EXPLAINABLE RECOMMENDER SYSTEMS

HASAN SERHAT GÜNDÜZ

BOĞAZİÇİ UNIVERSITY

2025

EXPLAINABLE RECOMMENDER SYSTEMS

Thesis submitted to the

Institute for Graduate Studies in Social Sciences

in partial fulfillment of the requirements for the degree of

Master of Arts

in

Business Information Systems

by

Hasan Serhat Gündüz

Boğaziçi University

2025

Explainable Recommender Systems

The thesis of Hasan Serhat Gündüz

has been approved by:

Assist. Prof. Aysun Bozanta Hakyemez _____
(Thesis Advisor)

Assoc. Prof. Nazım Ziya Perdahçı _____

Prof. Birgül Kutlu Bayraktar _____
(External Member)

January 2025

DECLARATION OF ORIGINALITY

I, Hasan Serhat Gündüz, certify that

- I am the sole author of this thesis and that I have fully acknowledged and documented in my thesis all sources of ideas and words, including digital resources, which have been produced or published by another person or institution;
- this thesis contains no material that has been submitted or accepted for a degree or diploma in any other educational institution;
- this is a true copy of the thesis approved by my advisor and thesis committee at Boğaziçi University, including final revisions required by them.

Signature.....

Date.....

ABSTRACT

Explainable Recommender Systems

Recommender systems generally produce successful recommendations using rating data. However, recommendation systems that rely solely on rating data cannot capture users' feelings and thoughts in detail, so they may exhibit limited performance in some cases. To overcome this limitation, the use of textual data, especially user reviews, in recommender systems provides greater insight into individual preferences and sentiments. In this way, the accuracy and level of personalization of recommendations can be significantly increased. Explainability of recommender systems is another critical issue. Explaining the decision-making mechanism and the reasoning behind recommendations enhances the fairness and transparency of these systems, which increases trust in the system and helps reveal any biases that may be present in the system. This study introduces a novel recommendation algorithm built on user-generated comments. Language models determine the similarity between users' comments. Instead of relying on explicit user ratings, sentiment scores derived from these comments reflect users' opinions about products. Recommendations for the target user are generated by leveraging the sentiment scores of the most similar users. Ten different language models are employed to calculate similarity scores, and their performances are evaluated based on recommendation accuracy. We selected Sentence-BERT (SBERT), the best-performing language model, to calculate similarity scores for our algorithm. Using the attention mechanisms of text representation models and the SHAP method, the system provides reasons behind each recommendation, resulting in an explainable recommendation system.

ÖZET

Açıklanabilir Tavsiye Sistemleri

Tavsiye sistemleri genellikle derecelendirme verilerini kullanarak başarılı tavsiyeler üretmektedir. Ancak, yalnızca derecelendirme verilerine dayanan tavsiye sistemleri kullanıcıların duygularını ve düşüncelerini ayrıntılı olarak yakalayamaz, bu nedenle bazı durumlarda sınırlı performans gösterebilirler. Bu sınırlamanın üstesinden gelmek için, tavsiye sistemlerinde özellikle kullanıcı yorumları gibi metinsel verilerin kullanılması, bireysel tercihler ve duygular hakkında daha fazla bilgi sağlar. Bu sayede, tavsiyelerin doğruluğu ve kişiselleştirme düzeyi önemli ölçüde artırılabilir. Tavsiye sistemlerinin açıklanabilirliği ise başka bir kritik konudur. Karar alma mekanizmasının ve tavsiyelerin arkasındaki mantığı açıklamak, bu sistemlerin adaletini ve şeffaflığını artırır böylece sisteme olan güven artar ve sistemde mevcut olabilecek önyargıları ortaya çıkarmaya yardımcı olur. Bu çalışma, kullanıcı tarafından üretilen yorumlara dayalı yeni bir tavsiye algoritması sunmaktadır. Dil modelleri, kullanıcı yorumları arasındaki benzerliği belirlemektedir. Açık kullanıcı derecelendirmelerine dayanmak yerine, bu yorumlardan türetilen duygu puanları, kullanıcıların ürünler hakkındaki görüşlerini yansıtmaktadır. Hedef kullanıcı için öneriler, en benzer kullanıcıların duygu puanlarından yararlanılarak oluşturulmaktadır. Benzerlik puanlarını hesaplamak için on farklı dil modeli kullanılmış ve bu modellerin performansları öneri doğruluğuna göre değerlendirilmiştir. Algoritmamız için, en iyi performansı gösteren dil modeli olan Sentence-BERT (SBERT) seçilmiştir. Metin temsil modellerinin dikkat mekanizmaları ve SHAP yöntemi kullanılarak sistem, her önerinin arkasındaki nedenleri açıklamakta ve sonuç olarak açıklanabilir bir tavsiye sistemi sunmaktadır.

ACKNOWLEDGMENTS

Today, I am happy to have completed my three-year enjoyable and educational graduate journey. I would like to thank the people who have supported me throughout this process.

Firstly, I would like to express my deepest gratitude to my thesis advisor Assist. Prof. Aysun Bozanta Hakyemez for her valuable guidance, encouragement, support and patience. Her contribution to the thesis is truly invaluable. This thesis could not be written without her.

I would like to express my endless gratitude to Osman Yücel, Ph.D. for his valuable contributions and support throughout the thesis process.

I would like to thank Assist. Prof. Ahmet Onur Durahim, who has always guided me and supported me throughout my graduate program.

I am grateful to Boğaziçi University Department of Management Information Systems and the Institute of Social Sciences Graduate Studies for creating an environment that encourages research and questioning, and for the resources and support necessary for this study.

I would like to thank my very valuable wife Gülçin Avcı Gündüz, who has always been the invisible hero of this process with her love and psychological support.

I would like to thank my mother Perihan Gündüz, my father İsmail Gündüz, my brother Serdar Gündüz and his family for always believing in me and encouraging me to start my master's program.

With love..

TABLE OF CONTENTS

CHAPTER 1 : INTRODUCTION 1

CHAPTER 2 : LITERATURE REVIEW 6

2.1 Recommender systems 6
2.2 Text-based recommender systems with language models 10
2.3 Explainability in recommender systems 13

CHAPTER 3 : METHODOLOGY 16

3.1 Data collection 16
3.2 Data preprocessing 17
3.3 Base recommender systems 20
3.3 Development of the recommender systems 20
3.4 Evaluation metrics 27
3.5 Explainability methods 30

CHAPTER 4 : RESULTS 33

4.1 Results of base recommender systems 33
4.2 Results of review and rating-based recommender systems 37
4.3 Results of review and sentiment-based recommender systems 39
4.4 Results of explainability of recommender systems 41

CHAPTER 5 : DISCUSSION 48

CHAPTER 6 : CONCLUSION 50

5.1 Limitations 52
5.2 Future works 52
5.3 Business implications 53

APPENDIX A : AMAZON DATASET RESULTS OF ALL RECOMENNDER SYSTEMS 55

APPENDIX B : YELP DATASET RESULTS OF ALL RECOMENNDER SYSTEMS 56

APPENDIX C : IMDB DATASET RESULTS OF ALL RECOMENNDER SYSTEMS 57

REFERENCES 58

LIST OF TABLES

Table 1. Literature Review Summary 15

Table 2. Selected Subcategory Raw Dataset Dimensions 18

Table 3. Dimensions of Datasets After the Data Elimination Process 19

Table 4. Confusion Matrix 29

Table 5. Amazon Dataset Results of Base Recommender Systems 34

Table 6. Yelp Dataset Results of Base Recommender Systems 35

Table 7. IMDB Dataset Results of Base Recommender Systems 36

Table 8. Amazon Dataset Results of Review and Rating-Based Recommender Systems 37

Table 9. Yelp Dataset Results of Review and Rating-Based Recommender Systems 38

Table 10. IMDB Dataset Results of Review and Rating-Based Recommender Systems 38

Table 11. Amazon Dataset Results of Review and Sentiment-Based Recommender Systems 39

Table 12. Yelp Dataset Results of Review and Sentiment-Based Recommender Systems 40

Table 13. IMDB Dataset Results of Review and Sentiment-Based Recommender Systems 41

Table 14. Recommendation Information 42

Table 15. The Reviews of Similar Users on the Recommended Product 43

Table 16. The Similarity Score Between the Main User and the Similar/Causing Users 45

Table 17. Reviews Made Previously on the Same Product 45

LIST OF FIGURES

Figure 1. Types of developed recommender systems 21

Figure 2. Sentiment analysis of the recommended product's review 43

Figure 3. Sentiment analysis of the first similar user's review 44

Figure 4. Sentiment analysis of the second similar user's review 44

Figure 5. Top 10 words having high attention score for main user’s review 46

Figure 6. Top 10 words having high attention score for first similar user’s review . 46

Figure 7. Top 10 words having high attention score for second similar user’s review 47

CHAPTER 1

INTRODUCTION

Recommendation systems are used in many areas such as e-commerce, social media, and online content platforms. Recommendation systems aim to predict the items that the user may be interested in by examining users' interactions with services or products and thus provide a better user experience (Lu, Wu, Mao, Wang, & Zhang, 2015).

Recommendation systems are generally developed using one of three main approaches namely Collaborative Filtering, Content-Based Filtering or hybrid methods (Adomavicius & Tuzhilin, 2005). The appropriate method is chosen depending on the application area and data structure, and it is aimed at recommender systems to provide personalized and effective recommendations.

Recommendation systems work by using users' interactions with products, purchasing behaviors, rating scores they give to products or review data they write about products (Ansari, Essegaier, & Kohli, 2000). The fact that the user purchases or interacts with the product does not alone indicate the user's satisfaction. For this reason, many recommendation systems also analyze the rating data and take into account the satisfaction levels of the users. However, the fact that users generally tend to make excessively high or low ratings causes a bias problem (Shani & Gunawardana, 2011). This situation limits the generalizability of the rating data. In addition, the rating data does not capture the users' feelings and thoughts about the product in detail (Ghose & Ipeirotis, 2010).

To create a recommendation system that produces more successful results and overcomes data sparsity, different types of user and product data that will

increase the performance of systems are used in recommendation systems (Adomavicius & Zhang, 2012). Text data is also one of the data types used to increase the success of recommender systems (Kanwal, Nawaz, Malik, & Nawaz, 2021). User comments or reviews about products eliminate the problem of bias because they detail users' feelings, perspectives, and what they care about in products.

It is difficult for recommender systems to provide explanations about their recommendations due to the complex algorithms used and many recommender systems cannot explain the reason for the recommendation they produce. It becomes more difficult to establish causality with the use of advanced state-of-the-art models. Providing explainability is important because it enables transparency and accountability in recommendation systems (Abdollahi & Nasraoui, 2018).

In this thesis, comments made by users on products were used as text-type data, and novel user-based collaborative filtering recommender systems were developed. Two types of recommendation systems have been developed: review and rating-based recommendation systems and review and sentiment-based recommendation systems. For these systems, similarities between comments made by users about the same products were calculated. In order to determine the similarities in the reviews; Natural Language Processing based Text Representation models namely Bag of Words (BoW), Word2Vec, ELMo, GloVe, BERT, DistilBERT, SBERT, RoBERTa, BERT Large, and SBERT Large were used (Sarzynska-Wawer, et al., 2021; Liu Y. , 2019).. Similarity calculations were made with the cosine similarity metric. The similarity score between two users was calculated by taking the average of the similarity scores in the reviews made by two users for the same products.

Review and rating-based recommendation system was created by combining the calculated similarity scores between users and rating data. Review and sentiment-based recommendation system was created by combining the calculated similarity scores between users and sentiment scores of review data. The two newly created types of recommender systems were compared with the base recommendation systems. Three different datasets, Amazon, Yelp and IMDB, were used to test the results of the models developed in the study.

