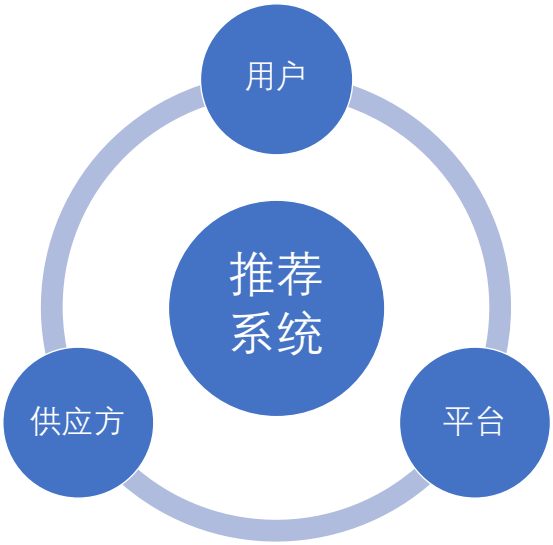
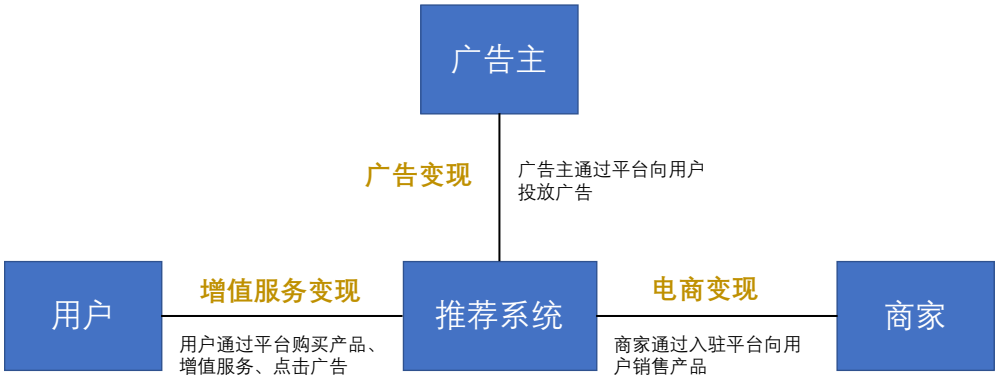
推荐系统的商业价值：什么是好的推荐系统？

- 一个好的推荐系统应该是能让用户、供应方（商家/广告主）、平台**三方共赢**的系统
  - 准确的预测并不代表好的推荐[i]

