1 Growth Engine of the Internet

Recommender Systems

This is an era where life is being influenced by recommender systems everywhere. For online shopping, the recommender system will pick thoughtful products; for searching for news, the recommender system will prepare interesting headlines; for learning something new, the recommender system will provide the most suitable courses; for looking for some relaxation, the recommender system will stream addicting short videos; or, if you simply want to close your eyes and rest, the recommender system will play the most appropriate music. In short, recommender systems have never impacted people's lives as much as they do now.

And the machine learning engineers behind the recommender systems have never been chasing the ever-changing recommendation technology like they are now. If recommender systems are the growth engine of internet development, then the machine learning engineer is the development engine of the recommender systems. In this chapter, we will start from the specific scenarios of recommender systems, introduce what recommender systems are, why recommender systems are called the "growth engine" of the internet, and how to view recommender systems from a technical point of view and build the overall architecture of recommender systems.

1.1 Why Are Recommender Systems the Growth Engine of the Internet?

For internet industry practitioners, the word "growth" is like a spear in the heart, constantly stimulating and motivating us. My understanding of the word "growth" comes from an experience in college. Sogou, Inc. has a long-term partnership with Tsinghua University's Department of Computer Science. As a result, my classmates from the same lab often talk about the cooperation projects with Sogou. There was one sentence that I remember to this day: "If we can recommend more suitable ads for Sogou users and increase their click-through rate by 1%, we can increase the company's profit by tens of millions." Since then, the word "growth" has been ingrained in my heart. This word has almost become the only criterion for the success of IT companies, and it has also become the eternal pursuit of all IT practitioners. The desire to "magically" achieve "growth" through algorithms and models has also guided me on my career path as a machine learning engineer.

https://doi.org/10.1017/9781009447515.003 Published online by Cambridge University Press

1.1.1 The Role and Significance of Recommender Systems

The role and significance of recommender systems can be explained from the perspectives of users and companies.

User perspective: The recommender system solves the problem of how users can efficiently obtain interesting information in the case of "information overload." In theory, application scenarios for recommender systems are not limited to the internet. However, the massive information problem brought by the internet often causes users to get lost in the sea of information and unable to find the target content. This makes the internet the best scenario for recommender system applications. Like the fish that represents this book on the cover, it emerges from the shoal of fish, traverses the digital web and leaps onto the paper. I hope it can become the knowledge "koi" that was screened out for you. From the perspective of user needs, the recommender system works as a filter when the user's needs are not very clear. Therefore, compared with a search system (in which users input a clear "search keyword"), the recommender system makes more use of the historical information from the user to "guess" what they might like. This is the basic assumption that must be paid attention to when solving recommendation problems.

Company perspective: The recommender system solves the problem that products can attract users to the greatest extent, retain users, increase user stickiness, and improve user conversion rate, so as to achieve the purpose of continuous growth of the company's business goals. Companies with different business models define different optimization goals. For example, video companies pay more attention to users' viewing time; e-commerce companies pay more attention to users' conversion rate (CVR); and news companies pay more attention to users' click-through rate (CTR), and so on. It should be noted that the ultimate goal of designing a recommender system is to achieve the company's business goals and increase the company's revenue. It should be the starting point for engineers to consider problems from the company's perspective.

Because of this, the recommender system is not only an "engine" for users to efficiently obtain interesting content, but also an "engine" for internet companies to achieve business goals. These two aspects are two dimensions of the same problem and complement each other. Next, I will try to use two application scenarios to further explain how the recommender system plays the key role of "growth engine."

1.1.2 Recommender Systems and YouTube Watch Time Growth

As already mentioned, the "ultimate" optimization goal of a recommender system should include two dimensions: optimizing user experience and satisfying the company's business interests. For a healthy business model, these two dimensions should be in harmony. This is fully reflected in the YouTube recommender system.

