
To determine which feature extraction method performs better in our experiment, we used two different approaches and computed the evaluation metrics for each approach. The results show that using TF-IDF yielded better outcomes compared to using BoW as a feature extraction technique. Figure 3.11 presents the evaluation metrics for the machine learning classifiers used in sentiment classification using TF-IDF feature extraction technique.

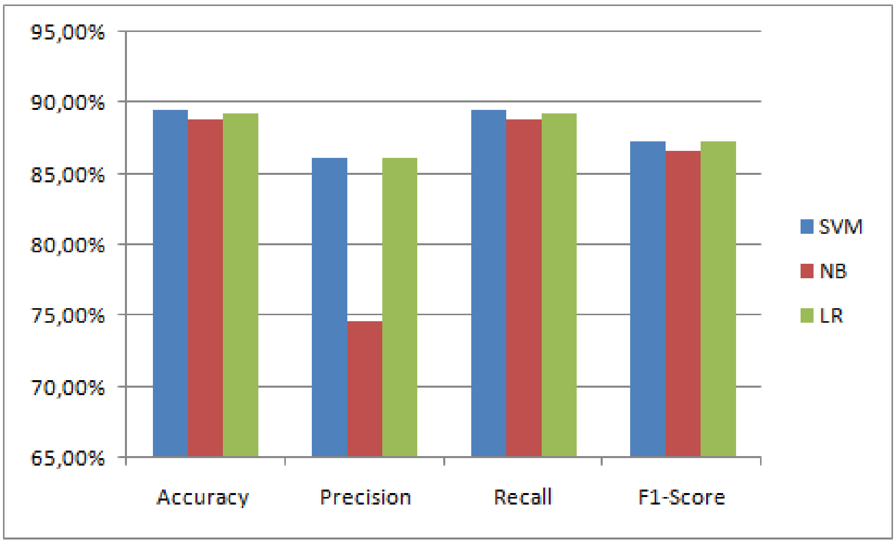


Figure 3.11: *Performance Metrics for the machine learning classifiers used in sentiment classification*

Given the multi-class nature of the classification, it is necessary to incorporate averaging to compute precision, recall, and F1-score. We opted for the '*weighted*' average as it accounts for dataset imbalance.

Figure 3.12 presents the evaluation metrics of sentiment classification using the DistilBERT-based model.

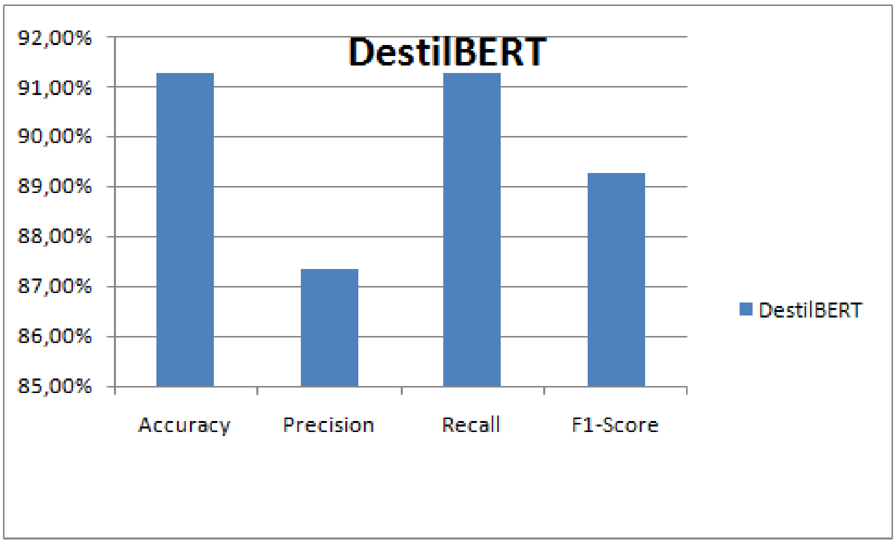


Figure 3.12: *Performance Metrics of DistilBERT-based model used in sentiment classification*

3.5 Recommendation system

3.5.1 Dataset preparation

Before feeding the dataset to the recommendation model, the dataset needs to be prepared. Firstly, we merged the original dataset with the result of the SVD prediction: Initially, we load a dataset and select a subset of 1000 random users. Using Surprise, a Python library for recommendation systems, we construct a user-item matrix from the data and train an SVD model on this matrix. Then, we randomly choose 25 businesses and predict ratings for each user-business pair using the trained SVD model.

3.5.2 Training and test

Training

The neural collaborative filtering model is trained on the merged dataset of the original dataset and the one predicted through the SVD model we used. We split the merged dataset into training (70%), validation (10%), and test sets (20%).

The Values of training parameters that are defined in section 3.4 are represented in Table 3.4. By incorporating Early Stopping and Checkpoint callbacks, we ensured that

Training Parameter	Value
Batch Size	32
Epochs	50
Learning Rate	0.0001

Table 3.4: *Training Parameters for NCF model*

the training process halted while preserving the best-performing model. As illustrated in Figure 3.13, the best model was reached in the eightieth epoch with the minimum validation loss.

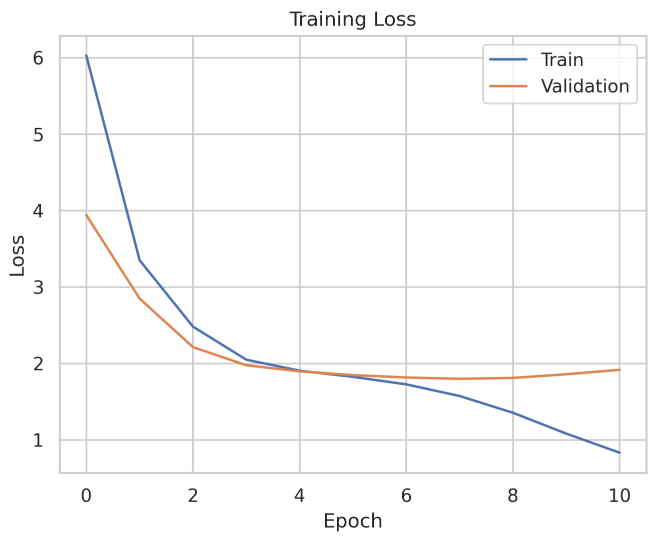


Figure 3.13: *Training and validation loss of NCF model*

Testing

In this stage, we use the model we trained earlier to make predictions on the test dataset. This allows us to measure how well the model performs in practical applications. So we tested it on the 20% of test data of the merged dataset.

