第 1 章 推荐系统概览

1.1 什么是推荐系统

随着互联网、大数据以及人工智能技术的快速发展，我们日常使用的绝大多数互联网产品，例如短视频平台、新闻资讯平台、电商平台、生活服务平台以及社交媒体平台，都已经深度依赖推荐系统来完成海量信息的组织与分发。无论是抖音、快手、微信视频号、美团，还是淘宝、京东、小红书，推荐系统都已经成为这些平台最核心的基础设施之一。

从本质上来看，推荐系统承担着连接用户与内容的重要桥梁作用。面对海量且持续增长的信息供给，用户往往无法依靠主动搜索的方式发现所有感兴趣的内容。推荐系统则通过分析用户历史行为、兴趣偏好以及上下文环境等信息，从海量候选内容中筛选出最有可能满足用户需求的内容，并以个性化的方式呈现给不同用户，实现“千人千面”的内容分发。

换句话说，推荐系统所解决的核心问题是：

定义 1.1

如何在海量内容与海量用户之间建立高效且精准的匹配关系，从而在合适的时间，将合适的内容推荐给合适的用户。

♣

对于推荐系统的初学者而言，第一次接触这一领域时往往会产生许多疑问：

笔记 什么是推荐系统？

为什么称之为“推荐系统”而不是“推荐模型”？

推荐系统由哪些模块组成？

推荐结果是如何一步一步计算出来的？

推荐系统每天都在优化什么目标？

工业界的大规模推荐系统究竟是如何运行的？

事实上，这些问题恰恰揭示了推荐系统与传统机器学习任务之间的重要区别。很多初学者在学习推荐系统时，往往首先接触的是协同过滤（Collaborative Filtering）、FM、DeepFM、DIN、Transformer 等模型，并容易形成一种误解：推荐系统就是利用某个模型预测用户是否会点击某个内容。

然而在真实工业场景中，一个推荐系统远远不只是一个模型。它通常是一个由数据采集、特征工程、候选召回、排序决策、流量调控、在线服务、实验评估以及业务生态治理等多个模块共同组成的**大规模复杂系统工程**。模型只是其中的一个或多个组成部分，而非推荐系统的全部。因此，本书希望从**系统视角**而非单纯模型视角来理解推荐系统。

在本章中，我们将首先从宏观层面介绍推荐系统的整体架构，帮助读者建立对推荐系统的全局认知。随后，我们将进一步介绍推荐系统的发展历程、推荐系统中的业务目标、工作原理以及工程挑战等内容。在后续章节中，我们将逐步深入推荐系统最常见的级联架构，系统介绍召回、粗排、精排、重排、混排以及生成式推荐等核心模块，并进一步分析各模块中的经典模型、关键算法及其演进过程。同时，我们还将讨论工业界和学术界长期关注的重要研究问题，以及大语言模型浪潮下推荐系统领域正在发生的新变化。

如图1.1所示，为了更直观地理解推荐系统的作用，我们以短视频平台为例进行说明。如今，一个大型短视频平台每天可能拥有数亿活跃用户，同时平台内容库中存储着数十亿甚至上百亿规模的视频内容。对于任何一个用户而言，其每天能够消费的内容数量却极其有限，通常只有几十到数百条视频。这意味着平台必须从海量内容中快速筛选出极少量最有价值的内容呈现给用户。如果不存在推荐系统，那么对于数亿用户和数十亿内容而言，平台需要在用户与内容之间进行近似笛卡尔积规模的匹配计算，其计算复杂度将达到天文数字级别：

$$O(User \times Item) \tag{1.1}$$

图 1.1: 推荐系统是链接用户和内容的桥梁。

这样的计算规模在工业实践中显然是无法接受的。

推荐系统正是在这种背景下诞生的。它通过多阶段筛选、兴趣建模以及排序优化等技术手段，在保证计算效率的同时，尽可能提升用户与内容之间的匹配质量，从而实现大规模内容分发。更进一步地说，推荐系统的重要性已经远远超出了“内容推荐”本身。对于现代互联网平台而言，**推荐系统实际上掌握着平台最核心的流量分配权**。它不仅决定用户能够看到什么内容，也会间接影响用户兴趣的形成、内容创作者的成长路径、社区生态的发展方向以及平台商业化能力的提升。

