UNIVERSITY OF CAPE TOWN
IYUNIVESITHI YASEKAPA • UNIVERSITEIT VAN KAAPSTAD

UNIVERSITY OF CAPE TOWN

MASTER'S DISSERTATION

A Review-Aware Multi-Modal Neural Collaborative Filtering Recommender System

---

Author:
Pavan SINGH

Supervisor:
Assoc Professor Ian DURBACH
Dr. Allan E CLARK

A dissertation presented for the degree of
Master of Science Advanced Analytics

from the

Department of Statistical Sciences

January 12, 2024

Chapter 1

Introduction

In an age where there is an exponentially increasing volume of data being produced, the ability for a user to get the information they seek has become ever more challenging (SINTEF, 2013). The abundance of information which gives an overwhelming number of options to a user when making a decision is known as the information overload problem (Bawden and Robinson, 2020). With the vast amount of information available, it can be challenging for users to find items or products that meet their preferences or needs. Recommender systems are tools which directly aim to address the challenge of information overload (O'Donovan and Smyth, 2005). The general idea behind a recommender system is to narrow down the perceived available options and present to a user a limited set of personalised choices (recommendations) based on said user's preferences, behaviour and or other relevant factors (O'Donovan and Smyth, 2005). In this light, a recommender system can be seen as a filtration tool which greatly washes out undesirable results and brings forth content desired or more relevant to a user's current interests and needs. In this way they are able to help to reduce information overload and make the decision-making process easier and more efficient. A lucrative byproduct of this is an improved overall user experience, increased engagement and user satisfaction. Given these outcomes, recommender systems have become a highly researched area over recent years (Seth and Sharaff, 2022). Recommender systems are a crucial tool in the arsenal of e-commerce platforms which try to help consumers navigate through an abundance of product options. To this end, recommender systems are capable of enhancing the customer experience by showing products that they are inclined to want and thus also boost sales of these product. The apparent effectiveness of these systems has led to this surge in research in this domain. In this thesis, we aim to directly contribute to this area of research by developing a recommender system that incorporates data from multiple modalities and investigates the potential impact of incorporating product review text and sentiment in improving the accuracy of recommendations.

The specific type of recommender system used in this thesis is known as a neural collaborative filtering. This chapter will introduce recommender systems in Section 1.1 along with the different recommender paradigms which have gained a lot of attention in this study area. In Section 1.2 we draw light to the research problem and objectives of this paper. The specific research questions and significance of this paper is explained in Section 1.3 in detail before concluding the chapter by explaining the structure of the remaining sections of this paper in Section

1.1 Recommender Systems: definition and types

A recommender system is a type of information filtering system that seeks to predict user preferences or interests by generating *recommended* items such as products, services or content that the user is likely to be interested in (Seth and Sharaff, 2022). "Item" is the general term used to denote what the system recommends. Instead of sifting through irrelevant information and products, users are presented with content and products that are more likely to be of interest to them. Recommender Systems have quickly become a necessity, given that users cannot search through millions of content to connect with products, services or knowledge (i.e., items) that is important to them.

Recommender systems deal with two types of information: characteristic information, which includes details about the objects or products, such as keywords or categorisation; and user-item interactions, which encompass data like user ratings or likes and so on. With respect to the types of recommender systems available, the data (ratings data, item characteristics data, etc.) also plays an important role in determining which recommender system would be effective. In the field of recommender systems, there exist three main branches: collaborative filtering, content-based filtering and hybrid systems (Thorat et al., 2015).

Collaborative filtering, probably the most extensively implemented type of recommender, is based on user-item interactions' data and helps you find what you like by looking for users, or items, who are similar to you (Thorat et al., 2015). So, the algorithm is based on using a similarity measure to determine how much a user will like an item. At a high level, there are two types of collaborative filtering algorithms which have been proposed in the literature: memory-based algorithms, which rely on computing the similarity between users or items in order to make recommendations, and model-based algorithms, which rely on building a model or mathematical representation of the data to make recommendations. Both memory-based and model-based algorithms have their strengths and weaknesses which we discuss further later on.

Content based filtering, by contrast, is based on characteristic information and works by understanding the underlying features (or characteristics) of each item (Thorat et al., 2015). There is no general high-level categories for which content-based filtering algorithms are universally recognised as like we have for Collaborative based filtering (model-based vs memory based). Regardless, the idea is that a content-based filtering algorithm analyses the features of an item and then recommends other items with similar features. These features could be the attributes of an item, like the colour or style of a product, or they could be keywords, like the genres or artist names of a song.

We often find in practice that these two types of recommender system techniques (collaborative and content based) are combined to form **hybrid systems**. These use both types of information, with the idea to leverage the strengths of each individual method to improve the accuracy and relevance of the recommendations, whilst also avoiding problems that are generated when working with just one kind of system (Thorat et al., 2015). These hybrid modelling approaches have been shown to outperform individual algorithms in many cases, however they also carry with it additional complexities in terms of model building and efficiency (Cano and Morisio, 2017). Furthermore, whilst combining different models can often produce a more accurate model, there are other approaches that can be used to improve the accuracy and relevance of recommendations. One such approach is multi-modal recommender systems, which leverage multiple types of data or information sources to make recommendations. Figure 1 shows at a very high level, the main types of recommender systems and the popular subcategories within these branches. Note that hybrid recommender, again, do not necessarily have recognised high level categories, however some approaches can be bundled into some high-level categories such as weighted approaches.

Multi-modal refers to the integration of different types or modes of data in a system (Truong et al., 2021). In the context of recommender systems, multi-modal simply refers to the use of multiple types of data to make recommendations. By combining information from different modalities, such as text, images, audio, and video, multi-modal recommender systems can provide a more comprehensive understanding of user preferences and offer more personalised recommendations (Truong et al., 2021). Using multi-modalities also poses the opportunity to alleviate data sparsity[1] by leveraging or including auxiliary information that may encode additional clues on how users consume items. Examples of such data (referred to as modalities) are social networks, item's descriptive text, or customer product reviews (Truong et al., 2021). Studies on multi-modal recommender systems have quickly become a hot topic in the field as many understood the potential upside of incorporating multiple data types into a prediction algorithm (Truong et al., 2021). In fact, multi-modal recommendation systems have become the industry standard and are widely used in across many domains as companies try to personalise their

[1] the situation where the available data is highly incomplete, resulting in numerous missing values in the user-item interactions. This occurs due to the vast number of items and limited user interactions, making accurate predictions for less observed items challenging

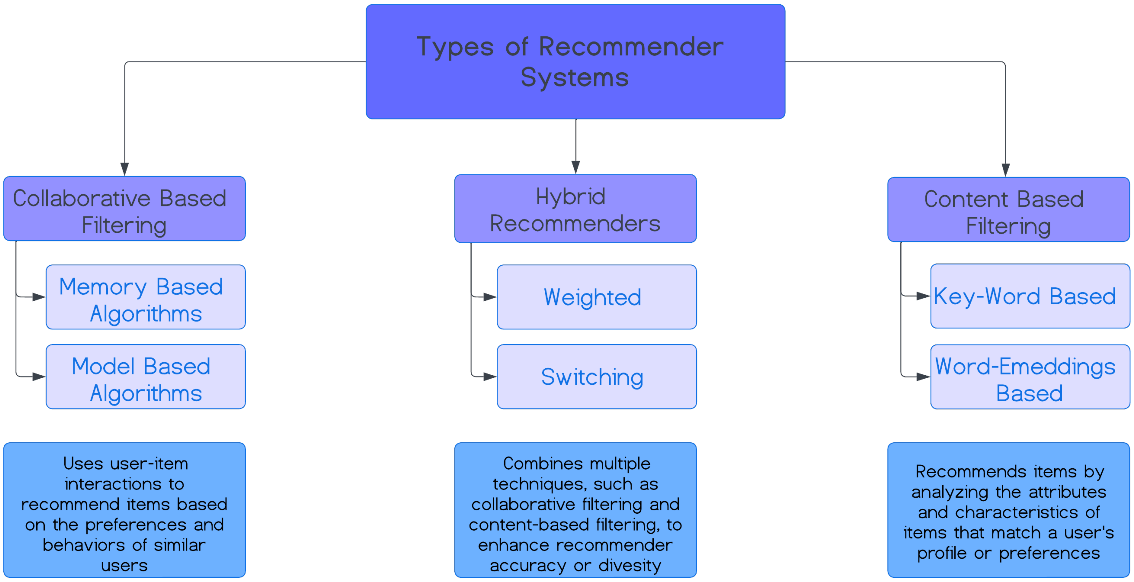

Figure 1.1: The different types of recommender systems.

products and services to better meet the needs and preferences of their customers (Liu et al., 2023). For example, in e-commerce, a multi-modal recommender system can take advantage of product images and descriptions, user reviews, and purchase history to make recommendations (Liu et al., 2023). Beyond multi-modalities recommender systems have evolved to incorporate deep learning technologies. Traditional recommender systems have shown to be effective in many cases, however the recent advances in deep learning over the past decade have opened up new opportunities for improving the accuracy and personalisation of recommendations.