3.5.3 Results

In this step, we show the final results of the recommendation system model. As for the way of evaluating the results, several of the typical evaluation metrics used in recommender systems are used. In particular, we computed the RMSE and the MAE, and in addition, we have drawn the AUC graph, which provided a better picture of the overall predictions. The value of RMSE found is 0.67 and the value of MAE found is 0.35.

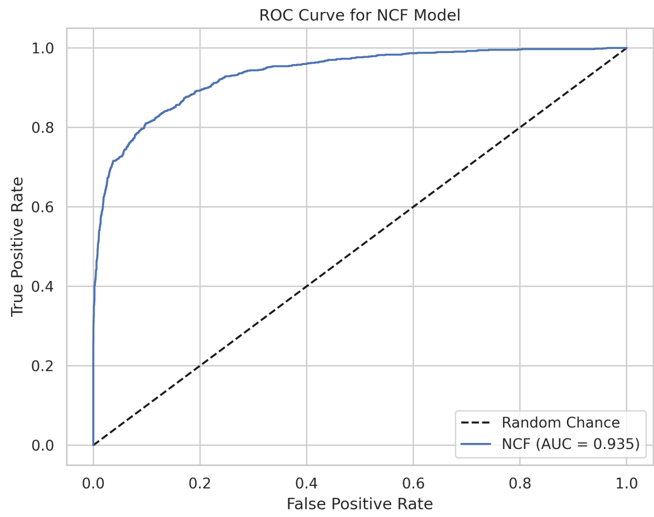


Figure 3.14: *Test AUC curve of the proposed recommender system*

3.6 Discussion

3.6.1 Sentiment Analysis Results Discussion

Based on the results obtained in Section 3.4.2, DistilBERT-based model showed higher overall classification accuracy (91.27%) than the machine learning models. This makes it our best classifier subsequently for sentiment scoring in recommendation system. Moreover, the classification accuracies of machine learning classifier based algorithm SVM, NB, and LR were 89.46%, 89.33% and 88.89% respectively.
We used two methods of feature extraction to determine the method that improves the classification the most. In conclusion, we found that the TF-IDF is suitable to the SVM and NB and the BoW is suitable to the LR in the classification respectively.

3.6.2 Recommendation System Results Discussion

The recommender system obviously performs well in metrics analysis across presumably other evaluation metrics. First, a 0.65 RMSE implies that the recommender system has a reasonable level of accuracy in its predictions. This value indicates that, overall (on average), the predictions are not far off from the actual weights - which may also be considered to be a good performance of the system-.
Also, there is pretty low MAE of 0.35, which implies that the predictions by the recommender system are very much close to the actual weights on average. This indicates that the recommendation algorithm proposed is highly accurate and reliable.
AUC of 93.5 Meaning that the score will be able to differentiate between a positive and negative occurrence with an ability of 93%.

3.7 Conclusion

In this chapter, we detail the implementation process for the proposed recommendation system based on sentiment analysis. We begin by describing the environment in which our experiment was conducted, including the programming language and libraries used. Next, we present the preparation of the dataset, followed by a comprehensive overview of the training and testing of sentiment analysis techniques, and the assessment of their performance.

The implementation of the recommendation system is then described, starting with dataset preparation, followed by the training and testing of the model, and the evaluation of its performance. We conclude the chapter with a discussion about different results obtained.

General Conclusion

The vast amount of exploitable data on the internet is constantly increasing. Utilizing this data is a meaningful practice for understanding what people like or dislike and gauging their sentiments toward products or services. Integrating sentiment analysis into recommender systems significantly enhances their performance, offering more accurate and relevant suggestions to users.

The results of this work demonstrated that by leveraging natural language processing techniques to analyze user sentiments, the system could better understand user preferences and improve recommendation accuracy.

We performed sentiment analysis on a subset of the Yelp business dataset, as we wanted to examine this approach in the healthcare domain using three machine learning techniques (SVM, Naive Bayes, and Logistic Regression) and a deep learning pre-trained model, DistilBERT, which gave the best performance.

The use of Singular Value Decomposition (SVD) further addressed the issue of data sparsity, leading to better performance metrics after utilizing the principles of the Neural Collaborative Filtering technique for the recommendation process. The experimental results validate the proposed approach's effectiveness, highlighting its potential for practical application in personalized recommendation services.
Future work could explore:

- Additional sentiment analysis techniques and their impact on recommendation quality.
- Personalized Sentiment Analysis: Enhance sentiment analysis and consider individual user sentiment expressions and preferences, thereby tailoring recommendations more accurately.
- Applying the proposed approach to other language data, such as Arabic.
- Handling the Cold Start Problem by Developing strategies to address this problem for new users and items.
- Real-time Recommendation: Optimize the system's computational efficiency to reduce latency for instant user feedback.
- Cross-Domain Recommendations: Extend the system to provide cross-domain recommendations, for instance, in e-commerce.

It is worth mentioning that one of the good results of this thesis have been the subject of participation in an international conference (Scopus Indexed):

Mehenaoui.Z., Merabti.C., Tadjer.H.,Lafifi.Y., (2024). A Comparative Study on Sentiment Lexicons For Automatic Labelling, The 13th International Conference on Research in Computing at Feminine (RIF24), Constantine, Algeria.

Bibliography

[Abdi, 2007] ABDI, H. (2007). Singular value decomposition (svd) and generalized singular value decomposition. *Encyclopedia of measurement and statistics*, 907(912):44.

[Adomavicius et Tuzhilin, 2005] ADOMAVICIUS, G. et TUZHILIN, A. (2005). Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering*, 17(6):734–749.

[Akinsola *et al.*, 2019] AKINSOLA, J., AWODELE, O., KUYORO, S. et KASALI, F. (2019). Performance evaluation of supervised machine learning algorithms using multi-criteria decision making techniques. *In Proceedings of the International Conference on Information Technology in Education and Development (ITED)*, pages 17–34.