因此，推荐系统不仅是一个算法问题，也不仅是一个工程问题，而是一个融合了人工智能、系统工程、产品设计、商业目标以及生态治理的复杂决策系统。理解推荐系统，实际上是在理解现代互联网平台如何利用数据与算法组织和分配注意力资源。

1.2 推荐系统发展历程

推荐系统并不是近年来随着人工智能兴起才出现的新技术。事实上，早在互联网发展的早期阶段，人们便已经开始尝试利用各种方法帮助用户从海量信息中发现自己感兴趣的内容。随着互联网规模、内容规模以及用户规模的不断增长，推荐系统也经历了从人工运营到机器学习、从深度学习到大模型驱动的发展过程。从整体上来看，推荐系统的发展历程如图1.2所示。其演化过程大致可以划分为四个阶段：人工推荐时代、机器学习时代、深度学习时代以及大模型时代。每一个阶段都对应着不同的互联网发展背景、数据规模、算力水平以及算法范式。

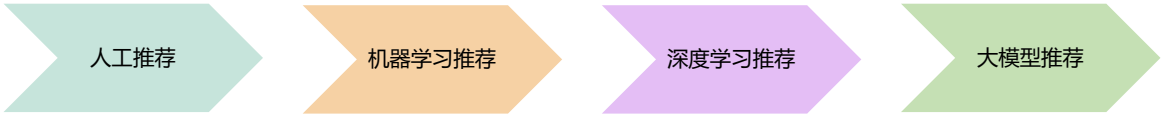

图 1.2: 推荐系统的发展历程。

1.2.1 人工推荐时代

在互联网发展的早期阶段，网站和应用中的内容规模相对有限，用户数量也远未达到今天的规模。此时，大多数内容分发主要依赖编辑运营和人工推荐完成。例如早期门户网站首页的新闻栏目、论坛社区中的精华帖推荐以及视频网站中的热门内容推荐，本质上都属于人工推荐。运营人员根据自身经验和平台策略挑选内容，并将其展示给所有用户。这种推荐方式的优点在于简单直接，能够保证内容质量和平台导向。但随着用户规模和内容规模不断增长，其缺点也逐渐显现出来：

- 无法满足不同用户的个性化需求；
- 推荐结果高度依赖人工经验；
- 内容规模扩大后人工筛选成本急剧增加；
- 推荐效率难以支撑大规模内容分发。

因此，当互联网开始进入大规模增长阶段后，人工推荐逐渐难以满足平台发展的需求。

1.2.2 机器学习时代

进入 2010 年前后，移动互联网开始快速发展，大量互联网企业相继诞生。用户规模和内容规模呈现爆发式增长，传统人工推荐已经无法支撑海量信息分发。以短视频、资讯、电商等场景为例，一个平台可能拥有数千万甚至上亿用户，同时每天新增海量内容。如果仍然依赖人工筛选和分发，将面临极其高昂的人力成本和极低的分发效率。

在这一阶段，推荐系统开始广泛采用机器学习方法进行自动化推荐。协同过滤（Collaborative Filtering）、矩阵分解（Matrix Factorization）、逻辑回归（Logistic Regression）、GBDT 等方法逐渐成为推荐系统的重要组成部分。这一时期推荐系统最核心的思想是：

定义 1.2

利用用户历史行为数据自动学习用户兴趣，从而实现个性化推荐。 ♣

推荐系统开始从“运营驱动”逐步转向“数据驱动”，个性化推荐也逐渐成为互联网产品的重要竞争力。

1.2.3 深度学习时代

随着互联网进一步发展，用户行为数据和内容数据规模持续增长，传统机器学习方法逐渐面临表达能力不足的问题。与此同时，GPU 计算能力的快速提升以及深度学习技术的突破推动推荐系统进入新的发展阶段。自 2016 年前后开始，以 Wide&Deep、DeepFM、DIN、DIEN 等为代表的大量深度学习推荐模型相继出现。推荐系统开始利用深层神经网络学习用户兴趣、内容特征以及复杂的用户行为模式。

这一时期推荐系统的核心特点包括：

- 从人工特征工程逐步转向自动特征学习；
- 用户兴趣建模能力显著增强；
- 多任务学习广泛应用于工业系统；
- 推荐系统逐渐形成召回、粗排、精排、重排等多阶段级联架构。