In this study, in order to provide explainability for the novel recommender systems created, attention mechanisms of text representation models are analyzed using the Transformers-Interpret approach (Pierse, 2021). Subsequently, the SHAP method is used to identify the words that have the most significant impact on sentiment analysis (Diwali et al., 2023). This approach provides in-depth insights regarding the reasons for the recommendations by identifying the specific users and comments that influenced each recommendation, as well as highlighting the words within the comments that had the greatest impact.

This study aims to develop a novel and explainable recommender system that can be an alternative for the systems developed with the rating data and examines the following research questions:

1. How do recommender systems that solely rely on user-generated textual data perform in comparison to rating-based recommender systems?
2. What is the impact of employing different language models on the performance of similarity calculations in a sentiment-based recommender system?

3. How can attention mechanisms and SHAP-based explanations improve the transparency of recommender systems by providing interpretable insights into recommendation decisions?

The scientific contributions of this study can be listed as follows:

- The developed novel review and sentiment-based recommender system, using text similarities and sentiment analysis, showed comparable performance with rating-based recommendation systems without the need for traditional rating data, and in some cases, surpassed.
- The review and sentiment-based recommender system was evaluated on three separate datasets. The system proved to be a robust alternative to traditional recommendation methods by providing generalizable and reliable results.
- Since the developed recommender system is explainable, it specifies the specific users, reviews, sentiments, and keywords that affect each recommendation. Therefore, the system increases the transparency, interpretability, and reliability of the recommendation process.

The recommender systems developed in this study offer significant benefits to users, businesses, and developers. For businesses, these systems enhance customer satisfaction by delivering personalized recommendations, leading to increased customer loyalty and repeat purchases. Additionally, by understanding user preferences more effectively, businesses can also identify emerging trends and tailor their product offerings to better meet market demands.

The remaining structure of this thesis is organized as follows: Chapter 2 presents a comprehensive literature review highlighting related research and developments in the field. Chapter 3 outlines the methodology, detailing the dataset

characteristics, models used, and approaches developed. Chapter 4 presents the results of the proposed approach, including performance comparisons and key findings, and its comparison with baseline models. Chapter 5 offers a summary of the findings and Chapter 6 concludes the thesis with explain of directions for future research.

CHAPTER 2

LITERATURE REVIEW

This chapter provides a comprehensive review of the literature on the methods and algorithms employed in this study. It examines the evolution of research in the fields of recommender systems and explainability, highlighting the foundational studies and key scientific publications referenced. Specifically, significant contributions in the literature on recommender systems, text-based recommender systems utilizing language models, and explainability in recommender systems are examined in detail. Each subheading provides an analysis of the relevant research in these areas and evaluates their influence on the development of this thesis.

2.1 Recommender systems

With the advancements in technology and software development, recommender systems aim to eliminate people's indecisiveness and improve decision-making processes in daily life.

If there were no recommender systems, users would spend more time choosing products and services and would need detailed information about the products or services to make the right choices. This would cause an information overload problem (Khusro, Ali, & Ullah, 2016). Because the recommendations produced by recommender systems, on average, give better results than people's recommendations (Krishnan, Narayanashetty, Nathan, & Davies, 2008). In a world without recommender systems, businesses would need more marketing budgets to attract user attention and more operations budgets to improve user experience.

Each recommender system aims to produce the best recommendations in its field of work. In order to produce better recommendations, different algorithms and approaches can be developed using different datasets. Recommender systems are divided into 3 main types according to their working principles namely collaborative filtering, content-based filtering and hybrid recommender systems (Adomavicius & Tuzhilin, 2005).

Collaborative filtering generates recommendations based on the similarity between users' preferences (Adomavicius & Tuzhilin, 2005; Herlocker, J.L. et al., 2004). The term collaborative filtering was first used in the Tapestry study (Goldberg, Nichols, Oki, & Terry, 1992). Tapestry focused on sending e-mails based on user interest, driven by the increasing use of e-mail (Goldberg, Nichols, Oki, & Terry, 1992). This method assumes that users with similar preferences in the past are likely to have similar preferences in the future. The content-based filtering method focuses on the similarity between items. Users are suggested products that are similar to the products they have preferred or rated highly in the past (Adomavicius & Tuzhilin, 2005). In other words, the user profile is matched with similar products. In the content-based filtering approach, there is an assumption that users who prefer a product or give high scores will also be satisfied with similar products (Pazzani, 2007). Hybrid systems aim to minimize the disadvantages of both methods by combining the strengths of collaborative filtering and content-based filtering methods (Burke, 2002).

The integration of deep learning into recommender systems has greatly accelerated developments in this field. Deep learning techniques are successful in capturing intricate and non-linear user-item relationships, enabling the modeling of more complex abstractions through hierarchical data representations in deeper

network layers. Thanks to the developments in deep learning-based recommender systems, the quality of recommendations has significantly enhanced compared to traditional model-based approaches. Furthermore, deep learning-based recommender systems can process various forms of data including contextual, textual and visual inputs, allowing them to develop different types of recommender systems (Zhang, Yao, Sun, & Tay, 2019).

Textual data has served as a rich source of user preferences and product features with the development of language models in the field of neural networks. Because language models particularly transformer-based models, have demonstrated remarkable success in the field of recommender systems by having their advanced capabilities in understanding and processing textual data. The broad language understanding capabilities of these models increase the accuracy of recommendations. However, language models require large datasets and significant computational resources to deliver high-quality results in recommender systems (Zhang, et al., 2021).

The integration of pre-trained Large Language Models (LLMs) and prompt learning into recommendation systems has been another trending area of study in the field of recommendation systems in recent years. According to Zhao et al. (2023), LLMs have introduced unprecedented capabilities in understanding and generating human-like text, enabling systems to better capture nuanced user preferences and provide context-aware recommendations.

Reinforcement learning (RL) has gained significant attention in recommender systems for its ability to model dynamic user interactions and treat recommendations as a sequential decision-making process. Unlike traditional methods like collaborative or content-based filtering, RL considers the evolving nature of user

preferences and engagement. The development of deep reinforcement learning (DRL) has handled the scalability challenges of RL and facilitated its use in recommender systems (Afsar, Crump, & Far, 2022). Recent research underscores DRL's potential to improve personalization, adaptability, and efficiency in recommendations.

The concepts of bias and fairness are among the current research topics in the field of recommender systems. Since recommendations may involve the allocation of social resources, they must be fair. Unfair recommendations not only raise ethical concerns but also threaten the long-term sustainability of the recommender systems. As a result, fairness in recommender systems has received increasing attention in recent years. Biases can exacerbate fairness problems and make it harder to ensure fairness in recommendation systems (Wang, Ma, Zhang, Liu, & Ma, 2023). Therefore, bias related to the concept of fairness has also received significant attention in studies in the field of recommender systems.

Explainable recommendation systems are designed to help users understand why the recommendations made by the algorithm are recommended. In this way, it helps to increase the transparency, persuasiveness, effectiveness, reliability and satisfaction of user systems (Zhang & Chen, 2020). The success of recommendation systems has increased with the complex algorithms used, but explainability has become difficult due to complexity. So, one of the trending research topics in the recommender system field is explainability.

2.2 Text-based recommender systems with language models

The use of text-based data in recommender systems is an important research area in modern recommender systems. Text representation models convert text data into vectors while preserving their semantic meaning.

In recent years, many language models have been developed to process text-based data. The BoW model is one of the simplest and most widely used methods for text representation. It represents texts as words and their frequencies within a document (Gabrilovich & Markovitch, 2006). However, BoW cannot capture word order, syntactic structure, or semantic relationships. In contrast to BoW, Word2Vec represents a significant improvement by using deep learning models to generate vector representations of words, and places semantically related words closer together in the vector space. This enables Word2Vec to capture semantic relationships and word analogies (Mikolov, Sutskever, Chen, & Corrado, 2013). GloVe uses a local context window to learn word representations and relies on global word co-occurrence statistics, unlike Word2Vec. By leveraging both local and global information, GloVe provides a more robust representation of word meanings (Pennington, Socher, & Manning, 2014).

ELMo (Embeddings from Language Models) generates context-dependent word embeddings by training a bidirectional LSTM. Unlike traditional static embeddings, ELMo determines word meanings based on their context, making it effective for words with multiple meanings. ELMo has outperformed its predecessors on many language processing tasks namely sentiment analysis, named entity recognition, and question answering (Sarzynska-Wawer, et al., 2021). BERT (Bidirectional Encoder Representations from Transformers) pre-trains deep language representations by utilizing a bidirectional transformer architecture. It predicts

missing words in a sentence (masked language modeling) and sentence relationships (next sentence prediction), achieving greater effectiveness in tasks of text classification, sentiment analysis, and question answering (Devlin, 2018). SBERT (Sentence-BERT) is a model based on the BERT model that generates sentence embeddings instead of word embeddings. This makes SBERT especially useful for semantic textual similarity, paraphrase identification, and sentence clustering (Reimers, 2019). The aim of RoBERTa (Robustly Optimized BERT Pretraining Approach) improves BERT by optimizing its training process, achieving superior performance in many natural language processing tasks due to its different training process (Liu Y. , 2019).

Pre-trained language models, namely BERT, RoBERTa, and GPT, have been widely tested in recommender systems. The results show that the models can be adapted for recommendation purposes and are promising, but that these models need to be customized and biases reduced (Zhang, et al., 2021).

Designing recommender systems for specific domains using language models is a growing area of research. UNBERT is a user-news matching model that enhances text representation by using a pre-trained BERT model (Zhang, et al., 2021). Besides improving text representation, UNBERT captures user-news matching signals at both the word and news levels. The model has been tested on the Microsoft News Dataset (MIND). The results highlight a significant improvement in the accuracy of news recommendations using language models (Zhang, et al., 2021).

Sentiment analysis can be performed on text data using language models. The sentiments extracted from text data can be utilized in recommendation systems. Zhang and Zhang (2022) extracted topics and emotional tendencies from movie reviews using BERT and developed a user interest model and a product feature

model based on these emotional tendencies to improve content-based recommendation algorithms. Thus, a movie recommendation algorithm based on sentiment analysis and the LDA (Latent Dirichlet Allocation) topic model was designed. The results of the study show that the performance of the recommendation system was significantly improved, and the variety of recommendations increased (Zhang & Zhang, 2022).

Conversational Recommender Systems (CSR) have the potential to improve traditional recommender systems through the use of interaction. The BERT and RoBERTa models have been investigated for use in conversational recommender systems (CRS). It has been understood that BERT acts on the knowledge stored in its parameters in CRS. It has also been shown that BERT is effective in distinguishing relevant answers from irrelevant answers (Penha & Hauff, 2020).

The integration of LLM into recommendation systems is another trend that has become evident in recent years. Studies have shown that the performance of recommendation systems improves with better understanding of text-based data. This performance increase is because LLMs capture the relationships between words and provide more personalized recommendations. Although the performance of recommendation systems has improved, it has been stated that there are still challenges in interpretability and the explainability of these systems (Zhao, et al., 2023).

Recommendation systems can be created by using the similarities between texts. Similarities between texts can be detected by language models' ability to analyze texts and vectorize them (Koroteev, 2021). There are computational approaches such as Cosine similarity, Jaccard and Euclidian to express similarities between texts with a score (Wang & Dong, 2020).

2.3 Explainability in recommender systems

Despite being widely used, recommender systems often operate as "black boxes" that do not explain the decision-making process behind the recommendation. This lack of transparency can decrease trust and engagement in recommender systems when the recommendations are inaccurate (Herlocker, Konstan, & Riedl, 2020). Therefore, there is increasing interest in the area of explainability studies that make recommendation systems transparent.