[Akpatsa *et al.*, 2022] AKPATSA, S. K., LEI, H., LI, X., OBENG, V.-H. K. S., MARTEY, E. M., ADDO, P. C. et FIAWOO, D. D. (2022). Online news sentiment classification using distilbert. *Journal of Quantum Computing*, 4(1).

[Al Maruf *et al.*, 2022] AL MARUF, A., ZIYAD, Z. M., HAQUE, M. M. et KHANAM, F. (2022). Emotion detection from text and sentiment analysis of ukraine russia war using machine learning technique. *International Journal of Advanced Computer Science and Applications*, 13(12).

[Al-Mashhadany *et al.*, 2022] AL-MASHHADANY, A. K., SADIQ, A. T., ALI, S. M. et AHMED, A. A. (2022). Healthcare assessment for beauty centers using hybrid sentiment analysis. *Indonesian Journal of Electrical Engineering and Computer Science*, 28(2): 890–897.

[Al-Shabi, 2020] AL-SHABI, M. (2020). Evaluating the performance of the most important lexicons used to sentiment analysis and opinions mining. *IJCSNS*, 20(1):1.

[ali *et al.*, 2021] ALI, M., KASRI, M. et BENI-HSSANE, A. (2021). A comprehensive survey on sentiment analysis: Approaches, challenges and trends. *Knowledge-Based Systems*, 226:107134.

[Aliman *et al.*, 2022] ALIMAN, G., NIVERA, T. F. S., OLAZO, J. C. A., RAMOS, D. J. P., SANCHEZ, C. D. B., AMADO, T. M., ARAGO, N. M., JORDA JR, R. L., VIRREY, G. C. et VALENZUELA, I. C. (2022). Sentiment analysis using logistic regression. *Journal of Computational Innovations and Engineering Applications*, 7(1):35–40.

[Asani *et al.*, 2021] ASANI, E., VAHDAT-NEJAD, H. et SADRI, J. (2021). Restaurant recommender system based on sentiment analysis. *Machine Learning with Applications*, 6:100114.

[Avinash et Sivasankar, 2019] Avinash, M. et Sivasankar, E. (2019). A study of feature extraction techniques for sentiment analysis. *In Emerging Technologies in Data Mining and Information Security: Proceedings of IEMIS 2018, Volume 3*, pages 475–486. Springer.

[Belete et Huchaiah, 2022] Belete, D. M. et Huchaiah, M. D. (2022). Grid search in hyperparameter optimization of machine learning models for prediction of hiv/aids test results. *International Journal of Computers and Applications*, 44(9):875–886.

[Bernal et Pedraz, 2024] Bernal, Á. I. M. et Pedraz, C. G. (2024). Sentiment analysis of the spanish financial stability report. *International Review of Economics & Finance*, 89:913–939.

[Bewick *et al.*, 2005] Bewick, V., Cheek, L. et Ball, J. (2005). Statistics review 14: Logistic regression. *Critical care*, 9:1–7.

[Bobadilla *et al.*, 2013] Bobadilla, J., Ortega, F., Hernando, A. et Gutierrez, A. (2013). Recommender systems survey. *Knowledge-Based Systems*, 46:109–132.

[Breese *et al.*, 1998] Breese, J. S., Heckerman, D. et Kadie, C. (1998). Empirical analysis of predictive algorithms for collaborative filtering. *In Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence*, pages 43–53, Madison, Wisconsin, USA.

[Burke, 2002] Burke, R. (2002). Hybrid recommender systems: Survey and experiment. *Journal of User Modeling and User Adapted Interaction*, 4(12):331–370.

[Calderon Vilca *et al.*, 2023] Calderon Vilca, H. D., Satornicio Medina, A. L. et Sucari Leon, R. (2023). Music recommender system based on sentiment analysis enhanced with natural language processing technics. *Computación y Sistemas*, 27(1).

[Cam *et al.*, 2024] Cam, H., Cam, A. V., Demirel, U. et Ahmed, S. (2024). Sentiment analysis of financial twitter posts on twitter with the machine learning classifiers. *Heliyon*, 10(1).

[Cervantes *et al.*, 2020] Cervantes, J., Garcia-Lamont, F., Rodríguez-Mazahua, L. et Lopez, A. (2020). A comprehensive survey on support vector machine classification: Applications, challenges and trends. *Neurocomputing*, 408:189–215.

[Choudhary *et al.*, 2023] Choudhary, C., Singh, I. et Kumar, M. (2023). Sarwas: Deep ensemble learning techniques for sentiment based recommendation system. *Expert Systems with Applications*, 216:119420.

[Dang *et al.*, 2021] Dang, C. N., Moreno-García, M. N. et Prieta, F. D. l. (2021). An approach to integrating sentiment analysis into recommender systems. *Sensors*, 21(16):5666.

[Dang *et al.*, 2020] Dang, N. C., Moreno-García, M. N. et De la Prieta, F. (2020). Sentiment analysis based on deep learning: A comparative study. *Electronics*, 9(3):483.

[Demircan *et al.*, 2021] Demircan, M., Seller, A., Abut, F. et Akay, M. F. (2021). Developing turkish sentiment analysis models using machine learning and e-commerce data. *International Journal of Cognitive Computing in Engineering*, 2:202–207.

[Devlin *et al.*, 2018] Devlin, J., Chang, M.-W., Lee, K. et Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

[Dey *et al.*, 2016] Dey, L., Chakraborty, S., Biswas, A., Bose, B. et Tiwari, S. (2016). Sentiment analysis of review datasets using naive bayes and k-nn classifier. *arXiv preprint arXiv:1610.09982*.

[Elfajr et Sarno, 2018] Elfajr, N. M. et Sarno, R. (2018). Sentiment analysis using weighted emoticons and sentiwordnet for indonesian language. *In 2018 International Seminar on Application for Technology of Information and Communication*, pages 234–238. IEEE.

[Feng, 2023] Feng, S. (2023). Job satisfaction, management sentiment, and financial performance: text analysis with job reviews from indeed. com. *International Journal of Information Management Data Insights*, 3(1):100155.

[F.O. Isinkaye et c, 2015] F.O. Isinkaye, Y. F. et c, B. O. (2015). Recommendation systems: Principles, methods and evaluation. *Egyptian Informatics Journal*, 16:261–273.