YouTube is the world's largest UGC (User Generated Content) video-sharing platform (as shown in Figure 1.1). The most direct manifestation of its optimized user

https://doi.org/10.1017/9781009447515.003 Published online by Cambridge University Press

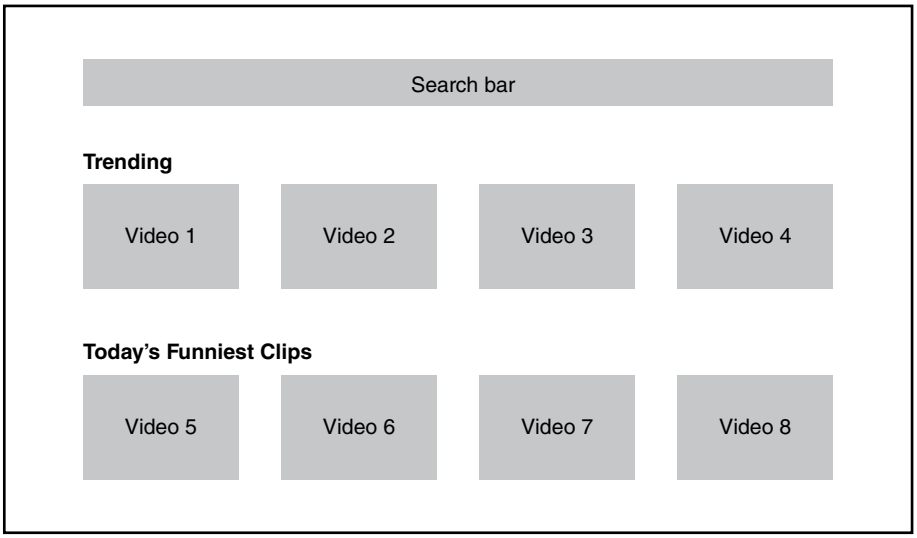

Figure 1.1 Homepage of YouTube.

experience is the increase in user viewing time. As an advertising-based company, YouTube's business interests are also based on the growth of user viewing time, because the user's viewing time is proportional to the total advertisement exposure. Only by continuously increasing the exposure of advertisements can the company's profits continue to grow. Therefore, YouTube's user experience and the company's interests are aligned on the "watch time".

Because of this, the main optimization goal of YouTube's recommender system is viewing time rather than the "click-through rate" that traditional recommender systems value. In fact, in a well-known engineering paper, *Deep Neural Networks for YouTube Recommendations* [1], YouTube engineers explicitly proposed a modeling method that uses watch time as an optimization objective. The general recommendation process is: first, build a deep learning model to predict the duration that the user watches a candidate video, and then sort the candidate videos according to the predicted duration to form the final recommendation list. The technical details of the YouTube recommender system are discussed in the following chapters.

1.1.3 Recommender Systems and Revenue Growth for E-Commerce Sites

While the recommender system plays a relatively indirect role in achieving YouTube's business goals, it directly drives the company's revenue growth on the e-commerce platform. This is because whether the products recommended by the recommender system for users are suitable directly affects the user's purchase CVR.

In 2019, the turnover of Tmall's "Double 11" was 268.4 billion yuan. What drives Tmall to achieve such an amazing turnover is Alibaba's famous "Thousands of People, Thousands of Faces" recommender system (as shown in Figure 1.2 on the homepage of Tmall's mobile terminal). Comparing the Tmall homepage seen by a man and

https://doi.org/10.1017/9781009447515.003 Published online by Cambridge University Press