Shapley Additive Explanations (SHAP) explains the model's decisions by calculating the contribution of each feature to the model's prediction. SHAP provides an explanation of particularly complex model outputs, and these explanations help users better comprehend the model's decision-making process (Lundberg, 2017). When integrated with sentiment analysis of text representation models, SHAP demonstrates the impact of each word on overall sentiment in texts. Thus, explanations of the outputs of text-based systems are provided. This approach helps uncover hidden biases or inconsistencies (Diwali et al., 2023).

Additionally, explainability research are also carried out in transformer-based language models. BERTViz, a tool that visualizes attention mechanisms in transformer-based models, is one such method used to explain text-based recommender systems. This method provides a deeper understanding of how different words or expressions contribute to a particular recommendation (Vig, 2019).

Recently, explainability has been further advanced in this field with the integration of LLMs in recommender systems. A study in this direction is the Chat-Rec system, which augments traditional recommender systems and introduces explainability using ChatGPT. According to the results of the study, Chat-Rec

significantly improved the recommendation results and performed well in zero-shot rating prediction tasks (Gao, Sheng, Xiang, Xiong, & Wang, 2023).

Explainability studies have also been conducted for recommender systems where visual and textual data are used together, by utilizing hybrid attention mechanisms. Textual attributes extracted from user reviews and visual aspects derived from product images were analyzed together to provide explainable outputs (Liu, Zhang, & Gulla , 2020).

Personalized Transformer for Explainable Recommendation (PETER) is a combined novel model that can produce both recommendations and explanations (Li, Zhang, & Chen, 2021). PETER is a small, untrained Transformer with only 2 layers. The Transformer personalizes and produces explainable recommendations. PETER displays recommendation explanations in textual form. It has achieved strong results according to both explainability and text quality metrics (Li, Zhang, & Chen, 2021).

Recent studies in literature that utilize text data and language models were selected for comparison. These studies were compared with this thesis study in terms of datasets, algorithms, sentiment analysis and explainability. The recommendation system developed in this study was tested on more datasets and with more algorithms than other studies. In addition, the study in this thesis was the only study that both applied sentiment analysis and produced explainable recommendations. A summary of the developed study compared with the studies in literature is given in Table 1 below.

Table 1. Literature Review Summary

Authors	Datasets	Algorithms	Sentiment Analysis	Explainability
Liu, et al. (2020)	MovieLens, Pinterest	AnteRNN	X	√
Zhang, et al. (2021)	MovieLens	BERT, GPT2	X	X
Zhang, et al. (2021)	MIND	BERT	X	X
Zhang, et al. (2022)	Movie Data	BERT	√	X
Jeong, et al. (2023	Amazon	RoBERTa and VGG-16	X	X
Javaji, et al. (2023)	Goodreads	RoBERTa and SBERT	X	X
This Study	Amazon, Yelp, IMDB	BagofWords, Word2Vec, GloVe, ELMo, BERT, DistilBERT, RoBERTa, SBERT	√	√

CHAPTER 3

METHODOLOGY

The methodology of the study is discussed in three different sections data collection and preprocessing, recommendation system development, and explainability of the recommendation system. All technical steps within the methodology were implemented using the Python programming language.

3.1 Data collection

Amazon (Hou, et al., 2024) , YELP (Yelp Dataset, 2022), and IMDB (Pal, Barigidad, & Mustafi, 2020) datasets were used for this study. These datasets include different types of businesses, products, and services. The reason of using datasets from different types of businesses is to make the findings of the study more generalizable.

The Amazon raw dataset contains data between the years 2000 and 2023, the Yelp raw dataset contains data between the years 2005 and 2022, and the IMDB raw dataset contains data between the years 1999 and 2019.

The criteria for selecting the dataset were that the dataset should include user, item, rating and review attributes.

Since the raw datasets are very large, long process time and high computation power are required for processing the data. For this reason, the data sizes were reduced by filtering or eliminating the subcategories from the raw datasets. The Beauty category from the Amazon dataset source, accommodation data in the state of Illinois from the YELP data source and data from 20,000 random users from the IMDB dataset were used.

Variables included in the datasets:

- User ID
- Rating
- Time
- Title
- Review
- Product/Service ID

3.2 Data preprocessing

Amazon Beauty raw dataset contains 701,528 reviews, 631,986 unique users, and 115,709 unique products.

YELP raw dataset consists of business_data, checkin_data, review_data, tip_data, and user_data subdatasets. For Yelp dataset, the business dataset with the review dataset was used along. The raw dataset contains 6,990,280 reviews, 1,987,929 unique users, and 150,346 unique businesses. Illinois state data was filtered on the YELP dataset. Illinois state dataset contains 51,832 reviews, 22,248 unique users, and 2,145 unique businesses.

IMDB raw dataset contains 932,464 reviews, 427,646 unique users, and 1,150 unique movies. A dataset containing 45,121 reviews and 1,150 unique movies was used in this study by randomly taking 20,000 unique users from the IMDB raw dataset. The raw datasets used in this study were summarized in Table 2 below.

Table 2. Selected Subcategory Raw Dataset Dimensions

	Amazon Beauty	YELP-Illinois	IMDB (20k Users)
Reviews	701.528	51.832	45.121
Unique Users	631.986	22.248	20.000
Unique Products/Services/Films	115.709	2.145	1.150

In all 3 datasets, the attribute names were assigned as follows:

- user_id: user identification key
- rating: score given by the user
- title: user comment title
- text: review made by the user
- timestamp: review time
- asin: product, service, or movie identification key

Data elimination was performed on 3 datasets according to the following conditions. After the eliminations are made, the dimensions and information of the data are available in Table 3 below.

- Comments made in languages other than English were removed.
- Data with empty text (review) columns were deleted.
- Rows containing null values were excluded from the dataset.
- Since the similarity between users will be determined from the similarity between their reviews of the same product, users with more than 5 reviews and their comments were selected. Users with fewer reviews and their data were deleted.

Table 3. Dimensions of Datasets After the Data Elimination Process

	Amazon Beauty	YELP - Illinois	IMDB (20k Users)
Reviews	11.287	19.418	45.121
Unique Users	966	1.381	791
Unique Products/Services/Films	5.377	2.018	1.141

Each rating given by each user was adjusted according to all ratings of that user, and a variable called adjusted_rating was created in Eq 1.

$$adjusted\_rating = \frac{rating - \mu}{\sigma} \quad (1)$$

The adjusted_rating variable was created to determine how positive or negative each rating given by the user was according to their own rating standards.

The IMDB dataset has rating values between 0 and 10. For the base recommender system, which only works with ratings, this data is reduced to a scale of 0 to 5. While doing this reduction, the rating values are divided by 2 by rounding up. The reason for rounding up is to get rid of numbers with decimal places and to create the format of the rating variable in other datasets.

Sentiment analysis was performed for each comment of the user, using the fine-tuned Bert model (Mukherjee, 2024), to determine whether it was positive or negative, and the bert_sentiment variable was created. The bert_sentiment variable was assigned a value of 1 for positive reviews and -1 for negative reviews. The bert_sentiment variable was created to enable the user experience to be determined based on reviews.

For this study, the datasets were first divided into two parts, 70% train and 30% test (Du, et al., 2021; Pujahari & Sisodia, 2020). While doing this separation, each user's data was sorted chronologically, and the first 70% remained in the train, while the last 30% was in the test section.

3.3 Base recommender systems

As the base recommendation systems, matrix factorization (MF) based algorithms, namely Singular Value Decomposition (SVD) and Truncated Singular Value Decomposition (TSVD), and K-Nearest Neighbor (KNN) algorithms were applied to all datasets. For all three algorithms, firstly, recommender systems were created for the ratings given by the users. Then, a version of these recommendation systems using the adjusted_rating variable, which is the adjusted version of user ratings, was designed.

Recommendation systems were created using the default hyperparameters of SVD, T-SVD and KNN algorithms. Recommendation systems using the T-SVD algorithm produce recommendations by selecting the 5 most similar users. In the recommender system established with the KNN algorithm, similarity between users was calculated through cosine similarity, and the 5 nearest users were selected using the brute-force method (Adeniyi, Wei, & Yongquan, 2016).

3.3 Development of the recommender systems

Two different types of user-based collaborative filtering recommender system algorithms were designed for this study (Figure 1).

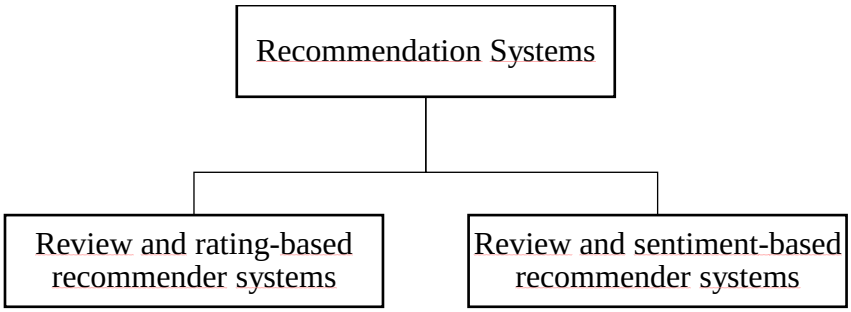

Figure 1. Types of developed recommender systems

The created recommender systems focused on the similarities between users' reviews of the same products while detecting similarities between users. After the similarity between users was detected, the products reviewed by users with a similarity score above 0.50 were selected.

When recommending products between similar users, the first type of recommender system, "Review and rating-based recommender systems", used the ratings given by users. The second type of recommender system, "Review and sentiment-based recommender systems", recommended products using the score called bert_sentiment that is the sentiment scores of user reviews.

3.3.1 Language models

The two recommender systems that were developed first detected the similarity between the reviews that the users had made in the past and thus calculated the similarity score between the users. For similarity, the similarity scores of the reviews that the users had made to the common products in the train dataset were first calculated. Since these similarity scores were text-based, text representation models were used.

These models are as follows:

- BoW (BoW)
- Word2Vec
- ELMo
- GloVe
- BERT
- BERT (large version)
- DistilBert
- SBERT
- SBERT (large version)
- RoBERTa

These models analyze reviews using their algorithms to calculate similarity scores, employing the cosine similarity method for comparison. Cosine similarity is particularly effective in natural language processing as it assesses similarities at both the word and sentence levels, yielding more accurate and meaningful results. Text-based sentiment analysis and recommendation systems leverage this technique to examine relationships between word and sentence vectors (Mikolov, Sutskever, Chen, & Corrado, 2013; Pennington, Socher, & Manning, 2014).

BoW stores texts as a numerical vector-based on the frequency of words. BoW is a simple and fast model, but it cannot capture syntactic structure, and semantic relationships between words because it does not consider the order of the words. Another disadvantage of BoW is that it cannot account for polysemy (words with more than one meaning) or synonymy (words with more than one meaning) (Gabrilovich & Markovitch, 2006).

Word2vec is a deep learning model that converts each word in the text into a high-dimensional numerical vector and detects relationships between words (Mikolov, Sutskever, Chen, & Corrado, 2013). It has the ability to detect similar-sounding words using cosine similarity. The model takes context into account by learning words together with their surroundings. In this study, a pre-trained word embedding model called "GoogleNews-vectors-negative300" was used, which was trained with skip-gram architecture and learned on a large text corpus (a dataset of approximately 100 billion words compiled from Google News articles). It contains 3 million unique words and phrases, and each word or phrase is represented by a 300-dimensional dense vector (Sutskever, Mikolov, & Le, 2013).

