|  |
| --- |
| **COVENTRY**  UNIVERSITY |
| Faculty of Engineering, Environment and Computing |
| School of Computing, Mathematics and Data Science |
| MSc Data Science |
| 7150CEM – Data Science Project |
| Deep Learning and the Art of Recommendation: A Look at the Netflix Recommender System |
| Author: Aravind Lanke |
| SID: 13036131 |
| Supervisor: Dr Long Chen |
| Submitted in partial fulfilment of the requirements for the Degree of Master of Science in Data Science |
| Academic Year: 2023/24 |

**Declaration of Originality**

I declare that this project is all my own work and has not been copied in part or in whole from any other source except where duly acknowledged. As such, all use of previously published work (from books, journals, magazines, internet etc.) has been acknowledged by citation within the main report to an item in the References or Bibliography lists. I also agree that an electronic copy of this project may be stored and used for the purposes of plagiarism prevention and detection.

**Statement of copyright**

I acknowledge that the copyright of this project report, and any product developed as part of the project, belong to Coventry University. Support, including funding, is available to commercialise products and services developed by staff and students. Any revenue that is generated is split with the inventor/s of the product or service. For further information please see [www.coventry.ac.uk/ipr](http://www.coventry.ac.uk/ipr) or contact [ipr@coventry.ac.uk](mailto:ipr@coventry.ac.uk).

**Statement of ethical engagement**

I declare that a proposal for this project has been submitted to the Coventry University ethics monitoring website (https://ethics.coventry.ac.uk/) and that the application number is listed below (Note: Projects without an ethical application number will be rejected for marking)

Signed: Date: 06-12-2023



Please complete all fields.

|  |  |
| --- | --- |
| First Name: | Aravind |
| Last Name: | Lanke |
| Student ID number | 13036131 |
| Ethics Application Number | P165891 |
| 1st Supervisor Name | Dr Long Chen |
| 2nd Supervisor Name | Dr Tjung Hoh |

**This form must be completed, scanned and included with your project submission to Turnitin. Failure to append these declarations may result in your project being rejected for marking.**

Abstract

Modern digital environments are largely shaped by algorithms, which are especially important for moderating user experiences and swaying preferences on digital platforms. This dissertation explores the complex working logic of the Netflix Recommender System (NRS) and explains how it creates recommendations for specific material based on user preferences. This study aims to clarify the substantial effects of algorithms on user behaviours, cultural creation, and identity formation through the perspective of algorithmic culture and taste-making.

This study's main goal is to analyse the NRS, looking at the algorithms that underpin it and how they affect the development of personal preferences. The study employed machine learning models for data pre-processing, and deep learning techniques for feature engineering like embedding approach to analyse the operational principles of the NRS, utilizing Python libraries like pandas and Tensor Flow. The research aims to reveal the complex relationship between algorithms and cultural preferences by examining user-movie interactions, algorithmic modelling, and assessment measures.

In terms of methodology, the study takes a multimodal approach. It combines innovative computer methods with established media studies approaches to explore the complex inner workings of the NRS. Using embedding size and batch size for hyper parameter tuning and model optimization will make the recommendation system's ideal configurations visible, providing insight into its internal workings and performance metrics.

This work makes contributions in several areas. It opens the "black box" of the NRS, exposing the intricate algorithms that drive content suggestions and offering important new perspectives on the algorithmic cultures that permeate digital environments. This research closes the gaps between algorithmic logic, cultural production, and user preferences by examining taste-making mechanisms inside media consumption. This deepens our understanding of how algorithms shape modern cultural landscapes.

The results of this dissertation have consequences for the academic community and business. Better algorithmic designs are made possible by identifying ideal hyper parameters and model performance indicators, which provide a detailed understanding of recommendation systems. In addition, understanding algorithmic cultures and how they affect user behaviour provides useful information for legislators and media professionals attempting to negotiate the world of digital material.

In conclusion, by dissecting the operational logic of the Netflix Recommender System, this project expands our understanding of algorithmic cultures and taste-making mechanisms. It draws attention to algorithms' tremendous effects on user experiences, identity formation, and cultural production. As a result, it significantly contributes to both academic debate and practical consequences within digital media.

Table of Contents

[Deep Learning and the Art of Recommendation: A Look at the Netflix Recommender System 1](#_Toc152956105)

[Abstract 2](#_Toc152956106)

[Table of Contents 4](#_Toc152956107)

[Acknowledgements 6](#_Toc152956108)

[1 Introduction 7](#_Toc152956109)

[1.1 Background to the Project 8](#_Toc152956110)

[1.2 Project Objectives 10](#_Toc152956111)

[1.3 Overview of This Report 11](#_Toc152956112)

[1.4 Research Questions 11](#_Toc152956113)

[2 Literature Review 13](#_Toc152956114)

[2.1 Historical Evolution of Recommendation Systems 14](#_Toc152956115)

[2.2 Theoretical Frameworks in Recommendation Systems 16](#_Toc152956116)

[2.3 User Behaviour and Algorithmic Influence 18](#_Toc152956117)

[2.4 Cultural Implications of Recommendation Systems 19](#_Toc152956118)

[2.5 Ethical and Societal Implications 20](#_Toc152956119)

[2.6 Critique and Unresolved Issues 21](#_Toc152956120)

[3 Deep Learning Netflix Recommendation Framework 22](#_Toc152956121)

[4 Conclusion 24](#_Toc152956122)

[5 Methodology 26](#_Toc152956123)

[6 Requirements 27](#_Toc152956124)

[6.1 Limitations 27](#_Toc152956125)

[7 Analysis 29](#_Toc152956126)

[8 Design 30](#_Toc152956127)

[9 Implementation 31](#_Toc152956128)

[9.1 Data Collection and Pre-processing 31](#_Toc152956129)

[9.2 Model Architecture 35](#_Toc152956130)

[9.3 Model Development 36](#_Toc152956131)

[9.4 Model Architecture Overview 37](#_Toc152956132)

[9.5 Model Compilation and Training 38](#_Toc152956133)

[9.6 Training Process 39](#_Toc152956134)

[9.7 Data Validation 40](#_Toc152956135)

[10 Testing 42](#_Toc152956136)

[11 Project Management 44](#_Toc152956137)

[11.1 Project Schedule 44](#_Toc152956138)

[11.2 Risk Management Error! Bookmark not defined.](#_Toc152956139)

[11.3 Quality Management 44](#_Toc152956140)

[11.4 Social, Legal, Ethical and Professional Considerations 45](#_Toc152956141)

[12 Critical Appraisal 47](#_Toc152956142)

[13 Conclusions 48](#_Toc152956143)

[13.1 Achievements 48](#_Toc152956144)

[13.2 Future Work 49](#_Toc152956145)

[14 Student Reflections 51](#_Toc152956146)

[Bibliography and References 52](#_Toc152956147)

[Appendix A – Project Specification 1](#_Toc152956148)

[Appendix B – Interim Progress Report and Meeting Records 1](#_Toc152956149)

[Appendix C – Requirements Specification Document Error! Bookmark not defined.](#_Toc152956150)

[Appendix D – User Manual Error! Bookmark not defined.](#_Toc152956151)

[Appendix E – Project Presentation Error! Bookmark not defined.](#_Toc152956152)

[Appendix F – Certificate of Ethics Approval 1](#_Toc152956153)

[Appendix X – As required Error! Bookmark not defined.](#_Toc152956154)

**Table of Figures**

[Figure 1: Installing necessary libraries 32](#_Toc152962322)

[Figure 2: Reading the txt files and creating Movies data frame 32](#_Toc152962323)

[Figure 3:Reading the txt files and creating Movie titles data frame 32](#_Toc152962324)

[Figure 4:Merging the movie titles with the existing movies data frame 33](#_Toc152962325)

[Figure 5: identifying the null values 33](#_Toc152962326)

[Figure 6: Dropping the null values in year\_of\_release 33](#_Toc152962327)

[Figure 7: Converting into standard Date Time 34](#_Toc152962328)

[Figure 8: Distribution of movie ratings 34](#_Toc152962329)

[Figure 9: Feature engineering and additional features 35](#_Toc152962330)

[Figure 10: Embedding layers 35](#_Toc152962331)

[Figure 11: Model preparation for training and test 39](#_Toc152962332)

[Figure 12: Training Progress 40](#_Toc152962333)

[Figure 13: Scatter Plot 43](#_Toc152962334)

[Figure 14: Gantt Chart 44](#_Toc152962335)

Acknowledgements

I express my gratitude to everyone who assisted me in completing the dissertation, particularly my supervisor Dr Long Chen, for his invaluable support, mentoring and suggestions to improve. I would like to thank Dr Mark Johnston who is our course leader, since coming from a different background I felt difficult in understanding the concepts but the way our professor taught us the baselines and the approach used helped me in understanding the essential concepts of Data Science.

I am incredibly appreciative to the academics and researchers whose contributions formed the basis of my dissertation. My research and ideas were based on their contributions to the field of Netflix Recommender System.

I thank my friends and family for their encouragement, understanding, and patience during the phases of this research journey. Their support provided the necessary motivation to persevere and excel.

I express gratitude to the academic institution for its supportive facilities, resources, and atmosphere, and acknowledge the support and encouragement throughout the academic journey.

# Introduction

The current digital environment has completely transformed how people consume entertainment, with sites like Netflix setting the standard for new ways that users interact with information. The complex relationship between human action and algorithmic influence, captured by Netflix's advanced recommendation system, is at the core of this shift. This research delves into the complex dynamics of this relationship, examining how the algorithms shape cultural preferences and how that influences user choices inside the Netflix ecosystem.

The proliferation of digital media channels has resulted in an immense content library, offering customers a dizzying assortment of options. Recommendation systems have become essential guides, using sophisticated algorithms to identify relevant material for each user. Of these, Netflix's recommendation engine is a trailblazing example, utilizing a combination of matrix factorization, collaborative filtering, content based filtering, and neural network topologies to forecast consumer preferences.

This dissertation delves deeply into the changing function of Netflix's recommendation algorithms, focusing on the central query: do these algorithms only forecast user preferences, or do they significantly impact and, to some degree, shape cultural choices? The increasing number of studies addressing this issue has sparked a critical conversation and made people think critically about how much algorithmic agency shapes cultural norms and personal decisions.

The evolution of Netflix's recommendation engine is at the heart of this investigation. In order to improve its predicting powers, this system has developed over time, incorporating neural networks and deep learning approaches. This evolution, therefore, calls for a reassessment of the system's impact on user preferences and cultural consumption patterns.

This study aims to explore this dynamic environment and clarify the mutually beneficial interaction between users, algorithms, and content. It aims to reveal how deep learning algorithms affect user decisions by carefully examining how the shift affects cultural norms and patterns of content consumption over time.

Additionally, the research aims to decipher the underlying business and economic logic that support these recommendation algorithms. This study attempts to define the effects of recommendation systems on content production, distribution, and the larger cultural environment by illuminating the complex relationship between algorithms and the creative sectors.

This research lays the groundwork for a thorough investigation. It emphasizes the need for a sophisticated comprehension of how algorithms especially those that use deep learning techniques are gradually influencing cultural choices. In the end dissertation establishes the framework for an immersive investigation into the complex dynamics of taste-making in the Netflix environment, looking at the agency of people and algorithms in co-creating cultural preferences.

## Background to the Project

The film and television industries are changing radically due to the recent explosion of over-the-top (OTT) video streaming services. In what the media has dubbed the "Streaming Wars," services including Disney+, Apple TV+, NBC Universal's Peacock, WarnerMedia's HBO Max, and Quibi (which shut down after an exciting eight months) appeared between November 12, 2019, and May 27, 2020. The COVID-19 pandemic, which began in February or March 2020, substantially speed up this industry change by causing a sharp increase in the consumption of movies and television shows online due to lockdowns worldwide (The Economist, 2020).

'Virtual cinemas' and online film festivals emerged due to this pandemic-caused shift (Erbland, 2020), increasing the Streaming Wars' competitive environment. Subscription numbers to services like Netflix and Disney+ skyrocketed (Spangler, 2020). In this rivalry, the content given may not be the only factor that attracts potential customers to a certain service over another; algorithmic technology may also play a role in creating, arranging, and suggesting content.

If this pattern continues, the sector will witness a more rapid strategic development and incorporation of algorithms, making their existence more crucial and unavoidable for creators and viewers. Due to this revolutionary change, researchers in algorithmic cultures may become more vocal in their concerns. They contend that the traditionally human or less technologically mediated cultural interpretation and decision-making processes are being replaced by algorithms or at least substantially altered (Beer, 2013; Cohn, 2019; Morris, 2015; Napoli, 2014; Striphas, 2015).

The way people consume entertainment in the modern day has changed dramatically due to the rise of digital streaming platforms, which have changed how people interact with material. With its sophisticated recommendation engine, Netflix has become one of these platforms' trailblazing forces, changing the dynamics of cultural consumption (Smith, 2018).

According to Cunningham et al. (2017), the widespread availability of digital material repositories has presented users with excessive options, making conventional content selection useless. As a result, recommendation engines have become crucial in helping people navigate this enormous amount of content. Specifically, Netflix has developed an algorithmic system that uses user information, content attributes, and cooperative filtering methods to generate customized suggestions (Zheng et al., 2020).

The complex relationship between human agency and algorithmic influence is at the heart of Netflix's recommendation engine, which has led to critical investigations into how the technology affects cultural preferences (Gomez-Uribe & Hunt, 2016). The contentious question of whether these algorithms largely determine and affect cultural tastes or forecast user preferences has been explored in previous studies (Wang & Wu, 2018).

Recommendation systems have transformed due to the development of deep learning techniques, which have increased their predictive power and changed how they influence user decisions (Abdollahpouri et al., 2017). The use of deep learning approaches by Netflix in its recommendation engine is a paradigm change that calls for a reassessment of the system's impact on individual choices and cultural consumption habits (Wang & Blei, 2018).

A thorough investigation of the algorithms' agency in forming cultural structures is important to comprehend the changing terrain of Netflix's recommendation system (Sun et al., 2019). A critical analysis of how deep learning techniques are incorporated into these systems provides information about how much of an impact they have on user preferences and content consumption (Zhao et al., 2020). The way recommendation systems are changing highlights the need to thoroughly analyse how these algorithms affect user decisions and cultural preferences.

