**说 明 书**

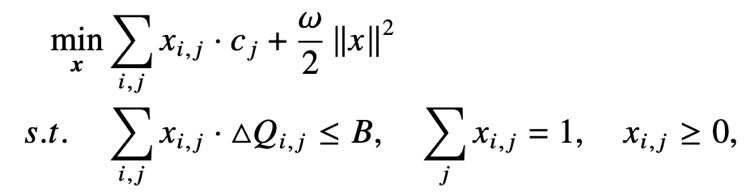
**多阶段算力回收框架**

**技术领域**

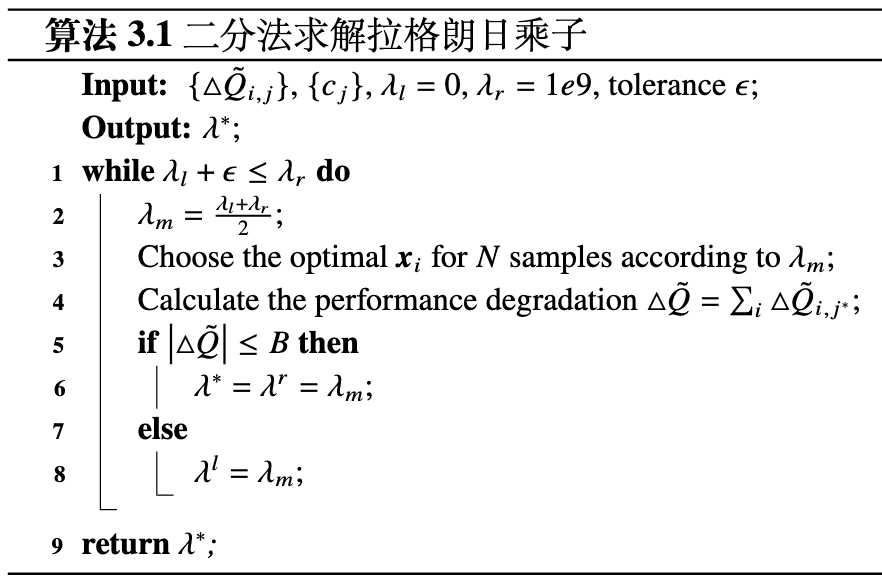
1. 本发明涉及的是一种推荐系统领域的方法，具体是一种多阶段智能算力回收方法，在级联式推荐系统中缓存各个阶段模型的输出结果，并根据本发明设计的算法在适时读取缓存复用，达到保持推荐效果不变，提高系统资源利用率的作用。