Deep learning is a a subset of machine learning that deals with models that are comprised of multiple layers (hence the “deep” in Deep Learning). In the context of recommender systems, deep learning algorithms can be used to effectively extract complex patterns and relationships from large amounts of data (He et al., 2017). This is particularly attractive since the goal is to predict user preferences based on past behaviours or interactions. We shall explore the application of deep learning in the context of our recommender system by employing a neural collaborative framework which essentially generalises the matrix factorisation approach (which is extensively used in collaborative filtering) to improve the accuracy of recommendations (He et al., 2017). Amazon, amongst many other internet giants, also use neural networks in their recommendation engine to make product recommendations (Steck et al., 2021).

Although these systems can easily be mistaken as simply a tool for information retrieval and data discovery - in industry their importance is unwavering. With the tremendous amount of data available online, recommender systems have become the forefront of large internet corporations research and investment (Steck et al., 2021).

1.2 Research Problem and Objectives

The aim of this thesis is to directly enhance the predictive accuracy of traditional recommender systems by developing a neural collaborative filtering (NCF) model that incorporates data from multiple modalities (types), exploiting explicit numeric product ratings and product review text data as well user review sentiments. The idea is using this auxiliary information we can aid the NCF system account for the nuances and complexities of user preferences that may be expressed through textual data. We build on the existing work done by He et al. (2017) on NCF systems and several other works (Srifi et al. (2020); Zhang et al. (2014)) which laid the groundwork to effectively incorporate auxiliary information into collaborative filtering systems (including user reviews). Ultimately, thesis seeks to investigate the potential of incorporating product review text and review text sentiment as additional sources of information to improve the accuracy of recommendations.

This is achieved by building a neural collaborative filtering recommender system which processes the explicit ratings data and is also supercharged with textual features (product reviews and review sentiment analysis) to form a multi-modal system. Building this neural network architecture (for collaborative filtering) enables the system to learn a non-linear mapping between the input features (i.e., item attributes) and the target output (i.e., user preferences). Through comparative evaluation with benchmark recommender models, the efficacy of the proposed multi-modal recommender will be assessed, with a particular focus on its ability to provide predict accurate ratings for unseen items. To this end, a number of different models will be trained, evaluated and compared to one another in an effort to establish the performance benefits of different models. The research questions presented in Section 1.3 will guide this process.

Ultimately, this thesis aims to contribute to the development of a more effective and efficient recommender system that can potentially leverage the additional textual information provided by users for products which can potentially aid in generating recommendations tailored to the unique preferences and needs of individual users.

1.3 Research Questions and Significance

This paper seeks to explore the impact of incorporating product reviews and review sentiments into a recommender system model whilst also evaluating the performance of the neural collaborative filtering model. Recommender systems have become pivotal to the way companies interact and sell their products to their consumers (Steck et al., 2021). A recommender system's ability to accurately predict consumer interests and desires on a highly personalised level make them a very valuable tool for content and product providers like Amazon, Google amongst other technology giants. The key research questions addressed in this paper are discussed and explained below.

Here we are interested in determining whether integrating user reviews into a recommendation algorithm leads to more accurate and relevant product suggestions for users. Another way of looking at this is examining the accuracy of a multi-modal recommender system compared to that of a single-modal recommender system - i.e., a recommender that relies on only one type of data. User reviews for products could potentially provide additional insights into user preferences or product features that are not captured by the metadata. However, reviews may also introduce noise or biases into the recommendation algorithm which can lead to less accurate suggestions. Findings from this study could provide further scope for incorporating review text into product recommendation designs or highlight potential pitfalls to avoid when integrating user reviews. The study also may prove to provide further support for multi-modal recommender systems.

2. **How does incorporating review sentiment as well as review text in a recommender system impact the accuracy of the system's recommendations? Additionally, how does the accuracy of a recommender system vary based on the sentiment analysis techniques used to interpret the reviews?**

Incorporating sentiment analysis into a recommender system introduces an avenue for the integration of additional features that can contribute to enhancing recommendation accuracy. Sentiment analysis, as a technique to assess the emotional tone and polarity of textual content such as product reviews (Medhat, Hassan Korashy, 2014), offers a means of extracting valuable insights from user-generated content. By incorporating sentiment analysis outcomes alongside traditional recommendation factors, the system gains access to a richer set of attributes that can potentially capture nuanced user preferences and sentiments.

The impact of sentiment analysis on accuracy of recommender systems provides scope to further utilising review text in possibly improving recommendation accuracy. In addition, by comparing the accuracy of the system when different sentiment analysis techniques are used, the study can identify which techniques are most effective in improving recommendation accuracy. Potential findings from this study could provide insights into the most effective sentiment analysis techniques for collaborative-based recommender systems and inform the design of more accurate and effective recommendation algorithms. Conversely, the study could highlight potential challenges and limitations of using sentiment analysis in a recommender system, such as accuracy issues due to sarcasm, ambiguity, or variations in language use.

3. **How does the performance of the collaborative-based filtering recommender system using neural collaborative filtering compare to that of popular benchmark recommender systems?**

The general idea is that incorporating deep learning into collaborative filtering model should alleviate the fragility of simpler, more traditional collaborative filtering models in capturing use preference profiles and as such improve accuracy of recommendations, however this is not guaranteed. This is another key area of interest in our study - examining the potential capacity of neural collaborative filtering system to outperform other popular recommender systems. In evaluating our recommender system, we assess its predictive accuracy against other benchmark models. The architecture and technical details of all the developed models are provided in detail in Chapter 4 - the methodology. Deep learning models are by design more complex and often have greater computational expense and slower processing than more traditional models under collaborative filtering. By investigating the potential advantage these deep learning systems have could highlight the potential trade-off between the level of desired complexity and performance. Whilst this will be domain specific and depend on the needs and resources available, the study will contribute to providing additional insight into this field.

4. **What are the potential trade-offs of incorporating product reviews into a recommender system, such as increased complexity or potential biases in the recommendations?**

Product reviews can often contain large amounts of unstructured data which is cause for concern for recommender systems needing to quickly process and integrate this information in their algorithm. Another potential concern may be the bias inherent in user reviews. These will be discussed in depth later on in this paper. However, findings from this study could prove to be useful in highlighting the potential benefits and drawbacks of incorporating product reviews into a recommender system. Furthermore, it could open up paths for discussion on potential strategies for mitigating the trade-offs of using reviews in recommendation algorithms, such as developing more sophisticated sentiment analysis techniques.

These key research questions will be referenced greatly throughout the whole paper. The significance of the findings from the study ultimately provides further support for possible future design and implementations in this space. More specifically, findings will directly contribute to the existing literature on recommender systems and provide insights into the effectiveness and limitations of incorporating product reviews and sentiment analysis features in collaborative models, as well as the trade-offs associated with these approaches.

1.4 Dissertation Outline

In this chapter we introduced what recommender systems are and the importance of them as tools to help users find the content and services they seek (or may want). Specifically, recommenders alleviate the information overload problem directly and enable users to find information as well as discover items that align to their preferences. We have also described the aim of this thesis, and the research questions it will seek to answer in 1.2. It has also introduced the primary collaborative-based multi-modal recommender system which we will be developing and examining in this thesis.

Chapter 2 is a detailed literature review which provides context for this research by first addressing the history and application of recommender systems in E-commerce. This history will start with a brief exploration of the fundamental concepts in recommender systems which have remained relevant since their inception in Section 2.1. We shall also describe the immediate benefit and advantage that recommender systems have provided for the E-commerce industry. From here, in Section 2.2, we shall explore the main recommender system paradigm of concern for this paper: collaborative filtering. Here, a brief history of the algorithms are discussed as well as the latest breakthroughs. This will lead into a discussion on the deep learning era in Section 2.3, where recent advances are of particular importance. Additionally, we explore the prevalence of text and its applications for recommender systems in 2.4, other types of recommenders in section 2.5 and finally the key evaluation methods used in recommender system research in Section 2.6. The chapter closes with addressing the various limitations and challenges faced by collaborative filtering and recommender systems as a whole. Chapter 3 addresses all matters relating to data. In order to critically examine our collaborative filtering recommender system and assess impacts of the textual features, it is necessary to have a source of product data. Our source is the Amazon Product Review dataset publicly available and is discussed in further detail in Section 3.1. In addition to describing (Section 3.2) and explaining the data collection process (Section 3.3), we explore the dataset in Section 3.4. Finally, we summarise our findings and partitioning process, as well as how we shall use the data and select features for our recommendation system in Sections 3.5 and 3.6 respectively. Chapter 4 looks at the methods used in our research. It provides all the technical details necessary to understand the models utilised and trained in Chapter 5. We begin the chapter (4) by detailing out our overall modelling approach in Section 4.1 before diving into the details. The groundwork for this will be built in Section 4.2, where the basic architecture and training procedure for neural collaborative filtering will be discussed in length. Additionally, Section 4.3 discusses the various benchmark models that shall be built to assess the performance of the neural collaborative model. Finally, Section 4.4 details the evaluation criteria that shall be employed to assess our recommender models. Chapter 5 documents the observed results (Section 5.1) as well as a detailed discussion of them (Section 5.2) with frequent reference to the research questions raised from this chapter. Chapter 6 will summarise and conclude the dissertation. This shall be done by first summarising the overall results in conjunction with our research question in Section 6.1 and then we also address the limitations and make suggestions for future work in Section 6.2

Chapter 2

Literature Review and Related Work

Chapter 1 briefly introduced the key concepts around recommender systems, the problem they solve and the main ideas relevant to this thesis. Chapter 2 will build on this by providing historical context through an overview of recommender systems with a focus on e-commerce applications, reviewing literature done on recommenders incorporating similar features and characteristics relevant to very own. We divide this overview into several different sections, each addressing key discoveries and background information relevant to this thesis. To this end, the sections covered will focus on work which harnessed either collaborative based filtering, deep learning, text analysis or a combination of these approaches all within the context of recommender systems.