[Gholamalinezhad et Khosravi, 2020] Gholamalinezhad, H. et Khosravi, H. (2020). Pooling methods in deep neural networks, a review. *arXiv preprint arXiv:2009.07485*.

[Gräßer *et al.*, 2018] Grässer, F., Kallumadi, S., Malberg, H. et Zaunseder, S. (2018). Aspect-based sentiment analysis of drug reviews applying cross-domain and cross-data learning. *In Proceedings of the 2018 international conference on digital health*, pages 121–125.

[Hacene et Tekia, 2021] Hacene, E. et Tekia, S. (2021). Développement d'un système de recommandation dans le domaine médical. Master's thesis, Université Abou Bakr Belkaïd de Tlemcen, Tlemcen, Algerie.

[He *et al.*, 2017] He, X., Liao, L., Zhang, H., Nie, L., Hu, X. et Chua, T.-S. (2017). Neural collaborative filtering. *In Proceedings of the 26th international conference on world wide web*, pages 173–182.

[Herlocker *et al.*, 2004] Herlocker, J. L., Konstan, J. A., Terveen, L. G. et Riedl, J. T. (2004). Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems (TOIS)*, 22(1):5–53.

[Jim *et al.*, 2024] Jim, J. R., Talukder, M. A. R., Malakar, P., Kabir, M. M., Nur, K. et Mridha, M. (2024). Recent advancements and challenges of nlp-based sentiment analysis: A state-of-the-art review. *Natural Language Processing Journal*, page 100059.

[Karthik et Ganapathy, 2021] Karthik, R. et Ganapathy, S. (2021). A fuzzy recommendation system for predicting the customers interests using sentiment analysis and ontology in e-commerce. *Applied Soft Computing*, 108:107396.

[Khurana *et al.*, 2023] Khurana, D., Koli, A., Khatter, K. et Singh, S. (2023). Natural language processing: State of the art, current trends and challenges. *Multimedia tools and applications*, 82(3):3713–3744.

[Kim *et al.*, 2020] Kim, J., Lee, J., Park, E. et Han, J. (2020). A deep learning model for detecting mental illness from user content on social media. *Scientific reports*, 10(1): 11846.

[Kumar *et al.*, 2020] Kumar, S., De, K. et Roy, P. P. (2020). Movie recommendation system using sentiment analysis from microblogging data. *IEEE Transactions on Computational Social Systems*, 7(4):915–923.

[Leelawat *et al.*, 2022] Leelawat, N., Jariyapongpaiboon, S., Promjun, A., Boonyarak, S., Saengtabtim, K., Laosunthara, A., Yudha, A. K. et Tang, J. (2022). Twitter data sentiment analysis of tourism in thailand during the covid-19 pandemic using machine learning. *Heliyon*, 8(10).

[Liu, 2022] Liu, B. (2022). *Sentiment analysis and opinion mining.* Springer Nature.

[Liu *et al.*, 2010] Liu, B. *et al.* (2010). Sentiment analysis and subjectivity. *Handbook of natural language processing*, 2(2010):627–666.

[Luţan et Bădică, 2023] Luţan, E.-R. et Bădică, C. (2023). Emotion-based literature books recommender systems. *In 2023 18th Conference on Computer Science and Intelligence Systems (FedCSIS)*, pages 275–280. IEEE.

[Majda, 2016] Majda, M. (2016). *Une Technique Hybride pour les Systèmes de Recommandation.* Thèse de doctorat, Université Badji Mokhtar-Annaba.

[Majeed *et al.*, 2020] Majeed, A., Mujtaba, H. et Beg, M. O. (2020). Emotion detection in roman urdu text using machine learning. *In Proceedings of the 35th IEEE/ACM International Conference on Automated Software Engineering*, pages 125–130.

[Medhat *et al.*, 2014] Medhat, W., Hassan, A. et Korashy, H. (2014). Sentiment analysis algorithms and applications: A survey. *Ain Shams engineering journal*, 5(4): 1093–1113.

[Mehenaoui, 2018] Mehenaoui, Z. (2018). *Recommandation de collaborateurs pertinents dans un environnement d'apprentissage collaboratif.* Thèse de doctorat, Université Badji Mokhtar-Annaba.

[Mehenaoui *et al.*, 2024] Mehenaoui, Z., Merabti, C., Tadjer, H. et Lafifi, Y. (2024). A recommendation approach based on sentiment analysis: Application to the healthcare domain. *Journal of King Saud University-Computer and Information Sciences.* Submitted article for publication.

[Meyer et Wien, 2001] Meyer, D. et Wien, F. (2001). Support vector machines. *R News*, 1(3):23–26.

[Mouzhi Ge et D'Auria, 2023] Mouzhi Ge, Giovanni Pilato, F. P. et D'Auria, D. (2023). Recommender system for social media: Research challenges and future applications. *In 22023 Fifth International Conference on Transdisciplinary AI (TransAI).*

[Nandwani et Verma, 2021] Nandwani, P. et Verma, R. (2021). A review on sentiment analysis and emotion detection from text. *Social network analysis and mining*, 11(1):81.

[Noyunsan *et al.*, 2018] Noyunsan, C., Katanyukul, T. et Saikaew, K. (2018). Performance evaluation of supervised learning algorithms with various training data sizes and missing attributes. *Engineering and Applied Science Research*, 45(3):221–229.

[Ojeda-Hernández *et al.*, 2023] Ojeda-Hernández, M., López-Rodríguez, D. et Mora, Á. (2023). Lexicon-based sentiment analysis in texts using formal concept analysis. *International Journal of Approximate Reasoning*, 155:104–112.

[Ortis *et al.*, 2020] Ortis, A., Farinella, G. M. et Battiato, S. (2020). Survey on visual sentiment analysis. *IET Image Processing*, 14(8):1440–1456.

[Pandey et Jain, 2017] Pandey, A. et Jain, A. (2017). Comparative analysis of knn algorithm using various normalization techniques. *International Journal of Computer Network and Information Security*, 11(11):36.

[Pandey *et al.*, 2019] Pandey, M., Williams, R., Jindal, N. et Batra, A. (2019). Sentiment analysis using lexicon based approach. *IITM Journal of Management and IT*, 10(1):68–76.