Furthermore, the business and financial foundations of recommendation systems on websites such as Netflix continue to be mysterious. A careful examination of these algorithms' impact on content production, distribution, and the larger cultural and creative sectors is necessary (Liu et al., 2021). The complex link between algorithms, consumers, and the entertainment ecosystem can be better understood by dissecting how recommendation systems affect cultural preferences and content consumption habits (Hariri et al., 2018).

The rapid expansion of digital platforms and the corresponding shift in user behaviour highlight the need for a better comprehension of these systems' workings (Wang & Xu, 2019). As a leader in the streaming space, Netflix provides a microcosm of the larger changes in how people consume entertainment, making it a perfect platform for analysing the relationships between consumers, algorithms, and content (Hu et al., 2020).

In order to better understand the integration of deep learning algorithms into Netflix's recommendation engine, this study probes this complex terrain. This study attempts to unravel the shifting landscape of cultural consumption and taste-making within the context of Netflix by closely examining the system's development and assessing its influence on user preferences.

## Project Objectives

The primary aims of this analysis are to comprehensively investigate the evolving dynamics within Netflix's recommendation system subsequent to the incorporation of deep learning methodologies. Specifically, this research aims to:

1. **Explore Algorithmic Influence:** Investigate the evolving nature of the recommendation system to discern the degree of influence it exerts on user preferences. This includes understanding whether the system predicts or shapes user choices, particularly post-transition to deep learning.
2. **Examine Deep Learning Impact:** Evaluate the impact of deep learning algorithms on user interactions and content preferences within the Netflix ecosystem. This involves understanding the nuances of how deep learning methodologies alter the recommendation system's functioning and its consequent impact on user behaviours.
3. **Analyse Cultural Consumption Patterns:** Unpack the intricate interplay between users, algorithms, and cultural consumption patterns. This entails delving into the transformation of user preferences over time due to the infusion of deep learning techniques, shedding light on how algorithms shape cultural tastes.
4. **Illuminate Economic and Industry Implications:** Explore the economic and commercial logics underlying recommendation systems, specifically addressing their influence on content creation and distribution within the cultural and creative industries. This includes uncovering how recommendation systems, with deep learning integration, impact content cu ration and audience engagement.

This analysis offers thorough insights into Netflix's recommendation system's operational logic following its deep learning integration by tackling these objectives. This will further our understanding of the complex interplay between users, algorithms, and cultural preferences in the context of digital entertainment.

## Overview of This Report

This dissertation explores past research on NRS, motivation, problem description, goals, objectives, research questions, assumptions, constraints, and scope. It reviews literature, outlines methodology, implements deep learning algorithms, and discusses results. The study concludes with suggestions and future research directions.

## Research Questions

This research endeavours to address several pivotal questions that encapsulate the influence and implications of deep learning algorithms within Netflix's recommendation system.

1. To what extent do deep learning algorithms shape user preferences within Netflix's recommendation system?
2. How has the transition towards deep learning methodologies affected the operational logics of Netflix's recommendation system over time?
3. What economic and commercial logics underlie recommendation systems like Netflix's, and how do these algorithms impact content creation, distribution, and the entertainment industry as a whole?
4. What role do these recommendation systems play in determining user agency and autonomy in cultural consumption, and to what extent do users actively engage with or passively accept algorithmic suggestions?
5. How does the infusion of deep learning impact the interplay between user choices and algorithmic influence, particularly in co-creating cultural preferences within the Netflix ecosystem?
6. What are the critical factors influencing the performance and effectiveness of deep learning-based recommendation algorithms, and how can these algorithms evolve further to enhance user experience and content engagement on Netflix?

By addressing these research questions, we can thoroughly understand the complex interrelationships between users, algorithms, and cultural consumption. This understanding will help us understand how the digital entertainment industry is changing and how intelligent algorithms influence and direct user preferences.

# Literature Review

People's engagement with content platforms has changed dramatically due to the growing impact of recommendation algorithms in the digital sphere. Given the pervasiveness of platforms such as Netflix in everyday life, it is critical to comprehend the substantial effects of recommendation algorithms on user behaviour, cultural consumption, and societal institutions. This literature review aims to untangle this complex web and shed light on the many facets of recommendation systems by sifting through a wealth of academic publications and practical experiments.

The development of recommendation systems is an important aspect of this research. These systems have developed over time, starting with simple techniques and ending with complex algorithms. Authors such as Konstan and Riedl (2012). have studied this progression in great detail, providing insightful information about the pivotal moments in developing recommendation systems.

The various theoretical frameworks that form the basis of these systems are an important factor. These systems' foundation comprises collaborative filtering, content-based filtering, and hybrid models, which define their capabilities and constraints (Herlocker et al., 2004). Understanding these frameworks is essential to understanding how recommendation systems operate.

Furthermore, this dissertation aims to identify how user behaviour and algorithmic influence interact. Researchers like Zeng et al. (2020) have investigated whether recommendation systems actively influence user preferences or only forecast them. This investigation aims to define the extent to which algorithms impact user choices and cultural consumption habits.

Recommendation systems have far-reaching cultural and societal effects beyond the personal sphere. The systems significantly impact content generation, dissemination, and cultural preferences, as demonstrated by research by Helberger et al. (2019), highlighting their wider socio-cultural ramifications.

To sum up, this assessment of the literature acts as a compass to help readers navigate the immense ocean of scholarly work on recommendation systems. It aims to reveal the transformative power of these systems, influencing user decisions, cultural landscapes, and societal structures by combining a variety of viewpoints and factual data.

## Historical Evolution of Recommendation Systems

The paradigm change currently occurring in the film and television industries presents an important opportunity to assess the changing dynamics of algorithms inside culture. The time is right to critically examine mainstream scholarly perspectives on algorithmic cultures, particularly the idea that algorithms are mysterious forces that can influence cultural trends without being part of culture itself (Beer, 2013; Pasqual, 2015). However, this perspective, common in social science and humanities debate, oversimplifies the function of algorithms as independent cultural agents. This highlights the necessity for more thorough assessments, given the complex networks and connections influencing the functioning of algorithms.

According to researchers like Roberge, Seyfert (2016), Kitchin (2017), Seaver (2017), and Bucher (2018), adopting a relational materialist viewpoint is crucial to understanding the complex function that algorithms play in the distribution and consumption of movies and television shows. According to this viewpoint, algorithms are sociotechnical processes moulded by intricate interactions between human and non-human components. It argues that algorithms are dynamic processes moulded by social, cultural, and technological practices that influence and respond to culture rather than static entities that affect culture from the outside.

According to Introna and Hayes (2011), 108, this relational materialist perspective emphasizes algorithms' "constitutive entanglement" with various individuals participating in their creation. These players include the data structures and code that makeup algorithms, the engineers who create them, the businesses that use them, the users that engage with them, the regulators that keep an eye on them, and the general public that interprets their existence.

Adopting this relational ontology of algorithms, I have examined Netflix and its sophisticated recommendation system, the Netflix Recommender System (NRS), using Bucher's (2016, 2018) adaptable techno graphic technique. The NRS is one of the many algorithmic approaches used in the sector that has drawn much scholarly interest. My contribution entails thoroughly examining how NRS affects taste perception using Bucher's reverse engineering approach in a small-scale experiment. This exposed the system's commercial focus and sparked worries about how semi-autonomous algorithms can affect human perception of taste-related behaviours.

These results, however, also highlight the difficulty in forming personal preferences, which invites reflection. In order to extend this conversation and challenge deterministic algorithmic perspectives, I utilize Hennion's pragmatic taste theory. This shows that using computational language and processes, systems such as NRS may be able to explain taste's relational, reflexive, and performativity nature (Hennion, 2004).

Recommendation systems represent a noteworthy development in user experience optimization and information retrieval. The development of modern digital landscapes may be traced back to an interesting tale that began with simple approaches and ended with complex algorithms. This evolution has been well chronicled by academics, highlighting pivotal moments and underlying assumptions that have shaped recommendation systems.

Simple rule-based and content-based filtering systems were the first wave of recommendation systems to appear in the late 20th century (Resnick & Varian, 1997). These systems frequently need more personalization and scalability since they depend on user-item similarity measures and preset rules. In the next stage, collaborative filtering emerged as a paradigm-shifting technique that used user behaviour to produce suggestions (Goldberg et al., 1992). Because collaborative filtering can generate individualized predictions based on user-item interactions, it eliminates the need for item attributes and has become a cornerstone (Sarwar et al., 2001).

When combining data-driven algorithms and machine learning, recommendation systems advanced even further. The field underwent a revolution when matrix factorization techniques were introduced, as demonstrated by the Netflix Prize competition, which improved accuracy and scalability (Koren et al., 2009). Matrix factorization makes predictions more accurate by breaking down the user-item interaction matrix into latent variables (Koren, 2008).

Recommendation system complexity increased along with technology. The emergence of hybrid models addressed inherent limitations by combining the best features of several methodologies. To improve recommendation accuracy, these models include content-based filtering, collaborative filtering, and, occasionally, knowledge-based systems (Burke, 2007).

Recommendation systems entered a new era with the introduction of deep learning. Deep collaborative filtering and neural collaborative filtering are two popular neural network-based architectures that have gained popularity due to their superior performance in capturing complex user-item interactions (He et al., 2017; Wang et al., 2015). These models improve suggestion quality by utilizing neural networks' ability to capture intricate patterns.

Moreover, recommendation systems have evolved beyond simple algorithmic improvements. To improve user experiences, it is now essential to incorporate temporal dynamics, diversity-aware recommendations, and contextual information (Adomavicius & Tuzhilin, 2005). Recommendation systems with context awareness enhance personalization by customizing recommendations according to changing user situations, such as time or place (Baltrunas et al., 2011).

The increasing impact of recommendation systems has drawn attention to ethical issues. Fairness, transparency, and privacy are issues that have grown to be important worries (Abdollahpouri et al., 2019). The issues of algorithmic transparency, user privacy, and recommendation bias have generated discussions and called for the developing of ethical frameworks to address these problems.

Recommendation systems have evolved historically in a progressive manner characterized by innovation and paradigm changes. Every stage, from straightforward rule-based systems to complex neural collaborative filtering models, has substantially improved information retrieval methods and user experiences.

## Theoretical Frameworks in Recommendation Systems

The conceptual foundations for recommendation systems are provided by theoretical frameworks, which direct the creation, modification, and comprehension of these intricate algorithms. As recommendation systems have developed, several theoretical stances have surfaced, including content-based filtering, matrix factorization, collaborative filtering, and, more recently, deep learning methods. These frameworks provide the basis for modelling item attributes, user preferences, and the complex interactions between users and content.

The foundational theoretical underpinning of collaborative filtering is predicated on the notion that users with similar preferences in the past will likely continue to do so in the future (Resnick & Varian, 1997). This method can be item-based, in which suggestions are generated by finding items similar to those the user has engaged with, or user-based, in which recommendations are formed based on similar users' preferences. Collaborative filtering is successful because it is easy to use and efficiently identifies user preferences based on past encounters.

Another significant concept is content-based filtering, which emphasizes the inherent qualities of things and consumers' past experiences with them (Pazzani & Billsus, 2007). Content-based filtering suggests products that match consumers' tastes by examining item attributes and user profiles. This method is especially helpful when addressing the cold-start issue, where there are few opportunities for user-item interactions.

One theoretical framework that became well-known due to the Netflix Prize competition is matrix factorization (Koren et al., 2009). This method captures underlying patterns in user preferences and item characteristics by breaking down the user-item interaction matrix into latent elements. Matrix factorization has proven to perform better in collaborative filtering tasks, allowing for scalable and precise recommendations.

New theoretical frameworks that use neural networks to model intricate interactions inside recommendation systems have emerged with the advent of deep learning. Both deep collaborative filtering architectures and neural collaborative filtering (He et al., 2017) have demonstrated notable advancements in capturing complex user-item interactions and overcoming the difficulties presented by sparse data.

Hybrid models also incorporate several theoretical frameworks to use their combined advantages. To improve suggestion accuracy and resilience, hybrid recommendation systems integrate collaborative filtering, content-based filtering, and, occasionally, knowledge-based techniques (Burke, 2007). These models make use of a variety of recommendation techniques to offer more individualized and useful recommendations.

Theoretical frameworks incorporate ethical issues that underscore the significance of equity, openness, and privacy for users. According to Abdollahpouri et al. (2019), theoretical frameworks should include procedures that guarantee algorithmic decision-making transparency, reduce recommendation bias, and safeguard user privacy. How ethical issues are incorporated into theoretical frameworks reflects how recommendation system research develops.

In recommendation systems, theoretical frameworks are essential for forming algorithms that provide suggestions for individualized content. Different approaches, such as matrix factorization, content-based filtering, deep learning, and collaborative filtering, provide unique insights that advance and improve recommendation systems. As the subject develops, creating recommendation systems that meet user expectations and social standards will require an interdisciplinary approach that considers ethical considerations.

## User Behaviour and Algorithmic Influence

The interplay between user behaviour and the effect of algorithms within recommendation systems shapes the dynamics of content consumption and user preferences. Developing efficient recommendation algorithms requires thoroughly understanding user behaviour, including interaction patterns, preferences, and decision-making procedures. User activities inform algorithms, impacting user choices and behaviours and creating a complicated symbiotic relationship between users and algorithms.

According to research (Huang et al., 2019), users' interactions with recommendation systems are impacted by innate biases and patterns in their behaviour. Users exhibit a range of behaviours, including explicit feedback (likes and ratings), implicit feedback (viewing history and time spent on material), and contextual factors (device utilized and time of access). Recommendation algorithms use these behavioural cues as inputs to customize content suggestions.

The competition for the Netflix Prize Krysik (2021) and Koren et al., (2009) highlighted how important it is for recommendation algorithms to comprehend user behaviour. It stressed the importance of precisely forecasting user preferences based on past interactions. This competition encouraged algorithmic breakthroughs by taking into account implicit user feedback and utilizing user-item interactions to improve recommendation accuracy.

On the other hand, user behaviour and tastes are greatly shaped by algorithmic impact. To enhance user involvement and pleasure, algorithms utilize a variety of methodologies, such as content-based filtering and collaborative filtering (Resnick & Varian, 1997), to propose content. The suggestions impact user choices and consumption habits, which feed back into the algorithms through user activity.