**背景技术**

1. 推荐技术是互联网平台的重要基石，它对平台的运营和用户的留存具有深远影响。目前，工业界推荐系统部署了越来越多复杂的推荐模型来提升系统的推荐效果，却鲜少有研究进行系统的效率提升研究。
2. 随着算力投入对效果提升的边际效益递减效应越来越显著，我们应更优先关注系统的资源利用率。然而，虽然现有的级联式推荐系统架构将单条流量上的效率优化到了极致——可以在数百甚至数十毫秒内响应用户的请求并返回结果，它却为多条流量的分配同样的算力资源。因此现有推荐系统在多条用户流量上仍存在效率优化的空间。
3. 我们先在工业界推荐系统中进行资源利用率分析，发现存在算力资源浪费的现象。具体来说，在我们的场景中，用户可能在一次会话（Session）中向服务器发起多次请求。对于每个请求，在线推荐系统会完整运行级联式推荐模型，包括召回、粗排、精排和重排模型，以生成高质量的推荐。
4. 然而，这种“公平”的算力分配方法忽略了请求之间的重复性——系统为用户多次请求的返回结果可能相同；也忽略了不同请求的价值不同，即不同请求投入资源获得的回报比（ROI）不同。
5. 目前，有一些工作进行了智能算力的研究。这些工作提出了“弹性队列”、“弹性通道”、“弹性模型”作为手段对算力分配进行调控。具体来说，这些调控手段包括调整推荐模型的候选集合大小、调整召回通道数、调整精排模型的复杂性，实现对不同请求分配不一样的算力资源。
6. 然而这些工作仅从调整算力资源分配的角度考虑提高系统资源利用率的问题，并未从“回收”浪费的“资源”考虑新的解决方案。
7. **发明内容**
8. 本发明从全新的视角考虑提高系统资源利用率的问题。具体来说，本发明提出了多阶段算力回收框架，对不同阶段推荐模型的输出进行缓存，再根据本发明设计的算法，对缓存结果再次利用。从而实现请求个性化的算力资源分配，以及算力资源的回收利用，达到推荐效果不变，推荐系统资源利用率获得提升。
9. 本发明是通过以下技术方案实现的：
10. 本发明涉及一种多阶段智能算力回收框架，包括回收策略设计、离线学习与在线服务三部分：
11. 回收策略设计部分，用于设计细粒度的算力回收策略，同时指导了多阶段智能算力回收框架的设立，具体包括；
12. 步骤1、 根据级联式推荐系统架构，在每一处推荐模型做输出结果的缓存。
13. 所述的级联式推荐系统架构通常包括召回模型、粗排模型、精排模型、重排模型，前一处模型的输出结果是后一模型的输入候选集合，实现流量的高效处理。
14. 所述的做结果缓存可以用不同的缓存策略，例如：缓存所有结果；缓存部分结果；或缓存个性化数量的结果。
15. 步骤2、 读取缓存，并将读取的缓存结果作为候选物品集合，送入下一处推荐模型。
16. 具体来说，假设有 M 阶段推荐模型，由于缓存了多处推荐模型的结果，那么可以有 M+1 种算力回收策略：1. 不采用算力回收，使用完整的级联式推荐系统架构；2. 读取某一阶段推荐模型的缓存，并将缓存作为候选物品集合送入下一阶段推荐模型，此处共有 M 中算力回收策略。
17. 步骤3、 计算不同算力回收策略的算力开销 。
18. 所述的算力开销为常量，可离线计算。算力开销可以认为是浮点运算量（FLOPS）也可以是响应延时（RT）还可以是CPU核数。
19. 离线学习部分，用于训练在线服务所使用的效果损耗预估装置及计算策略打分参数，具体包括：
20. 步骤1、 利用采集的数据训练效果损耗预估装置。
21. 所述的采集的数据为不同用户请求采用不同算力回收策略的推荐效果，包含描述静态用户信息的用户特征、描述请求冗余性的请求特征。
22. 所述的效果损耗预估装置为一类深度学习模型，可以预测对某个请求采用 M+1 种算力回收策略可能导致推荐效果的损耗量。
23. 步骤2、 对在线算力回收策略决策问题进行问题建模，并对建模的数学问题进行对偶问题求解，得出实现推荐效果基本持平前提下提高资源利用率的策略打分公式。
24. 所述的问题建模具体为：建模成约束最优化问题。假设共有 N 次广告请求，请求 i 采用算力回收策略 j 时的效果损耗为 ，系统要求推荐系统损耗不超过阈值 B。策略分配 表示算法决策请求 i 采用算力回收策略j，则问题建模为



1. 所述的打分公式为 ，其中 为拉格朗日乘子，需要离线计算最优取值。
2. 步骤4、 二分求解最优拉格朗日乘子。
3. 所述的二分求解法包含如下流程：
4. 1. 收集时间段内的请求日志；
5. 2. 根据图示二分法进行最优拉格朗日乘子求解。



1. 在线服务部分，用于服务在线用户请求，具体包括：
2. 步骤1、 将在线到来的用户请求及请求相关信息送入请求过滤装置，将请求分类为初始请求和重复请求。
3. 所述的用户请求指用户操作应用软件或浏览网页时，请求新的物品包括但不限于商品、音乐、视频等。
4. 所述的请求相关信息指可以用于判断请求重复性的客观信息，包括但不限于距前次请求的时间间隔、前次请求物品曝光率等。
5. 步骤2、 若该请求为初始请求，则调用完整的推荐链路，包括召回模型、粗排模型、精排模型、召回模型，生成物品推荐；同时，将各模型的输出结果进行缓存；之后结束服务。
6. 步骤3、 若该请求为重复请求，则利用离线训得的效果损耗预估装置与策略打分参数，计算不同策略的打分。
7. 所述的利用与计算方法为：
8. 1. 将模型所需特征送入效果损耗预估装置，用模型预测不同策略的效果损耗；
9. 2. 将预估值及打分参数带入打分公式，得到不同算力回收策略的打分。
10. 步骤4、 选取步骤3中打分最高的策略，并在推荐链路实施回收策略设计部分中的对应策略，进行缓存读取及复用。

**技术效果**

1. 本发明与现有技术相比，其优点表现为：提出了新的算力调控手段，第一次实现算力回收利用；利用约束最优化问题建模，实现效果持平条件下的算力资源利用率提升；提出全新的多阶段智能算力回收框架，实现多条请求流量间的效率提升。