[Pavitha *et al.*, 2022] Pavitha, N., Pungliya, V., Raut, A., Bhonsle, R., Purohit, A., Patel, A. et Shashidhar, R. (2022). Movie recommendation and sentiment analysis using machine learning. *Global Transitions Proceedings*, 3(1):279–284.

[Pazzani et Billsus, 2007] Pazzani, M. J. et Billsus, D. (2007). Content-based recommendation systems. *In* Brusilovsky, P., Kobsa, A. et Nejdl, W., éditeurs : *The Adaptive Web*, volume 4321 de *LNCS*, pages 325–341, Berlin Heidelberg. Springer-Verlag.

[Perkins, 2014] Perkins, J. (2014). *Python 3 text processing with NLTK 3 cookbook*. Packt Publishing Ltd.

[Poria *et al.*, 2017] Poria, S., Cambria, E., Bajpai, R. et Hussain, A. (2017). A review of affective computing: From unimodal analysis to multimodal fusion. *Information fusion*, 37:98–125.

[Poria *et al.*, 2016] Poria, S., Cambria, E., Howard, N., Huang, G.-B. et Hussain, A. (2016). Fusing audio, visual and textual clues for sentiment analysis from multimodal content. *Neurocomputing*, 174:50–59.

[Raiaan *et al.*, 2024] Raiaan, M. A. K., Sakib, S., Fahad, N. M., Al Mamun, A., Rahman, M. A., Shatabda, S. et Mukta, M. S. H. (2024). A systematic review of hyperparameter optimization techniques in convolutional neural networks. *Decision Analytics Journal*, page 100470.

[Ramadhan et Setiawan, 2022] Ramadhan, G. et Setiawan, E. B. (2022). Collaborative filtering recommender system based on memory based on twitter using decision tree learning classification (case study: Movie on netflix). *In 2022 International Conference on Advanced Creative Networks and Intelligent Systems (ICACNIS)*.

[Ramzy et Ibrahim, 2024] Ramzy, M. et Ibrahim, B. (2024). User satisfaction with arabic covid-19 apps: Sentiment analysis of users' reviews using machine learning techniques. *Information Processing & Management*, 61(3):103644.

[Rana et Singh, 2016] Rana, S. et Singh, A. (2016). Comparative analysis of sentiment orientation using svm and naive bayes techniques. *In 2016 2nd International Conference on Next Generation Computing Technologies (NGCT)*, pages 106–111. IEEE.

[Rao, 2008] Rao, K. N. (2008). Application domain and functional classification of recommender systems–a survey. *DESIDOC Journal of Library & Information Technology*, 28(3):17.

[Rasappan *et al.*, 2024] Rasappan, P., Premkumar, M., Sinha, G. et Chandrasekaran, K. (2024). Transforming sentiment analysis for e-commerce product reviews: Hybrid deep learning model with an innovative term weighting and feature selection. *Information Processing & Management*, 61(3):103654.

[Revathy *et al.*, 2023] Revathy, V., Pillai, A. S. et Daneshfar, F. (2023). Lyemobert: Classification of lyrics' emotion and recommendation using a pre-trained model. *Procedia Computer Science*, 218:1196–1208.

[Saabith *et al.*, 2021] Saabith, S., Vinothraj, T. et Fareez, M. (2021). A review on python libraries and ides for data science. *Int. J. Res. Eng. Sci*, 9(11):36–53.

[Saraswat *et al.*, 2020] Saraswat, M., Chakraverty, S. et Kala, A. (2020). Analyzing emotion based movie recommender system using fuzzy emotion features. *International Journal of Information Technology*, 12:467–472.

[Savci et Das, 2023] Savci, P. et Das, B. (2023). Prediction of the customers' interests using sentiment analysis in e-commerce data for comparison of arabic, english, and turkish languages. *Journal of King Saud University-Computer and Information Sciences*, 35(3):227–237.

[Sinha et Dhanalakshmi, 2022] Sinha, B. B. et Dhanalakshmi, R. (2022). Evolution of recommender paradigm optimization over time. *Journal of King Saud University – Computer and Information Sciences*, 34:1047—-1059.

[Souilah, 2019] Souilah, S. (2019). Nouvelle méthode de démarrage à froid pour les systèmes de recommandation. Master's thesis, Université de 8 Mai 1945 – Guelma, Guelma, Algeria.

[Su et Khoshgoftaar, 2009] Su, X. et Khoshgoftaar, T. M. (2009). A survey of collaborative filtering techniques. *Advances in Artificial Intelligence*, 2009(1):421425.

[Sun *et al.*, 2018] Sun, Q., Berkelbach, T. C., Blunt, N. S., Booth, G. H., Guo, S., Li, Z., Liu, J., McClain, J. D., Sayfutyarova, E. R., Sharma, S. *et al.* (2018). Pyscf: the python-based simulations of chemistry framework. *Wiley Interdisciplinary Reviews: Computational Molecular Science*, 8(1):e1340.

[Tabassum et Patil, 2020] Tabassum, A. et Patil, R. R. (2020). A survey on text preprocessing & feature extraction techniques in natural language processing. *International Research Journal of Engineering and Technology (IRJET)*, 7(06):4864–4867.

[Tyagi et Sharma, 2018] Tyagi, A. et Sharma, N. (2018). Sentiment analysis using logistic regression and effective word score heuristic. *International Journal of Engineering and Technology (UAE)*, 7(2.24):20–23.

[Vaidya et Khachane., 2017] Vaidya, N. et Khachane., A. R. (2017). Recommender systems - the need of the e-commerce era. *In 2017 IEEE International Conference on Computing Methodologies and Communication (ICCMC)*, pages 100–104.

[Vapnik, 2013] Vapnik, V. (2013). *The nature of statistical learning theory.* Springer science & business media.

[Vaswani *et al.*, 2017] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł. et Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, 30.

[Winkler *et al.*, 2019] Winkler, J. P., Grönberg, J. et Vogelsang, A. (2019). Optimizing for recall in automatic requirements classification: An empirical study. *In 2019 IEEE 27th international requirements engineering conference (RE)*, pages 40–50. IEEE.

[Wongkar et Angdresey, 2019] Wongkar, M. et Angdresey, A. (2019). Sentiment analysis using naive bayes algorithm of the data crawler: Twitter. *In 2019 Fourth International Conference on Informatics and Computing (ICIC)*, pages 1–5. IEEE.