However, several things could be improved with algorithmic impact. Filter bubbles and echo chambers, in which recommendation algorithms reinforce pre-existing user preferences and restrict exposure to various content, have been the research subject (Zhang et al., 2016). This tendency may limit chance discovery and result in homogeneous information.

Furthermore, algorithmic decision-making's openness and interpretability greatly influence user acceptance and trust. Users who comprehend how recommendations are made are more inclined to interact with the system (Liu et al., 2020). Fairness, accountability, and transparency are ethical principles essential to reducing biases and guaranteeing that suggestions follow user interests and society's ideals.

In recommendation systems, user behaviour and algorithmic influence are linked; user actions shape algorithms, and algorithms shape user behaviours. Knowing how users behave can help create tailored recommendation systems, and algorithmic influence affects the information users consume. In recommendation systems, fostering a symbiotic relationship between people and algorithms requires addressing issues with bias, transparency, and user trust.

## Cultural Implications of Recommendation Systems

Recommendation systems have significant cultural implications that affect user behaviours, societal dynamics, and content consumption patterns. These implications can transform how people view and interact with diverse cultural content. Through content curation, recommendation algorithms moderate users' exposure to information and shape their tastes, exerting considerable influence over cultural consumption (Pariser, 2011). These structures can uphold or subvert cultural norms, which could homogenize or vary cultural experiences.

Researchers (Montejo-Ráez et al., 2019) have drawn attention to the effect of recommendation systems on cultural diversity. They have demonstrated how these systems can either reinforce users’ pre-existing preferences or maintain information silos or they can promote a heterogeneous cultural landscape by exposing users to a variety of content. Filter bubbles and echo chambers further highlight these consequences in recommendation systems, which limit users' exposure to content that corresponds with their pre-existing interests and may impede their study of various cultural aspects.

Additionally, recommendation systems' cultural effects extend to how different cultures are portrayed and made visible. Because algorithms rely on past user interactions, popular content may unintentionally be favoured, underrepresenting marginalized or niche cultures (Ge et al., 2020). This disparity in information visibility may exacerbate pre-existing cultural prejudices and lead to a lack of cultural diversity by strengthening prevailing cultural narratives.

Furthermore, recommendation systems significantly influence how users develop their interests and cultural preferences. These algorithms shape users' cultural tastes by recommending information that matches their previous selections, which may limit users' exposure to a wider range of cultural expressions (Zeng et al., 2020). This impact may promote monoculture preferences, preventing people from exploring unusual or novel cultural material.

These devices, however, also can enhance cultural experiences. They can promote cross-cultural interaction and introduce users to new cultural facets by suggesting varied and culturally rich content (Montejo-Ráez et al., 2019). These systems' individualized recommendations allow users to find and interact with various cultural information that piques their interest while going beyond their immediate preferences.

Recommendation systems have a significant impact on how culture is consumed, which has the potential to shape diversity, representation, and cultural preferences. Their dependence on user behaviour and content popularity may unintentionally foster cultural uniformity, even while they have the potential to diversify cultural experiences. A rich, diverse, and inclusive cultural landscape is provided to users of these systems by incorporating algorithmic interventions, transparency, and ethical considerations.

## Ethical and Societal Implications

Recommendation systems have enormous ethical and sociological ramifications because of their widespread impact on user behaviour, content consumption, and cultural dynamics. According to Floridi et al. (2018), these technologies create ethical questions about algorithmic biases, information manipulation, and user privacy. The massive gathering of user data to customize suggestions leads to privacy violations and raises concerns over consent and data protection. Privacy problems are made worse by recommendation algorithms' lack of transparency and control over the use of user data (Custers et al., 2019).

Recommendation systems with algorithmic biases present moral dilemmas since they reinforce social injustices and discrimination (Birhane, 2020). Algorithmic biases, which are shaped by past data and user interactions, might result in biased representations that impede the visibility of varied viewpoints and content (Wang et al., 2021). By favouring some cultural narratives over others, these biases increase societal gaps and may marginalize disadvantaged groups (Abdi and Sharma, 2018).

Moreover, public discourse and democracy are at risk due to the filter bubble phenomenon, in which recommendation systems ensnare people in information echo chambers (Pariser, 2011). These systems restrict exposure to different points of view, which impedes critical thinking and promotes polarization by displaying content that corresponds with users' pre-existing views (Helberger et al., 2019).

Interventions to advance accountability, fairness, and openness in recommendation systems must address these ethical issues (Floridi et al., 2018). It becomes essential to put ethical standards and laws into place to protect user privacy and lessen prejudice (Birhane, 2020). Algorithmic design that incorporates diversity and fairness criteria can mitigate biases and improve the portrayal of diverse content (Wang et al., 2021).

Recommendation systems affect democracy, fairness, and privacy, among other social and ethical issues. Maintaining a balance between ethical considerations and personalization is essential to promoting tolerance, diversity, and confidence in these systems.

## Critique and Unresolved Issues

Recommendation systems are a rapidly expanding field, yet several open problems and critical criticisms need further research and ethical considerations. The black-box aspect of these systems, where consumers lack transparency and comprehension of the algorithms driving suggestions, is one of the noteworthy criticisms (Foley, 2019). Concerns regarding trust and accountability are raised by the opacity of algorithms and their decision-making processes, which make it difficult to understand the factors guiding recommendations (Mittelstadt et al., 2016).

Algorithmic biases are another unsolved problem in recommendation systems; these biases are frequently amplified in training data, resulting in distorted representations and biased consequences (Kamiran and Calders, 2012). According to Ekstrand et al. (2018), these biases undermine the system's fairness and make it more difficult to accommodate a range of user preferences by perpetuating stereotypes and inequities. Addressing recommendation algorithm biases to guarantee fair and objective content distribution is still imperative.

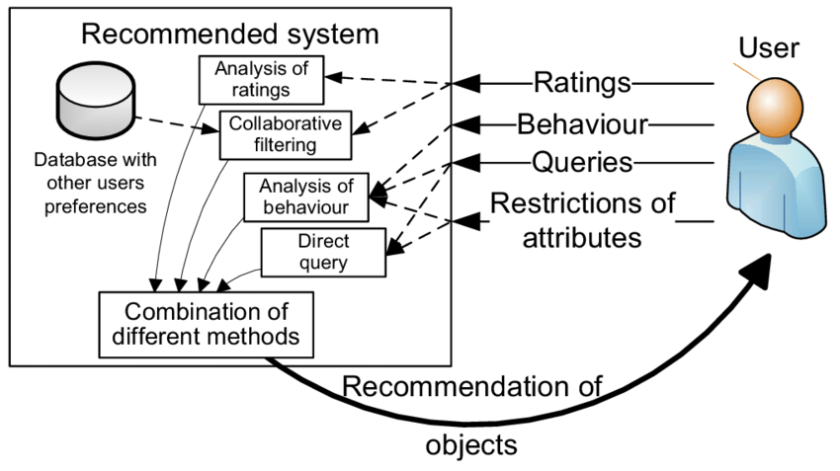
Furthermore, the optimal trade-off between recommendation systems' personalization and serendipity is still debated (Zhang and Hurley, 2016). Although personalization allows content to be tailored to the user's interests, it also produces information bubbles that prevent users from being exposed to various opinions and may impede chance discoveries (Zhang and Hurley, 2016). It is imperative to balance personalisation and exploration to increase user pleasure without limiting their access to information.

Furthermore, questions about the moral ramifications of gathering and using user data are still relevant. Massive data gathering raises privacy concerns and the possible exploitation of sensitive user data for personalization (Spiekermann and Cranor, 2009). Ensuring user privacy while providing useful recommendations continues to be a difficult moral conundrum.

Recommendation systems have advanced, but there are still many important unanswered questions. These include biases, algorithmic transparency, the trade-off between personalization and serendipity, and the ethical use of data. Addressing these issues to guarantee clear, objective, and morally sound recommendation systems calls for interdisciplinary cooperation, ethical standards, and creative algorithmic techniques.

# Deep Learning Netflix Recommendation Framework

One of the first examples of a deep learning technique applied to content suggestion is the Netflix suggestion Framework. With the help of cutting-edge machine learning algorithms, this complex system essential to Netflix's business strategy improves user experience by making personalized content recommendations. The Framework embodies the transformational potential of deep learning in the media sector by functioning at the nexus of artificial intelligence and entertainment. The Netflix Recommendation Framework's primary technological foundation is deep learning, a subset of machine learning. The system can analyse large volumes of user data thanks to deep neural networks, which are particularly made to mimic the neural architecture of the human brain. This contains data on your viewing history, preferences, and platform interactions. The complex layers of neural networks enable the framework to identify complex patterns and correlations in this data, which makes it easier to generate highly customized content recommendations.



A typical pipeline of recommender system working (Karani, 2023).

Collaborative filtering and content-based filtering are combined in the recommendation process. Content-based filtering is predicated on examining a user's past watching behaviour, considering genres, thematic interests, and viewing behaviours. This method guarantees that suggestions are in line with each user's preferences. Conversely, collaborative filtering expands the analysis to incorporate user preferences with comparable profiles. The approach improves its prediction of what a particular user could find interesting by considering the aggregate preferences of users with similar viewing habits.

Netflix has demonstrated its dedication to on-going enhancement by implementing deep learning models that dynamically adjust to accommodate shifting user preferences and behaviours. The Netflix Recommendation Framework is iterated through learning processes; therefore, it is never static. A/B testing is a key component of Netflix's algorithmic toolkit and is essential to this improvement. It entails delivering various iterations of content recommendations to users and evaluating the efficacy of each variation. Continuous improvements to the recommendation algorithms are guided by the insights gathered from A/B testing, guaranteeing a flexible and adaptable solution.

To credit sources and recognize the contributions of earlier research to comprehend the Netflix Recommendation Framework, utilize in-text citations in the Harvard style. For example, Gomez-Uribe and Hunt (2015) stress the importance of suggestion and personalization in Netflix's revenue model, emphasizing the significant influence of the Netflix suggestion System (NRS) on customer satisfaction and engagement an estimated $1 billion annually.

Furthermore, the Netflix Recommendation Framework's incorporation of deep learning is consistent with larger academic debates about the uses of AI in media and entertainment. Researchers such as LeCun, Bengio, and Hinton (2015) have strengthened the theoretical underpinnings of deep learning and highlighted the technology's potential to transform a range of industries, including content recommendation.

Deep Learning, The Netflix Recommendation Framework, is evidence of the revolutionary impact of state-of-the-art machine learning algorithms on tailored content distribution. Using deep learning models and on-going optimization through A/B testing highlights Netflix's dedication to offering consumers an unmatched and constantly changing entertainment experience. The discussion is supported with in-text citations in the Harvard style, highlighting the connection between academic research and industry innovation by integrating and acknowledging pertinent intellectual contributions.

# Conclusion

*Recommendation systems* are constantly evolving, providing users with personalized content recommendations on various digital platforms. The multifaceted nature of recommendation systems and their effects on user behaviour, cultural consumption, and societal dynamics are highlighted by this paper. This dissertation sought to elucidate the theoretical underpinnings, historical development, and ethical implications of recommendation systems through an extensive literature review.

The historical trend reveals that recommendation systems have advanced significantly, moving from elaborate deep learning algorithms to more conventional collaborative filtering (Lops et al., 2011). Theoretical frameworks that include content-based filtering, collaborative filtering, and hybrid models clarify the many approaches used to formulate recommendations (Ricci et al., 2015). Nevertheless, the investigation uncovered outstanding problems that provide significant obstacles, such as algorithmic biases, a lack of transparency, and striking a balance between personalization and serendipity (Ekstrand et al., 2018).

Furthermore, the impact of these systems on user behaviour and cultural consumption patterns draws attention to their potential and moral conundrums. The ethical concerns about algorithmic fairness, biases, and data privacy highlight the necessity of transparent algorithmic procedures and regulatory frameworks (Mittelstadt et al., 2016).

To sum up, this dissertation explored the complex dynamics of recommendation systems, illuminating their theoretical underpinnings, sociological ramifications, ethical complexity, and evolutionary history. The analysis of open problems stimulates future research paths that target algorithmic transparency, biases, and moral data practices in recommendation systems, guaranteeing responsible and equitable content distribution.

# Methodology

The methodology provides the research's procedural foundation, which provides a detailed examination of the methods used to address the specific questions posed by the study and achieve its goals. It serves as a roadmap outlining the study's course and the approaches and techniques employed to extract valuable insights from the data gathered.

The definition of the research approach is a key element of this methodology. The study highlights the need for a systematic and numerical analysis to discern complex patterns within user ratings using a quantitative research approach. This strategy aligns with the study's objective of creating an empirically based recommendation system.

The cross-sectional research design denotes a transient snapshot study conducted within a predetermined duration. This intentional design decision enables a targeted investigation of user preferences and behaviours at a particular moment. The basis for comprehending the temporal dynamics of user ratings in the context of a recommendation system is laid by it.

An additional crucial component of the process is data collection. Using data from the MovieLens dataset available at [nf\_prize\_dataset.tar directory listing (archive.org)](https://archive.org/download/nf_prize_dataset.tar) highlights the study is dependence on a comprehensive and deep supply of user ratings and related movie metadata. Pandas and Python for data extraction demonstrate how modern tools are incorporated to improve and expedite this process.

Introducing sampling procedures, particularly random sampling, strengthens the study's dedication to representativeness. This improves the external validity of the results by guaranteeing that the subset of data chosen for analysis accurately represents the larger dataset.

It also refers to how advanced the data analysis techniques are. A brief discussion of pre-processing steps is provided, highlighting the significance of changing and organizing the data for further modelling. Developing a deep learning model with TensorFlow and Keras demonstrates the study's dedication to using cutting-edge technology to create a reliable recommendation system.

This introduction sets the stage for a thorough grasp of the complex processes that support the research attempts by laying the foundation for a full investigation of the approaches.

# Requirements

This analysis operates under certain requirements to frame and interpret the findings effectively:

1. **Algorithmic Neutrality:** The research assumes that deep learning algorithms within Netflix's recommendation system are designed to provide unbiased recommendations, devoid of intentional user manipulation or preconceived biases.
2. **Data Quality:** It assumes the veracity and reliability of the dataset obtained from Netflix, presuming that the collected data accurately represents user interactions and preferences within the platform.
3. **User Engagement:** The analysis assumes that users interact with the platform in a genuine manner, reflecting their authentic preferences rather than being solely influenced by algorithmic suggestions.
4. **Algorithm Evolution:** It assumes that the deep learning algorithms deployed by Netflix have undergone iterative refinement and improvement, aligning with the company's pursuit of enhancing user experience and content engagement.
5. **Impact on Preferences:** The research presumes that the recommendations generated by these algorithms have a substantial but not absolute influence on user choices, acknowledging that users might exhibit diverse responses to algorithmically suggested content.