2.1 Recommender Systems: history and applications

Recommender systems have evolved significantly since their inception, becoming a pivotal component of contemporary information retrieval and personalised content delivery. Due to their success, they play a key part in many platforms offerings services or showcasing items to users across various different domains. In this section, we first look at the history and various applications of recommender systems (2.1.1). and then we dial in on looking at the literature behind recommender systems within E-commerce (2.1.2) specifically.

2.1.1 History and Applications

The roots of recommender systems can be traced back to the GroupLens project initiated at the University of Minnesota in the early 1990s, introducing the groundbreaking concept of collaborative filtering. This method involved generating recommendations by analysing users' historical preferences and behaviours, as exemplified in the development of MovieLens, an influential movie recommendation system based on user ratings (Konstan et al., 1997; Huang et al., 2004). A couple years later, a landmark application of recommender systems emerged with Amazon's adoption of personalised product recommendations to enhance user experience and boost sales. Leveraging collaborative filtering techniques, Amazon introduced the "Customers who bought this also bought" feature, marking a pioneering approach to personalised product discovery (Linden et al., 2003). Somewhere during the late 1990s, content-based filtering was introduced as a viable alternative to collaborative-based filtering, which further broadened the capabilities of recommender systems by proposing items to be recommended based on their inherent attributes and user preferences (Balabanović et al., 1997). The early 2000s witnessed a surge in interest and research on recommender systems, leading to the exploration of diverse methodologies and variations of collaborative-based or content-based (and sometimes both) recommender systems (Burke Robin, 2002). An important milestone in recommender systems occurred in 2006 when Netflix, initiated an open competition offering a $1000000 prize for an algorithm that could improve on the accuracy of their system, Cinematch[1]. This competition spurred significant interest and development in recommender system applications and use cases, with the prize claimed in 2009 by a group of three researchers (Bennett and Lanning, 2007).

[1] a recommendation algorithm developed by Netflix in 2000, which aimed to provide personalised movie recommendations to users based on their viewing history and preferences

Over the subsequent decade, recommender systems have become critical for the success of major Internet companies such as Netflix, Amazon, Facebook, Baidu, and Alibaba (Chen et al., 2012). The widespread adoption of internet-based services further fuelled the proliferation of recommender systems, extending their use across diverse domains. In broad terms, when a wide variety of products (or items rather) exist and customers differ from each other, personalised recommendations can assist in delivering suitable content to the respective individuals. Applications of such systems range from tourism, encompassing hotels, restaurants, and parks (Yang and Hwang, 2013; Loh et al., 2004); advertising (Cheung et al., 2003); business and retail (Ghani and Fano, 2002); medical diagnosis (Perez-Gallardo et al., 2013); and music selection (Bogdanov et al., 2013).

Table 2.1 shows some popular e-commerce sites using recommender systems and high-level description of what these systems recommend to users of the platform. Streaming platforms like Netflix leverage recommender systems to personalise content recommendations, significantly impacting user engagement and retention (Gomez-Uribe et al., 2015). Social media platforms like Facebook and LinkedIn employ recommender systems to curate users' news feeds and suggest connections (Aivazoglou et al., 2020). Recommendations were used so extensively by Amazon that a Microsoft Research report estimated that 30

Table 2.1: Content Recommendations on Different Platforms

Site/Platform	What is Recommended
Netflix	Movies, TV shows
Amazon	Books, Fashion, and other products
Facebook	Friends, posts, articles
LinkedIn	Posts, articles, jobs
Spotify	Music, podcasts

Ultimately, recommender systems continue to evolve and play a crucial role in delivering personalised content across various industries, adapting to the dynamic needs of users and businesses alike. The ongoing development and integration of these systems underscore their enduring significance in shaping the landscape of information retrieval and content delivery. We now dial our interest towards focusing on E-commerce domain specifically.

2.1.2 Recommenders in E-commerce: Amazon case study

The exponential growth of the internet has fundamentally transformed the operations of companies, particularly within the realm of e-commerce. E-commerce, characterised by the buying and selling of goods and services over the internet, has become platform ubiquitous with consumers exploring and purchasing products (Schafer et al., 1999). Users' navigating through these platforms are confronted with a multitude of decisions, prompting questions such as "Which product should I purchase?" or "What brand should I choose?" For a successful e-commerce platform, efficiency in showcasing relevant products is paramount, and recommender systems have emerged as a pivotal tool in achieving this goal (Schafer et al., 1999). Recommender systems have evolved from novelties to indispensable tools that shape the world of e-commerce (Schafer et al., 1999). They are proven to have significant impacts on sales, diversity, customer retention, and revenue generation (Linden et al., 2003). Today, prominent e-commerce websites, including Amazon, Netflix, eBay, Alibaba, and Etsy, leverage recommender systems to assist users in discovering products for purchase (Aivazoglou et al., 2020). Recommenders are identified as playing a vital role in turning browsers into buyers, addressing the challenge of users abandoning a platform when their desired item is not readily visible. In fact, research has demonstrated that users often exhibit a tendency to abandon a system when their desired item does not appear within the initial 5 or 10 search results and said user tries again on another system (Linden et al., 2003). However, recommender systems have the capability to assist these customers in discovering the products they wish to buy. Moreover, recommender systems enhance cross-selling efforts by suggesting supplementary products during the checkout process. For example, during the checkout process, a website could suggest additional items based on the products already present in the shopping cart. A study by McKinsey Company (2021) showed that recommender systems can contribute to a 10-30 percent increase in cross-selling revenue. Cultivating customer loyalty in a competitive online landscape is another notable outcome, as recommender systems establish value-enhanced relationships between platforms and users (Tsai and Hung, 2012). Platforms invest in tools or systems able to comprehend their user's preferences, and then accordingly translate this understanding into action, by offering tailored services or items that cater to a user's needs. In return, users tend to favour platforms that align well with their requirements.

Looking at the history of recommender systems within E-commerce, the paper by Schafer et al. (1999) offers a comprehensive exploration of how early E-commerce platform systems employ their technologies, exemplifying their application through notable e-commerce businesses like Amazon. The paper unveils key insights by scrutinising recommender system implementations and their contributions to platform profitability. Several of these recommendation strategies mentioned by Schafer et al. (1999) are still used today and stand out as revenue drivers, some of which are discussed below. The "Similar Items" approach prompts customers to explore related products based on their previous purchases, aligning with their expressed interests. "Text Comments" contribute to revenue generation by allowing users to discover desired items and read impartial reviews, leveraging the credibility associated with positive feedback. The "Average Ratings" strategy simplifies decision-making by offering numerical rankings, providing users with a convenient gauge of item quality and facilitating the conversion of browsers into buyers. Lastly, "Top-N Recommendations" personalise user experiences by offering lists of top unrated items, eliminating distractions caused by irrelevant products and focusing on items of genuine interest, thereby converting browsers into buyers and increasing exposure to the vendor's offerings. These strategies collectively contribute to the effectiveness of recommender systems in optimising the e-commerce landscape. Many of these strategies are still offered on Amazon's platform today. Amazon, a pioneer in the field, say that their recommender systems play a critical role in leveraging the "long tail" concept, contributing to substantial profits from their products with infrequent purchases (Leino et al., 2007). While individually rare, the collective presence of such items can yield substantial profits. For instance, Amazon attributes a noteworthy portion of its sales, ranging from 20 to 40 percent, to products that do not fall within its top 100,000 best-selling items (Brynjolfsson et al., 2003). The E-commerce giant's approach involves dynamically personalising the online store for each customer, aligning with the concept of having "2 million stores on the web" as articulated by Jeff Bezos, the CEO of Amazon (Linden et al., 2003). Amazon's technical implementation of recommender systems involves applying a variety of recommender algorithms, such as item-based collaborative filtering. Although not pioneered by Amazon, item-based collaborative filtering was definitely popularised by them. They developed these recommenders to scale to massive data sets and produce high-quality recommendations in real time (Smith and Linden, 2017). These recommender systems extend across various pages on Amazon's platform, including the homepage, search results page, and shopping cart, offering personalised recommendations based on customers' past behaviour and preferences. The paper by Linden et al. (2003) discusses the item-based collaborative filtering algorithm in great detail.