与此同时，短视频平台、直播平台以及内容社区的快速崛起，使推荐系统从传统的信息过滤工具逐渐演变为互联网平台最核心的流量分发基础设施。在这一阶段，推荐系统已经不仅仅影响内容曝光效率，而开始深刻影响用户行为、创作者生态以及平台的商业化变现能力。

1.2.4 大模型时代

近几年来，以 Transformer 和大语言模型（Large Language Model, LLM）为代表的生成式人工智能技术取得突破性进展。特别是在 Scaling Law 和 Next Token Prediction 训练范式的推动下，大模型展现出了强大的知识表示、涌现能力和序列建模能力。推荐系统也开始进入大模型和 Agent 驱动的新阶段。一方面，大模型被广泛应用于内容理解、用户画像构建、Embedding 生成以及特征增强等任务；另一方面，越来越多的研究开始探索生成式推荐（Generative Recommendation）、LLM4Rec 以及 Agent Recommendation 等新型推荐范式，希望利用统一的大模型完成用户兴趣建模与推荐决策。

与传统推荐系统相比，大模型时代的推荐系统呈现出以下几个特点：

- 推荐与自然语言理解逐渐融合；
- 推荐与搜索开始走向统一；
- 生成式推荐逐渐成为新的研究热点；
- Agent 开始参与用户决策与推荐流程；
- 推荐系统逐步向通用智能决策系统演进。

1.2.5 推荐系统发展的核心驱动力

回顾整个发展历程可以发现，推荐系统的每一次技术跃迁背后都存在三个重要驱动力：

1. 用户规模增长带来的个性化需求；
2. 内容规模增长带来的信息过载问题；
3. 算力提升带来的模型能力突破。

从人工推荐到机器学习，从深度学习到大模型，本质上都是为了在不断增长的用户规模和内容规模下，实现更加精准、高效和智能的用户与内容匹配。因此，推荐系统的发展史不仅是一部算法演化史，也是一部互联网内容分发方式不断升级的历史。随着大模型、Agent 以及具身智能等新技术的发展，推荐系统未来仍将持续演进，并在连接用户与信息的过程中发挥越来越重要的作用。

1.3 推荐系统的核心目标

很多推荐系统初学者在学习过程中，往往会将注意力集中在各种算法名词上，例如 CF、FM、DeepFM、DIN、MMoE，Transformer 等，并认为推荐系统的核心任务就是不断优化模型效果。然而在真实的工业场景中，模型从来都不是最终目标，模型只是实现业务目标的一种手段。

对于互联网企业而言，推荐系统本质上是一个服务于业务增长的智能决策系统。无论采用何种算法、何种模型架构，其最终目的都是帮助平台实现特定的业务价值。因此，在学习推荐系统之前，我们首先需要回答一个非常重要的问题：

笔记 *推荐系统究竟在优化什么？*

从宏观角度来看，一个成熟的推荐系统通常需要同时兼顾用户、平台以及内容生态三个方面的价值，因此推荐系统本质上是一个典型的多目标优化（Multi-Objective Optimization）问题。

1.3.1 用户价值

用户价值（User Value）是推荐系统最核心的优化目标之一。推荐系统存在的根本原因，就是帮助用户在海量信息中快速发现自己感兴趣的内容，从而降低信息获取成本，提升用户体验。从短期来看，用户价值通常体现在用户产生的各种**即时反馈行为**上，例如：点击率（CTR），视频播放时长（Watch Time），点赞率（Like Rate），评论率（Comment Rate），收藏率（Favorite Rate），分享率（Share Rate）。这些行为能够直接反映用户是否喜欢当前推荐内容。而从长期来看，平台更加关注**用户长期价值**，例如：用户留存率（Retention），日活跃用户数（Daily Active User，DAU），月活跃用户数（Monthly Active User，MAU），用户生命周期价值（Lifetime Value，LTV），用户满意度（User Satisfaction）。一个优秀的推荐系统不仅能够提升用户当前的点击和观看行为，更能够持续提升用户长期活跃度和忠诚度。