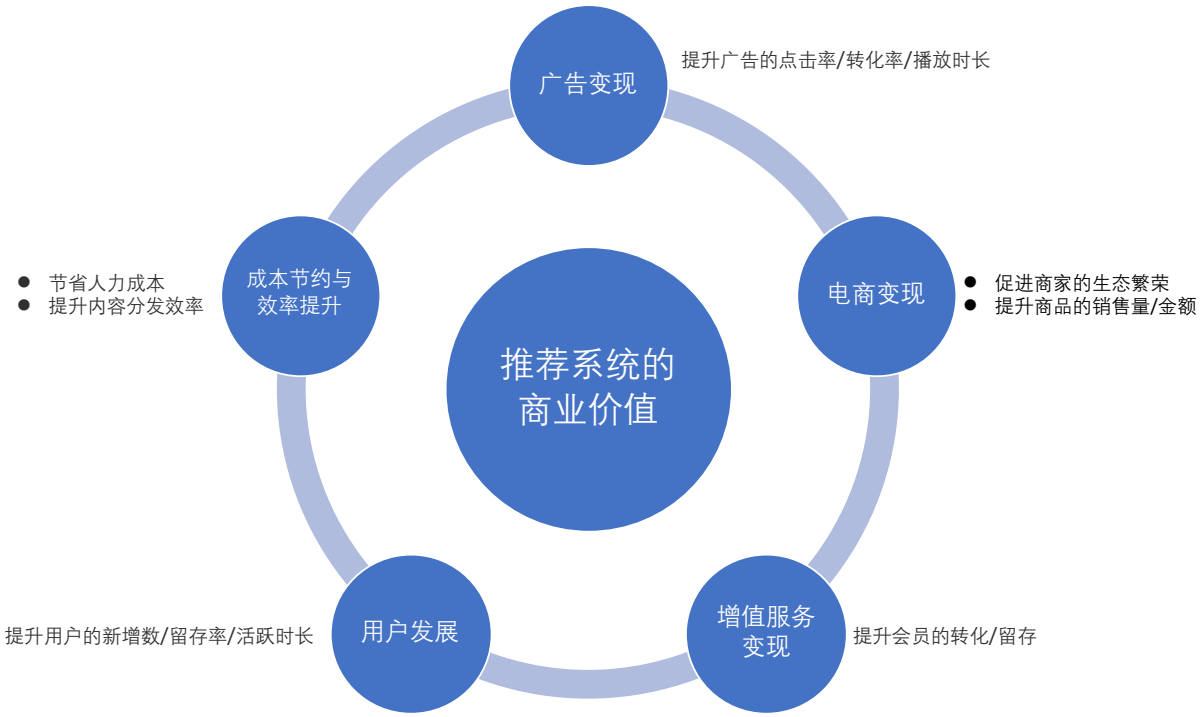
    比如，给用户推荐一本他本来就准备购买的书，无论是否给予推荐他都会购买。推荐系统并未使用户购买更多的书，而仅仅是方便用户购买一本他本来就准备买的书。对于用户来说，他会觉得这个推荐结果很不新颖，不能令他惊喜。同时，对于出版社来说，这个推荐也没能增加这本书的潜在购买人数。
  - 好的推荐系统不仅仅能够准确预测用户的行为，而且能够扩展用户的视野，帮助用户发现那些他们可能会感兴趣，但却不那么容易发现的东西
  - 同时，推荐系统还要能够帮助商家将那些被埋没在长尾中的好商品介绍给可能会对它们感兴趣的用户