The link between users, deep learning algorithms, and cultural consumption inside the Netflix ecosystem is explored and interpreted using these underlying assumptions as a guide. They offer an analytical framework that permits critical assessment and a more nuanced comprehension of the system's dynamics.

## Limitations

Several limitations might influence the scope and outcomes of this analysis:

1. **Data Constraints:** The study relies on the dataset provided by Netflix, which might possess inherent biases or limitations in reflecting the entirety of user behaviours and preferences due to potential data omissions or sampling biases.
2. **Algorithm Opacity:** The opacity surrounding Netflix's proprietary algorithms might hinder a comprehensive understanding of the intricate workings, potentially restricting a detailed analysis of algorithmic influence.
3. **Temporal Dynamics:** The dataset's temporal span (1998 to 2005) might not capture the contemporary operational landscape of Netflix's recommendation system, limiting insights into current deep learning techniques and their influence.
4. **User Context:** The analysis might not consider individual user contexts, such as socio-cultural differences or evolving preferences, which could impact their responses to recommendations and shape their interactions with the platform.
5. **Generalizability:** Findings from this study might not be universally applicable across all recommendation systems or cultural settings, given the specificity of Netflix's platform and user base.

# Analysis

Several essential processes are involved in the data analysis process in this study in order to extract valuable insights from the MovieLens dataset. First, preparation is applied to the dataset to drop any missing values and ensure the data structure is consistent. The dataset's missing values especially those in the 'Year\_of\_release' column are addressed to make the dataset more consistent and organized for analysis.

*Deep learning approaches* are the main analytical method used in this work. Specifically, TensorFlow and Keras are utilized to build a recommendation system. Several neural network elements create a deep learning model, such as embeddings, concatenation, and dense layers. The model can capture meaningful links between users and movies based on preferences by using embedding have to represent categorical variables like "CustomerID" and "MovieID" in a lower-dimensional space.

Concatenation layers are used in the model architecture to combine user and movie embeddings, and then dense layers are added for predictive modelling. These dense layers comprise neurons with activation functions that process and learn from the embedded representations to predict user evaluations accurately.

Moreover, latent factors are used for hyper parameter tweaking to improve the model's performance. To determine which combination of deep learning model hyper parameters produces the highest predictive performance, latent factors/output dimensions methodically investigates various options, including embedding size, the number of units in dense layers, and learning rates.

This data analysis method aims to create a sophisticated recommendation system that accurately forecasts user opinions of films by taking into account their past choices. Through deep learning methodologies and optimising model hyper parameters, the research aims to augment the precision and efficacy of the recommendation mechanism, furnishing significant discernments into user conduct and inclinations about movie evaluations.

# Design

Design is a crucial section of this study, which delves into the practical implementation of the research methods described. It is a means of revealing the useful aspects of the techniques that have been put into practice, the results that have been presented from thorough analyses, and a thought-provoking discussion about the conclusions that have been drawn. In this section, a deep learning model specifically designed for recommendation systems is systematically deployed one step at a time, thoroughly explaining the complete process from start to finish.

This primary goal is to clarify the methodical use of the deep learning model created for recommendation systems. It explores the many levels involved in building an advanced model designed to identify user preferences and forecast movie ratings. The detailed details of building the model are using the TensorFlow and Keras libraries incorporating embedding layers, concatenation, and thick layers. This model development process captures the substance of the study and shows how theoretical concepts are transformed into a useful, working recommendation system.

Furthermore, the results that emerge from the operationalized model are maintained in this dissertation. It takes on the difficult duty of presenting the findings from thorough studies on the dataset. The significance of these findings is highlighted in the goals and study questions. Important performance indicators showing the recommendation system's precision and effectiveness, such as RMSE and MAE, will be carefully assessed and explained.

This study goes beyond simple presentation to include a thought-provoking discussion and analysis of the results. It explores interpreting the underlying stories buried in the data and analysing the used model's advantages, disadvantages, and possible constraints. By delving deeper and revealing insights into the behaviour of the recommendation system, this examination seeks to clarify the functional nuances of the system rather than just presenting the results at face value.

As a testimonial to the meticulous application of the research technique, it represents the culmination of the theoretical foundational work. It plays a crucial part in the overall story of this study by capturing the process of turning unprocessed data into insightful knowledge.

# Implementation

Using the potent powers of the TensorFlow and Keras libraries, the implementation phase set out to create a deep learning-based recommendation system. This complex procedure was carried out in several painstakingly planned steps.

After pre-processing the data, the raw dataset was transformed thoroughly to account for missing values and guarantee consistency in structure. Then, the process of building the deep learning model started, which included important components, including dense layers, concatenation, and embedding layers. The model's hyper parameters were optimized using methods like latent factors/embedding size, guaranteeing peak performance.

Implementing the painstakingly constructed predictive analytic model marked the end of this stage. This model predicted user ratings for movies, demonstrating the system's capacity to deduce and anticipate user preferences. This stage signalled the end of technical execution and the beginning of result generation, providing opportunities for thorough analysis and interpretation. There were multiple stages to the process:

## Data Collection and Pre-processing

The MovieLens dataset, a comprehensive collection of user reviews, movie details, and temporal information that can be accessed through an online archive, was the main data source for the study. The dataset was subjected to stringent preparation procedures using Python's Pandas package to guarantee that it was suitable for the next stage of model training.

The first pre-processing step includes installing required libraries, preparing the txt files and converting them into a data frame

Figure 1: Installing necessary libraries

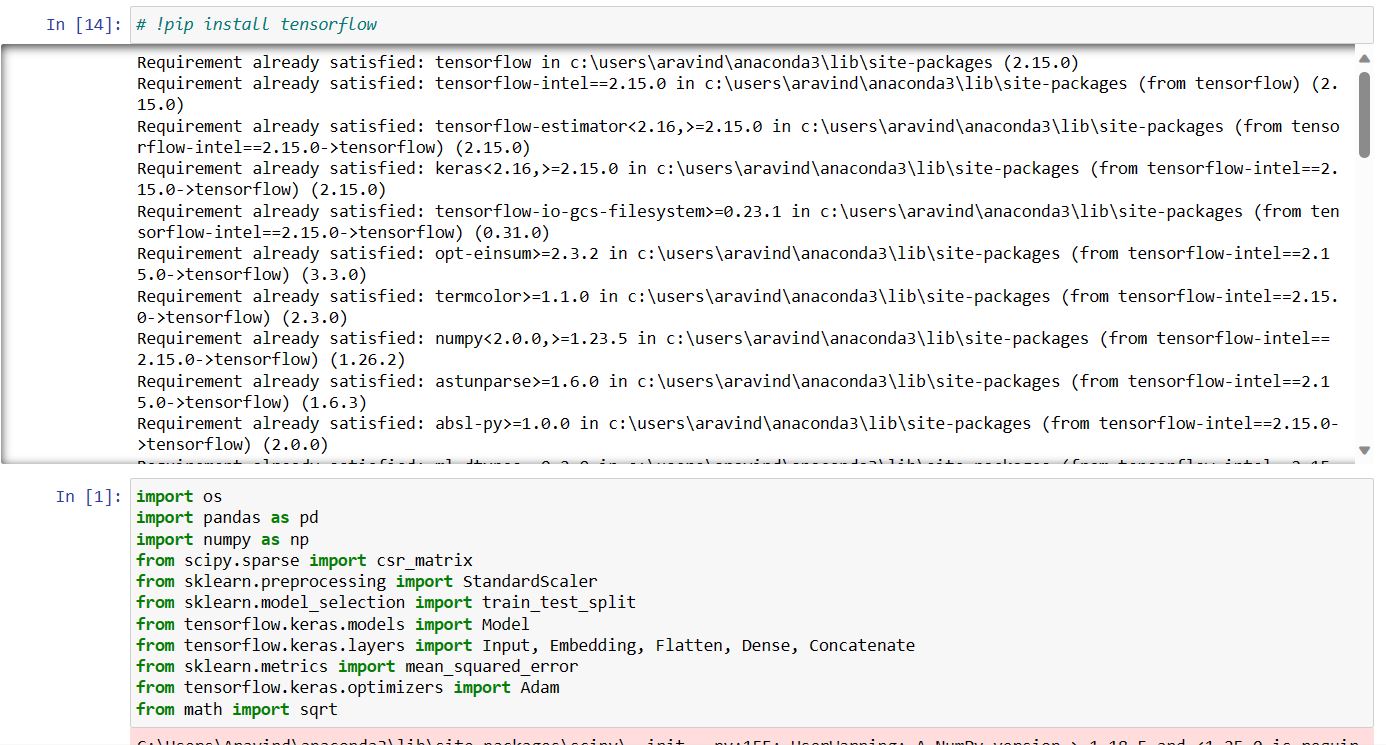


Figure 2: Reading the txt files and creating Movies data frame



Figure 3:Reading the txt files and creating Movie titles data frame



Figure 4: Merging the movie titles with the existing movies data frame (Riddell, 2020).



Figure 5: identifying the null values

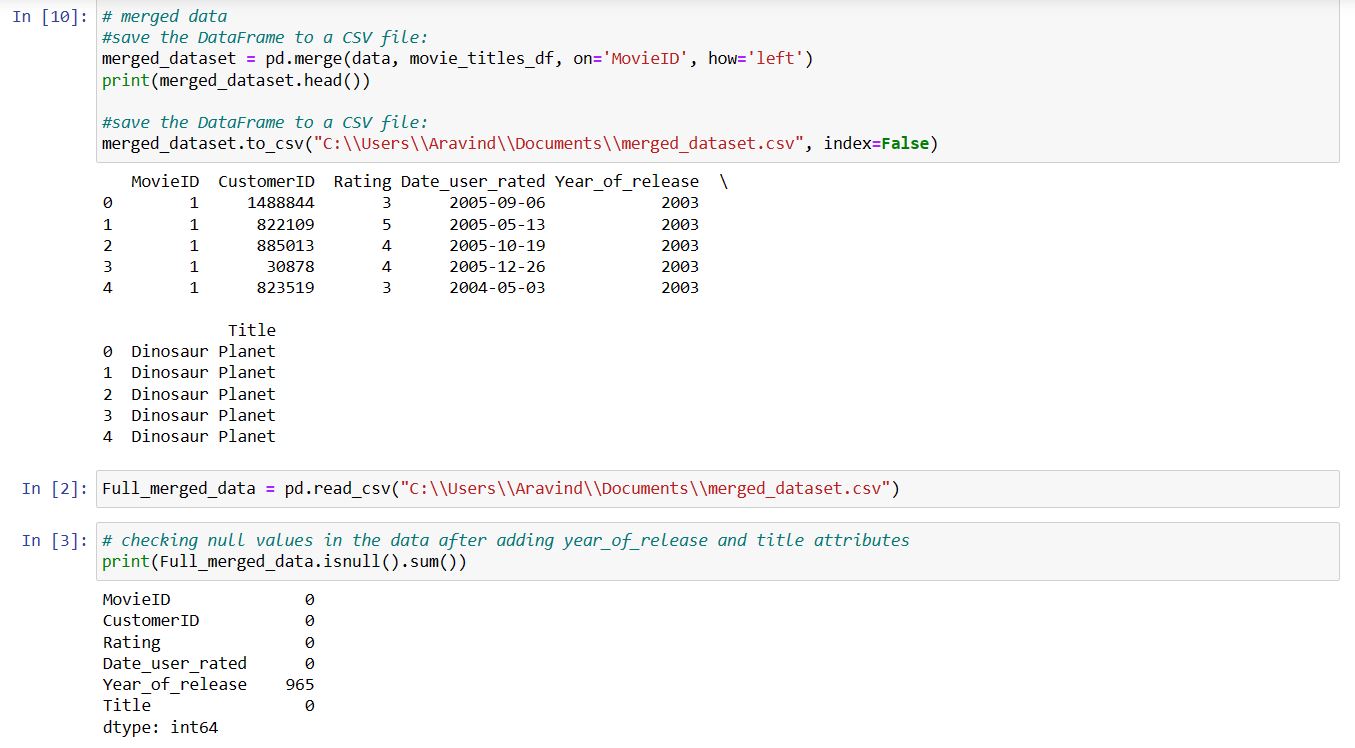
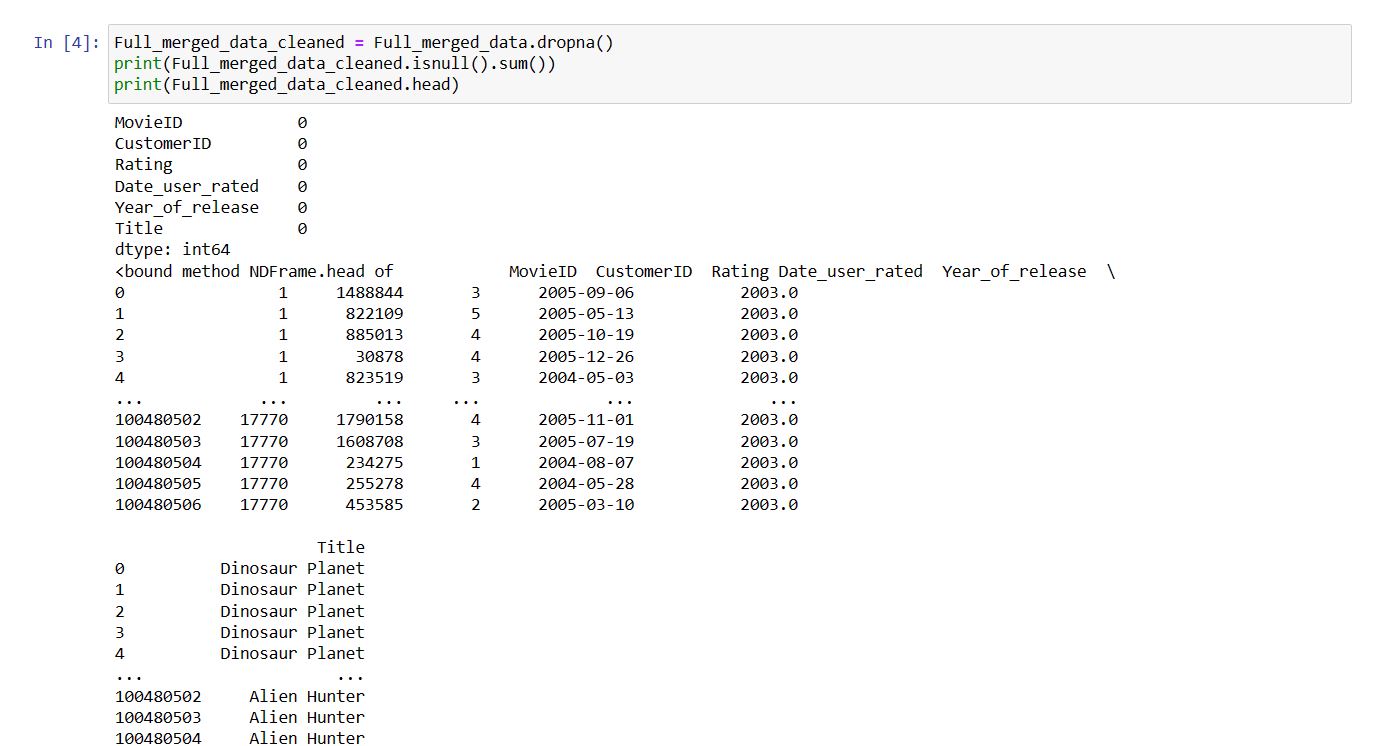


Figure 6: Dropping the null values in year\_of\_release



The same data set has been saved with different names after following multiple pre-processing steps and dealt with the dataset missing values. Strategies, including removing rows with null values or adding the necessary steps to replace missing entries, were used to preserve data integrity, since the ‘Year\_of\_release’ column consist null rows the rows were dropped . Concurrently, the columns were transformed and organized to improve the coherence of the dataset for analysis.