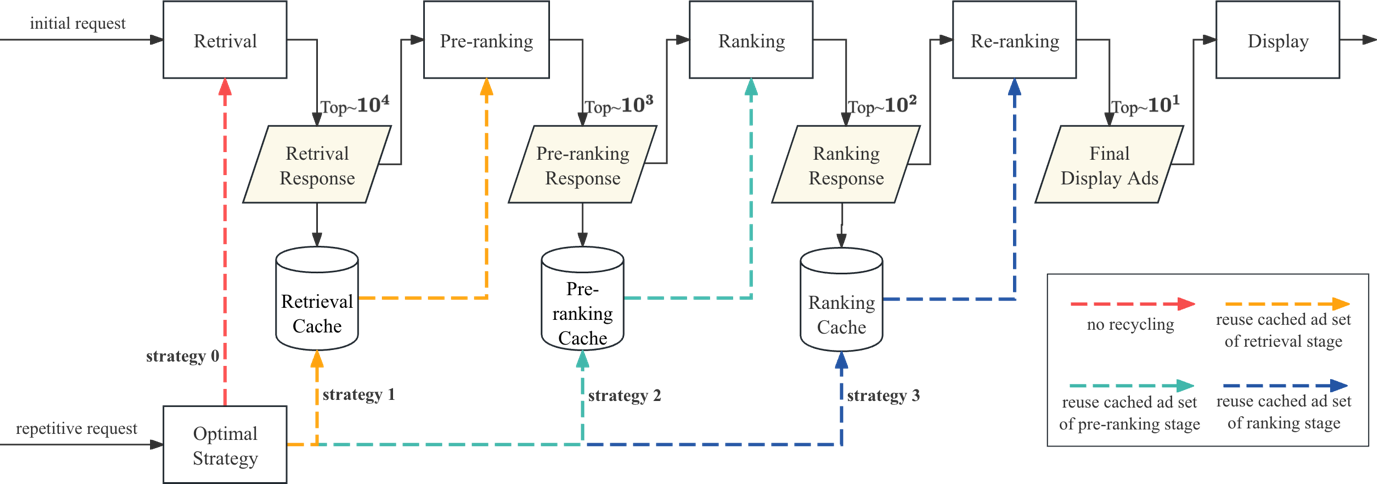
**附图说明**

**具体实施方式**

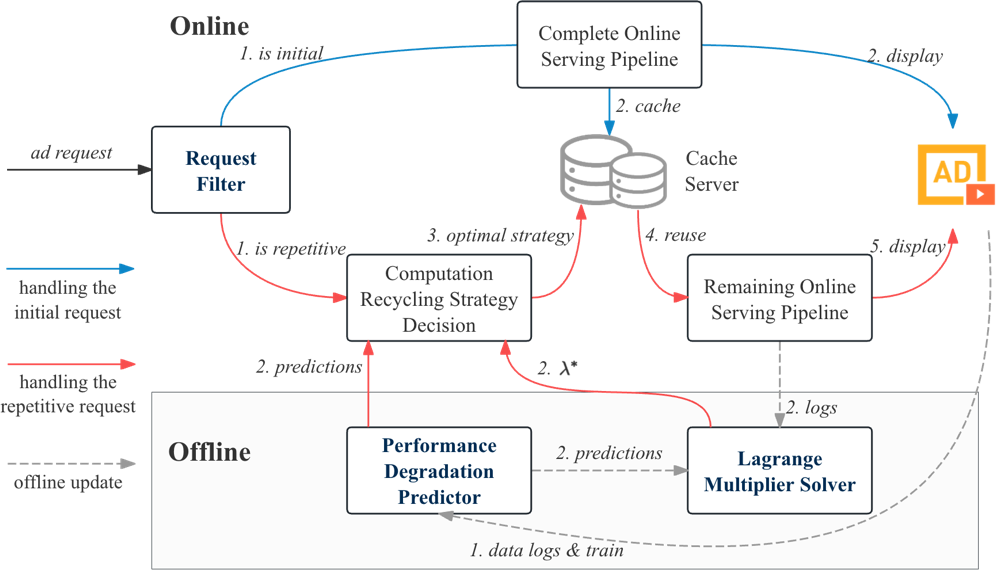
1. 下面对本发明的实施例作详细说明，本实施例在以本发明技术方案为前提下进行实施，给出了详细的实施方式和具体的操作过程，但本发明的保护范围不限于下述的实施例。