The paper by Schafer et al. (1999) paper offers a comprehensive exploration of how early E-commerce platform systems employ their technologies, exemplifying their application through notable e-commerce businesses like Amazon. The paper unveils key insights by scrutinising recommender system implementations and their contributions to platform profitability. Several of these recommendation strategies mentioned by Schafer et al. (1999) are still used today and stand out as revenue drivers, some of which are discussed below. The "Similar Items" approach prompts customers to explore related products based on their previous purchases, aligning with their expressed interests. "Text Comments" contribute to revenue

generation by allowing users to discover desired items and read impartial reviews, leveraging the credibility associated with positive feedback. The "Average Ratings" strategy simplifies decision-making by offering numerical rankings, providing users with a convenient gauge of item quality and facilitating the conversion of browsers into buyers. Lastly, "Top-N Recommendations" personalise user experiences by offering lists of top unrated items, eliminating distractions caused by irrelevant products and focusing on items of genuine interest, thereby converting browsers into buyers and increasing exposure to the vendor's offerings. These strategies collectively contribute to the effectiveness of recommender systems in optimising the e-commerce landscape.

Ultimately, recommender systems serve as communication bridges between products and users, enhancing customer experiences and increasing the likelihood of a sale. Amazon's success stands as a testament to the effectiveness of quality recommender systems, continuously adapting to user preferences and fostering customer loyalty. Within E-commerce, recommenders greatly impact the revenue generation and cross-selling ability of platforms by directly being able to understand customer preferences and show or highlight items aligning well to these user preferences. Having established the background and history of Recommender systems (with a focus on E-commerce), we now look toward one of the primary types of recommender systems - Collaborative-based Filtering.

2.2 Collaborative Filtering

The basic concept behind Collaborative Based Filtering methods is that if a user likes a certain item, he is most likely to like a similar item and if two users are similar, they're most likely to have a common interest. The seminal work titled "Using collaborative filtering to weave an information tapestry" by Paul Resnick and Hal R. Varian (1997) is considered foundational in the field of recommender systems, playing a pivotal role in popularising collaborative filtering techniques for recommenders. While not the absolute first use of recommender systems, it brought attention to the concept's potential applications, especially in personalised recommendations. In 1992, the Tapestry system (Goldberg et al., 1992) introduced collaborative filtering, leveraging users' experiences to tailor recommendations without requiring external information about items or users. Specifically, collaborative filtering is defined as a recommendation technique used to personalise the experience of users through recommendations tailored to their interests, leveraging the experiences of other users with similar profiles (Herlocker et al., 2004). It does this without need for exogenous information about either items or users (Ricci and Shapira, 2011). It comprises two primary approaches: memory-based, recommending based on similarity, and model-based, recommending by developing a predictive model based on user ratings (Symeonidis et al., 2008).

Both approaches have strengths. Both practical experience and related research have reported that memory-based algorithms (a.k.a. nearest-neighbor algorithms) present excellent performance, in terms of accuracy, for multi-value rating data. On the other hand, model-based algorithms are efficiently handle scalability to large data sets (Symeonidis et al., 2008). This is not to say that these methods have no weaknesses. Traditional collaborative filtering methods face two primary challenges in a growing e-commerce landscape: these are concerns related to real-time scalability and recommendation quality. More details about scalability are discussed later on in section 2.7.3. Regardless, a major appeal of collaborative filtering methods in general is that it is domain free, yet it can address data aspects that are often elusive and difficult to profile using other recommender techniques such as content filtering (Koren, 2009). In fact, collaborative filtering is recognised as the most successful recommender system, able to produce user-specific recommendations based on patterns of ratings or usage without the need for additional information (Tsai and Hung, 2012).

The collaborative filtering process begins by identifying a cohort of users or items with overlaps in interactions with the target user, pooling together items associated with similar users or items, and presenting recommendations to the user. However, it is crucial to note that collaborative filtering is computationally expensive, with different approaches facing varying performance challenges when scaling (Linden et al., 2003). The following subsections will outlay the details, background and literature into the two primary branches of collaborative filtering: memory-based and model-based approaches.

2.2.1 Memory-Based Algorithms

As mentioned, memory-based methods are techniques within collaborative filtering systems which involve recommending items based on finding users who have similar tastes in items as the target user or those who like similar items as the target user. One prominent family of methods within memory-based collaborative filtering is neighbourhood methods, which focus on creating a “neighbourhood” of users or items based on their similarity to a target user or item.(Adamopoulos, 2013). Specifically, these methods make recommendations by first identifying similar users or items to a target user or item and then utilises the preferences of these entities to make personalised recommendations. We look at the literature and history behind the two most prominent neighbourhood methods; user-based and item-based collaborative filtering.

The user-based approach for neighbourhood methods is one of the most popular applications of memory-based collaborative filtering, largely due to its simple implementation (SOURCE). In the mid-1990s, collaborative filtering predominantly followed a user-based methodology, where the initial step involved searching for other users with similar preferences, such as users having comparable purchase patterns. That is, identifying a set of customers whose purchased and rated items overlap with the target user’s preferences (perhaps they purchased similar items or rated common item’s similarly in the past). By aggregating items from these similar customers, the algorithm eliminates those already purchased or rated by the user and recommends the remaining items (Smith and Linden, 2017). The original GroupLens system (Resnick et al., 1994) implemented a user-based collaborative filtering algorithm in 1994, using users’ similarities to identify a neighbourhood of nearest users. Numerous improvements to user-based algorithms have since been suggested (Breese et al., 1998; Herlocker et al., 1999), enhancing the predictive capabilities of these methods. The original GroupLens system used Pearson correlations to weight user similarity, used all available correlated neighbour’s, and computed a final prediction by performing a weighted average of deviations from the neighbour’s mean. The Ringo music recommender (Shardanand and Maes, 1995) and the Bellcore Video Recommender (Hill et al., 1995) represented significant advancements beyond the original GroupLens algorithm. Ringo’s innovation involved achieving superior performance by calculating similarity weights through a constrained Pearson correlation coefficient. In contrast, the Bellcore Video Recommender employed a Pearson correlation to weigh a random sample of neighbors. It selected the most relevant neighbours, employing a full multiple regression on them to generate predictions. Since, “similarity” is a key component in this (and all memory-based) approach, the way in which similarity is measured had garnered significant research in the early beginnings of neighbourhood-based approaches. Breese et al. (1998) conducted an empirical analysis of various neighbourhood-based collaborative filtering algorithms and specifically looked at comparing similarity weighting techniques, namely Pearson correlation and cosine vector similarity. The findings indicated that Pearson correlation demonstrated superior performance, although subsequent research suggests potential equivalence with cosine similarity (Pennock et al., 2013). User-based collaborative filtering is still synonymous with collaborative filtering till this day. Due to its simplicity and ease of understanding, it is often employed in simple recommendation tasks, however the algorithm like any memory-based approach (as we will soon discuss in greater detail in Section 2.7.3) struggles to scale well.

The other neighbourhood based method is item based collaborative filtering which was first introduced in the early 2000s and was popularised by Amazon - it is based on the items’ similarities for a neighbourhood generation of nearest items (Sarwar et al., 2001). It is a prominent technique under neighbourhood methods, where it evaluates a user’s preference for an item based on the ratings of "neighbouring" items by the same user. Amazon has successfully employed item-item collaborative filtering since 2003, generating real-time, scalable, and high-quality recommendations. It's successful implementation has led a lot of research to be focused on this technique. In contrast to user-based approaches, item-to-item collaborative filtering focuses on finding similar items rather than similar customers, offering a unique recommendation approach (Linden et al., 2003). Specifically, the item-to-item collaborative filtering algorithm builds a similar-items table to recommend highly correlated items, leveraging the cosine measure for similarity calculations. It excels in recommendation quality even with limited user data and has gained widespread use across platforms like YouTube and Netflix due to its simplicity, scalability, explainability, and immediate updates based on new customer information (Davidson et al., 2010). However, like User-based approach, Item-based collaborative filtering has its flaws, such as the risk of trapping users in a "similarity hole" by offering overly similar recommendations (Rashid et al., 2001). The method, like user-based approach, also struggles to scale well to larger volumes of data.

Further details about these two memory-based approaches will be explored in Chapter 4, as they serve as important comparative benchmarks in our analysis. At this point, we can highlight that all memory-based algorithms face scalability challenges when dealing with large volumes of data. To address this, dimensionality reduction techniques have been proposed to balance the trade-off between accuracy and execution time of collaborative filtering algorithms (Sarwar et al., 2000). More details on this limitation (scalability) shall be explored in greater detail later on in this chapter, in Section 2.7.3).

2.2.2 Model-Based Algorithms

Model-based algorithms is the second branch of collaborative-based filtering systems. In contrast to memory-based methods, model-based algorithms harness the overall dataset of ratings to build predictive models, often employing statistical or machine learning techniques (Adomavicius and Tuzhilin, 2005). So, the key distinction between model-based techniques and memory-based approaches lies in their approach to generate predictions. Model-based techniques rely on models to derive predictions whilst memory-based methods generate predictions from the weighted similarity of preferences among users or items (Adomavicius and Tuzhilin, 2005).