[Yue *et al.*, 2019] Yue, L., Chen, W., Li, X., Zuo, W. et Yin, M. (2019). A survey of sentiment analysis in social media. *Knowledge and Information Systems*, 60:617–663.

[Zangerle et Bauer, 2022] Zangerle, E. et Bauer, C. (2022). Evaluating recommender systems: survey and framework. *ACM computing surveys*, 55(8):1–38.

Annex 1


RIF 2024
The 13th International Conference on
Research in computIng at Feminine
CERTIFICATE OF PARTICIPATION
awarded to
Mehenaoui Zohra MZ, Merabti Chayma, Tadjer Houda, Laffi Yacine
For participating in the 13th International Conference on Research in computIng at Feminine RIF, held on May 20-21, 2024 at University of Abdelhamid Mehri - Constantine 2.
For the short paper entitled :
A Comparative Study On Sentiment Lexicons For Automatic Labeling
THE GENERAL CHAIR
Pr F. MAGRA-BENCHIKHA
RIF
LA RECHERCHE AU FEMININ.

Annex 2: Start-up creation

A.1 Project presentation

A.1.1 Project idea (proposed solution)

A 2020 study found that nearly three-quarters (71%) of surveyed patients utilize online reviews as their initial step in finding a new doctor. This significant finding highlights the crucial role of online reviews in patients' decision-making processes.
GMHC, an acronym for Guide My HealthCare, is a service designed specifically for the healthcare domain in Algeria. It aims to assist patients and their families in deciding which medical center to choose. The idea originated from observing the struggles citizens face when seeking a diagnosis or healthcare service, particularly in identifying reputable centers that provide high-quality care. This often involves a lengthy decision-making process, including extensive searches through social media reviews and family opinions.
Additionally, an analysis of a dataset of user reviews revealed that many advised caution when selecting a medical center due to its direct impact on patient health outcomes.
Our proposed solution is a recommendation platform for medical centers that leverages artificial intelligence techniques (Machine Learning, Deep Learning) in the medical domain by performing sentiment analysis of people's reviews and feedback. The system will analyze user-generated content from various sources, such as social media, review sites, and forums, to determine the overall sentiment toward different medical centers. This approach enables the platform to offer personalized recommendations based on the collective experiences of previous patients to to simplify the decision-making process.

A.1.2 Proposed values

- **Modernity**
  GMHC uses the latest technology, like artificial intelligence and sentiment analysis, to provide up-to-date healthcare recommendations. We continually update our platform with the newest advancements to keep it modern and effective.

- **Performance**
  Our system works quickly and accurately, analyzing a large amount of user feedback to give reliable and fast recommendations. This efficiency helps users make decisions without long waits.

- **Flexibility**
  GMHC is very adaptable, allowing for a wide range of user preferences and needs. Our platform can handle different types of data and user inputs to provide personalized healthcare recommendations.

- **Task Accomplishment**
GMHC makes it easy to find and choose the right medical center. The platform organizes and presents information clearly, helping users quickly complete their search and make informed decisions.
- **Design**
Our platform has a clean and easy-to-use design. The user interface is attractive and simple to navigate, making it easy for users to find and choose medical centers.
- **Cost Reduction**
GMHC helps users save time and reduce costs associated with poor healthcare choices that lead to frequently changing healthcare providers and the high cost of medical treatments by providing reliable and efficient recommendations.
- **Risk Mitigation**
GMHC helps reduce risks by giving recommendations based on thorough analysis of patient reviews. This helps users avoid medical centers with poor reputations and ensures they receive quality healthcare.
- **Increased Trust**
Builds trust in the healthcare system by offering recommendations based on real patient experiences.
- **Improved Healthcare Outcomes**
By directing patients to highly-rated medical centers, it potentially improves their healthcare outcomes.
- **Transparency**
Offers transparency in the quality of services provided by medical centers through unbiased sentiment analysis.
- **Accessibility**
Our service is easy to access for everyone, including patients and their families across Algeria. GMHC is available on a web platform, so users can reach us anytime, anywhere.
- **User-Friendliness**
GMHC is designed to be very user-friendly. It is simple to use, so users can easily input their needs and get recommendations without needing technical skills.

A.1.3 Team members

1. **Merabti Chayma**: the main developer and project implementer, has competence in developing machine learning and deep learning algorithms and models and web development, including HTML, CSS, and JavaScript. She is responsible for the project's conception, system implementation, and platform development.
2. **Mehennaoui Zahra**: the supervisor of the project, an expert in the field of computer science, and responsible for project management. Her responsibility is to guide the process of realizing the project and provide advice.

Interaction and Communication Mode: The team communicates via weekly meetings (every Monday) or twice per week to ensure the smooth progress of the project and facilitate effective collaboration.

A.1.4 Project objectives

GMHC has the main objective of automating and modernizing Algerian patients' decision-making processes and enhancing their healthcare guidance. In terms of timeline, its objectives are:

Short Term (1 year)

1. Launch the process of performing a dataset collection of Algerian medical centers from various sources.

2. Develop the basic prototype of the recommendation system and make it real-time.

3. Conduct pilot tests with a small user base to gather feedback.

4. Attain an acceptance rate of 10% among Algerian people.

Medium Term (2-3 years)

5. Launch marketing campaigns to raise awareness and attract users.

6. Establish partnerships with healthcare providers and review platforms.

Long Term (4 years)

7. Expand the service to cover more regions in Algeria.

8. Achieve a significant user base and become the leading platform for medical center recommendations.

9. Expand the platform to become international.

Estimation of market share

GMHC aims to reach 10% of the Algerian population in the first year, with the goal of increasing this percentage to 20% by the end of the third year and 30% or more by the fourth year. These percentages are influenced by several factors, including the marketing strategy.

A.1.5 Timeline for project realisation

Phase	Task	**M1**	**M2**	**M3**	**M4**	**M5**	**M6**	**M7**
1	**User Interface Design and Development**	X	X					
2	**Enhancement of Recommendation System Algorithm**		X	X				
3	**Pilot Testing and Feedback Collection**				X	X	X	X
4	**Platform Adjustments Based on Feedback**					X	X	X
5	**Dataset Collection**						X	X
6	**Marketing and Launch Preparation**							X
7	**Expansion and Scalability Planning**							X

A.2 Innovative aspects

A.2.1 Nature of innovations

- **Radical Innovations**
  GMHC introduces a groundbreaking approach to healthcare decision-making in Algeria by utilizing sentiment analysis for medical center recommendations. This fundamentally changes how patients and their families choose healthcare providers, shifting from traditional word-of-mouth and personal research to a data-driven, automated system.