GloVe (Global Vectors for Word Representation) is a model that represents words in a meaningful way in a vector space by embedding; it can detect the relationships and matches between words with the help of co-occurrence matrix (Pennington, Socher, & Manning, 2014). The co-occurrence matrix shows the frequency of co-occurrence of words in a corpus. Like Word2Vec, it learns the relationships of words from the context, but unlike it, it bases these relationships on global context (global co-occurrence) information (Pennington, Socher, & Manning, 2014). GloVe can capture both semantic and syntactic relationships between words. In this study, glove.6B.300d model, which was pre-trained with 6 billion words on Wikipedia 2014 and Gigaword 5 corpora and where words are represented with 300-dimensional vectors, was used.

ELMo (Embeddings from Language Models) is a word embedding model that learns the meaning of each word-based on its position in the sentence with the help of birectional LSTM (Peters, Neumann, Iyyer, Gardner, & Clark C, 2018). Bidrectional, that is, it estimates the word meaning by extracting the context from the

previous (left) and next (right) words with both forward and backward LSTM (Peters, Neumann, Iyyer, Gardner, & Clark C, 2018). In addition, it converts it to a vector at the character level, that is, it learns by separating words into characters. It is sensitive to context and can distinguish different meanings of each word. It can update the meaning of a word according to the context of the sentence.

BERT is a contextual language model developed by Google that learns words by considering both the words before and after them in two directions using the Transformer architecture with the Masked Language Modeling method (Devlin, 2018). BERT learns by representing text as a sequence of vectors using self-supervised learning. While BERT is trained with masked token prediction, it has a structure that randomly masks some of the words and tries to predict these masked words, so it can focus on the entire sentence to learn the context of the word. With this masking architecture, it also solves problems such as coreference and polysemy and learns the relationships between two sentences with “Next Sentence Prediction (NSP)” approach (Devlin, 2018).

Two different pre-trained versions of BERT, "bert-base-uncased" and "bert-large-uncased", were used in this study. "bert-base-uncased" is a smaller and faster version of BERT with 12 Transformer encoder blocks and 110 million parameters. "bert-large-uncased" is a more advanced version of BERT with 24 Transformer encoder blocks and 340 million parameters, referred to as "BERT Large" in this study. (Devlin, 2018).

DistilBERT (Distilled version of BERT) is a smaller and faster version of BERT (Sanh, 2019). It uses the Knowledge Distillation method, which allows transferring the knowledge learned by a large model to a smaller model. DistilBERT has 66 million parameters (60% of BERT) and 6 layers. Therefore, it requires less

storage and memory. DistilBERT is 60% faster than BERT and reaches 97% of BERT's language understanding capabilities in terms of performance (Sanh, 2019).

SBERT is a BERT model optimized for calculating similarity between sentences. It is a modified version of BERT to work better at the sentence meaning level. It can calculate similarity between sentences faster and more successfully by creating dense vectors for sentence representations. It uses the cosine similarity metric in sentence similarities (Reimers, 2019). In this study, SBERT models named "paraphrase-MiniLM-L6-v2" and "all-mpnet-base-v2" were used. "paraphrase-MiniLM-L6-v2" is a small and fast SBERT model with 6 transformers layers. "all-mpnet-base-v2" is a SBERT model with 12 transformers layers based on MPNet (Masked and Permuted Pre-trained Network) developed by Microsoft, referred to as SBERT Large in this study. It is more powerful and more complex than "paraphrase-MiniLM-L6-v2" but also slower.

RoBERTa (Robustly Optimized BERT Pretraining Approach) is a language model developed by Facebook AI that aims to improve the performance of the BERT model by optimizing its training with more data and better methods without changing the architecture of the model. Roberta is based on transformers like BERT, but the hyperparameter tuning process is different while training and the “Next Sentence Prediction (NSP)” task in BERT has been removed (Liu Y. , 2019).

After calculating the similarities between the mentioned text representation models and the comments, the similarity scores between the users are calculated. The similarity score between the users is determined by taking the average of the similarity scores of the comments made by the users for the same products. This method produced the final similarity scores between the users.

3.3.2 Review and rating-based recommender systems

The first step of developing the review and rating-based recommender system is the calculation of adjusted ratings for each user – item pair (see Equation 1). Then the similarity scores between users calculated using reviews of users. For the target user, users with similarity scores above 0.50 were selected as similar users. For the target user, all products purchased by similar users in the past were added to the product recommendation pool. To predict the rating of target user to a specific item, the similarity score and adjusted rating of each user who purchased that item in the past are multiplied. This multiplication is done for all similar users and then the results of the multiplications are added together. The total value is divided by the sum of the users' similarity scores (see Equation 2). The weighted_rating represents the predicted adjusted rating of the target user to a specific item.

$$weighted\_rating = \frac{\sum(Similarity\ Score\ x\ Normalized\_Rating)}{\sum(Similarity\ Score)} \quad (2)$$

If the weighted_rating value for each product in the product pool is greater than 0, it is recommended to a target user and the recommendation_flag variable is assigned a value of 1 for that user. If the weighted_rating is less than 0, it is not recommended and the recommendation_flag variable is assigned a value of 0.

3.3.3 Review and sentiment-based recommender systems

In the second type of recommendation system, the similarity calculation between users based on the review data is the same with the previous algorithm. However, in this version of the algorithm, instead of users' adjusted ratings, we used the sentiment scores of reviews, which is calculated using BERT algorithm. We call this

variable as bert_sentiment. The bert_sentiment variable for each review can take the value of -1 or +1. The value of -1 indicates that the relevant review has a negative meaning; the value of +1 indicates that the review has a positive meaning.

The rest of the calculation is the same with the previous algorithm. Again, for the target user, users with similarity scores above 0.50 were selected as similar users. For the target user, all products purchased by similar users in the past were added to the product recommendation pool. To predict the rating of target user to a specific item, the similarity score and bert_sentiment score of each user who purchased that item in the past are multiplied. This multiplication is done for all similar users and then the results of the multiplications are added together. The total value is divided by the sum of the users' similarity scores (see Equation 3). The weighted_sentiment represents the predicted adjusted rating of the target user to a specific item.

$$weighted\_sentiment = \frac{\sum(Similarity\ Score\ x\ bert\_sentiment)}{\sum(Similarity\ Score)} \quad (3)$$

If the weighted_sentiment value for each product in the product pool is greater than 0, it is recommended to a target user and the recommendation_flag variable is assigned a value of 1 for that user. If the weighted_sentiment is less than 0, it is not recommended and the recommendation_flag variable is assigned a value of 0.

3.4 Evaluation metrics

Recommender systems produce recommendations for users, but how much these recommendations are liked by users is the most important criterion for the performance of recommender systems. The most important purpose of the evaluation

metrics used to measure the success rates of recommendations is to determine the users' interest in the recommendations in the best way.

In order to measure the performance of all recommender systems, confusion matrix was created. Accuracy, precision, recall and F1 score were used as evaluation metrics.

To create the confusion matrix, we used two flags: recommender_flag and like_flag. If ratings are used for base recommender systems, products with a rating of 4 or greater in the test dataset, this item is assumed as “liked” and the like_flag value is assigned as 1. If adjusted_rating was used for base recommender systems, products with a adjusted_rating value greater than 0 in the test dataset, this item is assumed as “liked” and the like_flag value is assigned as 1.

For review and rating based recommender system, since we used adjusted_ratings, if adjusted_rating is greater than 0, this item is assumed as “liked” and the like_flag value is assigned as 1. For the review and sentiment-based recommender algorithm, if the bert_rating variable is 1 in the test dataset, it is assumed that the user liked the product and the like_flag variable took the value of 1.

The explanation of confusion matrix (Table 4) is as follows:

- If prediction is 1, this means that this item is recommended to the target user.
- If prediction is 0, this means that this item is not recommended to the target user.
- If actual is 1, this means that this item is liked by the target user.
- If actual is 0, this means that this item is not liked by the target user.

True positives show the number of items that are recommended by the algorithm and liked by users. False positives show the number of items that are recommended by the algorithm but not liked by the user. True negatives show the number of items

that are not recommended by the algorithm and not liked by users. False negatives show the number of items that are not recommended by the algorithm but liked by users.

Table 4. Confusion Matrix

	Prediction		
Actual		1	0
	1	TP (True Positive)	FN (False Negative)
	0	FP (False Positive)	TN (True Negative)

Accuracy indicates the proportion of correct recommendations among all recommendations as follows in Equation 4.

$$Accuracy = \frac{True\ Positive\ (TP) + True\ Negative\ (TN)}{All\ Predictions\ (TP+FN+FP+TN)} \quad (4)$$

Precision is the ratio of true positive values to all positive predictions. It measures the accuracy of positive predictions in the recommender system as follows in Equation 5.

$$Precision = \frac{True\ Positive\ (TP)}{True\ Positive\ (TP) + False\ Postive(FP)} \quad (5)$$

Recall is the total number of correctly predicted positive values divided by the true positive values. This metric is also known as sensitivity as follows in Equation 6.

$$Recall = \frac{True\ Positive\ (TP)}{True\ Positive\ (TP) + False\ Negative(FN)} \quad (6)$$

The F1 score is the harmonic mean of precision and sensitivity. It takes both metrics into account and is useful when there is an imbalance between the number of positive and negative values as follows in Equation 7.

$$F1\ Score = 2\ x\ \frac{Precision\ x\ Recall}{Precision+Recall} \quad (7)$$

3.5 Explainability methods

One of the most current issues in recommendation systems is explaining the reasons for recommendations. Since the machine learning algorithms, especially the deep learning ones are very difficult to understand by the ordinary users, they are called as black-boxes. This situation affects the transparency and reliability of the recommendation system, making it difficult to use (Sinha & Swearingen, 2002).

Explainability is the presentation of the reasons or logic behind the recommendations produced by a recommender system in an understandable way to users or developers according to the demand. Explainability is one of the current trend and development areas of both artificial intelligence models and recommender systems. However, since recommendation systems are created with complex and advanced models, they usually cannot offer strong explainability.

In this study, the first step of explainability is to be able to explain the words that cause the comment made by the user to the recommended product to be characterized as positive or negative by sentiment analysis. The SHAP method is used to explain the words that affect whether the review is positive or negative. The pre-trained BERT model, which was previously used to assign the sentiment score for all reviews, is integrated into SHAP. In this way, whether the review is positive or negative and which words and phrases cause this result is visually explained with

SHAPley values (Diwali et al., 2023). Afterwards, the similar user that caused the recommendation is determined. The similarity score between the similar user and the main user is presented to the developer.

For example, when product X is recommended to User A, User A's review of Product X, sentiment analysis of this review, and words that affect sentiment analysis are explained. The SHAP method is used together with the pre-trained BERT model used to assign the sentiment score to explain the influencing words. Then, the similarity score between User B, who is similar to User A and is the reason for recommending product X, and users A-B is presented to developer.

Then, the same products that the main user and similar user have preferred in the past are determined. The reviews they have made for the same product are shown. The similarity score of the reviews made by the main user and similar user for the same products is presented. The average of the similarity scores of the reviews on the same products shows the similarity score between the users.

The attention mechanism of BERT-based models is examined for comments that cause similarity. The Transformers-Intepret method calculates the attention score of each word in the reviews (Pierse, 2021). The 10 words with the highest attention scores are identified for each review. These words have the greatest influence on the attention mechanism of the model and, therefore, have a significant impact on the similarity between reviews.

For example, User A and User B have preferred and commented on products Y and Z in the past. The average of the similarity score between the comments they made to Y and the comments they made to Z constitutes the similarity score between users A and B. There is a similarity score to understand how similar the comments made by these users to product Y are. However, to understand why they are similar,

the 10 words with the highest attention score are obtained with the transformers-interpret method (Pierse, 2021).