There are a variety of different model-based algorithms. Bayesian models (Chien and George, 1999), probabilistic relational models (Getoor and Sahami, 1999), linear regressions (Sarwar et al., 2001), and Latent Dirichlet Allocation (Marlin, 2003) are among the diverse array of model-based methods. Some of the most successful realisations of model-base methods (and collaborative filtering in general) are latent factor models, namely matrix factorisation (Koren et al., 2009). Matrix factorisation is characterised by transforming both items and users into the same latent factor space. These (latent factor) type models, seek to reveal latent features explaining observed ratings, characterising items and users based on a certain predefined number of factors inferred from rating patterns. Preferences for a particular item are determined by a small number of underlying factors or latent variables, capturing essential characteristics influencing user and item preferences. Once these latent factors are learned, the recommender system can provide personalised recommendations for each user. In the context of movies, the identified factors could encompass straightforward aspects like categorizing films into genres like comedy or drama, gauging the level of action, or assessing suitability for children. Additionally, these factors might delve into more ambiguous facets such as the depth of character development or the presence of unique qualities, or even include entirely unexplainable dimensions. Regarding users, each factor signifies the extent to which a user favours movies that align with the specific characteristics identified in the corresponding movie factor.

Matrix factorisation, particularly highlighted by its success in the Netflix competition, has surged in popularity, demonstrating superiority over classic nearest neighbour techniques for product recommendations (Koren et al., 2009). The acclaim for matrix factorisation stems from its exceptional scalability, effective handling of sparse data, and the ability to incorporate additional information, such as implicit feedback, temporal effects, and confidence levels (Koren et al., 2009). However, this is not to say that matrix factorisation techniques (and additionally all model-based) approaches do not have their own set of limitations. Model-based approaches, like all collaborative filtering approaches, struggle when it comes to recommending items to new users (the cold start problem), and data sparsity (Adomavicius and Tuzhilin, 2005). Details of these limitations are discussed in greater detail in section 2.7. Now, having set the stage by discussing the background and different types of collaborative filtering, we shall begin to explore deep learning and the impact it has had on the recommender landscape.

2.3 Deep Learning in Recommender Systems

Over the past few decades, deep learning has achieved remarkable success in a diverse range of domains like computer vision and speech recognition. Both academia and industry have fervently embraced its application, driven by its ability to tackle complex tasks and deliver state-of-the-art results (Cherkassky et al., 2012; Clark et al., 1999).

The most basic deep learning model entails a multilayer feed-forward neural network which undergoes training, employing back-propagation or other supervised algorithms, to create a predictive statistical model for a specific input–output mapping (Zhang et al., 2019). The network learns from a set of examples, embedding information in connection weights, allowing it to generalise to examples beyond the training set. Effectively, the neural network serves as a flexible tool that can apply its learned knowledge to new data points, showcasing its ability to generalize beyond the specific examples encountered during training.

Despite the extensive success of deep learning in various domains, the exploration of deep neural networks in recommender systems had received limited attention. However, the more recent advancements (in the last decade) in machine learning and artificial intelligence, have paved the way for refining and enabling to recommender systems to first, integrate deep learning models, and second achieve more accurate and relevant recommendations (He et al., 2017). The successful integration and apparent advantageous performance of deep learning in recommenders has led to there being a significant increase in the number of research publications on deep learning-based recommendation methods. The majority of which have provided further strong evidence of the inevitable pervasiveness of deep learning in recommender system research (Zhang et al., 2019).

Traditionally, (as discussed in Section 2.2) the state of the art for collaborative filtering has predominantly entailed matrix factorisation (or some derivative of it) for modelling the interaction between user and item features. This involved applying an inner product on the latent features of users and items. Much research effort has been devoted to enhancing MF, such as integrating it with neighbor-based models (Koren, 2008), combining it with topic models of item content (Wang et al., 2015), and extending it to factorization machines (Rendle, 2010) for a generic modelling of features. Despite these numerous efforts to enhance matrix factorisation, its performance has been limited by the simplicity of the inner product interaction function, which may not capture the complexity of user interaction data. The inner product, which simply combines the multiplication of latent features linearly, may not be sufficient to capture the complex structure of user interaction data. One approach to address this limitation is increasing the number of latent factors, but this can lead to overfitting, particularly in sparse settings (Rendle, 2010). To tackle this challenge, He et al. (2017) proposed a neural network-based collaborative filtering (NCF) framework, replacing the inner product with a neural architecture capable of learning arbitrary functions from data (Koren, 2008). They further presented a general framework named NCF, short for Neural network-based Collaborative Filtering. In their paper, their NCF model, leveraging a multi-layer perceptron for non-linearities, demonstrated consistent and statistically significant improvements over state of art matrix factorisation approaches such as eALS and BPR and basic baselines such as ItemPop[2] and ItemKNN[3] on MovieLens and Pinterest datasets. Moreover, evidence from He et al. (2017) paper suggests that incorporating deeper layers of neural networks enhances recommendation performance, indicating a promising direction for further exploration. In addition to NCF, neural networks have been applied to develop Deep Matrix Factorisation, a hybrid technique merging matrix factorisation and deep learning. In this approach, a deep neural network is employed to factorise the user-item interaction matrix, creating low-dimensional representations for both users and items. However, it is essential to note that this method is computationally demanding. This stands in contrast to NCF, where the approach involves using neural networks to directly learn representations of users and items, eliminating the need for explicit factorisation of the interaction matrix.

In the contemporary landscape, numerous companies leverage deep learning to enhance the quality of their recommendations, showcasing a notable shift in the recommender system paradigm (Cheng et al., 2016; Covington et al., 2016; Okura et al., 2017). Covington et al. (2016) introduced a deep neural network-based recommendation algorithm tailored for video recommendations on YouTube. Cheng et al. (2016) proposed a wide and deep model for an App recommender system on Google Play. Okura et al. (2017) presented an RNN-based news recommender system designed for Yahoo! News. These models underwent rigorous online testing and demonstrated significant improvements over traditional counterparts, illustrating the transformative impact of deep learning on industrial recommender applications. Zhang et al. (2019) conducted a comprehensive review, providing a taxonomy of deep learning-based recommendation models and summarising the state of the art. This survey, encompassing over 100 studies, serves as an invaluable resource for understanding the evolution of deep learning within the realm of recommender systems and provides a great background for further reading.

Ultimately, we follow the approach done in He et al. (2017) for NCF, to completely replace the matrix factorisation-based approach—i.e., embracing a neural network-based collaborative filtering paradigm that excels in capturing intricate user-item interactions through using multi-layer network, offering non-linear transformations and enhanced expressiveness in our recommender system. This can be particularly useful when the relationships between input features and user preferences are complex and non-linear, which may be the case in many real-world scenarios. Furthermore, neural networks and their inherent layered structure make it easy to incorporate auxiliary information since each layer can process different aspects of the data. This provides an effective platform for which we can evaluate the influence of textual features within our NCF model, aligning with our specified research objectives. In keeping with this, we will delve into the history and literature of text within recommender systems, also exploring the relevance of text-based recommender systems.

2.4 Text Analysis in Recommender Systems

Recommender systems face inherent challenges, we have already made mention (in Section 2.2) of collaborative filtering algorithms struggling when encountering sparseness, scalability, and cold-start problems (Dang et al., 2021). Sparsity arises from vast data volumes, scalability from missing rating data, and cold-start when new users or items are introduced. These issues compromise the effectiveness of these algorithms, especially in scenarios of sparsity (insufficient user feedback), impacting recommendation accuracy. Another important shortcoming of collaborative filtering systems is that they do not capture the rationale for a user's rating, and thus can not holistically capture a target user's preference. This is a key (and unique) problem for collaborative filtering, which generally rely on numerical ratings. To tackle these challenges, integrating additional information to collaborative filtering models have been

[2] recommendation method based on popularity
[3] recommendation method based on item-item collaborative filtering using k-nearest neighbours, as discussed before

looked at, including tags (Zhou et al., 2003), geo-location (Hu et al., 2014) and user reviews (McAuley and Leskovec, 2013). In this paper, we focus on the integration of textual information such as user reviews as a promising solution the the aforementioned challenges. Reviews, a rich source of information, offer nuanced insights into user preferences. These reviews, however often in unstructured text, present a complex yet valuable pool of information (Shoja and Tabrizi, 2019). In the realm of recommender systems, written reviews play a crucial role in enhancing both user understanding by providing information on what the customer thinks about a product or service. Compared to numerical ratings, textual reviews provide richer, more fine-grained, nuanced, and reliable user preference information enabling the system to construct detailed user preference representations (Zhang et al., 2014; Chen et al., 2015). User reviews have been utilised to assist recommender systems in many domains, for movies (Diao et al., 2014), hotels (Musat et al., 2013) and e-commerce (McAuley and Leskovec, 2013).