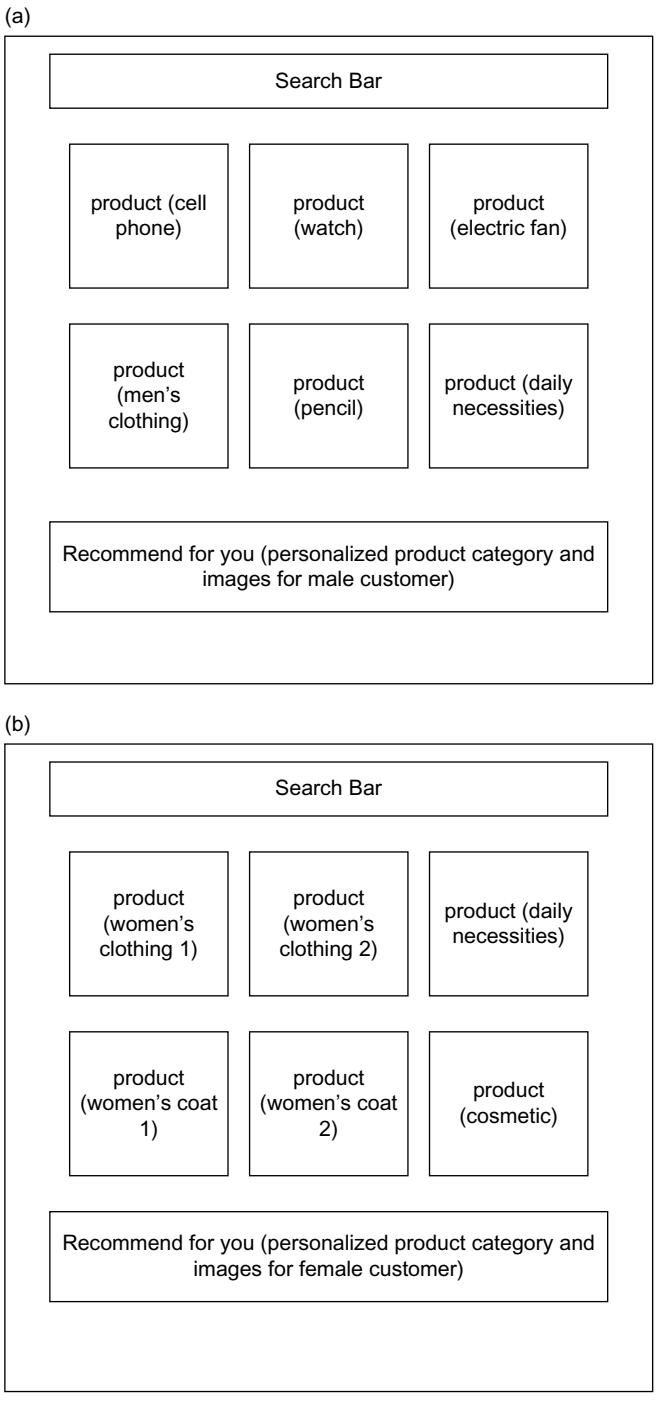

Figure 1.2 Homepage of Tmall's mobile terminal. (a) Tmall homepage seen by a male user. (b) Tmall homepage seen by a female user.

https://doi.org/10.1017/9781009447515.003 Published online by Cambridge University Press

a woman, it can be seen that Tmall's recommender system not only recommends different categories of products for different users (for example, in the "Quick Snap" module, mobile phones and watches are recommended for men, while women's clothing and pajamas are recommended for women), but also different thumbnails of the same category are generated according to the user's characteristics (for example, in the "Recommended for you" module, the thumbnails of the same channel are personalized).

It can be said that Tmall's recommender system truly realizes the personalized recommendation of all elements of the homepage, and achieves a veritable "Thousands of People, Thousands of Faces." Everything behind this is driven by recommendation algorithms that focus on improving CVRs and click-through rates. Assuming that an improvement in the recommender system increases the overall conversion rate of the platform by 1%, then with a 268.4-billion-yuan turnover, the increase will reach 2.684 billion yuan (268.4 × 1%). In other words, machine learning engineers created a value of 2.684 billion yuan just by optimizing the recommendation technology. This is undoubtedly the greatest charm of recommender engineers.