1.3.2 平台价值

推荐系统不仅服务于用户，同时也是互联网平台最重要的增长引擎和营收引擎之一。对于平台而言，推荐系统承担着流量分配和资源调度的重要职责，其优化目标通常包括：提高用户活跃度（DAU），提高用户留存率（Retention），提升平台整体使用时长（App Time），提升平台商业化收入（Revenue），提高流量利用效率，提升平台整体增长速度。例如在短视频平台中，推荐系统会努力提升用户观看时长和用户留存；在广告平台中，则更加关注广告收入和广告消耗；在电商平台中，则更关注订单成交和商品交易规模。因此，同一个推荐算法在不同业务场景下，其优化目标往往并不完全相同。

1.3.3 内容生态价值

除了用户和平台之外，推荐系统还会深刻影响内容生态的发展。在内容社区中，推荐系统实际上掌握着平台最重要的流量分配权。它决定哪些内容能够获得曝光，哪些创作者能够获得成长机会，也决定整个社区最终

会形成怎样的内容生态。因此，一个健康的推荐系统通常还需要关注：新作者成长，优质内容扶持，内容多样性（Diversity），内容公平性（Fairness），社区生态健康度。如果推荐系统只追求短期点击率，可能会导致内容同质化、标题党泛滥不断博人眼球以及内容生态失衡等问题。因此，现代推荐系统往往需要在用户体验、平台收益以及生态建设之间进行长期平衡。

1.3.4 不同业务场景的核心目标

不同业务场景关注的核心目标并不相同。以短视频平台为例，视频内容通常是整个产品的核心载体，因此推荐系统更加关注：视频播放时长（Watch Time），视频曝光（Video Views），App 使用时长（App Time），用户留存率（Retention），DAU 。对于广告业务而言，更关注商业化收益相关指标，例如：CPM（Cost Per Mille），CPC（Cost Per Click），CPA（Cost Per Acquisition），CTR（Click Through Rate），CVR（Conversion Rate），广告总消耗（Spend）等。对于直播业务而言，推荐系统则更关注：直播 DAU，直播观看时长，直播 CTR，打赏用户数，打赏总金额等。而对于电商业务，推荐系统最终需要促进商品成交，因此更加关注：商品交易总额（GMV），千次曝光订单数（OPM），下单率，加购率，支付转化率。可以看到，虽然不同业务使用的推荐技术体系高度相似，但由于业务目标不同，其最终优化方向往往存在明显差异。

为了方便大家理解，这里给出 GMV、OPM、CPM、CPC、CPA 这些术语的具体定义和计算公式。其中，GMV 的定义如下：

定义 1.3 (GMV)

GMV 即商品交易总额，是指在一定时间内，通过平台成交的订单总金额（通常包含已付款 + 未付款但已下单的订单）。具体计算公式为：

$$GMV = \sum(\text{商品单价} \times \text{购买数量}) \tag{1.2}$$

OPM 的定义如下：

定义 1.4 (OPM)

OPM 即千次展示订单数，是指每 1000 次内容 / 商品展示带来的订单数量。具体计算公式为：

$$OPM = \frac{\text{总订单数}}{\text{总展示次数}} \times 1000 \tag{1.3}$$

OPM 与 GMV 的关系可以用如下公式表示：

$$GMV = OPM \times \text{展示次数} \times \text{客单价}/1000 \tag{1.4}$$

上述两个指标 GMV 和 OPM 是电商推荐业务中非常重要的业务目标。需要注意 GMV 并不是平台最终的收入，平台收入 $= GMV \times$ 佣金率（或广告费）。除此之外，GMV 也包含了退款和取消订单的金额，如果退款率很高的话，GMV 也会存在虚高的问题。对于电商业务来讲，推荐系统需要进行具备精准且极致的流量分发机制，确保能够最大化整个电商业务的成交金额，同时也需要关注流量分发的效率问题，不能搞“大水漫灌”，因此 OPM 指标可以精准地衡量电商推荐中的流量效率，OPM 越大，说明用户看到电商的内容之后更容易下单，转化效率也更高。

而 CPM、CPC、CPA 都是广告场景中的业务目标，具体定义如下：

定义 1.5 (CPM)

CPM 即千次展示成本，是指广告主为每 1000 次广告展示所支付的费用。具体计算公式为：

$$CPM = \frac{\text{广告总花费}}{\text{总展示次数}} \times 1000 \tag{1.5}$$

CPC 的具体定义如下：

定义 1.6 (CPC)

