1. 排序有哪些
2. 文档检索（Document retrieval）

系统保持一个文档候选集，当一个查询（query）到达时，系统根据这个查询从候选集中查找文档，然后生成一个top ranked list of documents给用户。

1. 机器翻译（Machine translation）

首先通过生成式模型将输入的句子转换为目标语言的表达形式，之后通过判别式模型对生成的句子进行筛选。涉及candidate generation and re-ranking。

1. 元搜索（Meta-search）

将用户搜索的关键字发送给

1. 搜索过程与LTR方法简介

搜索这一过程的本质是自动选取与用户输入的关键词（query）最相关的一组文档（docs，或称网页, urls）的过程，如图1所示。目前主要通过如下两个步骤实现：

·query-doc匹配：寻找与当前输入的query相关度高的docs；

·高相关度docs精确排序：对①中返回的docs，选取更多特征并按照用户点击该doc的可能性大小精确排序，如图2所示。有时我们还会选择不同的特征，召回多组①并将它们通过排序算法融合为一组。

1. point\_wise方法

pointwise单文档方法顾名思义：对于某一个query，它将每个doc分别判断与这个query的相关程度，由此将docs排序问题转化为了分类（比如相关、不相关）或回归问题（相关程度越大，回归函数的值越大）。但是pointwise方法只将query与单个doc建模，建模时未将其他docs作为特征进行学习，也就无法考虑到不同docs之间的顺序关系。而排序学习的目的主要是对搜索结果中的docs根据用户点击的可能性概率大小进行排序，所以pointwise势必存在一些缺陷。

1. pair\_wise方法

pairwise文档对方法并不关心某一个doc与query相关程度的具体数值，而是将排序问题转化为任意两个不同docs [公式]和[公式]谁与当前query更相关的相对顺序的排序问题，一般分为 [公式]比[公式] 更相关、更不相关和相关程度相等三个类别，分别记为{+1, -1, 0}，由此便又转化为了分类问题。本文重点关注的RankNet和LambdaRank算法同属于pairwise方法，是很多排序算法的雏形。

1. RankNet算法

根据docs与query的相关程度，比较任意两个不同文档和的相对位置关系，并将query更相关的doc排在前面。一般使用比更相关、比更相关和相关程度相等三个类别，并分别适用作为对应的类别标签，然后使用文档对作为样本的输入特征，由此将排序问题转换为分类问题。这样做的好处是无需对每一个doc与query的相关性进行精确标注，只需要获得docs之间的相对相关性，相对容易获得，可通过搜索日志、点击率数据等方式获得。