The value of recommender systems goes far beyond that. In 2018, the global online advertising market reached 220 billion US dollars, and the driving force behind this is the advertising recommender systems of major companies. In the same year, the user viewing time of short video applications in China increased by 89.2%, for which video recommendation engines played an irreplaceable role. Since 2015, personalized information applications have overwhelmingly surpassed traditional portal websites and news applications, becoming the most important way for users to obtain information. It can be said that the recommender system has become the core technology system driving almost all fields of applications of the internet, and it deserves to be the strong engine that boosts the growth of the internet today.

1.2 Recommender Systems Architecture

Through the introduction of Section 1.1, the reader should already understand the following two points:

(1) The core demand of internet companies is "growth," and the recommender system is at the center of the "growth engine."
(2) The "user pain point" to be solved by the recommender system is how to efficiently obtain the information of interest for the user in the case of "information overload."

The first point tells us that the recommender system is important and indispensable. The second point clearly explains the basic problem to be solved in building it; that is, the recommender system needs to deal with the relationship between "people" and "information."

The "information" here refers to "product information" in product recommendation, "video information" in video recommendation, and "news information" in news recommendation. In short, it can be collectively referred to as "item information."

https://doi.org/10.1017/9781009447515.003 Published online by Cambridge University Press

From the perspective of "user," in order to more reliably infer the interests of "user", the recommender system hopes to use a large amount of information related to "user," including past behavior, demographic attributes, relationship networks, and so on, which can be collectively referred to as "user information."

In addition, in a specific recommendation scenario, the user's final selection is generally affected by a series of environmental information such as time, location, and user status, which may be referred to as "scenario information" or "context information."

1.2.1 The Logical Framework for Recommender Systems

Based on the knowledge of "user information," "item information," and "context information," the problem to be handled by the recommender system can be formally defined as: for user U, under a specific context C, for massive information of "item," construct a function f(U, I, C), to predict the user's preference for a specific candidate "item" I, and then sort all candidate items according to the preference to generate a recommendation list.

According to the definition of the recommender system problem, the abstract logical framework can be obtained (as shown in Figure 1.3). Although this logical framework is highly generalized, it is on this basis that the entire technical architecture of the recommender system is produced by refining and expanding each module.

1.2.2 The Technical Architecture for Recommender Systems

To develop a functional recommender system, engineers must translate abstract concepts and modules into concrete implementations. Based on Figure 1.3, there are two types of problems that engineers need to focus on solving:

(1) Questions related to data and information, that is, what are "user information," "item information," and "context information"? How are they stored, updated, and processed?
(2) Issues related to the recommender system algorithms and models, that is, how to train the recommendation model, how to predict, and how to achieve a better recommendation?

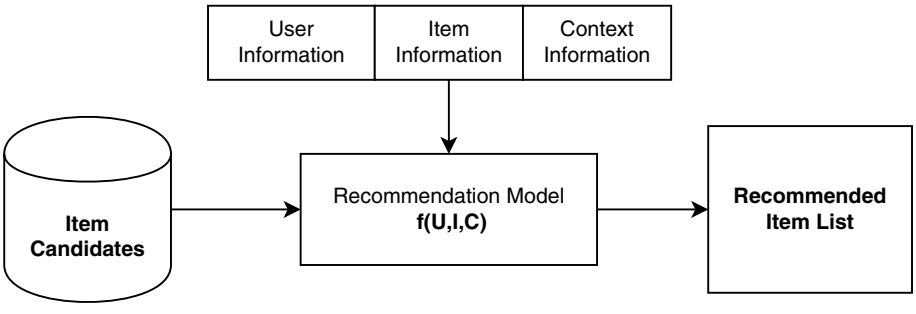

Figure 1.3 The logical framework for recommender systems.

https://doi.org/10.1017/9781009447515.003 Published online by Cambridge University Press

These two types of problems can be divided into two parts: the "data and information" part has gradually developed into a data flow framework that integrates offline batch processing and real-time stream processing in the recommender system; the "algorithm and model" part is further refined as a model framework that combines training, evaluation, deployment, and online inference of the recommender system. Specifically, the schematic diagram of the technical architecture for recommender systems is shown in Figure 1.4.