CPC 即每次点击成本，是指广告主为每次用户点击广告所支付的费用。具体计算公式为：

$$CPC = \frac{\text{广告总花费}}{\text{总点击次数}} \tag{1.6}$$

CPA 的定义如下：

定义 1.7 (CPA)

CPA 即每次转化成本，是指广告主为每次有效转化行为（如下单、注册、提交表单）所支付的费用。具体计算公式为：

$$CPA = \frac{\text{广告总花费}}{\text{总转化次数}} \tag{1.7}$$

这里需要注意，CPM、CPC 和 CPA 代表了广告领域中几种不同的计费模式，它们的区别不仅在于计算方式，更在于业务目标、风险承担和优化逻辑。它们的具体区别如下表 1.1所示。

表 1.1: CPM、CPC 与 CPA 的具体区别。

维度	CPM	CPC	CPA
计费基础	曝光	点击	转化
广告主风险	低（展示就付费）	中（点击可能无效）	极低（只为结果付费）
平台风险	低	中	高（需保证转化）
适用广告类型	品牌广告、开屏广告、视频前贴片	搜索广告、信息流广告	效果广告、电商、游戏
是否需要预估模型	否	是（需预估 CTR）	是（需预估 CTR+CVR）
优化目标	提升曝光量、人群覆盖	提升点击率	控制成本、提升 ROI

接下来我们简要介绍一下 CPM、CPC 和 CPA 与广告模型经常需要预估的 CTR 与 CVR 的关系。我们知道：

$$\begin{aligned} CTR &= \frac{\text{点击数}}{\text{曝光数}} \\ CVR &= \frac{\text{转化数}}{\text{点击数}} \end{aligned} \tag{1.8}$$

那么有如下等式成立：

$$\begin{aligned} CPC &= CPA \times CVR \\ CPM &= CPC \times CTR \times 1000 \\ CPM &= CPA \times CVR \times CTR \times 1000 \end{aligned} \tag{1.9}$$

从公式 1.9 我们可以看到，要优化的 CPM 这个广告的业务指标，最终会和广告的 CTR、CVR 这些指标联系起来，这两个指标也是广告算法模型持续去优化的指标。

上述的 CPM，CPC，CPA 都是广告主侧计费相关的指标。而对于平台侧，通常会关注 eCPM(effective Cost Per Mille) 的指标，具体定义如下：

定义 1.8 (eCPM)

eCPM 即千次展示有效收益，指的是每一千次广告曝光，平台能赚到的收入，是广告业务最核心的收益指标之一。具体计算公式为：

$$eCPM = \frac{\text{广告总收入}}{\text{总曝光次数}} \times 1000 \tag{1.10}$$

eCPM 指标通常是指千次广告曝光之后的平台实际收入用来衡量广告的流量价值，而 CPM 是广告主侧使用的指标，通常指广告主愿意为 1000 次曝光付多少钱。这两个指标的衡量视角是有所差异的。在广告推荐系统的

竞价排序中，同时会根据预估的 eCPM 进行排序，如果我们采用最简单的 CPC 计费模式，那么预估的 eCPM 为：

$$eCPM_{预估} = CTR \times bid \times 1000 \tag{1.11}$$

其中 bid 是广告主设置单次点击最高愿意出的价格（预想出价），但并不等于最终结算扣费之后的真实广告成本。如果我们采用 CPA 计费模式，那么预估的 eCPM 为：

$$eCPM_{预估} = CTR \times CVR \times bid \times 1000 \tag{1.12}$$

这里的 bid 是广告主设置的单次转化出价（CPA），与 CPC 计费模式下的 bid 含义有所不同。

1.3.5 北极星指标

在工业界，大家经常会提到一个非常重要的概念——北极星指标（North Star Metric）。所谓北极星指标，是指能够最直接反映业务长期价值的核心指标。推荐系统中的所有模型、策略和算法优化，最终都应该服务于北极星指标的提升。例如：

- 短视频业务的北极星指标可能是观看时长和用户留存；
- 广告业务的北极星指标可能是广告收入和 ROI；
- 电商业务的北极星指标可能是 GMV 和订单量；
- 社区产品的北极星指标可能是用户活跃度和内容消费时长。