As a result, each recommendation and its reasons are examined for the best recommender system. This examination goes backwards from the recommendation generation step.

CHAPTER 4

RESULTS

In this section, we compared the performances of recommendation algorithms with baseline algorithms using different evaluation metrics. After the confusion matrix was created, accuracy, F1 score, recall and precision were calculated. The developed review and rating-based recommender system and review and sentiment-based recommender system were compared with the base recommender systems in terms of these metrics.

To provide a clearer explanation of the results, this section has been divided into four main headings: results of the base recommender systems, results of the review and rating-based recommender system, results of the review and sentiment-based recommender system, and results of explainability.

4.1 Results of base recommender systems

The performance of base recommendation systems that include user-based collaborative filtering methods are discussed in detail in this section. SVD, KNN and T-SVD algorithms were used. Since the base recommender systems were used first with the rating data and then with the adjusted_rating variable, each algorithm produced two different results on each dataset.

The success of each model was tested using Amazon, Yelp and IMDB datasets. The success of the created recommender systems in predicting user preferences in the test datasets is compared below.

The findings obtained as a result of running the base recommender systems on the Amazon dataset are summarized as follows in Table 5.

Table 5. Amazon Dataset Results of Base Recommender Systems

		Amazon			
		Accuracy	F1 Score	Precision	Recall
Adjusted rating	SVD	0.389	0.430	0.443	0.418
	KNN	0.560	0.619	0.594	0.645
	T-SVD	0.569	0.634	0.604	0.667
Rating	SVD	**0.767**	**0.855**	**0.851**	0.858
	KNN	0.746	0.846	0.842	0.849
	T-SVD	0.746	0.847	0.834	**0.861**

For base recommender systems constructed with adjusted rating on the Amazon dataset, the T-SVD model achieved the best results. The accuracy of the T-SVD model was 0.569, F1 score was 0.634, precision was 0.604 and recall is 0.667. Among the recommendation systems created with Adjusted Rating, the T-SVD model was followed by the KNN model in terms of performance.

For base recommender systems created with rating data on the Amazon dataset, the base recommender system utilizing the SVD model achieved the highest performance across three out of four metrics. The accuracy of the SVD model was 0.767, F1 score was 0.855, precision was 0.851 and recall was 0.858. However, the performance differences between the SVD model and other models using rating data were minimal across all evaluation metrics.

When comparing the two approaches, models using rating gave better results than the versions created using adjusted rating for all three models, so it is clear that models using rating in Amazon dataset for base recommender systems creates more efficient results. The best among all models by a small margin is the recommendation system that uses rating data and uses the SVD model. KNN and T-SVD models using rating data also showed competitive performance.

The performance of the base recommender system on the Yelp dataset was examined as follows in Table 6.

Table 6. Yelp Dataset Results of Base Recommender Systems

		Yelp			
		Accuracy	F1 Score	Precision	Recall
Adjusted rating	SVD	0.529	0.567	0.589	0.547
	KNN	0.554	0.598	0.600	0.596
	T-SVD	0.558	0.601	0.605	0.597
Rating	SVD	0.574	0.530	**0.845**	0.386
	KNN	**0.599**	**0.687**	0.664	**0.711**
	T-SVD	0.593	0.665	0.677	0.653

For base recommender systems constructed with adjusted rating on the Yelp dataset, the T-SVD model achieved the best results. The accuracy of the T-SVD model was 0.558, the F1 score was 0.601, the precision was 0.605, and the recall was 0.597. The performance difference between T-SVD and KNN models was minimal for all evaluation metrics.

When examining the results of recommender systems created using rating data on the Yelp dataset, the base recommender system utilizing the KNN model achieved the highest performance across three out of four metrics. The accuracy of the KNN model was 0.599, F1 score was 0.687, precision was 0.664 and recall was 0.711.

Although the base recommender system with rating variable and SVD model received the highest value in precision metric, it showed very low performance in recall metric. The recommender system with KNN model created using rating achieved the best result in all three metrics except precision among all of 6 models. T-SVD and KNN models using rating gave better results than the versions created using adjusted rating.

In the tests conducted on the IMDB dataset, the performance of the base recommender systems created using the adjusted rating and rating data was examined in detail as shown in Table 7.

Table 7. IMDB Dataset Results of Base Recommender Systems.

		IMDB			
		Accuracy	F1 Score	Precision	Recall
Adjusted rating	SVD	0.600	0.632	0.644	0.621
	KNN	0.576	0.635	0.607	0.667
	T-SVD	0.548	0.564	0.605	0.529
Rating	SVD	0.658	0.705	**0.903**	0.579
	KNN	0.637	0.739	0.751	0.727
	T-SVD	**0.646**	**0.754**	0.742	**0.765**

For base recommender systems constructed with adjusted rating in the IMDB dataset, The SVD and KNN models achieved better results than T-SVD. When comparing these models, the SVD model achieved the highest results in terms of accuracy and precision metrics, while the KNN models achieved the best results in terms of F1 score and recall metrics.

When the results of recommender systems using rating data in the IMDB dataset are examined, the KNN and T-SVD models achieved better results in all metrics compared to their versions using adjusted rating data. The base recommender system created using the rating data with the SVD model is also much more successful than the version using the adjusted rating, except for recall.

The base recommender system using the rating variable and the SVD model achieved the highest value in precision and accuracy metrics and the base recommender system using the rating variable and T-SVD model achieved the best results in F1 score and recall evaluation metrics among of 6 models. According to the evaluation metrics for the IMDB dataset, similar to the Amazon and Yelp datasets, the recommender system versions using adjusted ratings showed more limited and lower success.

4.2 Results of review and rating-based recommender systems

At this point, the performance of recommender systems created for each algorithm used in calculating similarity between users was compared. Recommender systems created using different text-based models were tested on Amazon, Yelp and IMDB datasets and the results were compared according to Accuracy, F1 score, precision and recall evaluation metrics.

Review and rating-based recommender systems were applied to the Amazon dataset and the results obtained were examined as follows in Table 8.

Table 8. Amazon Dataset Results of Review and Rating-Based Recommender Systems

	Amazon			
	Accuracy	F1 Score	Precision	Recall
BoW	0.517	0.618	**0.667**	0.576
Word2Vec	**0.554**	**0.641**	0.632	**0.650**
ELMo	0.553	0.639	0.630	0.648
GloVe	**0.554**	**0.641**	0.632	**0.650**
BERT	**0.554**	**0.641**	0.632	**0.650**
DistilBERT	**0.554**	**0.641**	0.632	**0.650**
SBERT	0.525	0.614	0.626	0.603
ROBERTA	0.553	0.639	0.629	**0.650**
BERT Large	0.553	0.639	0.629	**0.650**
SBERT Large	0.552	**0.641**	0.650	0.633

Review and rating-based recommender systems created using the BoW model have the highest precision among the recommender systems applied to the Amazon dataset, with a precision of 0.667. For other metrics, the Word2Vec, GloVe, BERT, and DistilBERT models achieved the highest values.

Review and rating-based recommender systems were applied to the Yelp dataset as follows in Table 9.

Table 9. Yelp Dataset Results of Review and Rating-Based Recommender Systems

	Yelp			
	Accuracy	F1 Score	Precision	Recall
BoW	**0.581**	**0.647**	**0.628**	**0.667**
Word2Vec	0.580	0.629	0.626	0.633
ELMo	0.580	0.630	0.625	0.634
GloVe	0.579	0.628	0.625	0.631
BERT	**0.581**	0.630	0.626	0.633
DistilBERT	0.580	0.629	0.625	0.632
SBERT	0.567	0.624	0.619	0.630
ROBERTA	0.579	0.628	0.625	0.631
BERT Large	0.580	0.629	0.625	0.632
SBERT Large	0.570	0.623	0.620	0.626

The recommender system created using the BoW model ranked first on the Yelp dataset for all metrics. The accuracy of BoW model was 0.581, the F1 score was 0.647, the precision was 0.628 and the recall was 0.667.

Review and rating-based recommender systems were applied to the IMDB dataset and performance metrics were calculated as shown in Table 10.

Table 10. IMDB Dataset Results of Review and Rating-Based Recommender Systems

	IMDB			
	Accuracy	F1 Score	Precision	Recall
BoW	0.633	0.676	**0.665**	0.687
Word2Vec	0.632	0.677	0.660	0.695
ELMo	0.633	0.678	0.661	0.696
GloVe	0.631	0.677	0.660	0.695
BERT	0.632	0.678	0.660	0.696
DistilBERT	0.632	0.678	0.660	0.696
SBERT	0.620	0.668	0.660	0.676
ROBERTA	0.632	0.677	0.660	0.695
BERT Large	0.632	0.678	0.661	0.697
SBERT Large	**0.637**	**0.683**	0.666	**0.701**

The recommender system using the SBERT Large model achieved the best results for all evaluation metrics, with the accuracy rate of 0.637, the F1 score of 0.683, the precision of 0.666, and the recall of 0.701 on the IMDB dataset.

4.3 Results of review and sentiment-based recommender systems

In this section, the results of review and sentiment-based recommendation systems are examined. Similarities between users and reviews made by users were calculated with ten different text-based models. Then, sentiment scores were calculated based on sentiment analysis of the texts and were determined. Sentiment scores were used instead of rating scores and recommendations were produced.

The results of the review and sentiment-based recommendation systems on the Amazon dataset are shown in Table 11.

Table 11. Amazon Dataset Results of Review and Sentiment-Based Recommender Systems

	Amazon			
	Accuracy	F1 Score	Precision	Recall
BoW	0.621	0.748	0.766	0.731
Word2Vec	0.697	0.806	0.811	0.801
ELMo	0.696	0.805	0.811	0.800
GloVe	0.697	0.806	0.811	0.801
BERT	0.697	0.806	0.811	0.801
DistilBERT	0.697	0.806	0.811	0.801
SBERT	**0.724**	**0.824**	0.824	**0.824**
ROBERTA	0.695	0.805	0.809	0.802
BERT Large	0.698	0.807	0.812	0.802
SBERT Large	0.720	**0.824**	**0.826**	0.822

For review and sentiment-based recommender systems on the Amazon dataset, the recommender system established with the SBERT model was 0.724 accuracy, F1 score 0.824, precision 0.824, and recall 0.824, making it the best performer among all models. SBERT Large model closely followed SBERT in performance.

The results of the review and sentiment-based recommendation systems on the Yelp dataset are shown in Table 12.

Table 12. Yelp Dataset Results of Review and Sentiment-Based Recommender Systems

	Yelp			
	Accuracy	F1 Score	Precision	Recall
BoW	0.656	0.765	0.726	0.808
Word2Vec	0.641	0.749	0.704	0.799
ELMo	0.641	0.749	0.704	0.799
GloVe	0.641	0.749	0.705	0.799
BERT	0.641	0.749	0.704	0.799
DistilBERT	0.641	0.749	0.704	0.799
SBERT	0.650	0.760	0.724	0.799
ROBERTA	0.641	0.749	0.704	0.799
BERT Large	0.642	0.749	0.705	0.799
SBERT Large	0.648	0.756	0.710	0.808

For review and sentiment-based recommender systems on the Yelp dataset, the recommender system established with the BoW model achieved an accuracy of 0.656, an F1 score of 0.765, a precision of 0.726, and a recall of 0.808, making it the best performer among all models. The BoW model was followed by the SBERT and SBERT Large models.

The results of the review and sentiment-based recommendation systems on the IMDB dataset are shown in Table 13.