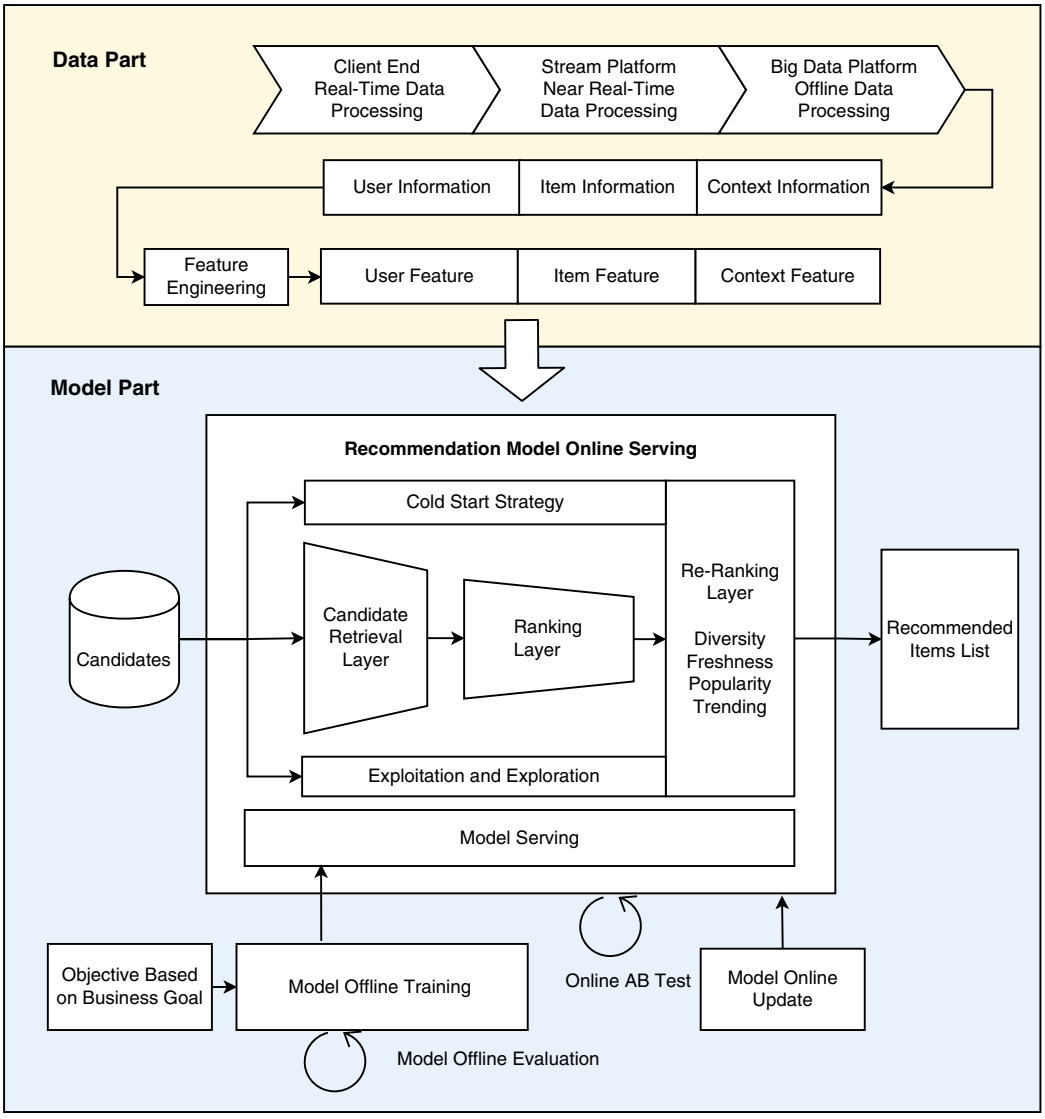

Figure 1.4 Schematic diagram of the technical architecture for recommender systems.

https://doi.org/10.1017/9781009447515.003 Published online by Cambridge University Press

1.2.3 Data Part of the Recommender System

The data part of the recommender system (as shown in beige in Figure 1.4) is mainly responsible for the information collection and processing of "users," "items," and "context." Specifically, the three platforms responsible for data collection and processing are ranked according to the ability of real-time performance, namely "client-side and server-side real-time data processing," "quasi-real-time data processing on stream processing platform," and "offline data processing on big data platform." While the real-time performances decrease, the massive data processing capabilities of the platforms increase. Therefore, the data flow framework of a mature recommender system will use the three platforms to complement each other and utilize them together.