**实施例1**

1. 本实施例设计的多阶段智能算力回收工程链路如图，表示了四种策略对应的工程链路：



1. 本实施例设计的多阶段智能算力算法流程如图，各部分设计具体如下：



1. 回收策略设计部分：
2. 步骤1、 设计四种算力回收策略，包括不进行算力回收、复用召回缓存结果、复用粗排缓存结果、复用精排缓存结果。
3. 步骤2、 利用系统统计的响应时间作为算力开销 。
4. 离线学习部分：
5. 步骤1、 设计了如图的效果损耗预估模型。具体来说，模型采用用户侧特征如性别、年龄、国籍等，请求侧特征如距前次请求时间间隔、前次请求曝光数、点击数等，用户历史点击行为等。我们用GRU对用户历史行为进行序列建模，之后将抽取出的特征与其他映射为低维嵌入的特征拼接，送入多任务学习的门控多专家混合模块（MMoE）。MMoE会输出 4 个值，包括 。，，，最后， 经过激活函数最终得到输出结果——预估的效果损耗值。

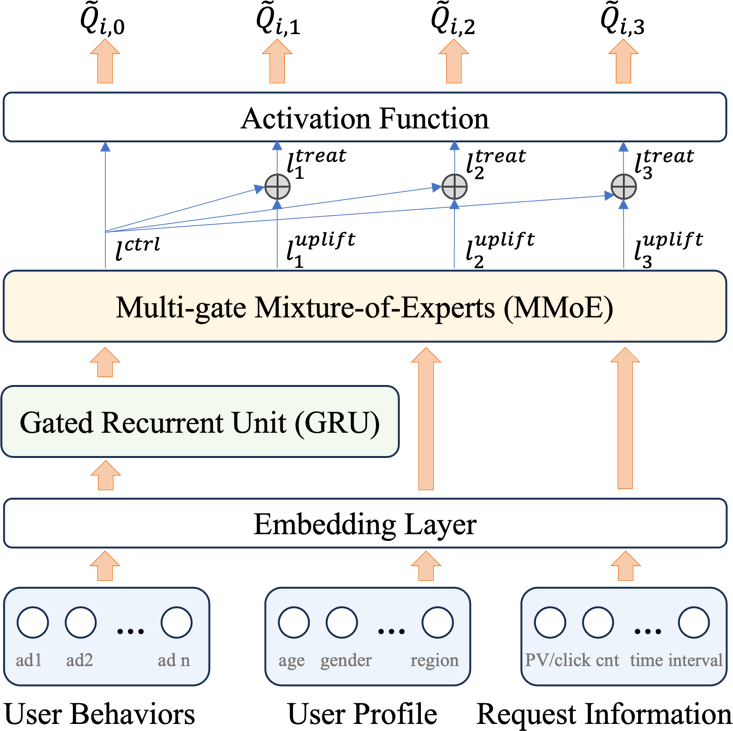


Figure 效果损耗预估模型

1. 步骤2、 根据实际业务需求，设定阈值B具体数值。
2. 步骤3、 由于在线系统存在流量波动，因此我们每15分钟采集数据，并用二分法求解最新拉格朗日乘子，用于接下来的15分钟到达的请求。
3. 在线服务部分：
4. 步骤1、 根据请求距前次请求的时间间隔，划分初始请求和重复请求。具体来说，时间间隔在2分钟内的，作为重复请求，时间间隔超过2分钟的，作为初始请求。
5. 步骤2、3、 均与技术方案中步骤相同。

**模拟实验结果**

1. 本实施实例的模拟实验对比了几个基线方法：动态规划DP，根据预估的推荐效果下降量离线为所有请求求解算力回收策略；算力回收ComRecycle，根据预估的推荐效果下降量二分求解拉格朗日乘子，并根据打分公式为每个请求分配算力回收策略；固定策略方法，为所有请求分配相同的算力回收策略；最优动态规划DP\*，根据实际广告效果下降量，离线为所有请求求解算力回收策略，这给出了原始问题的最优解；最优算力回收ComRecycle\*，根据实际广告效果下降量二分求解拉格朗日乘子，并根据打分公式为每个请求分配算力回收策略。
2. 通过调整阈值B，我们可以建立效果和算力开销之间的关系如图所示。我们总结了在相同算力开销下不同方法的推荐效果，以及实现相同推荐效果不同方法所需的算力开销如表所示。