Table 13. IMDB Dataset Results of Review and Sentiment-Based Recommender Systems

	IMDB			
	Accuracy	F1 Score	Precision	Recall
BoW	0.659	0.769	0.716	0.832
Word2Vec	0.666	0.778	0.716	0.852
ELMo	0.666	0.778	0.716	0.852
GloVe	0.666	0.778	0.716	0.852
BERT	0.667	0.779	0.716	0.853
DistilBERT	0.666	0.778	0.716	0.853
SBERT	0.660	0.772	**0.723**	0.828
ROBERTA	0.666	0.779	0.716	0.854
BERT Large	0.666	0.778	0.716	0.853
SBERT Large	**0.670**	**0.781**	0.720	**0.854**

On the IMDB dataset, the SBERT Large model achieved the highest accuracy, F1 score and recall with an accuracy rate of 0.670, an F1 score of 0.781, a precision of 0.720 and a recall of 0.854. Other BERT-based models have been the most successful models after the SBERT large model.

When the results of the applications of the review and sentiment-based recommender systems to all 3 datasets are compared, it is understood that the SBERT and SBERT Large models give good results as shown in the comprehensive table in Appendix A, B, C.

4.4 Results of explainability of recommender systems

In this thesis, an explainability framework was developed to explain the reasons for the recommendations of review and sentiment-based recommender systems. The explainability approach explains the reasons for each produced recommendation in detail with the following steps.

- Firstly, information about the generated recommendation namely user_id, asin, rating, sentiment_score, predicted_rating, recommendation_flag, like_flag and text are displayed on the screen as follows in Table 14.

Table 14. Recommendation Information

user_id	AE53545QTFI7RPWHC2BSGOAD3NYA
asin	B0190K63GA
rating	4
sentiment_score	1
predicted_rating	1.0
recommendation_flag	1
like_flag	1
text	i have been using this body wash for about a week and i have to say my skin is looking fabulous. as far as a body wash goes this one has a very light scent. while it definitely cleans the skin it is a low lather soap. you do not get a lot of bubbles. there are three different types of oils in this body wash coconut, argan, and tea tree. they work well in terms of moisturization. one of my favorite features is the pump that is included to dispense the soap. it is really well made. it doesn't allow for any drips and dispenses accurately. i received a discounted sample of this product in exchange for an unbiased and honest review.

  - user_id: user identification key
  - asin: product, service, or movie identification key
  - rating: score given by the user
  - sentiment_score: sentiment of user review (-1: negative, +1: positive)
  - predicted_rating: predicted sentiment score
  - recommendation_flag: 1: recommended, 0: not recommended
  - like_flag: 1: liked, 0:not liked

- Secondly, the review for the recommended product made by the user who is given the product recommendation is examined. Whether the user's review is positive or negative is obtained by sentiment analysis. Sentiment analysis is explained and visualized with the SHAP method. Words that positively or negatively affect the meaning of the comment are determined as illustrated in Figure 2. Words marked in blue (my skin, a looking fabulous, favorite, really well, honest review) affected the emotion of the sentence positively, while words marked in red (you, do, not) affected it negatively.

Figure 2. Sentiment analysis of the recommended product's review

- The users similar to the main user who caused the product to be recommended are detected. The reviews of similar users on the recommended product are displayed. Two similar users led to this recommendation as follows in Table 15.

Table 15. The Reviews of Similar Users on the Recommended Product

causing_user	recommended_product	text
AE644JVRCJETRVH5MHJ55NHVPOFQ	B0190K63GA	i love it leave my skin soft i receive it fre...
AGTIC36N7QHDFOXP562DICQFYUOQ	B0190K63GA	this is an excellent natural body wash that is...

- Sentiment analysis is performed on the reviews of similar users and explained with the SHAP method. The Shapley values of the words that most affect whether the reviews of similar users are positive or negative are displayed on the screen. The first similar user's review about the recommended product is visualized with the SHAP method as shown in Figure 3. Words marked in red (leave) have affected the meaning of the review negatively, and words marked in blue (soft, skin, love, it, like etc.) have affected it positively.

Figure 3. Sentiment analysis of the first similar user's review

- The second similar user's review about the recommended product is visualized with the SHAP method as shown in Figure 4. Words marked in red (to kill, not sure, etc.) have affected the meaning of the review negatively, and words marked in blue (powerful, absolutely worth, but, the price, etc.) have affected it positively.

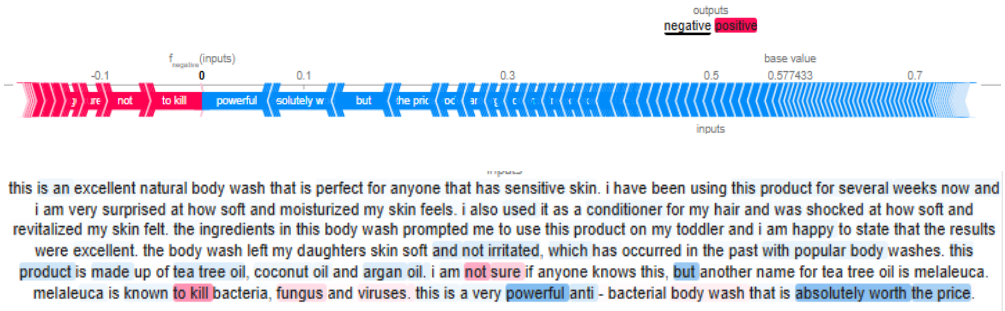

Figure 4. Sentiment analysis of the second similar user's review

- The similarity score between the main user and the similar users are shown as follows in Table 16. The reviews they have made for the same product, which are used to calculate this similarity score, are displayed as follows in Table 17.

Table 16. The Similarity Score Between the Main User and the Similar/Causing Users

main_user	causing_user	similarity score
AE53545QTFI7RPWHC2BSGOAD3NYA	AE644JVRCJETRVH5MHJ55NHVPOFQ	0.782930
AE53545QTFI7RPWHC2BSGOAD3NYA	AGTIC36N7QHDFOXP562DICQFYUOQ	0.823025

Table 17. Reviews Made Previously on the Same Product

main_user	causing_user	asin	main_user's review	causing_user's review
AE53545QTFI7RPWHC2BSGOAD3NYA	AE644JVRCJETRVH5MHJ55NHVPOFQ	B01AYTGWA8	this nail polish remover is a fascinating pr...	it are the vest nail polish i have before it r...
AE53545QTFI7RPWHC2BSGOAD3NYA	AGTIC36N7QHDFOXP562DICQFYUOQ	B01AYTGWA8	this nail polish remover is a fascinating pr...	this was a great buy as i have been on the loo...

- The attention mechanism analyzes the main and similar users' reviews made on the same product in the past. It calculates the attention score of words in reviews. The 10 words with the highest attention scores for the main user's review are shown in Figure 5, the first similar user's review is shown in Figure 6, the second similar user's review are shown in Figure 7.

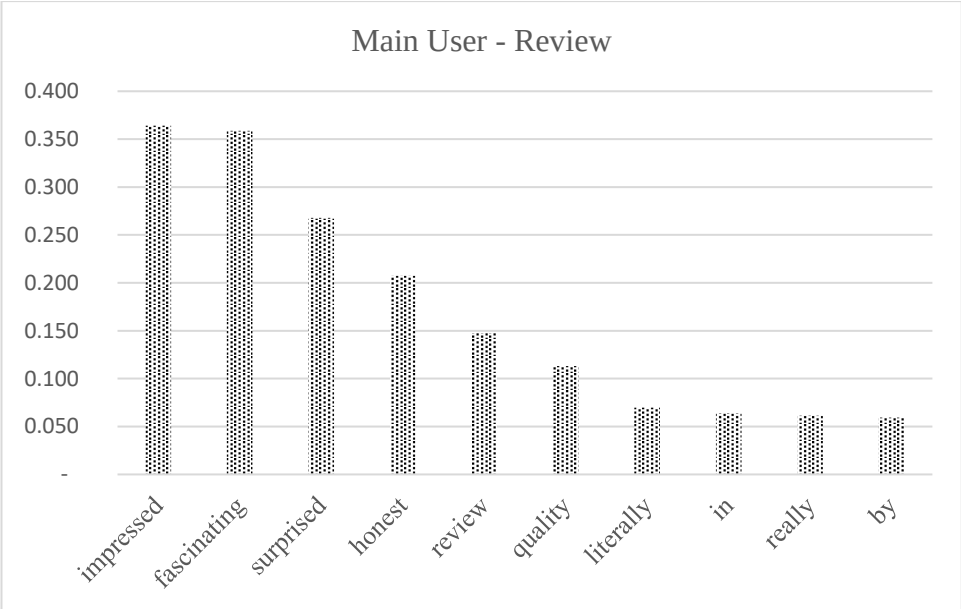

Figure 5. Top 10 words having high attention score for main user's review

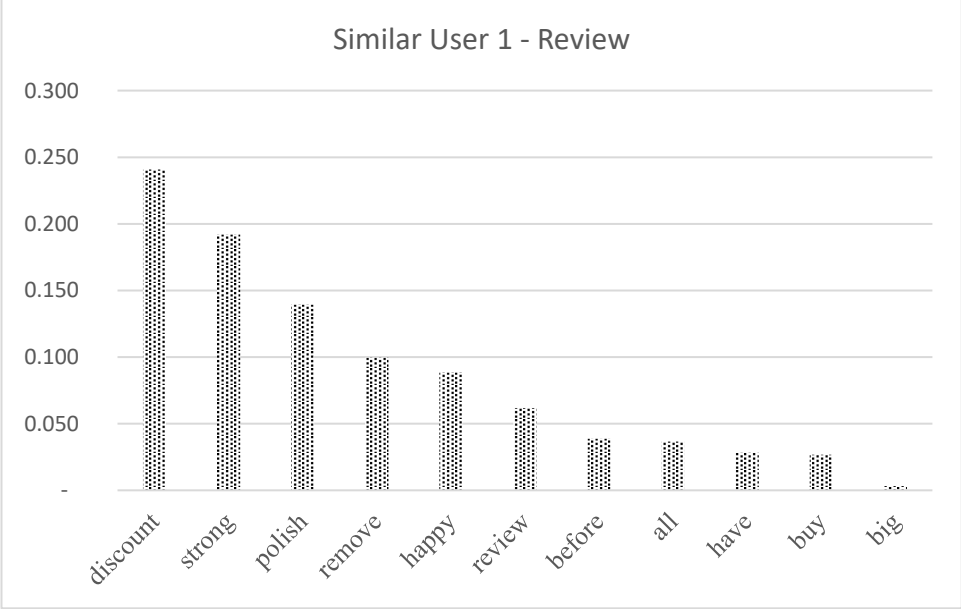

Figure 6. Top 10 words having high attention score for first similar user's review

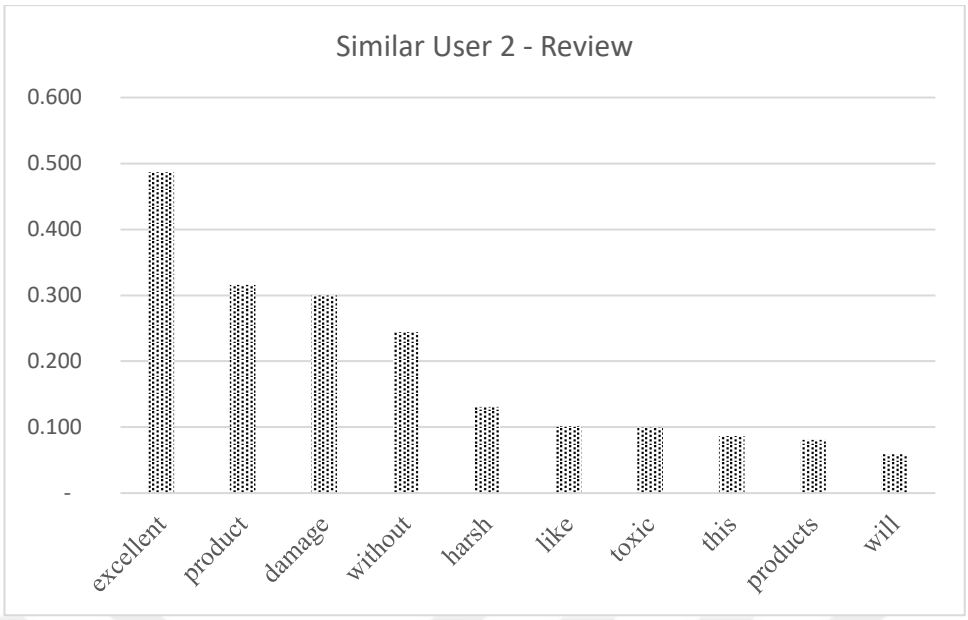

Figure 7. Top 10 words having high attention score for second similar user's review

CHAPTER 5

DISCUSSION

This study develops an explainable recommendation system that relies solely on user-generated reviews and is competitive with rating-based recommendation systems. Furthermore, the effects of different language models on performance, the contributions of the Attention mechanism and SHAP methods to explainability are examined in the light of research questions.