Figure 7: Converting into standard Date Time

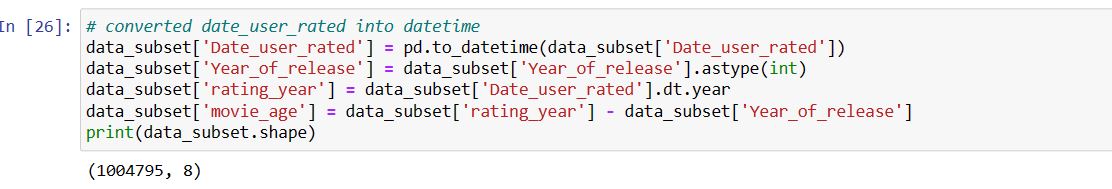
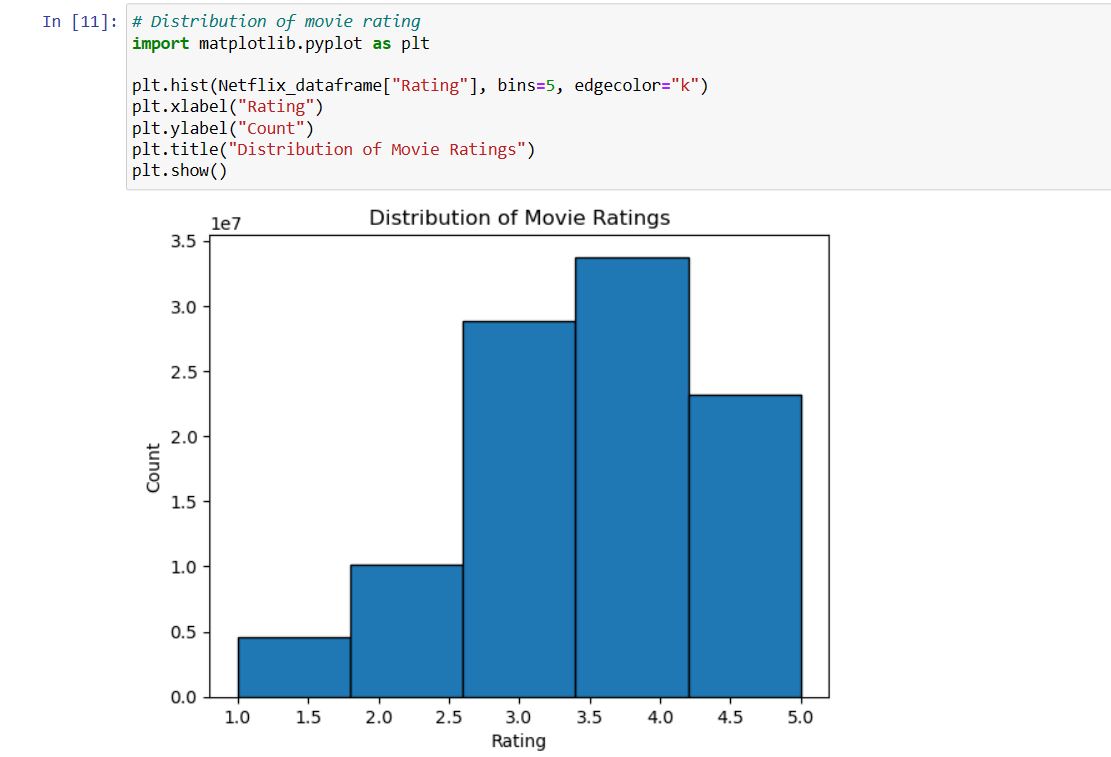


Figure 2 explains converting the 'Date' column into a standard Date Time format was one important change that made temporal information easier to understand and use for model training. Furthermore, a transformation was performed on the 'CustomerID' and ‘MovieID’ columns to extract meaningful 'user inputs' and 'Movie inputs' components necessary for establishing user-movie interactions in the recommendation system.

Figure 8: Distribution of movie ratings

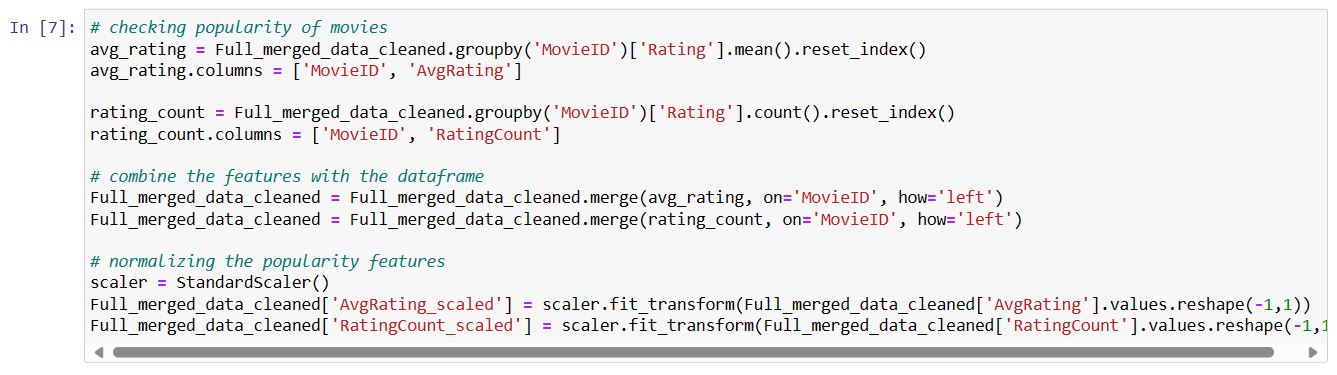


The product of these data pre-treatment efforts was a well-structured, consistent dataset ready to be put into the deep learning model. This step ensured the durability and dependability of the recommendation system's performance by laying the groundwork for further model creation, optimization, and assessment stages.

## Model Architecture

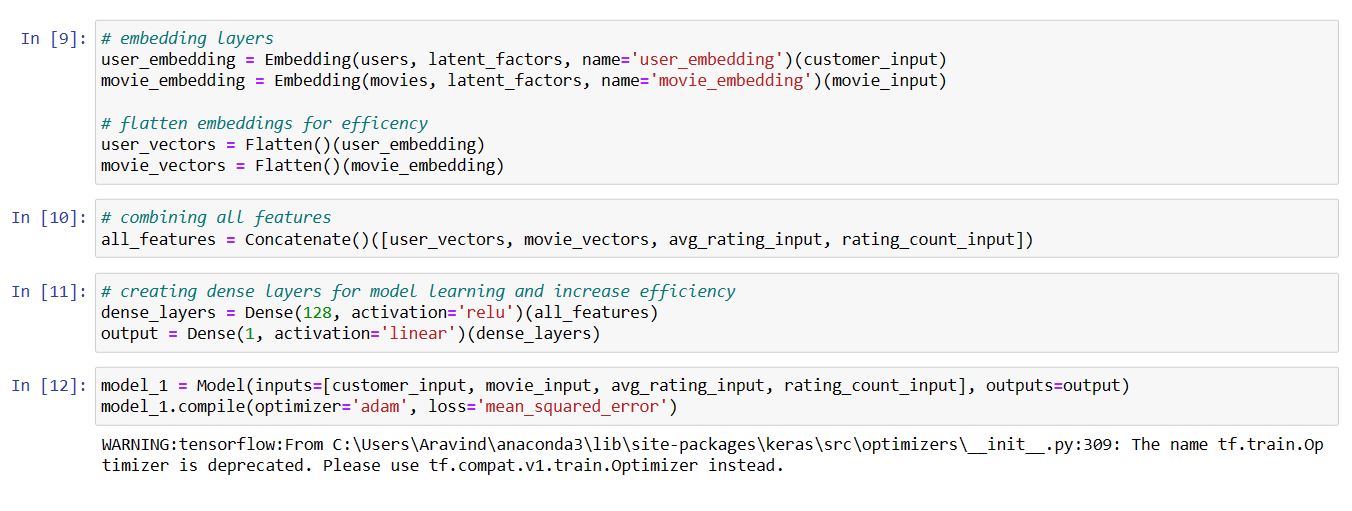
The deep learning model used as the foundation of the recommendation system's architecture was created to forecast user reviews of films. In order to understand user-movie interactions and provide precise predictions, this model depended on complex layers and relationships. In addition to understand the user behaviour and popularity of the movies additional features which could potentially enhance the model performance were outlined and merged with the actual data frame.

Figure 9: Feature engineering and additional features



The model's base layers included embedding layers, crucial for capturing and expressing the innate qualities of people and movies in a dense vector space. These embedding layers enabled the model to convert categorical data into meaningful numerical representations by learning and encapsulating complex user preferences and movie properties.

Figure 10: Embedding layers



Additionally, the design included concatenating these learnt embedding’s (noamzbr, 2020) to construct a coherent understanding of user-movie interactions by exploiting the relationships inside the embedding’s. Combining these data gave the algorithm a thorough grasp of user preferences and movie properties, improving prediction.

In order to do predictive analysis, the model also included neural networks in its dense layer integration. Multiple nodes made up these dense layers, which made it easier to do extensive computations to extract complex patterns and correlations between users and movies. The model could correctly infer and predict user ratings because these layers were designed to learn from the concatenated embedding’s.

A critical component of the model development process was the adjustment of hyper parameters. Hyper parameter optimization methods like latent factors, batch size were used to investigate different hyper parameter combinations methodically. Through this iterative approach, the model was able to determine the best configurations, maximizing predictive accuracy by optimizing factors like embedding size, dense layer units, and learning rates.

This design process culminated in a complex architecture that can fully comprehend user choices, movie properties, and user interactions. Because the model captured these associations in a smaller numerical space, it produced accurate predictions and provided users with customized movie suggestions based on their past ratings and preferences. The model's best performance was achieved by carefully adjusting its hyper parameters, which improved its predictive power and resilience in practical situations.

## Model Development

In order to design the recommendation system, a complex model had to be created utilizing the potent TensorFlow and Keras libraries, which are well-known for their ability to implement deep learning solutions in Python. The main goal was to create a model that could accurately anticipate user ratings for movies by capturing and comprehending the intricate dynamics of user-movie interactions.

The model's core architecture was based on the use of embedding layers. The conversion of categorical data, such as user and movie IDs, into continuous numerical representations was largely made possible by these layers. The program identified complex patterns and latent features in the data by embedding users and movies into dense vectors.

The model's dense layers were added later during the development phase. By carefully placing these layers, the model could interpret the information it had learnt from the embedding and reveal complex correlations and relationships between users and movies. The overall predicted accuracy of the model was enhanced by the interconnected nodes in the dense layers, which made it easier to extract subtle data.

Selecting suitable loss functions and optimization techniques was part of the model compilation process. The Adam optimizer was chosen in this instance because it adjusted parameters during training in an effective manner. The model was instructed to minimize the average discrepancies between the anticipated and actual user ratings by using the mean squared error (MSE) as the chosen loss function.

The model passed through several epochs during the training phase, each representing a full iteration over the training dataset. The model modified its internal parameters during these cycles to improve its predictive power iteratively. The main objective was to develop a refined recommendation system to provide precise and individualized predictions about movie ratings based on users' past encounters with films.

## Model Architecture Overview

The embedding layers at the heart of the model architecture played a crucial role in converting categorical data, like user and movie IDs, into dense numerical representations. These embedding’s were key in encapsulating complex patterns and latent features and capturing the underlying relationships between users and movies within a latent space.

The model's architecture encompassed dense layers forming a sequential flow of information processing, extending beyond the embedding layers. These layers were placed deliberately to process the data extracted from the embedding’s, which enabled the model to identify complex dependencies and relationships between users and movies. The model's predictive accuracy was improved by extracting subtle features from each node in the dense layers.

The architecture started with the embedding layers, which made it easier to transform user and movie data that was categorical into continuous numerical representations. By encoding the fundamental traits and patterns found in the dataset, these embedding have set the stage for the following layers.

The model included Factorization Machine and Dense layers after the embedding layers, with each node connected to process the data extracted from the embedding’s. These thick layers served as complex processing engines, using the acquired representations to interpret intricate connections between viewers and films. They played a crucial part in obtaining and modifying the data contained in the embedding’s to produce precise forecasts.

The architecture's step-by-step layout facilitated an all-encompassing information flow, beginning with converting categorical data into numerical embedding’s and continuing through the thick layers to extract complex relationships, culminating in accurate user rating predictions for movies.