Chen et al. (2015) identified key elements extracted from review texts that can be exploited by RS for efficient user modelling, including review words, review topics, and overall opinions. Leveraging TF-IDF measures, review words capture representative terms, while topics, detected through methods like latent dirichlet allocation, reveal aspects discussed in reviews. Overall opinions, reflecting sentiments, can be deduced using sentiment analysis techniques. Several works have used review texts and their related rich information like review words, review topics and review sentiments, for improving the rating-based collaborative filtering recommender systems. A recent addition to several e-commerce based recommender systems is to incorporate or augment their user profiles with review text for their collaborative filtering model. The paper by (Srifi et al., 2020) provides a detailed survey of recent works that integrate review texts and also discusses how these review texts are exploited in order to mitigate the main issues of the standard rating-based systems like sparsity and prediction accuracy problems. The paper reiterated that the elemetns that can be extracted and used for recommenders are the actual review words, topics and opinions. They also concluded that the inclusion of textual features was associated with positive outcomes in terms of recommender accuracy. Moreover, the paper by Shoja and Tabrizi (2019), looked at employing a deep neural network to extract deep features from the reviews-characteristics matrix to deal with sparsity, ambiguity, and redundancy. They then applied matrix factorization as the collaborative filtering method to provide recommendations. The experimental results from the paper (on the [Amazon.com](http://amazon.com/) dataset) demonstrated that the performance of the recommender system by incorporating information from reviews and produces recommendations with higher quality in terms of rating prediction accuracy compared to the baseline methods (which included matrix factorisation).

As mentioned, user sentiment or opinion is another way in which review text can be used for recommender systems. To extract this opinion from the review we use sentiment analysis, which to this day is one of the more conventional approaches towards the incorporating review text within recommenders (Kim et al., 2016). Traditional recommendation algorithms heavily depend on users' ratings, which are often inadequate and limited. In contrast, sentiment-based ratings derived from reviews offer a valuable alternative, capable of providing improved recommendations to users. Techniques like sentiment-based matrix factorisation and convolutional matrix factorisation highlight the synergy of sentiment analysis with recommendation models (Shen et al., 2019; Kim et al., 2016). Specifically, Shen et al. [Shen et al., 2019] developed a sentiment-based MF model that incorporates reviews' sentiments. To infer the review's overall sentiments scores, this model sums the sentiment score of each keyword in the target review based on the score obtained from a constructed sentiment dictionary. To perform rating prediction, these sentiment scores are converted into real values and then fused with the users' explicit ratings into an extended probabilistic matrix factorisation model. Kim et al. (2016) proposed a Convolutional Matrix Factorization (ConvMF) model, which utilises reviews text as complementary information. Firstly, this model utilises convolutional operations and word embedding for capturing the items' latent characteristics from their review texts. After that, the inferred latent features are integrated into a matrix factorization model to compute the users' ratings on target items. In fact, the extracted review words may be used to compute the similarity among users, rather than utilizing numerical ratings in CF (Kim et al., 2016). Effectively, sentiment analysis poses as a viable option to not only addresses sparsity and ambiguity but also as a tool to complement traditional ratings, improving recommendation quality (Shoja and Tabrizi, 2019).

Ultimately the integration of review texts into collaborative filtering recommender systems has shown positive impacts on system performance (Dang et al., 2020; Hernández-Rubio et al., 2019) both from a perspective of augmenting the existing user feedback (review words) and incorporating the user sentiment from review texts. As such we pursue integrating both user reviews and review sentiment into our neural collaborative filtering architecture - the details of which will be discussed further in Chapter 4.

2.5 Other Recommenders: a brief overview and history

While this paper shall focus solely on the application and enhancement of established techniques in collaborative filtering, it is essential to acknowledge the broader landscape of recommender systems, encompassing various methodologies beyond collaborative filtering. For recommender systems, there are three predominant categories: collaborative filtering, content-based filtering, and hybrid models (Sarwar et al., 2000). Collaborative filtering, the central theme of this paper, leverages user preferences and behaviours to provide personalised recommendations (Smith et al., 2017). In contrast, content-based filtering relies on the intrinsic attributes and characteristics of items to offer suggestions (Lops et al., 2011). Hybrid models, as the name implies, integrates elements of both collaborative and content-based approaches, aiming to harness the strengths of each (Thorat et al., 2015). In this section, we briefly explore the historical development and distinctive features of content-based filtering and hybrid models within the realm of recommender systems, emphasising how they differ from collaborative filtering. Hereby, we can provide a greater depth of understanding of recommender systems and also place collaborative filtering within the broader spectrum of methodologies.

2.5.1 Content-Based Filtering

Content-based filtering approaches the recommendation problem as a quest to find related items, constructing search queries based on the user's purchased and rated items to find others with similar attributes, such as the same author, artist, or director, or sharing similar keywords or subjects (Balabanović Shoham, 1997). The method relies on the item's information, neglecting contributions from other users - in contrast to the workings of collaborative filtering - focusing on recommending items akin to those the user has liked in the past (Pazzani, 1999; Balabanovic Shoham, 1997). The simplicity of content-based filtering lies in its three-step process: extracting item attributes, comparing them with user preferences, and recommending items based on matching features (Linden et al., 2003). One notable example of successful content-based filtering is the Music Genome Project, used by Pandora for its internet radio service (Koren et al., 2009). Content based methods are also a very popular approach when it comes to domains such as streaming platforms for TV shows, music or movies (Chen et al., 2017).

Research suggests that most content based recommender system use relatively simple retrieval models, such as keyword matching or the Vector Space Model (VSM) with basic TF-IDF weighting (Musto et al., 2015). These models operate by representing user preferences and item descriptions as vectors in a high-dimensional space, measuring their similarity using cosine similarity or other distance metrics (Weihong Yi, 2006). Recent developments in content-based filtering include word embedding techniques like Latent Semantic Indexing, Random Indexing, and Word2Vec. These techniques represent items and user profiles in a low-dimensional vector space, allowing for measuring similarity based on semantic understanding and improving the quality of content-based recommendations (Mikolov et al., 2013).

Despite its success in several domains, collaborative filtering methods often outperform content-based ones. However, content-based filtering, unlike collaborative methodologies, are unaffected and alleviate the cold-start problem (Chen et al., 2017) - the first-rater problem. In addition to alleviating the cold-start problem, content based filtering methods also offer transparency in recommendation explanations by explicitly listing content features that influenced the recommendations, unlike collaborative systems, which are often perceived as black boxes (Lops et al., 2011). While content-based algorithms scale well and perform computationally efficiently, they often fall short in delivering recommendations that are either too broad, like best-selling drama DVDs, or overly specific, such as all books by the same author (Linden et al., 2003) - this problem is termed overspecialisation. The recommendations tend to be confined to items similar to those already rated by the user, limiting the system's ability to suggest novel or unexpected items, a phenomenon known as the serendipity problem. Introducing randomness or filtering out items that are too similar to those the user has seen before are potential solutions (Lops et al., 2011).

2.5.2 Hybrid Models

Despite the successes of content-based filtering and collaborative-based approaches, each method has its own limitations, including issues like overspecialisation, cold-start problems, sparsity, and scalability. These challenges hinder their seamless integration into live production systems. This was acknowledged early into these model inceptions and led to hybrid filtering implementations (Vassiliou et al., 2006).

Hybrid filtering, entails a recommender system which combines two or more filtering techniques (such as collaborative and content-based filtering), with the aim to enhance recommender system performance and mitigate the limitations associated with individual approaches (Sanchez Sanchez, 2013; Zhou et al., 2010). These hybrid systems leverage the strengths of each method while compensating for their inherent weaknesses. Several comparitave studies demonstrate the superior performance of hybrid recommender systems compared to standalone approaches (Burke and Robin, 2007). The modern day Netflix serves as an exemplary case of a platform using a hybrid recommender system; they use a variation of collaborative filtering by comparing exploration habits of similar users and also incorporate content-based filtering by recommending movies similar to those highly rated (liked) by users (Thorat et al., 2015).

Various hybrid approaches exist, allowing flexibility in their implementation. The prominent variations of these approaches are listed below (Thorat et al., 2015; Mobasher et al., 2007):

1. Implementing collaborative and content-based methods separately and combining their predictions. 2. Incorporating some content-based characteristics into a collaborative approach. 3. Incorporating some collaborative characteristics into a content-based approach. 4. Constructing a general unifying model that incorporates both content-based and collaborative characteristics.

Ultimately, hybrid recommender systems offer a beneficial and effective method of alleviating various limitations inherent in individual standalone recommender system approaches. By combining the strengths of different methods, hybrid recommender systems can enhance accuracy and generate more tailored recommendations. However, they are not innocent of challenges. One notable challenge is the increased complexity of implementation and maintenance (Thorat et al., 2015). Integrating diverse methods requires careful coordination and might result in intricate systems that are challenging to manage, as well as computationally demanding. Additionally, selecting appropriate weighting or integration strategies for combining different recommendation sources can be non-trivial and may impact system performance (Thorat et al., 2015).