RQ 1: How do recommender systems that solely rely on user-generated textual data perform in comparison to rating-based recommender systems?

The review and sentiment-based recommender system developed in this study outperformed traditional rating-based recommendation systems on the IMDB and Yelp datasets, while also achieving comparable results on the Amazon dataset. This is consistent with prior studies on the importance of richness of textual data in capturing user preferences (Zhang, et al., 2021). The performance of the models varied due to the differing characteristics of user reviews in each dataset. Factors such as the length, grammar structure, semantic depth, and emotional richness of user comments (Li, Jin, & Goh, 2020) differed across platforms, influencing the performance of the models.

RQ 2: What is the impact of employing different language models on the performance of similarity calculations in a sentiment-based recommender system?

The language models employed in the study produced varying results in detecting text similarities, which, in turn, influenced the performance of the recommendation systems. The SBERT model generally performed better than other language models, owing to its ability to capture semantic meaning at the sentence

level (Reimers, 2019). SBERT delivered the best results in datasets with well-structured reviews, such as IMDB. However, in datasets where reviews contained shorter and less carefully constructed sentences, such as the Yelp, Bag of Words, a word-level language model (Gabrilovich & Markovitch, 2006), performed best, with SBERT ranking second.

RQ 3: How can attention mechanisms and SHAP-based explanations improve the transparency of recommender systems by providing interpretable insights into recommendation decisions?

The SHAP approach revealed the influence of each word on the sentiment score, showing whether words positively or negatively impacted user satisfaction (Lundberg, 2017). This made the sentiment score, which reflects user satisfaction, more transparent. Additionally, the attention mechanism, analyzed using the Transformer-Interpret approach, identified the words that most influenced the similarity calculations between users (Pierse, 2021). This allowed system developers to determine the reasons behind user similarities. The integration of these two methods provided step-by-step explanations for each recommendation, thereby enhancing confidence in the system.

In summary, this study demonstrates the potential of review and sentiment-based recommender systems. The integration of explainability approaches has increased the transparency and the trust (Zhang & Chen, 2020) in the system.

CHAPTER 6

CONCLUSION

This study aims to develop a novel explainable recommender system that uses only text data that can compete with the base recommender systems created with rating data.

The created recommendation systems were tested on Amazon, Yelp and IMDB datasets. The results of recommender systems were compared based on accuracy, F1 score, recall and precision metrics.

Base recommender systems developed using rating data outperformed those using adjusted rating across all three datasets. On the Amazon dataset, the base recommender system using the SVD algorithm with rating data yielded the best results. On the Yelp dataset, the base recommender system utilizing rating data with the KNN algorithm achieved the highest performance. In the tests conducted on the IMDB dataset, the recommender system developed with the T-SVD algorithm demonstrated the best performance in terms of F1 score and recall. Although the SVD model performed well in terms of accuracy and precision, its performance in the recall metric was relatively low.

Review and rating-based recommender systems generally achieved comparable results on the Amazon dataset, with no model showing significant differences. However, the results from the IMDB dataset indicate that language models, specifically SBERT Large, outperformed all other models across all evaluation criteria. In terms of performance, review and rating-based recommender systems did not achieve the success of review and sentiment-based or base recommender systems.

For the review and sentiment-based recommender systems, the SBERT model achieved the best results on the Amazon dataset in terms of Accuracy, F1 Score and Recall, and the second-best result in Precision. The recommender system developed with the SBERT Large model showed the highest performance in F1 Score and Precision, with the second-best performance in Accuracy and Recall. For the YELP dataset, the version using the BoW algorithm had the highest performance metrics, followed by the SBERT and SBERT Large models. Finally, in the IMDB dataset, the recommender system based on the SBERT Large model ranked first across three evaluation metrics, with the SBERT model coming in second.

As shown in the comprehensive table in Appendix C, the review and sentiment-based SBERT models outperform all other models on the IMDB dataset. As shown in Appendix B, in the Yelp dataset, SBERT-based review and sentiment-based recommender systems performed as the best model after the BoW-based model. On the Amazon dataset, while the review and sentiment-based SBERT models did not outperform the base recommendation systems, they demonstrated competitive performance, as shown in Appendix A. More effective results were obtained in recommendation systems by classifying the sentiment in user reviews as positive or negative.

Another focus of this study is the explainability of the recommendations. An explainability approach was developed using the SHAP and Transformers-Interpret methods, allowing the reasons for each recommendation to be explained.

As a result, this study has shown that review and sentiment-based recommender systems, which utilize only text data, can compete with base recommender systems using rating data and produce explainable results.

5.1 Limitations

Although this thesis provides comprehensive research, several limitations remain that need to be addressed.

The first limitation is the use of data filtering and elimination steps to reduce processing time, due to limited computational power. As a result, the systems were tested on smaller datasets. Additionally, the developed recommendation systems were not tested on more else datasets beyond the 3 datasets used in this study.

Another limitation is that only user reviews were utilized, which led to a cold-start problem in the recommender systems. Moreover, since product data was not included, only user-based collaborative filtering was implemented, while hybrid or content-based approaches were not explored.

Furthermore, the sentiment analysis in this study classified user comments as either positive or negative, without considering more detailed emotions such as anger, sadness, or happiness.

These limitations highlight areas for further development and should be considered in future research.

5.2 Future works

The recommendation systems developed in this study have yielded successful results in terms of prediction accuracy and explainability. However, new scientific contributions can be made in the future by addressing the mentioned limitations and areas of development.

Firstly, if computation constraints are overcome with more advanced hardware, the developed recommender systems can be tested on larger datasets without filtering. This would improve both data quality and density, leading to more

reliable results. Additionally, the recommender systems developed in this study could be tested on new datasets, making the results more generalizable.

This study developed a recommender system using user reviews, which faces the cold-start problem arising from the lack of data for new users. To overcome this issue, future work could focus on incorporating text-based product data, such as product descriptions, and developing a hybrid or content-based recommender system.

LLMs are more effective in detecting semantic relationships in text data compared to language models. In the future, by using LLMs in review and sentiment-based recommender systems, both the recommendation quality and the overall performance of the system can be improved.

Finally, in addition to sentiment-based approaches, emotion detection methods could be employed to identify more detailed emotions and integrate them into recommender systems. This would improve the sensitivity of the systems by addressing a wider range of emotions, such as anger and sadness, rather than just positive and negative. Future research could explore these aspects to further diversify the literature.

5.3 Business implications

The explainable review and sentiment-based recommendation systems created in this study can offer more personalized recommendations by analyzing the sentiment in user comments and the similarity between those comments. Additionally, the explainability of the designed recommendation system allows businesses to explore the logic behind the recommendations and gain a deeper understanding of customer preferences. These explanations of recommendations can help businesses identify

industry trends, address customer issues, and meet their needs effectively (Konstan, Riedl, & Schafer, 2001). Businesses can improve customer experience, enhance customer satisfaction, and customer loyalty (Sharma, Shaikh, & Li, 2021) through these type of recommendation systems. Consequently, improved customer satisfaction is expected to increase sales and profitability (Panniello, Hill, & Gorgoglione, 2016).

APPENDIX A

AMAZON DATASET RESULTS OF ALL RECOMENNDER SYSTEMS

		Amazon			
		Accuracy	F1 Score	Precision	Recall
Base:Adjusted rating	SVD	0.389	0.430	0.443	0.418
	KNN	0.560	0.619	0.594	0.645
	T-SVD	0.569	0.634	0.604	0.667
Base: Rating	SVD	0.767	0.855	**0.851**	0.858
	KNN	**0.746**	0.846	0.842	0.849
	T-SVD	**0.746**	**0.847**	0.834	**0.861**
Review and Rating-Based	BoW	0.517	0.618	0.667	0.576
	Word2Vec	0.554	0.641	0.632	0.650
	ELMo	0.553	0.639	0.630	0.648
	GloVe	0.554	0.641	0.632	0.650
	BERT	0.554	0.641	0.632	0.650
	DistilBERT	0.554	0.641	0.632	0.650
	SBERT	0.525	0.614	0.626	0.603
	ROBERTA	0.553	0.639	0.629	0.650
	BERT Large	0.553	0.639	0.629	0.650
	SBERT Large	0.552	0.641	0.650	0.633
Review and Sentiment-Based	BoW	0.621	0.748	0.766	0.731
	Word2Vec	0.697	0.806	0.811	0.801
	ELMo	0.696	0.805	0.811	0.800
	GloVe	0.697	0.806	0.811	0.801
	BERT	0.697	0.806	0.811	0.801
	DistilBERT	0.697	0.806	0.811	0.801
	SBERT	0.724	0.824	0.824	0.824
	ROBERTA	0.695	0.805	0.809	0.802
	BERT Large	0.698	0.807	0.812	0.802
	SBERT Large	0.720	0.824	0.826	0.822

APPENDIX B

YELP DATASET RESULTS OF ALL RECOMENNDER SYSTEMS

		Yelp			
		Accuracy	F1 Score	Precision	Recall
Base: Adjusted rating	SVD	0.529	0.567	0.589	0.547
	KNN	0.554	0.598	0.600	0.596
	T-SVD	0.558	0.601	0.605	0.597
Base: Rating	SVD	0.574	0.530	**0.845**	0.386
	KNN	0.599	0.687	0.664	0.711
	T-SVD	0.593	0.665	0.677	0.653
Review and Rating-Based	BoW	0.581	0.647	0.628	0.667
	Word2Vec	0.580	0.629	0.626	0.633
	ELMo	0.580	0.630	0.625	0.634
	GloVe	0.579	0.628	0.625	0.631
	BERT	0.581	0.630	0.626	0.633
	DistilBERT	0.580	0.629	0.625	0.632
	SBERT	0.567	0.624	0.619	0.630
	ROBERTA	0.579	0.628	0.625	0.631
	BERT Large	0.580	0.629	0.625	0.632
	SBERT Large	0.570	0.623	0.620	0.626
Review and Sentiment-Based	BoW	**0.656**	**0.765**	0.726	**0.808**
	Word2Vec	0.641	0.749	0.704	0.799
	ELMo	0.641	0.749	0.704	0.799
	GloVe	0.641	0.749	0.705	0.799
	BERT	0.641	0.749	0.704	0.799
	DistilBERT	0.641	0.749	0.704	0.799
	SBERT	0.650	0.760	0.724	0.799
	ROBERTA	0.641	0.749	0.704	0.799
	BERT Large	0.642	0.749	0.705	0.799
	SBERT Large	0.648	0.756	0.710	**0.808**