After obtaining the original data information, the data processing system will further process the original data. There are three main data exports after processing:

(1) Generate the sample dataset required by the recommendation model for training and evaluation.
(2) Generate the "features" required for the recommendation model serving for online inference of the recommender system.
(3) Generate statistical data required for system monitoring and business intelligence (BI) systems.

To some extent, the data part of the recommender system is the "water source" of the entire system. Only by ensuring the continuity and purity of the "water source" can the recommender system be continuously "nurtured" to operate efficiently and output accurately.

1.2.4 Model Part of the Recommender System

The "model part" is the main body of the recommender system (as shown in light blue in Figure 1.4). The structure of the model is generally composed of the "Retrieval Layer," "Ranking Layer" and "Re-ranking Layer."

- The "Retrieval Layer" generally uses efficient rules, algorithms or simple models to quickly retrieve items that users may be interested in from a massive candidate set.
- The "Ranking Layer" uses the sorting model to fine-sort the candidate sets that are initially screened.
- The "Re-ranking Layer" can combine some supplementary methods and algorithms to make certain adjustments to the recommendation list, so that additional factors such as "diversity," "popularity," and "freshness" of the results are taken into account, before finally forming a user-visible recommendation list.

This process, from the recommendation model receiving the set of all candidate items, to finally generating the recommendation list, is generally referred to as the model-serving process.

https://doi.org/10.1017/9781009447515.003 Published online by Cambridge University Press

Before performing model services in an online environment, model training is required to determine the model structure, the parameter weights in the structure, and the parameters in related algorithms and strategies. According to the different training environments of the model, the training methods can be divided into two parts: "offline training" and "online updating." The advantage of offline training is that it can utilize the entire set of samples and features to make the model approach the global optimum; while online updating focuses on "digesting" new data samples in quasi-real-time and reflecting new data trends more quickly to meet the real-time requirements of the model.

In addition, to evaluate the recommendation model and facilitate iterative optimization, the model part of the recommender system provides various evaluation modules such as "offline evaluation" and "online A/B testing." These offline and online evaluation indicators are used to guide the next iterative model optimization.

All of these modules together constitute the technical framework of the model part of the recommender system. The model part, particularly the "Ranking Layer" model, is its focus, and it is also the focus of research in the industry and academia. Therefore, the following chapters focus on the model part, especially the mainstream technology of the "Ranking Layer" models and their evolution trends.

1.2.5 The Revolutionary Contribution of Deep Learning to Recommender Systems

The revolutionary contribution of deep learning to recommender systems lies in the improvement of the recommendation model part. Compared with traditional recommendation models, deep learning models are more capable of fitting data patterns and mining feature combinations. In addition, the flexibility of the deep learning model structure enables it to adjust the model according to different recommendation scenarios, making it a "perfect" fit with specific business data.

At the same time, the requirements of deep learning for massive training data and real-time data also pose new challenges to the recommender system's data flow. How to achieve real-time processing of massive data, real-time extraction of features, and real-time data acquisition in the online model service process are the difficult problems that need to be overcome in the data part of the deep learning recommender system.

1.2.6 See the Whole Picture, Supplement Details

The overall technical architecture of the recommender system and its corresponding technical details are extremely complex. It not only requires practitioners to have deep machine learning knowledge and theoretical understanding of recommendation models, but also demands the practitioners' engineering capabilities and the "business sense" to make the best choice by leveraging different technical solutions. Perhaps this is the charm of recommender systems.