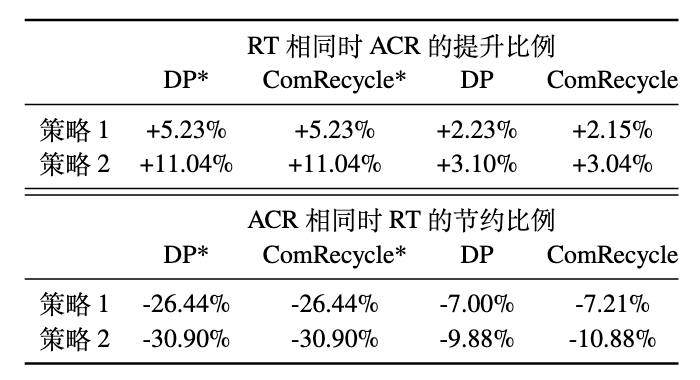
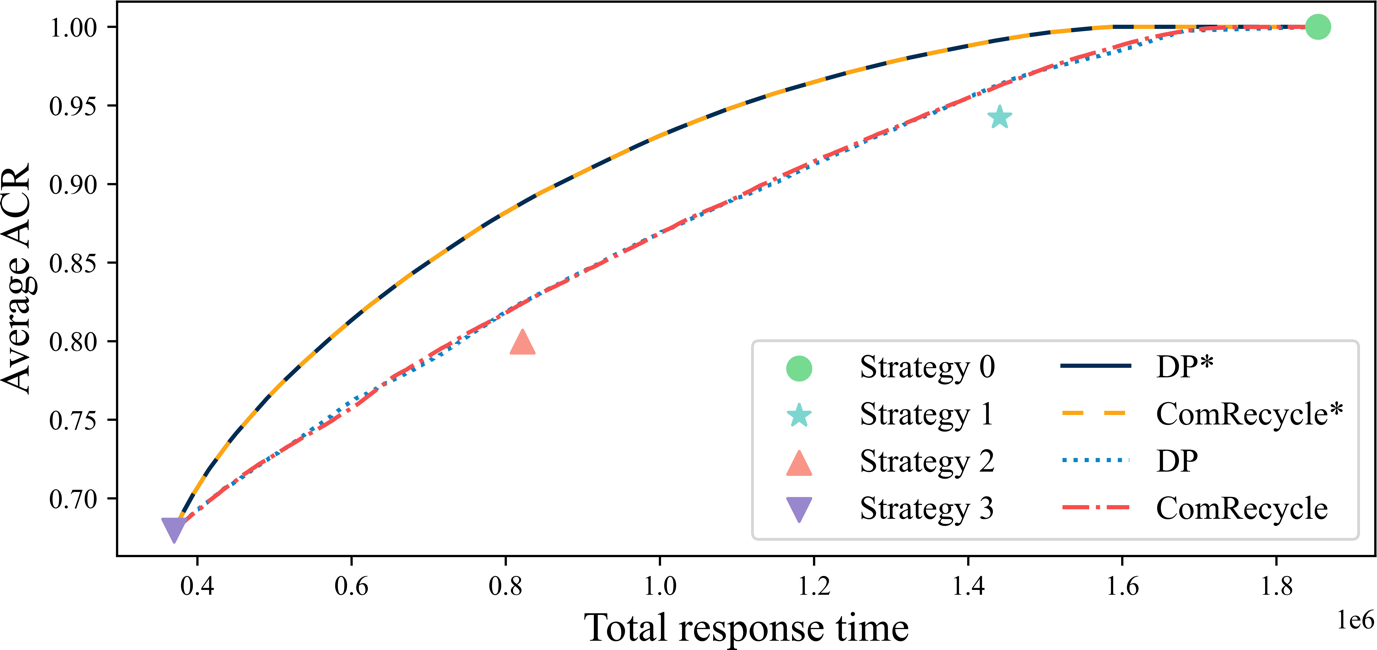


Figure 与固定策略方法的资源利用率比较。策略 1 和策略 2 表示为所有请求分配复用召回或粗排的算力回收策略

1. 通过以上模拟实验的结果，我们得到三点结论：
2. 1. DP\*和ComRecycle\*的曲线几乎一致，DP和ComRecycle的曲线也几乎一致，表明ComRecycle可以接近最优解；
3. 2. ComRecycle\*明显优于ComRecycle，表明估计的准确性限制了框架的有效性。另一方面，这也表明，如果模型的估计足够准确，资源利用率可以大大提高；
4. 3. ComRecycle优于固定策略方法（图中的蓝色和橙色点），表明 ComRecycle 提高了系统资源利用率。
5. 最终，我们得出结论：我们提出的多阶段智能算力回收框架可以有效地提升推荐系统资源利用率。

**说 明 书 附 图**

**权 利 要 求 书**

**说 明 书 摘 要**

一种推荐系统中的多阶段智能算力回收框架，ComRecycle。对级联式广告推荐系统架构进行了创新，通过“缓存” 与“回收”的方法，减少广告系统中不必要的算力开销。我们引入了细粒度算力回收策略，并将计算回收决策表述为在线约束优化问题。对优化问题求解，得到在线服务的策略打分公式，只要近在线更新打分参数，即可实现在线近似最优算力回收决策。最终，ComRecycle可以在保证相同水平的推荐效果的同时实现降低算力消耗的目标。