APPENDIX C

IMDB DATASET RESULTS OF ALL RECOMENNDER SYSTEMS

	IMDB				
		Accuracy	F1 Score	Precision	Recall
Base: Adjusted rating	SVD	0.600	0.632	0.644	0.621
	KNN	0.576	0.635	0.607	0.667
	T-SVD	0.548	0.564	0.605	0.529
Base: Rating	SVD	0.658	0.705	0.903	0.579
	KNN	0.637	0.739	**0.751**	0.727
	T-SVD	0.646	0.754	0.742	0.765
Review and Rating-Based	BoW	0.633	0.676	0.665	0.687
	Word2Vec	0.632	0.677	0.660	0.695
	ELMo	0.633	0.678	0.661	0.696
	GloVe	0.631	0.677	0.660	0.695
	BERT	0.632	0.678	0.660	0.696
	DistilBERT	0.632	0.678	0.660	0.696
	SBERT	0.620	0.668	0.660	0.676
	ROBERTA	0.632	0.677	0.660	0.695
	BERT Large	0.632	0.678	0.661	0.697
	SBERT Large	0.637	0.683	0.666	0.701
Review and Sentiment-Based	BoW	0.659	0.769	0.716	0.832
	Word2Vec	0.666	0.778	0.716	0.852
	ELMo	0.666	0.778	0.716	0.852
	GloVe	0.666	0.778	0.716	0.852
	BERT	0.667	0.779	0.716	0.853
	DistilBERT	0.666	0.778	0.716	0.853
	SBERT	0.660	0.772	0.723	0.828
	ROBERTA	0.666	0.779	0.716	0.854
	BERT Large	0.666	0.778	0.716	0.853
	SBERT Large	**0.670**	**0.781**	0.720	**0.854**

REFERENCES

Abdollahi, B., & Nasraoui, O. (2018). Transparency in fair machine learning: the case of explainable recommender systems. *Human and Machine Learning: Visible, Explainable, Trustworthy and Transparent*, 21-35.

Adeniyi, D., Wei, Z., & Yongquan, Y. (2016). Automated web usage data mining and recommendation system using K-Nearest Neighbor (KNN) classification method. *Applied Computing and Informatics*, 90-108.

Adomavicius, G., & 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.

Adomavicius, G., & Zhang, J. (2012). Impact of data characteristics on recommender systems performance. *ACM Transactions on Management Information Systems (TMIS)*, 1-17.

Afsar, M., Crump, T., & Far, B. (2022). Reinforcement learning based recommender systems: A survey. *ACM Computing Surveys, 55(7)*, 1-38.

Ansari, A., Essegaier, S., & Kohli, R. (2000). Internet recommendation systems.

Burke, R. (2002). Hybrid recommender systems: Survey and experiments. *User Modeling and User-Adapted Interaction*, 331-370.

Devlin, J. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding.

Diwali et al. (2023). Sentiment analysis meets explainable artificial intelligence: A survey on explainable sentiment analysis. *IEEE Transactions on Affective Computing*.

Du, X., Bhushanam, B., Yu, J., Choudhary, D., Gao, T., Wong, S., & Kejariwal, A. (2021). Alternate model growth and pruning for efficient training of recommendation systems. *20th IEEE International Conference on Machine Learning and Applications (ICMLA)*, 1421-1428.

Gabrilovich & Markovitch. (2006). Overcoming the brittleness bottleneck using Wikipedia: Enhancing text categorization with encyclopedic knowledge. *AAAI*, 1301-1306.

Gao, Y., Sheng, T., Xiang, Y., Xiong, Y., & Wang, H. (2023). Chat-rec: Towards interactive and explainable llms-augmented recommender system.

Ghose, A., & Ipeirotis, P. (2010). Estimating the helpfulness and economic impact of product reviews: Mining text and reviewer characteristics. *IEEE Transactions on Knowledge and Data Engineering*, 1498-1512.

Goldberg, Nichols, Oki, & Terry. (1992). Using Collaborative Filtering to Weave an Information Tapestry.

Golub, G. H., & Reinsch, C. . (1971). Singular value decomposition and least squares solutions. *Handbook for Automatic Computation: Volume II: Linear Algebra* (s. 134-15). içinde Berli.

Guo, G., Wang, H., Bell, D., Bi, Y., & Greer, K. . (2003). KNN model-based approach in classification. *In On The Move to Meaningful Internet Systems 2003: CoopIS, DOA, and ODBASE: OTM Confederated International Conferences, CoopIS, DOA, and ODBASE*, 986-996.

Herlocker, J., Konstan, J., & Riedl, J. (2020). Explaining collaborative filtering recommendations. *In Proceedings of the 2000 ACM Conference on Computer Supported Cooperative Work*, 241-250.

Herlocker, J.L. et al. (2004). Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems (TOIS)*, 5-53.

Hou, Y., Li, J., He, Z., Yan, A., Chen, X., & McAuley, J. (2024). Bridging language and items for retrieval and recommendation.

Kanwal, S., Nawaz, S., Malik, M., & Nawaz, Z. (2021). A review of text-based recommendation systems. *IEEE Access*.

Khusro, S., Ali, Z., & Ullah, I. (2016). Recommender systems: Issues, challenges, and research opportunities. *Information Science and Applications (ICISA)*, 1179-1189.

Klema, V., Laub, A. . (1980). The singular value decomposition: Its computation and some applications. *IEEE Transactions on automatic control*, 164-176.

Konstan, J., Riedl, J., & Schafer, J. (2001). E-commerce recommendation applications. *Data Mining and Knowledge Discovery*, 115-153.

Koroteev, M. (2021). BERT: A review of applications in natural language processing and understanding.

Krishnan, V., Narayanashetty, P., Nathan, M., & Davies, R. (2008). Who predicts better? Results from an online study comparing humans and an online recommender system. *Proceedings of the 2008 ACM Conference on Recommender systems*, 211-218.

Li, L., Jin, D., & Goh, T. (2020). How textual quality of online reviews affect classification performance: A case of deep learning sentiment analysis. *Neural Computing and Applications*, 4387-4415.

Li, L., Zhang, Y., & Chen, L. (2021). Personalized transformer for explainable recommendation.

Liu, P., Zhang, L., & Gulla , J. (2020). Dynamic attention-based explainable recommendation with textual and visual fusion. *Information Processing & Management, 57(6)*.

Liu, Y. (2019). Roberta: A robustly optimized bert pretraining approach.

Lu, J., Wu, D., Mao, M., Wang, W., & Zhang, G. (2015). Recommender system application developments: A survey. *Decision Support Systems*, 12-32.

Lundberg, S. (2017). A unified approach to interpreting model predictions.

Marlin, B. M. (2003). Modeling user rating profiles for collaborative filtering. *Advances in neural information processing systems*, 16.

Mikolov, T., Sutskever, I., Chen, K., & Corrado, G. (2013). Distributed representations of words and phrases and their compositionality. *Advances in Neural Information Processing Systems*, 26.

Mukherjee, S. (2024, 1). *bert-base-uncased-finetuned-sst2-v2*. Retrieved from HuggingFace: https://huggingface.co/sadhaklal/bert-base-uncased-finetuned-sst2-v2

Naseem, U., Razzak, I., Khan, S. K., & Prasad, M. (2021). A comprehensive survey on word representation models: From classical to state-of-the-art word representation language models. *Transactions on Asian and Low-Resource Language Information Processing*.

Pal, A., Barigidad, A., & Mustafi, A. (2020). *IMDB Movie Reviews Dataset*. Retrieved from IMDB: https://ieee-dataport.org/open-access/imdb-movie-reviews-dataset

Panniello, U., Hill, S., & Gorgoglione, M. (2016). The impact of profit incentives on the relevance of online recommendations. *Electronic Commerce Research and Applications*, 87-104.

Pazzani, M. J. (2007). Content-based recommendation systems.

Penha, G., & Hauff, C. (2020). What does bert know about books, movies and music? probing bert for conversational recommendation. *Proceedings of the 14th ACM Conference on Recommender Systems*, 388-397.

Pennington, J., Socher, R., & Manning, C. (2014). Glove: Global vectors for word representation. *In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, (s. 1532-1543).

Peters, M., Neumann, M., Iyyer, M., Gardner, M., & Clark C. (2018). Deep contextualized word representations.

Pierse, C. (2021). *Transformers Interpret (v0.5.2)*. Retrieved from GitHub: https://github.com/cdpierse/transformers-interpret

Pujahari, A., & Sisodia, D. (2020). Pair-wise preference relation based probabilistic matrix factorization for collaborative filtering in recommender system. *Knowledge-Based Systems*, 196.

Rahutomo, F., Kitasuka, T., & Aritsugi, M. (2012). Semantic cosine similarity. *In The 7th international student conference on advanced science and technology ICAST*, 1.

Reimers, N. (2019). Sentence-BERT: Sentence embeddings using siamese BERT-Networks.

Sanh, V. (2019). DistilBERT, a distilled version of BERT: Smaller, faster, cheaper and lighter.

Sarwar, B., Karypis, G., Konstan, J., & Riedl, J. . (2000). Application of dimensionality reduction in recommender system-a case study.

Sarzynska-Wawer, J., Wawer, A., Pawlak, A., Szymanowska, J., Stefaniak, I., Jarkiewicz, M., & Okruszek, L. (2021). Detecting formal thought disorder by deep contextualized word representations. *Psychiatry Research*, 304.

Shani, G., & Gunawardana, A. (2011). Evaluating recommendation systems. *Recommender Systems Handbook*, 257-297.

Sharma, R., Shaikh, A., & Li, E. (2021). Designing Recommendation or Suggestion Systems: Looking to the future. *Electronic Markets*, 243-252.

Sinha, R., & Swearingen, K. (2002). The role of transparency in recommender systems. *CHI'02 Extended Abstracts on Human Factors in Computing Systems*, 830-831.

Sutskever, I., Mikolov, T., & Le, Q. (2013). Exploiting similarities among languages for machine translation.

Vig, J. (2019). BertViz: A tool for visualizing multihead self-attention in the BERT model. *ICLR Workshop: Debugging Machine Learning Models (Vol. 3)*.

Wang, J., & Dong, Y. (2020). Measurement of text similarity: A survey. *Information, 11(9), 421*.

Wang, Y., Ma, W., Zhang, M., Liu, Y., & Ma, S. (2023). A survey on the fairness of recommender systems. *ACM Transactions on Information Systems 41(3)*, 1-43.

Yelp Dataset. (2022). Retrieved from Yelp: https://www.yelp.com/dataset/

Zhang, Q., Li, J., Jia, Q., Wang, C., Zhu, J., Wang, Z., & He, X. (2021). UNBERT: User-news matching BERT for news recommendation. *IJCAI Vol. 21*, 3356-3362.

Zhang, S., Yao, L., Sun, A., & Tay, Y. (2019). Deep learning based recommender system: A survey and new perspectives. *ACM computing surveys (CSUR)*, 1-38.

Zhang, Y., & Chen, X. (2020). Explainable recommendation: A survey and new perspectives. *Foundations and Trends® in Information Retrieval, 14(1)*, 1-101.

Zhang, Y., & Zhang, L. (2022). Movie recommendation algorithm based on sentiment analysis and LDA. *Procedia Computer Science*, 871-878.

Zhang, Y., Ding, H., Shui, Z., Ma, Y., Zou, J., Deoras, A., & Wang, H. (2021). Language models as recommender systems: Evaluations and limitations.

Zhao, Z., Fan, W., Li, J., Liu, Y., Mei, X., & Wang,, Y. (2023). Recommender systems in the era of large language models (LLMs).