By studying this chapter, you will gain an overall understanding of the framework of deep learning recommender systems. Don't worry if you are not clear about the

https://doi.org/10.1017/9781009447515.003 Published online by Cambridge University Press

technical terms and related concepts involved in this chapter. Just keeping the initial impression of the deep learning recommender system is good enough. I hope you can "see the whole picture" of the technical framework of recommender systems, and read specific chapters to "supplement details." I believe this book will help you answer the questions in your heart.

1.3 Structure of the Book

The contents of this book will follow the structure depicted in Figure 1.4, with an emphasis on introducing the applied knowledge and practical experience of deep learning in recommender systems. While introducing a specific technical topic, we will try to present a full picture of the development process and its cause and effect.

Since the ranking model is taking the absolute core position in the entire recommender system, the first few chapters of this book will focus on the technical evolution trend of the deep learning ranking model. In the following chapters, we will introduce some technical details and engineering implementations in the other modules of recommender systems, by presenting examples of industry-leading recommender systems. Specifically, the main content of this book is divided into nine chapters.

Chapter 1. The Growth Engine of the Internet: Recommender Systems
This chapter introduces the basic knowledge of recommender systems, their status and role in the IT industry. It introduces the main technical architecture of recommender systems so that readers can gain a high-level understanding and expand the content of this book from the whole to the details.

Chapter 2. Pre-Deep Learning Era: The Evolution of Recommender Systems
This chapter looks at the evolution history of the recommendation models in the pre-deep-learning era and introduces basic machine learning technology related to recommendation models, so as to lay a solid foundation for grasping the deep learning recommender system.

Chapter 3. Top of the Tide: Application of Deep Learning in Recommender Systems
This chapter examines the popular deep learning recommendation model structure in industry and the evolution maps among different models. It is hoped that readers can establish ideas and technical intuitions for improving recommendation models while mastering the main technical approaches of deep learning recommender systems.

Chapter 4. Application of Embedding Technology in Recommender Systems
This chapter focuses on the embedding technique, the core technique of deep learning, in recommender systems. This chapter includes the development process and technical details of the state-of-the-art embedding technique, as well as its applications.

https://doi.org/10.1017/9781009447515.003 Published online by Cambridge University Press

Chapter 5. Recommender Systems from Multiple Perspectives
If the deep learning recommendation model is the core of the recommender system, then this chapter will re-examine it from perspectives beyond this core. It covers the different technical modules and optimization ideas of recommender systems. These include feature engineering, retrieval layer strategies, real-time recommender systems, optimization goals, business understanding, cold start, "exploration and exploitation," and many other important recommender systems topics.

Chapter 6. Engineering Implementations in Deep Learning Recommender Systems
This chapter introduces the engineering implementation approach and main technical platform of the deep learning recommender systems. It includes three parts: data processing platforms; offline training platforms; and online deployment and inference methods

Chapter 7. Evaluation in Recommender Systems
This chapter takes a closer look at the main indicators and methods of recommender system evaluation. It explains how to establish a multilayer recommender systems evaluation framework from traditional offline simulation and evaluation, to fast online evaluation test methods, and finally to online A/B testing.

Chapter 8. Frontier Practice of Deep Learning Recommender Systems
This chapter introduces the technical framework and model details of the industry-leading recommender systems. It mainly includes the cutting-edge practice of recommender systems from industry giants such as YouTube, Airbnb, Facebook, and Alibaba.

Chapter 9. Build Your Own Recommender Systems Knowledge Framework
The final chapter summarizes the knowledge of recommender systems related to this book, and introduces the main technical and analytical methodologies required for recommendation engineers.

Reference

[1] Paul Covington, Jay Adams, Emre Sargin. Deep neural networks for YouTube recommendations. Proceedings of the 10th ACM Conference on Recommender Systems, September 7, 2016 (pp. 191–198).

https://doi.org/10.1017/9781009447515.003 Published online by Cambridge University Press