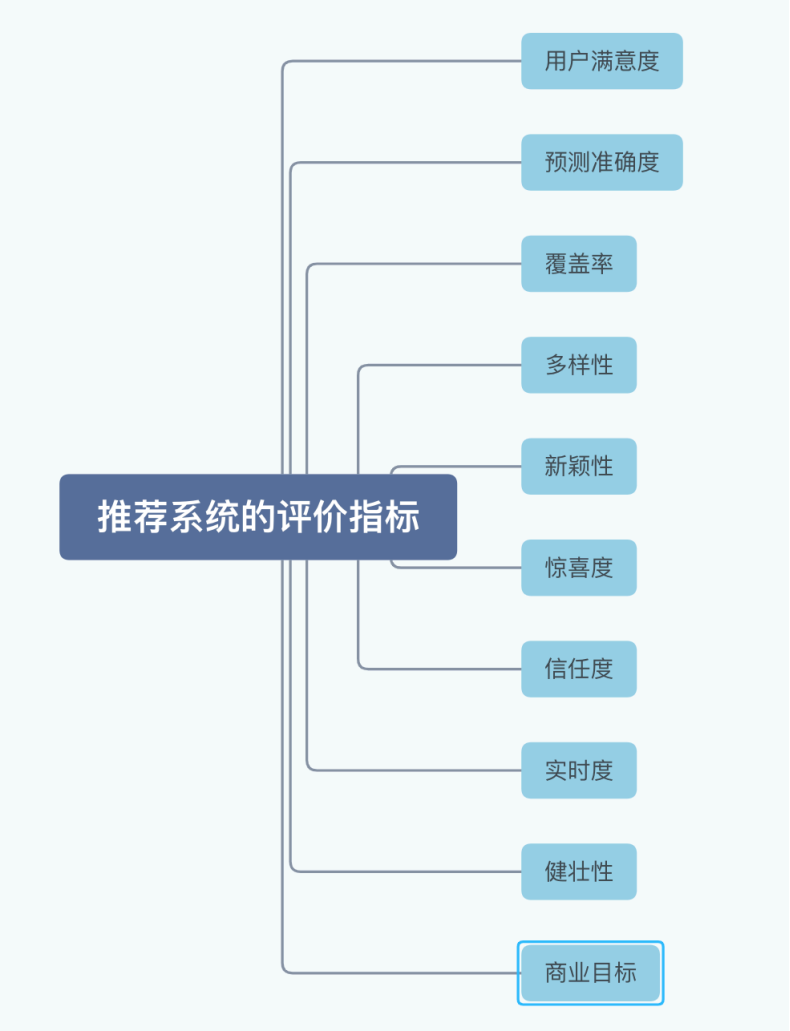
- 推荐系统相关的互联网产品盈利模式：广告、电商、增值服务



■ 推荐系统商业价值的体现维度[ii]



推荐系统的评价指标（如何评估推荐系统的效果？）[iii]



评估指标	具体含义
用户满意度	用户满意度无法进行离线计算，只能通过用户调查或在线实验获得： ● 用户调查获得用户满意度主要通过调查问卷的形式，设计调查问卷不是简单地询问用户对推荐结果是否满意，而是要考虑到用户各方面的感受 ● 通过在系统中设计一些用户反馈页面也可用于收集用户满意度 ● 更一般的情况下，可以通过点击率、用户停留时间和转化率等指标来度量用户满意度

预测准确度	评分预测	对于离线测试集中的某个用户u和物品i，令r_{ui}为用户u对物品i的实际评分，$\hat{r}_{ui}$为推荐算法给出度预测评分 ● 均方根误差（RMSE） $$\mathrm{RMSE}=\frac{\sqrt{\sum_{u,i\in T}(r_{ui}-\hat{r}_{ui})^2}}{|T|}$$ ● 平均绝对误差（MAE） $$\mathrm{MAE}=\frac{\sum_{u,i\in T}|r_{ui}-\hat{r}_{ui}|}{|T|}$$
	TopN 推荐	● Precision/Recall/F1 @N ● ROC AUC 衡量的是推荐系统将用户喜欢/不喜欢的商品区分出来的能力 ● Hit Rate (HR) $$\mathrm{HR}=\frac{\#\mathrm{hits}}{\#\mathrm{users}}$$ HR 是目前 TopN 推荐中十分流行的评级指标，其中#users是用户总数，而#hits是测试集中 item 出现在 TopN 推荐列表中的用户总数 ● Average Reciprocal Hit Rate (ARHR) $$\mathrm{ARHR}=\frac{1}{\#\mathrm{users}}\sum_{i=1}^{\#\mathrm{hits}}p_i$$ ARHR 是一种版本的 HR，度量的是一个 item 被推荐的强度，其中权重p_i是推荐列表中位置的倒数，可以用来衡量 Ranking 对该算法是否友好