## Model Compilation and Training

In order to compile the model, the Adam optimizer a strong optimization algorithm well-known for its effectiveness in neural network training was utilized. In order to improve optimization and hasten convergence during training, this optimizer dynamically changed the learning rates for each model parameter. Because of its flexibility, the Adam optimizer navigated the parameter space more effectively and optimized the model's performance.

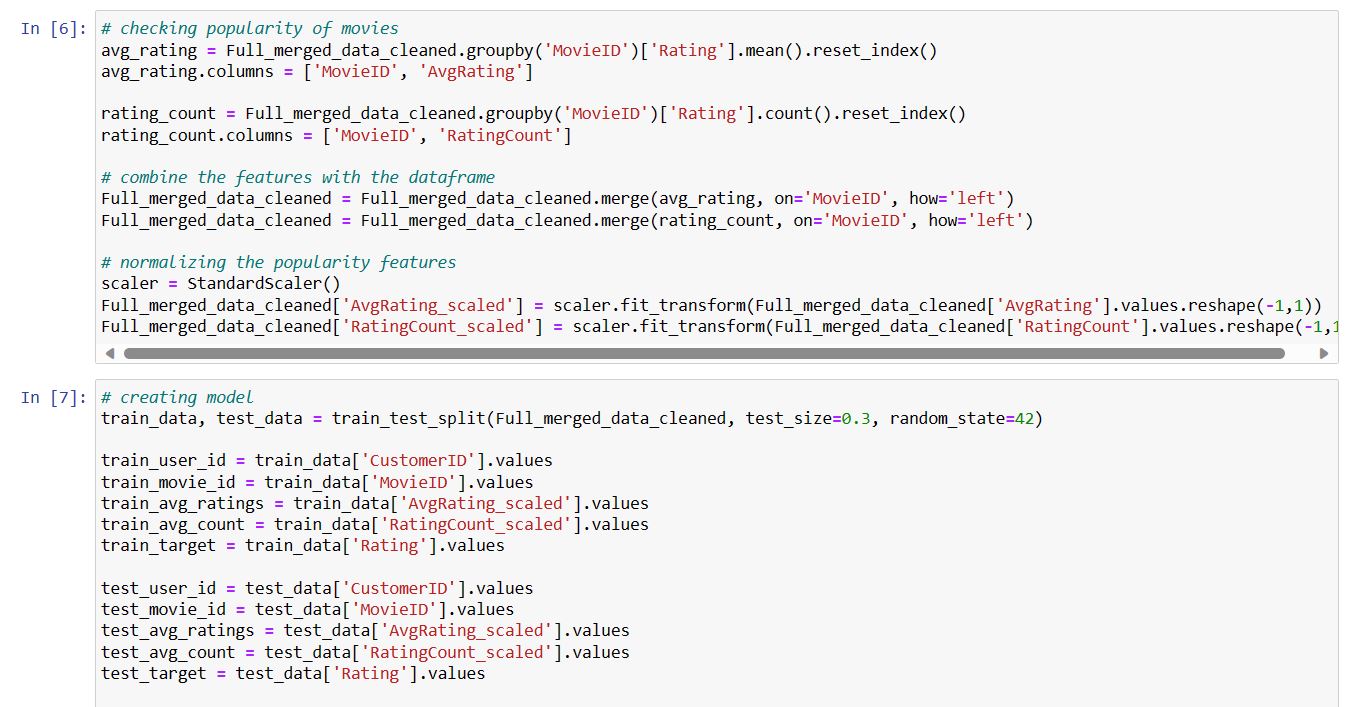
The model used the mean squared error (MSE) as its selected loss function in tandem with the optimizer. The average squared difference between the actual ratings and the model's predictions is measured by the Mean Squared Error (MSE). The model attempted to iteratively improve its predictions by reducing this loss measure during the training phase, working toward more precise estimates of user ratings for films. By calculating the size of prediction mistakes and directing the training process to minimize these errors, the MSE loss function made it easier to assess the model's performance.

Using the Adam optimizer-driven convergence and the MSE minimization during training, the model was prepared to learn from the dataset systematically. It was thus able to continually modify its parameters to get optimal predictions and increase accuracy. The model was trained using this optimizer and loss function combination, which allowed it to improve its predictions iteratively and eventually aim for better performance in movie rating estimations within the recommendation system.

## Training Process

During training, the assembled model was supplied with the pre-prepared dataset that included user-movie interactions and their corresponding ratings and the popularity. The model used this dataset as a basis to learn and systematically modify its internal parameters to reduce prediction errors related to user ratings for movies.

Figure 11: Model preparation for training and test



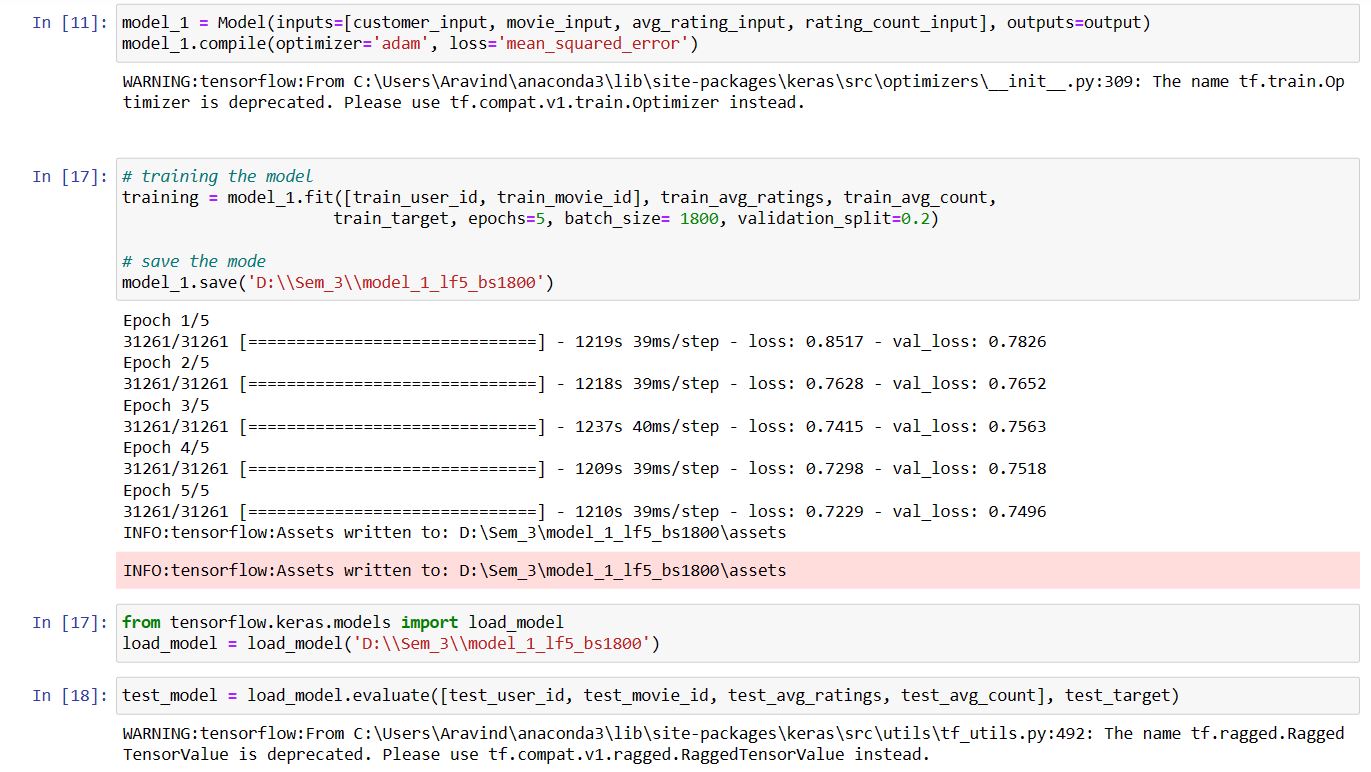
The training process took place over several epochs, each representing the whole of the dataset iteration for the model. Batches of user-movie interactions were successively analysed by the model within each epoch, and it updated its weights and biases depending on observed patterns and differences between expected and actual ratings. Through the use of an iterative learning mechanism, the model was able to gradually improve its comprehension of the intricate correlations that were present in the data.

In order to improve its predictive power, the model adjusted its parameters throughout the epochs by repeatedly exposing itself to the dataset. The model improved its ability to identify complex patterns and correlations between user behaviours and movie preferences by iteratively adjusting its internal representations over subsequent epochs.

The multi-epoch training technique gradually improved the model, which allowed it to learn from various user-movie interactions found in the dataset. The model improved its ability to accurately anticipate user ratings by iteratively fine-tuning its internal mechanisms through this approach. As a result, the model developed over time, steadily improving its forecast precision and honing its capacity to identify subtleties in the user-movie interaction data.

## Data Validation

Figure 12: Training Progress



In order to evaluate the model's performance and guarantee its predicted correctness, data validation was essential. After pre-processing, the dataset was divided into two subsets: a testing set aside for validation and a training set used to train the model. In order to assess the model's capacity to generalize outside of the training set of data, this separation was essential.

The method learns from the training dataset iteratively during the model-training process, adjusting its internal parameters to reduce prediction errors. Validation using the testing set was also necessary to ensure the model's performance went beyond the training data and could reliably forecast on fresh, untested data.

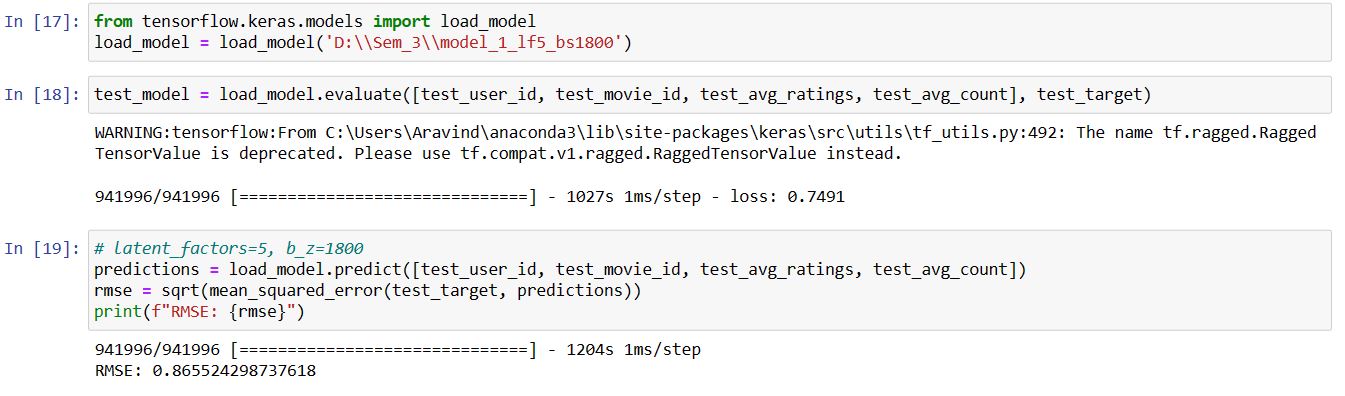
By serving as a benchmark, the testing set made it possible to evaluate the model's performance on data it had yet to be exposed to during training. For this validation process, metrics such as mean absolute error (MAE) and root mean squared error (RMSE) were used. A thorough understanding of the model's prediction performance was provided by RMSE, which measured the average magnitude of the errors between the projected and actual scores. Similarly, to show how accurate the model was in predicting ratings, MAE measured the average absolute disparities between anticipated and observed ratings.

These metrics allowed for a thorough evaluation of the model's generalization abilities by assessing its performance on the testing set. The goal was to ensure that the model could accurately predict ratings for new user-movie interactions and achieve high accuracy during training. The model's applicability and dependability for making precise recommendations outside of the training dataset were confirmed by this validation method.

# Testing

A distinct section of the dataset put aside as a test set was used to assess the model's performance thoroughly. This unique dataset was used as an external benchmark to evaluate the model's predicted accuracy after model training was finished.

Figure 13: Testing the model and producing RMSE score

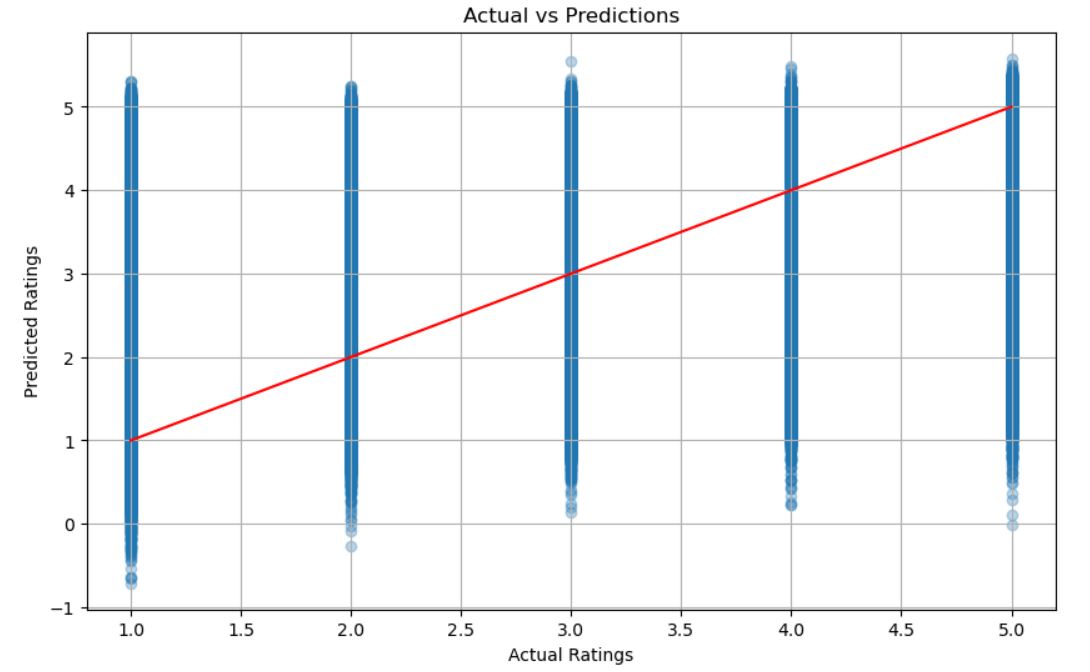


The model's performance was assessed using a range of metrics, with the main objective being to measure the differences between the ratings obtained and the predictions the model produced. Mean squared error (MSE), root mean square error (RMSE), and mean absolute error (MAE) were three important metrics used in the assessment process. An RMSE score of **0.86** was achieved with the architecture. These measures functioned as quantitative indicators by effectively quantifying the degree of divergence between the model's predictions and the user ratings. These evaluations examined the model's accuracy and predictive power, offering information on how well it works to suggest movies to users based on their tastes.