2.6 Evaluation Methods

Having covered the history and literature on recommender systems, particularly collaborative filtering, as well as providing context for text and deep learning in the recommender landscape, it would be amiss to not cover the literature pertaining to the evaluation of recommenders and draw focus to the various metrics and aspects evaluation entails.

In general, evaluating recommender systems poses inherent challenges, with algorithms exhibiting variable performance across different datasets (Herlocker et al., 2009) due to the fact that generally collaborative filtering models (and indeed all recommender models) have been developed or tailored for specific datasets. The diverse goals of evaluations further complicate matters, as early research focused on predictive accuracy, while more recent efforts have recognised the need to assess a recommender's impact on a users' decisions (Herlocker et al., 2009). Although accuracy metrics have facilitated early algorithm comparisons, they might not guarantee the generation of valuable recommendations (Adamopoulos, 2013). Historically, accuracy metrics had greatly helped the field of recommender systems; they had given us a way to compare algorithms and create robust experimental designs (McNee et al., 2006). Formally, predictive accuracy metrics measure how close the recommender system's predicted ratings are to the true user ratings (Herlocker et al., 2004). There have been many different predictive accuracy metrics applied to collaborative filtering results, including mean-absolute error (MAE) (Breese et al. 1998, Herlocker et al. 1999, Pennock et al. 2000b, Resnick et al. 1994, Shardanand and Maes 1995), correlation (Hill et al. 1995, Sarwar et al. 1998), ROC sensitivity (Good et al. 1999, Herlocker et al. 1999), mean squared error (MSE) (Shardanand and Maes 1995), precision/recall (Sarwar et al. 2000a), and a rank-utility metric (Breese et al. 1998). In fact, most research in the field has focused on improving accuracy of recommender systems, however that does not always result in a good recommender system. A more holistic view of a recommender system necessitates consideration of factors like diversity, serendipity, explainability, and responsiveness - more user-centric metrics (Herlocker et al., 2009). Since the early 2000s, there has been a gradual shift in the importance of these user-centric described metrics beyond accuracy (McNee et al., 2006; Herlocker et al., 2004).

Empirical experiments have shown that mean absolute error correlates strongly with many other proposed metrics for collaborative filtering (Herlocker 2000), yet is easier to measure and has well understood significance measures. Furthermore, mean absolute error is still the most frequently used metric among collaborative filtering researchers. We used MAE, MSE and RMSE as our chosen. predicitve accuracy metrics to report the performance of the prediction problem because they are most commonly used and easiest to interpret directly (Sarwar et al., 2000). Below we provide some benefits of MAE as discussed by Herlocker et al. (2004): First, the mechanics of the computation are simple and easy to understand. Second, MAE has well-studied statistical properties that provide for testing the significance of a difference between the mean absolute errors of two systems.

In addition to traditional predictive accuracy metrics, another impactful (particularly in E-commerce) evaluation method in recommender systems involves generating a list of items and assessing its quality – commonly known as Top-N evaluation. This approach, as defined by Herlocker et al. (2004), employs certain metrics to gauge the performance of recommender systems by evaluating their ability to recommend a specific number (N) of items to users. Top-N evaluation is particularly valuable because it aligns with real-world scenarios where users are presented with a limited set of recommendations - particularly pertinent in E-commerce settings. It reflects the practical utility of recommender systems in suggesting a curated list of items that are most likely to appeal to users' preferences. This evaluation approach captures the essence of personalized recommendation effectiveness in situations where users are typically interested in a manageable number of suggestions rather than an exhaustive list.

Ultimately, we acknowledge the importance of a holistic view of a recommender system, and not simply accepting that improving predictive accuracy builds a better recommender system. To facilitate this, we incorporate predictive accuracy metrics as well as Top-N evaluation metrics to paint a clearer picture as to the performance of our recommender. The aforementioned user-centric metrics (coverage, diversity etc.) are mentioned to bring awareness around the holistic view of evaluation, however they are not of the foremost concern for this paper. In this paper we focus primarily on predictive accuracy, such as MAE, but we still incorporate and assess our models using Top-N metrics to see the ability of our recommender to find good items.

2.7 Limitations and Challenges

Recommenders, particularly collaborative filtering, have achieved substantial success in diverse domains by delivering personalized content tailored to users on various platforms. Despite their widespread effectiveness, these systems are not immune to certain limitations. While we have briefly touched upon these limitations throughout this chapter, this section aims to provide more clarity on each of them. These limitations are specific to collaborative filtering models; however, we also provide some domain-specific challenges that e-commerce recommender applications pose.

2.7.1 Sparsity

Sparsity, formally, refers to the situation where the available data in a collaborative filtering recommender system is limited, resulting in a sparse user-item interaction matrix with a significant number of missing values. In scenarios where the pool of available items is exceptionally large, as observed in major E-commerce platforms with millions of items (Newman, 2005), the overlap between users becomes minimal or even nonexistent. That is to say that often we will have instances where users don't purchase the same items. In fact, the distribution of user evaluations per items follows a power-law distribution or a Weibull distribution, resulting in a majority of users/items expressing or receiving only a few ratings (Weibull, 1951; Huang et al., 2004) and hence majority of users (or items) may have expressed (or received) only a few ratings. This poses a significant challenge in the domain of collaborative filtering recommender systems, impacting the effectiveness of these systems in providing accurate and relevant recommendations. As mentioned, sparsity is particularly evident in e-commerce user-item interaction data, where a substantial number of ratings are missing, frequently exceeding 99 percent sparsity (Newman, 2005). Several factors contribute to this issue, including the challenge users face in expressing their interests numerically on products and the limited coverage of the recommendation space (Newman, 2005). This high level of sparsity critically hinders the effectiveness of collaborative filtering (CF) approaches (Newman, 2005), since it diminishes the ability of the (collaborative) system to identify meaningful patterns or similarities between users or items, ultimately impacting the quality and reliability of the generated recommendations. Specifically, it impedes the calculation of similarities among users, a fundamental aspect of collaborative filtering. When the sparsity is extensive, the system struggles to find a sufficient number of overlapping preferences between users, leading to ineffective similarity computations. And, even when similarities are calculable, they become unreliable due to the inadequacy of information caused by the sparsity (Newman, 2005).

Ultimately, the challenge when users rate on a few items makes it increasingly difficult to discern their interests accurately. While this paper did not primarily focus on this limitation, it is nonetheless addressed due to the inclusion of textual features, which are well-known for their utility in mitigating or assisting recommenders in coping with sparsity issues.

2.7.2 Cold Start Problem

The cold start problem is a persistent age-old challenge encountered in recommender systems when dealing with new users or items that lack sufficient information in the system. The first facet of this problem emerges when novel users or items are introduced to the rating matrix, preventing collaborative filtering (CF) methods from generating accurate recommendations due there being not enough ratings available about them (Huang et al., 2004). To address this issue, hybrid recommender techniques, combining both content and collaborative data, have been employed as solutions, and sometimes they are accompanied by asking for some base information (such as age, location and preferred genres) from the users [Burke et al., 2015; Zheng et al., 2010] - also known as active learning. Notably, research by Liu et al. (2014) has proposed a novel technique that tracks individual users' activities across multiple e-commerce sites, allowing recommendations for a cold-start user in one site to be informed by their records in other sites [Zhang et al., 2013].

The cold start problem encompasses three distinct scenarios within the recommender system domain (Huang et al., 2004). The first scenario arises when a new user joins the system, and no prior information is available about their preferences, constituting the new user cold-start problem. The second scenario emerges with the introduction of a completely new item to the system, lacking any associated ratings—an issue known as the cold start item problem. And finally, the third scenario occurs during the initial launch of the system when both user and item information are absent, characterising the cold start system problem. In these instances, content-based solutions, renowned for their effectiveness in handling information scarcity, can be applied to mitigate the challenges associated with cold start problems. This is because content-based filtering leverages the inherent characteristics of items, such as textual or numerical features, to provide recommendations even when user-item interactions are limited or absent.

In our paper, addressing the cold start problem is not a priority and is beyond the scope of the work done, as such we mitigate any risk of the cold start problem by selecting or using a pool of users and items with sufficient information available.

2.7.3 Scalability

Scalability is, formally, defined as the ability of a system or process to handle a growing amount of work, resources, or an expanding user base, without compromising performance, efficiency, or overall functionality (Burke et al., 2015). As such, scalability becomes a key concern for many platforms as they grow, and their user base or item catalog rises in numbers. For example, Amazon's deals with millions of customers and has a catalog of items equally as big, the enormity of this available data provides both opportunities and challenges. The challenge lies in utilising algorithms designed for smaller datasets by three orders of magnitude, demanding substantial computational resources. In collaborative filtering (CF) algorithms, particularly memory-baed methods, the computation of user similarity becomes prohibitively expensive with larger datasets, hampering scalability and necessitating significant memory and computational power (Smith and Linden, 2017). The scale at which recommender systems operate, especially for successful internet companies like Amazon, incurs substantial infrastructure costs and limitations due to the escalating volume of processed data. This added dimension makes more an interesting problem for building recommenders.