- **Technological Innovations**
  The platform leverages advanced technologies such as machine learning, natural language processing, and sentiment analysis to process and analyze large volumes of user-generated content. This technological innovation enhances the accuracy and reliability of the recommendations provided to users.

- **Market Innovations**
  GMHC addresses a significant gap in the Algerian healthcare market by providing a centralized platform for evaluating medical centers based on real patient experiences. This market innovation improves accessibility to reliable healthcare information and empowers patients with informed choices.

- **Incremental Innovations**
  The platform continuously improves its recommendation system by integrating user feedback, refining algorithms, and expanding data sources. These incremental in-

novations ensure that the service remains up-to-date and increasingly accurate over time.

- **Market Uncertainty**
  Entering a new market with a novel solution involves a degree of uncertainty. GMHC mitigates market uncertainty by conducting pilot tests and gathering user feedback to validate its approach and adapt to the specific needs and preferences of Algerian patients an their families. Thus, it aims to adapt to new technologies continually to ensure the good performance of the platform.

A.2.2 Fields of innovations

- **New processes**
  Increasing profitability through improved operational efficiency. This involves enhancing the methods and workflows within GMHC to reduce costs, speed up performance, and improve the quality of services. By leveraging modern technologies and optimizing processes, GMHC can achieve better outcomes with fewer resources, thereby increasing overall profitability.

- **New functionalities**
  GMHC introduce advanced functionalities such as personalized medical centers recommendations, sharing comments and reviews on them and communicating with each other, contacting healthcare providers and rating them. Thus, Our start-up plans to introduce new functionalities such as providing healthcare providers with sentiment analysis reports on their medical center, personalized health tips for patients, integration with health insurance information, appointment booking features, and real-time feedback options. These new functionalities will enhance user engagement and provide additional value to the platform's users.

- **New customers**
  GMHC targets a new customer segment by focusing on citizens who are searching for healthcare providers and currently lack an automated system for centralizing information and reviews about medical centers in Algeria.

A.3 Strategic analysis of the market

A.3.1 Market Segment

Potential market

1. **All Healthcare Seekers in Algeria**: Any Algerian looking for healthcare services, whether for diagnosis, treatment, or regular check-ups. This includes everyone in the country.

2. **Healthcare Providers**: Medical centers, hospitals, and clinics that want more patients based on positive reviews and recommendations. They can also benefit from sentiment analysis reports on their services. As of 31/12/2019, there are over 8,000 general and specialist doctors in Algeria.

3. **Health Insurance Companies**: Companies that want to work with a platform providing reliable recommendations. This can help them save money by guiding

patients to high-quality providers.

4. **Expats and Foreign Residents**: Foreigners living in Algeria who need help finding good healthcare options since they might not know the local providers well.
5. **Government and Public Health Agencies**: Organizations aiming to improve public health by ensuring people have access to trustworthy healthcare providers.

Target market

1. **Patients**: Individuals who require medical consultations and are actively seeking reliable healthcare providers to manage their conditions.
2. **Healthcare Advocates**: People who are actively involved in promoting healthcare quality and patient rights, and who can influence others' decisions based on their experiences.
3. **Middle and Upper-Income Groups**: Individuals who have the financial means to seek quality healthcare and are likely to value and use a service like GMHC.
4. **Educated Individuals**: People with higher education levels who are more likely to appreciate and use data-driven recommendations for healthcare services.

Choice of Target market

We choose this target market because we observed that citizens frequently seek advice from others regarding which medical center they should visit. Therefore, the target market is substantial, allowing our service to cover a wide segment of the Algerian market. By addressing this common need, GMHC can provide valuable, data-driven recommendations, making it an essential tool for a large number of potential users across the country.

A.3.2 Measuring the intensity of competition

Direct and Indirect Competitors

Direct competitors do not yet exist in the Algerian market, but there are several indirect competitors. These include social media and forums where users discuss healthcare experiences and provide recommendations informally. Additionally, recommendations from friends, family, and colleagues influence healthcare provider choices. Moreover, users may search on search engines like Google for healthcare providers and rely on reviews and ratings displayed in search results.

Strengths and weaknesses of Competitors

while indirect competitors have strengths in leveraging personal experiences and broad access to information, they also exhibit weaknesses related to reliability, bias, and scalability. GMHC can position itself strongly by addressing these weaknesses through its structured approach to sentiment analysis and recommendation system.

A.3.3 Marketing strategies

- **Digital Marketing**: GMHC will focus on content marketing and SEO (Search Engine Optimization) to attract organic traffic. By creating informative blogs and articles about healthcare provider selection and using PPC (pay-per-click) campaigns on platforms like Google Ads.
- **Social Media Marketing**: GMHC plans to engage users on social media through campaigns encouraging them to share healthcare experiences.
- **Partnerships and Collaborations**: GMHC will collaborate with healthcare providers and integrate a subscription-based recommendation system oriented to healthcare providers to enhance accessibility and credibility, expanding its user base and solidifying its market leadership.
- **Freemium Model for Citizens**: GMHC will provide basic healthcare provider recommendations for free to citizens, with premium subscribers receiving additional benefits such as scheduling a visit with a medical center.

Balanced marketing mix for GMHC

A balance in the marketing mix for our startup can be achieved by:

- Continuously updating the platform to retain customers and provide them with the best experience.
- Continuously study user feedback on the platform to understand their needs and preferences, and adjust strategies accordingly.
- Adjust platform pricing based on data analysis and revenue value, and modify free and paid features based on this analysis.
- Study the returns from digital marketing and social media marketing, and balance the overall returns with advertising and marketing campaigns.
- Seek out key partners (e.g., healthcare providers) and strengthen partnerships with them to ensure credibility, expand the user base, and build trust.