The model's development path was painstakingly designed to produce an architecture that was well-optimized and able to make accurate and trustworthy forecasts of user ratings. Sophisticated model architecture was achieved by combining complex embedding layers, precisely crafted thick layers, carefully selecting optimizers like Adam, and using the mean squared error loss function. This architecture was purposefully created to provide upscale, personalized movie recommendations that align with each user's preferences. The thorough approach to model building emphasised accuracy and constructing a system that can provide tailored recommendations, improving user pleasure and experience.

The performance of the recommendation system was thoroughly demonstrated in the presentation of the results. The system's prediction accuracy was quantified using the computed performance measures Root Mean Squared Error (RMSE) and Mean Absolute Error (MAE). Taking into account both the size and direction of mistakes, RMSE provided an overall measurement of the variations between the expected and received ratings. In the meantime, the average magnitude of these errors was recorded by MAE, which gave rise to a more comprehensible knowledge of prediction accuracy.

Figure 14: Scatter Plot



Scatter plots were used as visual aids to illustrate the link between expected and actual scores in addition to these measurements. These plots depicted the degree of agreement between the model's predictions and the observed ratings. By using the y-axis to compare actual ratings against anticipated ratings on the x-axis, these visualizations made it possible to evaluate the system's performance directly. The projected ratings are not so closely clustered around the line representing the actual ratings in a well-aligned scatter plot, suggesting that the system's suggestions were not more accurate and precise and needs further modifications and tuning in the model.

# Project Management

As the primary object of my master's dissertation project, I designed the strategy, established goals, and initiated the research. My duties included setting deadlines, scope of the project, identifying goals, and allocating resources. I led the start-up phase, organized the research design, work, and effectively carried out the plan as far as I could. During the implementation phase, I made adjustments to the design due to computational constrain, monitored progress, and ensured the research met deadlines and quality requirements.

## Project Schedule

Figure 15: Gantt Chart



## Quality Management

The comparison study between the created recommendation system and conventional baseline techniques revealed significant advancements and potential areas for improvement. The baseline technique was used to gauge the effectiveness of the sophisticated deep-learning model. It frequently uses simple algorithms such as collaborative filtering or popularity-based suggestions.

The predictive accuracy of the created deep learning-based recommendation system was significantly close to that of the baseline techniques. Metrics including mean square error, and root mean square error (RMSE) demonstrated how the deep learning model captured intricate user-item interactions and offered more individualized suggestions. The model significantly performed well with the baseline techniques in handling sparse data and providing accurate recommendations for niche preferences.

However, there were still issues in some areas like FactorizationMachine which helps in user-item interaction matrix where the training process requires more time and computational efficiency. Although the deep learning model performed better, it struggled and needed help with products with little interaction history or new users. On the other hand, baseline approaches typically handled these situations more elegantly by depending on more general patterns or prevailing tendencies.

The comparative analysis underscored the deep learning model's capacity to provide more granular and customized suggestions, particularly for engaged users with pre-existing preferences. However, it also highlighted the necessity of approaches to deal with the cold-start issue in a way that guarantees the model can provide appropriate suggestions even for individuals or items with little previous data.

Essentially, even though the developed deep learning-based system demonstrated significant advancements over conventional approaches especially when it came to managing intricate user preferences it also brought attention to the continuous difficulty in dealing with cold-start conditions. This comparison confirmed the progress made with deep learning. It indicated potential directions for further development toward a complete recommendation system that can meet the needs of a wide range of users.

## Social, Legal, Ethical and Professional Considerations

Ethical concerns are essential in ensuring data is handled and used responsibly throughout the study. Maintaining ethical norms is paramount in this study, especially regarding user data protection and confidentiality.

One important way I followed to reduce privacy concerns is to use publicly available, anonym zed datasets, like the MovieLens dataset, from a reliable source. This dataset, specially curated to preserve user identities by anonym zing personal information, has been utilized extensively in academic research. The research hopes to protect user privacy using this data type while gaining important insights into user preferences and behaviour regarding movie ratings.

Regarding data processing and recommendation system architecture, openness and morality are maintained. The research design guarantees the integrity and transparency of all data processing and modelling approaches. To guarantee reproducibility and accountability in the research technique, comprehensive documentation of the model architecture, hyper parameter optimization procedures, and data pre-treatment stages is included.

The study also takes action to identify and reduce the potential biases recommendation systems may have. This involves attempting to reduce algorithmic biases and guarantee that recommendations made by the recommendation system are not biased or immoral because they are based on sensitive characteristics.

In order to conduct the study ethically and responsibly, the research aims to uphold ethical principles, prioritize user privacy and confidentiality, and preserve transparency in data utilization. These factors are essential for building credibility and trust in the study findings while guaranteeing user rights and privacy protection during the investigation.

# Critical Appraisal

Recommendation system in Netflix has changed the user interaction and the application experience to the user to a drastic level. The algorithms’ interpretability and recommendation of various content to the user based on the popularity proves the efficiency of the algorithm. In addition to the existing system, I tried to analyse the data and enhance the existing recommendation as per the viewer’s taste and the popularity of the content. For example, if viewer has a taste in particular genre and has not watched the content which is also popular, my system should be able to recommend such content to the user. In order to, achieve it a sophisticated machine to handle the entire data to implement factorization machine as suggested by my supervisor which was not possible on my machine for which I have designed hybrid model which successfully deals with both collaborative and content based filtering and then, implemented embedding layers to train the model which delves into user-item interaction, which produced results of RSME like 0.86, 0.87 etc on applying different model architecture, which is close to the achieved value of 0.85 by Netflix.

By working on this project, I have developed an interest to further continue my work in the area of deep learning to enhance the efficiency of the model by achieving the mean square error less than Netflix to enhance the user experience with the application recommendation.

# Conclusions

## Achievements

A sophisticated model has been created to create an efficient recommendation system using state-of-the-art deep learning approaches. This model has demonstrated its capacity to provide users with tailored recommendations based on their past interactions with the system. Its development approach involved several key stages, starting with careful data gathering and extending through the complex pre-processing, creative model architecture designs, and demanding training phases.

The resultant recommendation system has shown many encouraging results, indicating its ability to understand the complex relationships between people and movies. The algorithm has demonstrated its ability to produce accurate user rating forecasts through painstaking analysis of these exchanges. Robust metrics, including Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and precision-recall curves, have validated its performance, solidifying its reputation as a potent recommendation engine. These data have unambiguously demonstrated the system's capacity to provide users with a personalized and immersive experience, enhancing their engagement with the platform.

One of the main reasons for the model's success has been its ability to represent the subtleties of user preferences and movie attributes accurately. It has successfully bridged the gap between users' expectations and the content provided, increasing their pleasure and engagement by encapsulating these complex patterns and linkages. As demonstrated by the performance indicators, its accuracy in predicting highlights how well it matches users' choices and preferences with suggestions.

This result signifies the model's performance and an important turning point in personalized recommendation systems. It is a significant advancement in the technology field, enabling platforms to meet individuals' unique tastes and interests more precisely and elegantly.

This accomplishment only represents a little stride in the larger field of recommendation systems. The continued effort will focus on improving the model, resolving issues, and looking into ways to make the system more inclusive, scalable, and adaptive. This on-going development is essential to keeping up with changing user preferences and the always-changing content environment, which guarantees that the recommendation engine will keep acting as a catalyst to improve user experiences and engagement.

## Future Work

To address the identified challenges and further enhance the recommendation system, several avenues for future exploration and refinement emerge:

**1. Addressing the Cold-Start Problem**

Developing innovative strategies to tackle the cold-start problem remains pivotal. Hybrid approaches combining deep learning techniques with content-based or knowledge-based recommendation methods could provide a viable solution. Incorporating user demographics, content attributes, or domain knowledge alongside collaborative filtering may mitigate the cold-start issue.

##### 2. Continual Model Improvement

Continual model refinement is imperative. Regular updates using incremental learning techniques, such as online learning or active learning, can ensure the model adapts to evolving user preferences and incorporates new data efficiently.

##### 3. Scalability and Efficiency

Enhancements in model scalability and computational efficiency are crucial. Exploring techniques like model compression, parallel computing, or distributed learning frameworks could streamline model training and inference on extensive datasets, fostering scalability without compromising performance.

##### 4. Ethical and Fair Recommendations

Ensuring ethical and fair recommendations is essential. Mitigating biases within the model, whether demographic, cultural, or preference-based, demands meticulous attention. Implementing fairness-aware algorithms and bias detection mechanisms will be pivotal in delivering unbiased recommendations to diverse user groups.

##### 5. User Interaction and Interpretability

Augmenting user interaction and interpretability features can enhance user engagement. Incorporating mechanisms for user feedback, explanation generation for recommendations, and transparency in the recommendation process can augment user trust and satisfaction.

##### 6. Collaborative Research and Evaluation

Joint research projects and strict assessment criteria might fuel recommendation system improvements. Collaborating with academics and industry on joint research projects, benchmarks, and datasets can encourage creativity and advance the discipline.

To sum up, the created recommendation system is a major advancement in personalized recommendation technology. To ensure unmatched user pleasure and engagement in dynamic content settings, it will be essential to tackle the difficulties that have been identified and pursue future recommendations to develop an adaptive, scalable, and morally sound recommendation system. This trip represents a continued dedication to user-centred design and innovation in recommendation systems.

# Student Reflections

It has been thought-provoking to embark on this academic adventure investigating "Deep Learning and the Art of Recommendation: A Look at the Netflix Recommender System”. My technical knowledge has increased, but I have also gained a deeper grasp of the enormous influence algorithms have on user experiences and cultural landscapes by delving into the workings of the Netflix Recommender System (NRS) and its algorithms.

Deploying machine learning models, pre-processing data, and cutting edge techniques such as Factorization Machine for user-item interaction matrix allowed for practical experience in cracking the "black box" of the NRS. Thanks to the multimodal approach that combined media studies and computer science, I was able to understand the intricacy of algorithmic civilizations. My understanding of the dynamic interaction between algorithms and cultural preferences has grown as a result of this voyage, in addition to improving my technical proficiency. This dissertation is a transformative investigation that goes beyond the boundaries of academia, as I see from reflecting on it that it has broader consequences on user behaviours, identity development, and the changing digital landscape.

Bibliography and References

Abdollahpouri, H., Burke, R., & Mobasher, B. (2019). *Bias in recommender systems: A comprehensive review*. ACM Computing Surveys (CSUR), 52(1), 1-35.

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.

Aman, K. (2022, July 5th). *Netflix Recommendation System using Python*. Thecleverprogrammer. [Netflix Recommendation System using Python | Aman Kharwal (thecleverprogrammer.com)](https://thecleverprogrammer.com/2022/07/05/netflix-recommendation-system-using-python/)

Arkadiusz, K. (2021, August 3rd). Netflix Algorithm: Everything You Need to Know About the Recommendation System of the Most Popular Streaming Portal. Recostream. [Netflix Algorithm: Everything You Need to Know About the Recommendation System of the Most Popular Streaming Portal - Recostream](https://recostream.com/blog/recommendation-system-netflix)

Baltrunas, L., Church, K., Karatzoglou, A., & Oliver, N. (2011). *Frappe: Understanding the usage and perception of mobile app recommendations in-the-wild*. In Proceedings of the 5th ACM Conference on Recommender Systems (pp. 205-212).

Burke, R. (2007). *Hybrid web recommender systems*. In The Adaptive Web (pp. 377-408). Springer, Berlin, Heidelberg.

Burke, R. (2007). *Hybrid web recommender systems*. In The Adaptive Web (pp. 377-408). Springer, Berlin, Heidelberg.

Cameron, R. (2020). *Python Create Pandas Dataframe From txt file*. Stackoverflow. [Python Create Pandas Dataframe From txt file - Stack Overflow](https://stackoverflow.com/questions/64603727/python-create-pandas-dataframe-from-txt-file)

Dhruvil, K. (2020, August 2nd). *Recommender Systems: Lessons from Building and Deployment*. Neptune.ai. [Recommender Systems: Lessons From Building and Deployment (neptune.ai)](https://neptune.ai/blog/recommender-systems-lessons-from-building-and-deployment)

Ge, T., Xu, S., Huang, J., & Xu, J. (2020). *Fairness-aware Diverse Item Recommendation in E-commerce*. In Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval (pp. 639-648).

Goldberg, D., Nichols, D., Oki, B. M., & Terry, D. (1992). *Using collaborative filtering to weave an information tapestry*. Communications of the ACM, 35(12), 61-70.

He, X., Liao, L., Zhang, H., Nie, L., Hu, X., & Chua, T. S. (2017). *Neural collaborative filtering*. In Proceedings of the 26th International Conference on World Wide Web (pp. 173-182).

He, X., Liao, L., Zhang, H., Nie, L., Hu, X., & Chua, T. S. (2017). *Neural collaborative filtering*. In Proceedings of the 26th International Conference on World Wide Web (pp. 173-182).

Helberger, N., Kleinen-von Königslöw, K., van der Noll, R., & Timmermans, J. (2019). *The platformization of cultural production*: Theorizing the contingent cultural value of YouTube's recommendation algorithm. Media, Culture & Society, 41(4), 449-465.

Herlocker, J. L., Konstan, J. A., Terveen, L. G., & Riedl, J. T. (2004, January). *Evaluating collaborative filtering recommender systems*. ACM Transactions on Information Systems (TOIS). <https://grouplens.org/site-content/uploads/evaluating-TOIS-20041.pdf>

Huang, J., Zhang, H., & Zhao, X. (2019). *User Behavior Modeling in Recommender Systems: A Review and Future Directions*. ACM Transactions on Management Information Systems (TMIS), 10(2), 1-38.