In practice, often overcome this issue of scalability, recommendation algorithms often divide the prediction generation algorithm into offline and online components, where the offline part requires extensive computation, and the online component dynamically generates predictions for users in real time (Sarwar et al., 2000). Additionally, it is also not uncommon for companies to address scalability concerns by incorporating standard collaborative filtering algorithms into distributed computing engines such as Apache Hadoop or Spark, leveraging their speed and efficiency in parallel large-scale data processing (Burke et al., 2015; Smith and Linden, 2017). Other more traditional techniques to handle scalability involve methods such as sampling users, data partitioning, and omitting high or low-frequency items to manage scalability, yet these strategies face the risk of compromising the quality of recommendations. Dimensionality reduction techniques like clustering and principal component analysis have also been considered to, however, similar to aforementioned techniques they can potentially adversely impact recommendation quality by eliminating low-frequency items. These issues have been long standing and have been present since the early days of collaborative filtering inception, however, these scalability issues have become more pertinent obstacles nowadays with the vast amounts of data readily available.

We acknowledge this limitation to raise awareness, though it is not an immediate concern for this paper. To address this potential obstacle, we opt to reduce our dataset to a more manageable size. This allows us to eliminate any concerns related to scalability, ensuring a smoother and more efficient execution of our analysis.

2.7.4 Domain and Data Challenges

Another key challenge faced by collaborative filtering methods, in particular, are rooted in their reliance on users' numeric ratings as the primary source of preference information, leading to a significant limitation in recommendation accuracy due to the often inadequate semantic explanation of scalar rating information (Leino et al., 2007; Shoja et al., 2019). Recognising this drawback, efforts have been made to enhance recommendation accuracy by integrating other information with ratings - such as user reviews. This was briefly touched on in Section 2.4, when discussing the role of text for recommenders.

Furthermore, each domain has their inherent (and sometimes unique) challenges. In the context of E-commerce there are a variety of characteristics of the field which make it slightly more challenging for collaborative filtering methods. These are discussed by Linden et al. (2003) and are summarised below.

1. Scalability and need for real-time results: As mentioned in the previous subsection, recommenders in E-commerce are often faced with scalability obstacles. Large retailers contend with extensive data, managing tens of millions of customers and millions of distinct catalog items. Navigating this vast landscape poses a considerable challenge to generate useful recommendations. This coupled with the fact that many applications demand real-time results, requiring recommendations to be generated in no more than half a second while maintaining high-quality outputs. This necessity adds a layer of complexity to the operational efficiency of recommendation algorithms. This challenge is reflective of the issues raises in Section 22.7.1.

2. Cold Start Problem: Another key concern is that there is limited information for new customers on an e-commerce platform - known as cold start problem as discussed in earlier subsections. New customers entering the e-commerce platform typically possess minimal information, often based on just a few initial purchases or product ratings. The challenge lies in delivering relevant recommendations with limited user history. This is inline with discussions on cold start problem in Section 2.7.2.

3. Volatile Customer Data: A new more niche challenge is that customer interactions in e-commerce are dynamic and volatile, with each new interaction offering valuable data. Recommendation algorithms must promptly respond to this evolving information to ensure relevance and accuracy.

Besides these aforementioned long-standing challenges from this section, many novel issues have begun to appear more recently. Generally, progress and propagation of new techniques brings new challenges. For example, the GPS equipped mobile phones have become mainstream and the Internet access is ubiquitous, hence the location-based recommendation is now feasible and increasingly significant[4]. Additionally, one of the more interesting challenges for research on recommender systems is that there are a lot of factors which affect recommendation quality; namely, data partitioning (train and test sizes), similarity measures, recommendation list size amongst some others. These factors are greatly explored in the paper by Symeonidis et al. (2008). We have thus used the literature from paper Symeonidis et al. (2008) to infer the most commonly used approaches and techniques used in selecting the optimum values for these certain factors. These will be discussed in detail in Chapters 3 and Chapters 4.

[4]Websites like Foursquare, Gowalla, Google Latitude, Facebook, Jiapang, and others already provide location-based services and show that many people want to share their location information and get location-based recommendations

Given these discussions on the specific limitations for this domain, the scope of this paper does not extend to a comprehensive discussion or addressing these issues at all in the context of recommender systems. Primarily, we are concerned with development and analysis of a neural collaborative filtering network together with multiple modalities (text and ratings data) to better improve the performance (in terms of predictive accuracy and Top-N generation) of recommender systems in retail applications. The limitations that these recommender systems approaches suffer from inherently as discussed in this section were to simply build a better picture of collaborative filtering approach.

2.8 Conclusion

Throughout this chapter we often made reference to past work which is related to our research and the results of which can be compared to our own results and findings from our research. We began this chapter by presenting a giving a brief history of recommender systems as well as their impact and application in e-commerce, where they play a significant role in better engaging and driving sales between content and consumer. Recommender Systems prove to be a key element in the operation of e-commerce platforms looking to meet their customer's needs. These recommender systems filter out information desired or would perhaps be of interest to the user. Ultimately, a good recommender system staves off a user's time, keeps them engaged, enhancing their experience and consequently leads to increased revenue.

In Section 2.2, we delved into the details and literature around collaborative based learning, one of the three main recommender system paradigms. We find that this method has it's own unique strengths as well as weaknesses. For example, CF can excel in providing accurate recommendations based on user preferences, historical behaviors, or item similarities, but its performance can be significantly negatively affected by user feedback sparsity.

We also looked at deep learning and text analysis for recommender systems in Sections 2.3 and 2.4 respectively, particularly focusing on the recent implementations of neural collaborative filtering. This historical analysis provides context for the choices made in this thesis, such as the decision to use a deep learning approach for collaborative filtering as well as why we augment our NCF model to incorporate textual features and user sentiments. Specifically, neural collaborative filtering alleviates limitations associated with traditional collaborative filtering methods by capturing intricate patterns and non-linear relationships in user-item interactions, while the incorporation of textual features and user sentiments enriches the model's understanding of user preferences, resulting in more nuanced and context-aware recommendations (as well as addressing sparsity concerns in the data).

We also addressed all the various aspects of evaluation for recommender systems in Section 2.6. This serves as a guide to help support the decision for the metrics used in this paper. We have decided to use predictive accuracy metrics as well as classification and Top-N metrics to assess our recommender systems going forward. Even though the rating prediction perspective is the prevailing paradigm in recommender systems, there are other perspectives growing in use and recognition that were discussed in length in this section. By us choosing accuracy metrics as well as Top-N metrics we get a slightly better perspective of the performance of the recommender system as well as alleviate the problems pertaining to the narrow rating prediction accuracy-based focus.

The final section in this chapter addressed the limitations and challenges faced by collaborative based filtering systems and recommender systems as a whole in general. This section discussed in length data sparsity, cold start problem, scalability as well as domain specific problems such as the volatile nature of customer preferences. To this end we outlined that the scope of this paper does not seek to answer or address all of these problems, however we will be touching on data sparsity as a consequence of exploring augmenting our collaborative filtering with textual features.

Ultimately, this detailed exploration on the general history of recommender systems has taken us on a journey through various methodologies and approaches, each contributing to the evolution of this dynamic field. From the foundational collaborative filtering methods, the integration of text into recommenders, as well deep learning-based approach for recommendations, we have established context for the rest of the paper. As we transition to the next chapter on data, it becomes evident that the success of these models is intricately tied to the quality and characteristics of the data they leverage. Understanding the nuances of data used in recommender systems is crucial for addressing challenges, refining methodologies, and ultimately delivering more effective and personalised recommendations. In the forthcoming section, we delve into the intricacies of the data landscape, exploring the diverse sources, formats, and considerations that shape the foundation of our recommender system research.

Chapter 3

Data Exploration and Analysis

3.1 Background

3.2 Variable Description

3.3 Data Collection and Preparation

3.4 Data Partitioning

3.5 Data Summary

3.6 Data Trends and Patterns

3.7 Conclusion

Chapter 4

Methodology

4.1 Modelling Approach

4.2 Neural Collaborative Filtering Model

4.2.1 Model Specification and Components

4.2.2 Regularisation

4.3 Content Based Filtering Model

4.4 Hybrid Model

4.4.1 Model Fusion

4.4.2 Model Specification

4.5 Benchmark Models

4.5.1 Non-negative Matrix Factorisation

4.5.2 User-Based Collaborative Filtering

4.5.3 Content Based Filtering

4.5.4 Neural Collaborative Filtering

4.5.5 Non-negative Matrix Factorisation and Content Based Filtering Hyrbid

4.5.6 User-Based Collaborative Filtering and Content Based Filtering Hyrbid

4.6 Model Criteria

4.6.1 Ranking Criteria

4.6.2 Evaluation Criteria

Chapter 5

Application

5.1 Example

5.1.1 Example

5.1.2 Example

5.2 Example

5.3 Example

Chapter 6

Results

6.1 Results

Chapter 7

Discussion

7.1 Example

7.2 Example

7.3 Example

Appendix A

A.1 Example

Appendix B

B.1 Example

B.2 Example