	排序	当推荐项目的数量很大时，用户会更加重视推荐列表中排在前面的物品，这些物品中发生的错误比列表中排在后面的物品中的错误更严重，因此需要按排名列表对推荐效果进行加权评估 ● Discounted Cumulative Gain (DCG) DCG 的主要思想是用户喜欢的商品被排在推荐列表前面比排在后面会更大程度上提升用户体验，定义为 $$\mathrm{DCG}(b,L)=\sum_{i=1}^{b} r_i+\sum_{i=b+1}^{L}\frac{r_i}{\log_b i}$$ 其中r_i表示排在第i位的商品是否被用户喜欢，b是自由参数一般设为 2，L为推荐列表长度 ● Rank-biased Precision (RBP) RBP 和 DCG 指标的唯一不同点在于 RBP 把推荐列表中商品的浏览概率按等比数列递减，而 DCG 则是按照 log 调和级数形式 ● Normalized Discounted Cumulative Gain (NDCG) 由于用户之间 DCG 没有直接的可比性，需要进行归一化处理 ● Mean Average Precision (MAP)
覆盖率		覆盖率最简单的定义是，推荐系统能够推荐出来的物品占总物品的比例。覆盖率越高表明模型能够针对更多的物品产生推荐，从而发掘长尾的能力就越强 $$\mathrm{Coverage}=\frac{|\mathrm{U}_{u\in U} R_u|}{|I|}$$ 其中U是所有用户的集合，I是所有物品的集合，R_u是给用户u推荐的所有物品集合。 从上面的定义也可以看到，热门排行榜的推荐覆盖率是很低的，它只会推荐那些热门的物品，这些物品在总物品中占的比例很小。但是上面的定义过于粗略，覆盖率 100%的系统可以有无数的物品流行度分布。 为了更细致地描述推荐系统发掘长尾的能力，需要统计推荐列表中不同物品出现次数的分布。如果所有的物品都出现在推荐列表中，且出现的次数差不多，那么推荐系统发掘长尾的能力就很好。信息论中的信息熵和 KL 散度可以用来定义覆盖率 ● 信息熵 $$\mathrm{Entropy}=-\sum_{i=1}^{n} p(i)\log p(i)$$ 其中$p(i)$是物品i的流行度除以所有物品的流行度之和 ● KL 散度

多样性	多样性描述了推荐列表中物品两两之间的不相似性。假设$s(i.j) \in [0,1]$定义了物品i和j之间的相似度，那么用户u的推荐列表$R(u)$的多样性定义如下 $$\text{Diversity} = 1 - \frac{\sum_{i,j \in R(u), i \neq j} s(i,j)}{\frac{1}{2}\lvert R(u)\rvert(\lvert R(u)\rvert - 1)}$$ 而推荐系统的整体多样性定义为所有用户推荐列表多样性的平均值 $$\text{Diversity} = \frac{1}{\lvert U\rvert}\sum_{u \in U} \text{Diversity}\big(R(u)\big)$$ 推荐系统列表多样性与用户实际兴趣多样性应尽可能接近。
新颖性	新颖的推荐是指给用户推荐那些他们以前没有听说过的物品。在一个推荐系统中实现新颖性的最简单办法是，把那些用户之前对其有过行为的物品从推荐列表中过滤掉。 评测新颖度的最简单方法是利用推荐结果的平均流行度，因为越不热门的物品越可能让用户觉得新颖。因此，如果推荐结果中物品的平均热门程度较低，那么推荐结果就可能有比较高的新颖性。 但是，用推荐结果的平均流行度度量新颖性比较粗略，因为不同用户不知道的东西是不同的。因此，要准确地统计新颖性需要做用户调查。
惊喜度	如果推荐结果和用户的历史兴趣不相似，但却让用户觉得满意，那么就可以说推荐结果的惊喜度很高，而推荐的新颖性仅仅取决于用户是否听说过这个推荐结果。 目前并没有什么公认的惊喜度指标定义方式，这里只给出一种定性的度量方式。用户满意度只能通过问卷调查或者在线实验获得，而推荐结果和用户历史上喜欢的物品相似度一般可以用内容相似度定义。也就是说，如果获得了一个用户观看电影的历史，得到这些电影的演员和导演集合 A，然后给用户推荐一个不属于集合 A 的导演和演员创作的电影，而用户表示非常满意，这样就实现了一个惊喜度很高的推荐。因此提高推荐惊喜度需要提高推荐结果的用户满意度，同时降低推荐结果和用户历史兴趣的相似度。