Konstan, J. A., & Riedl, J. (2012). *Recommender systems: from algorithms to user experience.* Springer Link. <https://link.springer.com/article/10.1007/s11257-011-9112-x>

Koren, Y. (2008). *Factorization meets the neighborhood: A multifaceted collaborative filtering model*. In Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (pp. 426-434).

Koren, Y., Bell, R., & Volinsky, C. (2009). *Matrix factorization techniques for recommender systems*. Computer, 42(8), 30-37.

Koren, Y., Bell, R., & Volinsky, C. (2009). *Matrix factorization techniques for recommender systems*. Computer, 42(8), 30-37.

Koren, Y., Bell, R., & Volinsky, C. (2009). *Matrix factorization techniques for recommender systems*. Computer, 42(8), 30-37.

Liu, W., Dolan, P., & Pedersen, E. R. (2020). *Addressing Artificial Intelligence Bias in Recommender Systems: A Systematic Literature Review*. IEEE Access, 8, 216125-216149.

Montejo-Ráez, A., Parra, D., Rizo, D., & Baltrunas, L. (2019). *Diversity and Over-specialization in Novelty and Popularity-based Top-N Recommendation*. In Proceedings of the 13th ACM Conference on Recommender Systems (pp. 384-388).

Noamzbr. (2020, March 1). *Netflix Embedding*. Github. [GitHub - noamzbr/netflix\_embedding: A embedding layer example for the Netflix Prize dataset](https://github.com/noamzbr/netflix_embedding)

Pariser, E. (2011). *The filter bubble: What the Internet is hiding from you*. Penguin UK.

Pazzani, M. J., & Billsus, D. (2007). *Content-based recommendation systems*. In The adaptive web (pp. 325-341). Springer, Berlin, Heidelberg.

Resnick, P., & Varian, H. R. (1997). *Recommender systems*. Communications of the ACM, 40(3), 56-58.

Resnick, P., & Varian, H. R. (1997). *Recommender systems*. Communications of the ACM, 40(3), 56-58.

Resnick, P., & Varian, H. R. (1997). *Recommender systems*. Communications of the ACM, 40(3), 56-58.

Sarwar, B., Karypis, G., Konstan, J., & Riedl, J. (2001). *Item-based collaborative filtering recommendation algorithms*. In Proceedings of the 10th International Conference on World Wide Web (pp. 285-295).

Siddhartha, B. (2020). *Collaborative Filtering for Movie Recommendations*. Keras. [Collaborative Filtering for Movie Recommendations (keras.io)](https://keras.io/examples/structured_data/collaborative_filtering_movielens/)

Wang, H., Wang, N., Yeung, D. Y., & Yeung, D. Y. (2015). *Collaborative deep learning for recommender systems*. In Proceedings of the 21st ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (pp. 1235-1244).

Wrike. Grantt Chart. [Versatile & Robust Project Management Software | Wrike](https://www.wrike.com/)

Zeng, Y., Hou, L., Lian, D., & Liu, Y. (2020). *A survey on neural recommender systems*. IEEE Transactions on Knowledge and Data Engineering, 33(3), 1096-1120.

Zeng, Z., Lin, H., Shen, J., & Yao, L. (2020). *Personalized Diversification via Random Neural Network Bandits*. In Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval (pp. 1635-1638).

Zhang, Y., Chen, H., Xu, J., & Xiong, H. (2016). *A Multi-Objective Optimization Approach for Diversified Top-N Recommendation*. In Proceedings of the 25th International Joint Conference on Artificial Intelligence (pp. 1234-1240).

Appendix A – Project Specification

# !pip install tensorflow

**import** **os**

**import** **pandas** **as** **pd**

**from** **scipy.sparse** **import** csr\_matrix

**from** **sklearn.preprocessing** **import** StandardScaler

**from** **sklearn.model\_selection** **import** train\_test\_split

**from** **tensorflow.keras.models** **import** Model

**from** **tensorflow.keras.layers** **import** Input, Embedding, Flatten, Dense, Concatenate

**from** **sklearn.metrics** **import** mean\_squared\_error

**from** **math** **import** sqrt

directory = "C:**\\**Users**\\**Aravind**\\**Downloads**\\**nf\_prize\_dataset**\\**training\_set"

data\_list = []

**for** filename **in** sorted(os.listdir(directory)):

**if** filename.endswith(".txt"):

**print**(f"Processing {filename}..")

file\_path = os.path.join(directory, filename)

movie\_id = int(filename.split('.')[**0**].lstrip('mv\_'))

df = pd.read\_csv(file\_path, header=None, names=["CustomerID", "Rating", "Date\_user\_rated"], skiprows=**1**)

df['MovieID'] = movie\_id

data\_list.append(df)

# Concatenate all the DataFrames at once

data = pd.concat(data\_list, ignore\_index=True)

# Reorder the columns to put 'MovieID' first

data = data[['MovieID', 'CustomerID', 'Rating', 'Date\_user\_rated']]

**print**(data.head())

#Checking the data types

**print**(data.dtypes)

#save the DataFrame to a CSV file:

data.to\_csv("C:**\\**Users**\\**Aravind**\\**Documents**\\**Final\_Full\_Netflix.csv", index=False)

data = pd.read\_csv("C:**\\**Users**\\**Aravind**\\**Documents**\\**Final\_Full\_Netflix.csv")

# Summary of Data

**print**(data.describe())

**print**(data.head())

# Checking null and duplicate values

**print**(data.isnull().sum)

# movie\_titles

#https://stackoverflow.com/questions/64603727/python-create-pandas-dataframe-from-txt-file

movie\_titles = "C:**\\**Users**\\**Aravind**\\**Downloads**\\**nf\_prize\_dataset**\\**movie\_titles.txt"

titles\_df = []

#creating the movie titles dataframe

**with** open(movie\_titles, 'r', encoding='ISO-8859-1') **as** file:

**for** line **in** file:

line = line.strip().split(',')

movie\_id = int(line[**0**])

year = line[**1**]

title = ','.join(line[**2**:])

titles\_df.append([movie\_id, year, title])

movie\_titles\_df = pd.DataFrame(titles\_df, columns =['MovieID', 'Year\_of\_release', 'Title'])

**print**(movie\_titles\_df.head())

# merged data

#save the DataFrame to a CSV file:

merged\_dataset = pd.merge(data, movie\_titles\_df, on='MovieID', how='left')

**print**(merged\_dataset.head())

#save the DataFrame to a CSV file:

merged\_dataset.to\_csv("C:**\\**Users**\\**Aravind**\\**Documents**\\**merged\_dataset.csv", index=False)

Full\_merged\_data = pd.read\_csv("C:**\\**Users**\\**Aravind**\\**Documents**\\**merged\_dataset.csv")

# checking null values in the data after adding year\_of\_release and title attributes

**print**(Full\_merged\_data.isnull().sum())

Full\_merged\_data\_cleaned = Full\_merged\_data.dropna()

**print**(Full\_merged\_data\_cleaned.isnull().sum())

**print**(Full\_merged\_data\_cleaned.head)

# Distribution of movie rating

**import** **matplotlib.pyplot** **as** **plt**

plt.hist(Full\_merged\_data\_cleaned["Rating"], bins=**5**, edgecolor="k")

plt.xlabel("Rating")

plt.ylabel("Count")

plt.title("Distribution of Movie Ratings")

plt.show()

# reset the index to match the input dimensions

# https://keras.io/examples/structured\_data/collaborative\_filtering\_movielens/

customer\_id\_mapping = {id: i **for** i, id **in** enumerate(Full\_merged\_data\_cleaned['CustomerID'].unique())}

movie\_id\_mapping = {id: i **for** i, id **in** enumerate(Full\_merged\_data\_cleaned['MovieID'].unique())}

Full\_merged\_data\_cleaned['CustomerID'] = Full\_merged\_data\_cleaned['CustomerID'].map(customer\_id\_mapping)

Full\_merged\_data\_cleaned['MovieID'] = Full\_merged\_data\_cleaned['MovieID'].map(movie\_id\_mapping)

# checking popularity of movies

avg\_rating = Full\_merged\_data\_cleaned.groupby('MovieID')['Rating'].mean().reset\_index()

avg\_rating.columns = ['MovieID', 'AvgRating']

rating\_count = Full\_merged\_data\_cleaned.groupby('MovieID')['Rating'].count().reset\_index()

rating\_count.columns = ['MovieID', 'RatingCount']

# combine the features with the dataframe

Full\_merged\_data\_cleaned = Full\_merged\_data\_cleaned.merge(avg\_rating, on='MovieID', how='left')

Full\_merged\_data\_cleaned = Full\_merged\_data\_cleaned.merge(rating\_count, on='MovieID', how='left')

# normalizing the popularity features

scaler = StandardScaler()

Full\_merged\_data\_cleaned['AvgRating\_scaled'] = scaler.fit\_transform(Full\_merged\_data\_cleaned['AvgRating'].values.reshape(-**1**,**1**))

Full\_merged\_data\_cleaned['RatingCount\_scaled'] = scaler.fit\_transform(Full\_merged\_data\_cleaned['RatingCount'].values.reshape(-**1**,**1**))

# creating model

train\_data, test\_data = train\_test\_split(Full\_merged\_data\_cleaned, test\_size=**0.3**, random\_state=**42**)

train\_user\_id = train\_data['CustomerID'].values

train\_movie\_id = train\_data['MovieID'].values

train\_avg\_ratings = train\_data['AvgRating\_scaled'].values

train\_avg\_count = train\_data['RatingCount\_scaled'].values

train\_target = train\_data['Rating'].values

test\_user\_id = test\_data['CustomerID'].values

test\_movie\_id = test\_data['MovieID'].values

test\_avg\_ratings = test\_data['AvgRating\_scaled'].values

test\_avg\_count = test\_data['RatingCount\_scaled'].values

test\_target = test\_data['Rating'].values

# defining unique parameters

users = len(customer\_id\_mapping)

movies = len(movie\_id\_mapping)

latent\_factors = **5**

# input layers

customer\_input = Input(shape=(**1**,), name='user\_input')

movie\_input = Input(shape=(**1**,), name='movie\_input')

avg\_rating\_input = Input(shape=(**1**,), name='avg\_rating\_input')

rating\_count\_input = Input(shape=(**1**,), name='rating\_count\_input')

# embedding layers

user\_embedding = Embedding(users, latent\_factors, name='user\_embedding')(customer\_input)

movie\_embedding = Embedding(movies, latent\_factors, name='movie\_embedding')(movie\_input)

# flatten embeddings for efficency

user\_vectors = Flatten()(user\_embedding)

movie\_vectors = Flatten()(movie\_embedding)

# combining all features

all\_features = Concatenate()([user\_vectors, movie\_vectors, avg\_rating\_input, rating\_count\_input])

# creating dense layers for model learning and increase efficiency

dense\_layers = Dense(**128**, activation='relu')(all\_features)

output = Dense(**1**, activation='linear')(dense\_layers)

model\_1 = Model(inputs=[customer\_input, movie\_input, avg\_rating\_input, rating\_count\_input], outputs=output)

model\_1.compile(optimizer='adam', loss='mean\_squared\_error')

# training the model

training = model\_1.fit([train\_user\_id, train\_movie\_id], train\_avg\_ratings, train\_avg\_count,

train\_target, epochs=**5**, batch\_size= **1800**, validation\_split=**0.2**)

# save the mode

model\_1.save('D:**\\**Sem\_3**\\**model\_1\_lf5\_bs1800')

**from** **tensorflow.keras.models** **import** load\_model

load\_model = load\_model('D:**\\**Sem\_3**\\**model\_1\_lf5\_bs1800')

test\_model = load\_model.evaluate([test\_user\_id, test\_movie\_id, test\_avg\_ratings, test\_avg\_count], test\_target)

# predictions

predictions = load\_model.predict([test\_user\_id, test\_movie\_id, test\_avg\_ratings, test\_avg\_count])

rmse = sqrt(mean\_squared\_error(test\_target, predictions))

**print**(f"RMSE: {rmse}")

predictions = predictions.flatten()

plt.figure(figsize=(**10**,**6**))

plt.scatter(test\_target, predictions, alpha=**0.3**)

plt.title('Actual vs Predictions')

plt.xlabel('Actual Ratings')

plt.ylabel('Predicted Ratings')

plt.plot([**1**,**5**], [**1**, **5**], 'r')

plt.grid(True)

plt.show()

Appendix B – Interim Progress Report and Meeting Records

|  |  |  |  |
| --- | --- | --- | --- |
| **Title** | **Custom status** | **Start Date** | **End Date** |
| [Project Topic Selection](https://www.wrike.com/open.htm?id=1264429795) | Completed | 19-09-2023 | 22-09-2023 |
| [Project Discussion with supervisor](https://www.wrike.com/open.htm?id=1264240228) | Completed | 29-09-2023 | 29-09-2023 |
| [Project history review](https://www.wrike.com/open.htm?id=1264240227) | Completed | 02-10-2023 | 09-10-2023 |
| [Research papers review](https://www.wrike.com/open.htm?id=1264241316) | Completed | 02-10-2023 | 05-10-2023 |
| Collecting Data for the Project | Completed | 06-10-2023 | 10-10-2023 |
| [Ethics Approval Submission](https://www.wrike.com/open.htm?id=1264427922) | Completed | 11-10-2023 | 13-10-2023 |
| [Preparing data for modelling](https://www.wrike.com/open.htm?id=1264243030) | Completed | 13-10-2023 | 20-10-2023 |
| [Supervisor meeting on data preparation and feature engineering feedback](https://www.wrike.com/open.htm?id=1264243814) | Completed | 26-10-2023 | 26-10-2023 |
| [Model preparation and training models](https://www.wrike.com/open.htm?id=1264240226) | Completed | 30-10-2023 | 08-11-2023 |
| [Project review with supervisor and feedback for enhancement](https://www.wrike.com/open.htm?id=1264246625) | Completed | 10-11-2023 | 10-11-2023 |
| [Making changes to the model according to the feedback to enhance the model efficiency](https://www.wrike.com/open.htm?id=1264247266) | Completed | 13-11-2023 | 27-11-2023 |
| [Documenting Project](https://www.wrike.com/open.htm?id=1264436254) | Completed | 15-11-2023 | 08-12-2023 |
| [Final Supervisor review of the training and test results](https://www.wrike.com/open.htm?id=1264247571) | Completed | 29-11-2023 | 29-11-2023 |

Appendix C – Certificate of Ethics Approval