需要特别强调的是：

定义 1.9

推荐系统优化的不是模型，而是业务目标；模型只是实现业务目标的工具。

很多初学者容易陷入“只关注模型指标”的误区，例如不断追求 AUC、LogLoss 等离线指标的提升，却忽略这些指标是否真正能够带来业务增长。在工业实践中，一个能够显著提升核心业务指标的简单策略，其价值往往远远高于一个仅仅提升少量离线指标的复杂模型。因此，在学习推荐系统的过程中，读者不仅需要掌握各种模型和算法，更需要始终思考：

笔记 这个模型究竟在解决什么业务问题？

它最终优化的业务目标是什么？

只有理解这一点，才能真正理解推荐系统的本质。

1.4 推荐系统是如何工作的

在前几节中，我们已经了解到：

推荐系统本质上是连接用户与内容之间的桥梁。

然而，仅仅理解这一点还远远不够。读者可能会进一步产生疑问：

笔记 推荐系统在整个软件系统中究竟处于什么位置？

用户每刷到一个视频，推荐系统都做了哪些事情？

推荐结果又是如何一步一步计算出来的？

为了回答这些问题，本节将从系统架构的宏观视角出发，介绍推荐系统在实际工业场景中的工作流程，帮助读者建立对推荐系统运行机制的整体认知。以短视频平台为例，当用户打开 App 开始浏览内容时，我们通常将用户从打开 App 到关闭 App 之间的这一段连续使用过程称为一次**会话（Session）**。在一次 Session 中，用户可能连续浏览几十条甚至上百条视频。对于用户而言，整个过程似乎是视频不断地自动出现在屏幕上，但从系统实现角度来看，推荐系统并不是每展示一个视频就向服务器发起一次请求。如果用户每看完一个视频都立即向

推荐系统发起请求，那么服务器将面临极高的网络通信开销和计算压力，同时也会增加用户等待时间，影响整体使用体验。

因此，在工业界实际系统中，推荐结果通常采用**批量返回（Batch Recommendation）**的方式进行下发。具体来说，当用户进入推荐页面时，客户端会向服务端发起一次推荐请求。服务端经过推荐系统计算后，通常会一次性返回若干条推荐内容，例如 10 条左右的视频。随后用户不断向下滑动浏览这些视频。当用户即将浏览完当前缓存的视频列表时，客户端会提前发起下一次推荐请求，系统再返回新的一批推荐结果。整个过程不断循环，从而实现用户侧近乎无感知的连续内容消费体验。从用户视角来看，视频似乎是源源不断地出现；而从系统视角来看，本质上是**一次又一次推荐请求不断驱动整个推荐系统运行**。

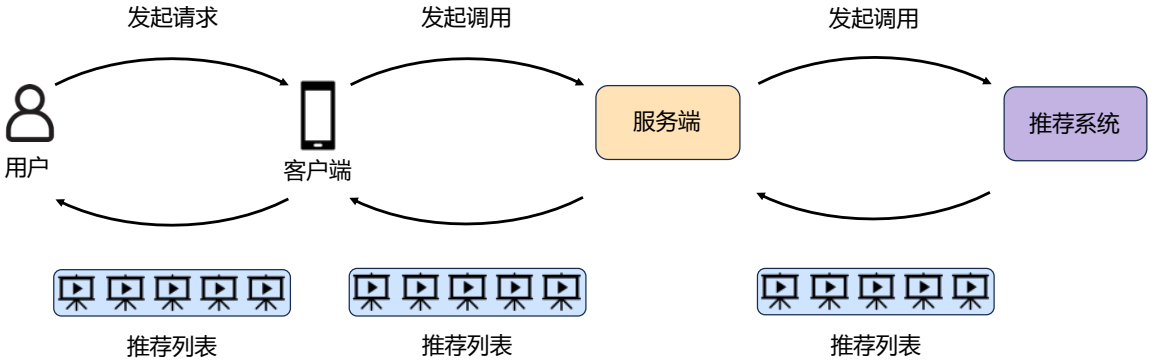

图 1.3: 一次完整的用户推荐请求流程。

整个推荐请求过程如图1.3所示。在整个调用链路中，首先与用户直接交互的是客户端（Client），即用户手机上的 App。当用户产生刷视频、点击、点赞等行为时，客户端会将相关请求发送给服务端（Server）。服务端接收到请求后，会完成请求解析、用户信息补充、上下文特征构建以及权限校验等工作，并进一步调用推荐系统服务。推荐系统接收到请求后，会结合用户画像、历史行为、实时特征以及海量候选内容进行复杂计算，最终生成当前最适合用户的一批推荐结果。随后，推荐结果会返回给服务端。服务端完成必要的数据封装和业务处理后，再将最终结果返回给客户端进行展示。