信任度	如果用户信任推荐系统，就会增加用户和推荐系统的交互。特别是在电商推荐系统中，让用户对推荐结果产生信任是非常重要的。同样的推荐结果，以让用户信任的方式推荐给用户就更能让用户产生购买欲，而以类似广告形式的方法推荐给用户就可能很难让用户产生购买的意愿。 度量推荐系统的信任度只能通过问卷调查的方式，询问用户是否信任推荐系统的推荐结果。 提高推荐系统的信任度主要有两种方法 ● 首先需要增加推荐系统的透明度，而增加推荐系统透明度的主要办法是提供推荐解释。只有让用户了解推荐系统的运行机制，让用户认同推荐系统的运行机制，才会提高用户对推荐系统的信任度 ● 其次是考虑用户的社交网络信息，利用用户的好友信息给用户做推荐，并且用好友进行推荐解释。这是因为用户对他们的好友一般都比较信任，因此如果推荐的商品是好友购买过的，那么他们对推荐结果就会相对比较信任
实时性	推荐系统的实时性包括两个方面 ● 首先，推荐系统需要实时地更新推荐列表来满足用户新的行为变化。比如，当一个用户购买了 iPhone，如果推荐系统能够立即给他推荐相关配件，那么肯定比第二天再给用户推荐相关配件更有价值。与用户行为相应的实时性，可以通过推荐列表的变化速率来评测。如果推荐列表在用户有行为后变化不大，或者没有变化，说明推荐系统的实时性不高 ● 实时性的第二个方面是推荐系统需要能够将新加入系统的物品推荐给用户，这主要考验了推荐系统处理物品冷启动的能力。而对于新物品推荐能力，我们可以利用用户推荐列表中当天新加入物品的比例来评测。
健壮性	任何一个能带来利益的算法系统都会被人攻击，这方面最典型的例子就是搜索引擎。搜索引擎的作弊和反作弊斗争异常激烈，这是因为如果能让自己的商品成为热门搜索词的第一个搜索果，会带来极大的商业利益。推荐系统目前也遇到了同样的作弊问题，而健壮性指标衡量了一个推荐系统抗击作弊的能力。 算法健壮性的评测主要利用模拟攻击。 ● 首先，给定一个数据集和一个算法，可以用这个算法给这个数据集中的用户生成推荐列表

	● 然后，用常用的攻击方法向数据集中注入噪声数据，利用算法在注入噪声后的数据集上再次给用户生成推荐列表 ● 最后，通过比较攻击前后推荐列表的相似度来评测算法的健壮性。如果攻击后的推荐列表相对于攻击前没有发生大的变化，就说明算法比较健壮 在实际系统中，提高系统的健壮性，除了选择健壮性高的算法，还有以下方法 ● 设计推荐系统时尽量使用代价比较高的用户行为 比如，如果有用户购买行为和用户浏览行为，那么主要应该使用用户购买行为，因为购买需要付费，所以攻击购买行为的代价远远大于攻击浏览行为。 ● 在使用数据前，进行攻击检测，从而对数据进行清理
商业目标	商业目标和平台的盈利模式是息息相关的。一般来说，最本质的商业目标就是平均一个用户给公司带来的盈利。不过这种指标不是很难计算，只是计算一次需要比较大的代价。因此，很多公司会根据自己的盈利模式设计不同的商业目标。 不同的平台具有不同的商业目标。比如电商平台的目标可能是销售额，基于展示广告盈利的平台其商业目标可能是广告展示总数，基于点击广告盈利的平台其商业目标可能是广告点击总数。因此，设计推荐系统时需要考虑最终的商业目标，而平台使用推荐系统的目的除了满足用户发现内容的需求，也需要利用推荐系统加快实现商业上的指标。

评估指标的影响因素（如何设计推荐系统的评估方式？）[iv]

[i] 参见 Sean M. McNee、John Riedl、Joseph A. Konstan 的论文 “Being accurate is not enough: how accuracy metrics have hurt recommender systems”。

[ii] https://www.infoq.cn/article/pwPEy-LjDtzo86FqR7Hh

[iii] https://zhuanlan.zhihu.com/p/67287992
https://www.infoq.cn/article/Hz_w4DA4RdeBvdBVVfbX
https://juejin.cn/post/6844904065638350861
https://blog.csdn.net/lly1122334/article/details/104221360

[iv] https://www.jiqizhixin.com/articles/2020-04-01-12