A.4 Production and Organization Plan

A.4.1 Production process

1. **System Design:** Creating user interface, developing technical requirements, and prototype.
2. **Development:** Building front-end, back-end systems, integrating sentiment analysis algorithms.
3. **Testing and Quality Assurance:** Conducting comprehensive testing, ensuring accuracy of sentiment analysis.
4. **Deployment:** Preparing scalable infrastructure, conducting user acceptance testing (UAT).

5. **Launch and Marketing:** Executing marketing strategy, emphasizing unique features and benefits.
6. **Post-Launch Optimization:** Gathering feedback, optimizing based on analytics and user interactions.
7. **Customer Support and Maintenance:** Providing ongoing support, ensuring platform stability and security updates.

A.4.2 Catering

Necessary Equipment

Purchasing Policy includes the acquisition of necessary technology and software required to operate the GMHC platform (Sentiment Analysis Software such as IBM Watson Natural Language Understanding or Google Cloud Natural Language API, Database Management System to handle the large volume of data, Data security and privacy tools to protect sensitive medical information, and more)

Key Suppliers

Key Suppliers might include Technology and software suppliers and cybersecurity suppliers.

A.4.3 Manpower

Key manpower roles for GMHC include technical team (Software developers, Data scientists and Database administrators) for the platform, marketing and sales team to handle digital, social media marketing and sales and project managers to coordinate tasks and timelines across different teams and projects.

A.4.4 Key Partnerships

- **Healthcare Providers**: Partnering with medical centers to integrate their services into the GMHC platform and collaborating with individual healthcare professionals (doctors, specialists) to enhance credibility and access to expert healthcare advice.
- **Technology Providers**: Partnering with software development companies to enhance the platform's capabilities in sentiment analysis and recommendation algorithms.
- **Government and Regulatory Bodies**: Collaborating with regulatory authorities to ensure compliance with healthcare regulations and standards. This allows the start-up to ensure that it operates in a legal and ethical manner, contributing to building trust between it and the regulatory authorities as well as the users.

A.5 Financial plan

A.5.1 Costs and charges

Costs Details

- **Development Costs**: this includes costs of Software development that are related to developing the GMHC platform, including Salaries of hired developers, purchasing software licenses, and infrastructure setup and technology Infrastructure for example for cloud hosting services and servers needed to support the platform.
- **Research Costs**: this includes costs of Data analytics oriented to acquire and analyze healthcare data, improving recommendation algorithms, and conducting market research.
- **Marketing and Promotion**: Expenses for SEO, PPC advertising, social media campaigns, and content creation to promote GMHC and attract users.
- **Legal and Compliance**: Costs for legal consultations, drafting contracts, and ensuring compliance with healthcare regulations and data protection laws.

Funding sources

- **Equity Financing**:
  Attracting capital from private investors or venture capitalists.
- **Loans and Bank Financing**:
  Obtaining loans from banks or financial institutions to finance company activities such as technological development and marketing.
- **Government Support and Grants**:
  Utilizing government support programs and grants available for startups working in fields of innovation, technology, and healthcare.

Repayment plan

Creating a clear and flexible repayment plan is essential for attracting investors and securing loans for GMHC. By establishing structured repayment schedules that align with our projected cash flow and growth, we can highlight this plan in marketing materials to build trust with lenders and investors, thereby enhancing GMHC's position as a credible and strategically managed startup.

A.5.2 Turnover

Optimistic and Pessimistic Scenarios

- **Optimistic Perspective**:
  GMHC is expected to experience significant growth in revenue, thanks to a strong marketing strategy and fruitful partnerships with healthcare providers and insurance companies. The increasing demand for sentiment analysis-based recommendations will attract a large number of users seeking reliable and well-documented information about healthcare providers. With the expansion of services and targeting new

markets, the revenue could reach higher levels than initially anticipated, strengthening the company's position in the market and attracting more investors and strategic partners.

- **Pessimistic Perspective**:
  GMHC might face substantial challenges in achieving the projected revenue. There could be a slowdown in user adoption of new recommendation services, especially being in competition from traditional alternatives like recommendations from friends and family and reviews on social media. Additionally, high operational costs and difficulties in securing sufficient funding may slow the company's growth. If GMHC fails to meet market expectations and secure a wide user base, it could result in lower-than-expected revenue, negatively impacting the company's sustainability and future growth.

Turnover Details

Year	Number of Premium Citizens	Premium Pricing Yearly (DA)	Number of Healthcare Providers Subscriptions	Subscription Pricing Yearly (DA)	Premium Services Revenue (DA)	Subscription Revenue (DA)	Total Revenue (DA)
N	200	4,800	100	9,600	960,000	960,000	1,920,000
N+1	400	4,800	200	9,600	1,920,000	1,920,000	3,840,000
N+2	600	4,800	300	9,600	2,880,000	2,880,000	5,760,000
N+3	1000	4,800	400	9,600	4,800,000	3,840,000	8,640,000
N+4	1200	4,800	600	9,600	5,760,000	5,760,000	11,520,000

A.6 BMC

Key Partners

-Hospitals and medical centers
-Sentiment analysis software providers
-Healthcare professionals
-Data storage and cloud computing providers
-Funders and investors.

Key Activities

-Data collection from hospitals and medical centers.
-Sentiment analysis of patient feedback and reviews.
-Developing and refining the recommendation algorithm.
-User interface design and development.
-Marketing and promotion

Key Resources

-Data scientists and machine learning engineers.
-Healthcare domain experts.
-Software developers and engineers.
-Marketing and sales team.
-Cloud computing infrastructure.

Value Proportions

- Improved patient satisfaction by recommending hospitals and medical centers based on sentiment analysis.
-Enhanced decision-making for patients in choosing healthcare facilities.
-Increased efficiency for hospitals and medical centers in addressing patient concerns and improving service quality.

Customer Relationship

-Automated recommendation system for patients.
-Direct communication with hospitals and medical centers

Channels

-Online platform for patients.
-Direct sales and partnerships with hospitals and medical centers.
-Social media marketing and digital advertising.

Customer Segments

Patients
Hospitals and medical centers

Cost Structure

-Development and maintenance of the recommendation system.
-Employee salaries and benefits.
-Marketing and advertising expenses.
-Cloud computing and infrastructure costs.
-Legal and regulatory compliance costs.

Revenue Streams

Subscription fees
Revenue earned from premium users' prices
Revenue generated from ads of healthcare-related businesses