因此，从宏观上看，一次完整的推荐请求流程可以表示为：

$$User \rightarrow Client \rightarrow Server \rightarrow Recommendation\ System \rightarrow Server \rightarrow Client \rightarrow User \tag{1.13}$$

虽然这一流程看起来比较简单，但其背后往往涉及数百台甚至成千上万台服务器协同完成计算，是一个典型的大规模分布式系统工程。需要特别注意的是，对于推荐系统而言，除了推荐结果的准确性之外，响应速度同样至关重要。

从用户发起请求到最终看到推荐结果，整个过程通常需要在几百毫秒内完成。对于短视频、直播等实时交互场景而言，推荐延迟甚至会被严格控制在 100 毫秒至 300 毫秒级别。如果某个推荐模型的在线推理耗时过高，即使模型精度有所提升，也可能导致用户等待时间增加，从而影响整体使用体验。用户通常不会感知到模型是否更加先进，但会立即感知到系统是否流畅。

因此，在工业界的推荐系统建设过程中，往往需要在推荐效果与系统性能之间进行平衡。一方面，推荐算法团队持续优化模型和策略，以提升点击率、观看时长、GMV 等业务指标；另一方面，推荐架构团队则需要持续优化系统架构、计算资源利用率以及在线推理效率，确保推荐服务能够在极低延迟下稳定运行。正是算法、工程以及系统架构三者的协同配合，才共同构成了现代互联网产品背后复杂而高效的推荐系统。

1.5 推荐系统的挑战

前面的章节中我们已经介绍了推荐系统的基本概念、核心目标以及整体工作流程。从表面上看，推荐系统似乎只是根据用户兴趣推荐内容，但在真实工业场景中，构建一个能够稳定服务亿级用户的大规模推荐系统远

比想象中复杂。推荐系统不仅需要保证推荐结果的准确性，还需要同时满足**高并发、低延迟、强实时性以及复杂业务目标**等多方面要求。因此，推荐系统既是一个算法问题，也是一个复杂的系统工程问题。

总体来看，工业级推荐系统主要面临以下几个核心挑战：

- 海量数据与海量内容带来的计算挑战；
- 高并发场景下的实时响应挑战；
- 多业务场景下的多目标优化挑战；
- 新用户与新内容带来的冷启动挑战。

1.5.1 海量数据挑战

推荐系统首先面临的是数据规模带来的巨大挑战。以大型短视频平台为例，平台往往需要服务数亿活跃用户，同时内容库规模可能达到数十亿甚至上百亿级别。用户每天产生的点击、播放、点赞、评论、分享等行为数据更是达到数十亿至上百亿条。如果采用传统用户-物品交互矩阵的方式进行存储，那么系统需要维护一个规模接近：$10^8 \times 10^9$ 的超大规模矩阵。无论是存储成本还是计算成本，这种方案在工业实践中都难以落地。因此，工业界通常不会直接存储完整的用户-物品交互矩阵，而是采用以用户为中心的行为序列存储模式。

具体而言，每个用户 ID 作为主键（Key），对应一个用户行为序列（User Action List）作为值（Value），记录用户历史浏览、点击、点赞、收藏、购买等行为。这种存储方式不仅能够显著降低存储成本，同时也更加符合现代推荐模型的输入形式。目前广泛应用的 DIN、DIEN、SASRec 等模型，本质上都建立在用户行为序列建模的基础之上。

与此同时，海量样本也给模型训练带来了巨大压力。面对每天新增的数十亿级训练样本，单机训练已经无法满足工业需求。因此工业界普遍采用分布式训练架构，通过 Parameter Server、AllReduce 等技术实现海量数据的并行训练，从而保证模型能够在合理时间内完成迭代更新。

1.5.2 实时响应挑战

除了海量数据之外，推荐系统还需要满足极高的在线响应速度要求。对于用户而言，从发起请求到看到推荐结果，整个过程通常只能容忍几百毫秒的延迟。如果推荐系统响应过慢，用户会明显感受到页面卡顿，从而影响整体使用体验。然而，推荐系统所面对的内容规模却往往达到十亿级别。如果对全量内容逐一调用复杂深度模型进行打分，其计算量将难以接受。因此工业界普遍采用多阶段级联架构（Multi-stage Cascade Architecture）来解决这一问题。

整个推荐流程通常包含：召回（Retrieval），粗排（Pre-ranking），精排（Ranking），重排（Re-ranking）。其中：

- 召回负责从十亿级内容中快速筛选数千个候选；
- 粗排进一步筛选出数百个高质量候选；
- 精排利用复杂模型进行精准排序；
- 重排结合业务规则进行最终调整。

这种“候选集逐级缩小、模型复杂度逐级提升”的设计，使推荐系统能够在有限时间内完成大规模内容筛选。因此，级联架构实际上是工业界在推荐效果与计算成本之间长期权衡的结果。

1.5.3 多目标优化挑战

推荐系统并不是一个单纯追求点击率最大化的问题。在真实业务场景中，推荐系统往往需要同时服务多个目标。例如：

- 用户希望看到感兴趣的内容；
- 平台希望提高留存与商业化收入；
- 创作者希望获得更多曝光机会；
- 社区希望保持内容生态健康。

然而这些目标之间并不总是一致的。

例如：

- 增加广告曝光可能提升收入，但会损害用户体验；
- 过度追求点击率可能导致标题党泛滥；
- 过度推荐头部作者可能压缩新作者成长空间。

因此，推荐系统本质上是一个典型的**多目标优化（Multi-Objective Optimization）问题**。工业界通常会通过多任务学习（Multi-Task Learning）、流量调控等技术，在用户价值、平台价值以及生态价值之间寻找平衡。事实上，许多推荐系统团队花费最多精力解决的问题并不是模型精度，而是如何协调不同业务目标之间的冲突。

1.5.4 冷启动挑战

冷启动（Cold Start）是推荐系统长期存在的经典问题。推荐系统的大部分能力来源于历史行为数据。当用户产生足够多的行为之后，系统可以逐渐学习其兴趣偏好；同样，当内容获得足够多曝光之后，系统也能够较准确地评估其质量。然而，对于新用户和新内容而言，历史数据往往非常有限甚至完全不存在。例如：

- 新注册用户没有浏览历史；
- 新发布视频没有点击和播放数据；
- 新商品没有成交记录；
- 新直播间没有用户互动数据。

这种缺乏历史反馈信息的情况被称为冷启动问题。为了缓解冷启动带来的影响，工业界通常会采用：

- 内容特征建模；
- 用户画像初始化；
- 探索与利用（Exploration & Exploitation）机制；
- 大模型内容理解与语义建模。

随着大语言模型的发展，利用文本、图像、视频等多模态信息直接理解内容语义，也逐渐成为解决冷启动问题的重要方向。

1.6 本章小结

本章首先阐释了推荐系统的定义与核心价值，点明其作为连接用户与内容的核心桥梁，旨在实现海量用户与海量内容之间高效、精准的匹配。随后梳理了推荐系统的四大发展阶段：人工推荐、机器学习推荐、深度学习推荐与大模型推荐，并总结出技术演进的三大核心驱动力，即用户规模扩张带来的个性化需求、信息过载问题，以及算力升级带来的模型能力突破。

接着，本章从用户、平台、内容生态三个维度拆解了推荐系统的多元优化目标，结合短视频、广告、电商等典型业务场景，讲解了各类核心业务指标的定义以及在工业界推荐优化中北极星指标的具体内涵，同时纠正了“唯模型指标论”的认知误区，强调模型只是落地业务目标的工具，在推荐系统中策略算法同样也发挥着重要作用。之后以真实业务场景为例，完整介绍了工业级推荐系统的整体工作流程与请求链路，分析了系统在**高准确率**与**低延迟**之间的取舍逻辑，进而演化出了工业界最常见的**多阶段级联式推荐架构**。最后，本章总结了落地大规模推荐系统需要面对的四大核心挑战：海量数据存储与训练、高并发下的实时响应、多目标冲突优化以及新用户与新内容的冷启动问题。