Suppose that we are given a set of ranked instances   a  from the space X ×Y . The task here is to select the best function f\* from F that minimizes a given loss function with respect to the given ranked instances. 假设我们从空间X×Y中得到一组排序实例a。这里的任务是从f中选择最佳函数f\*，使给定的损失函数相对于给定的排序实例最小化。

对于给定的排序实例。Herbrich等人[8]提出将上述学习问题形式化为对实例进行分类的学习问题。首先，我们假设f是一个线性函数。

## 学习排序

对于每个查询，根据上一节的方法计算了文档的相似度后，即可根据文档相似度得到一个初步的排序列表，但此时仅使用了文本语义相似度一个特征，为得到更准确的追溯性关系，下一步将此排序列表作为候选列表，利用在数据集文本中其它的特征，使用学习排序算法继续进行处理。

以下分别介绍学习排序算法和文档特征选择。

一、排序算法

排序算法的类型单文档方法仅考虑了单个文档与查询的绝对相关度，忽略了文档间的顺序关系；文档对方法考虑了任意两个文档之间的相对前后关系，相比单文档方法的效果更好；文档列表方法需要考虑每次对查询候选结果列表，当文档数量较大时，需要考虑的数量较大，相对而言没有文档对方法的效率高。综合以上原因，Tr-WELR方法模型中使用了文档对方法

此处我们选择的具体排序方法是IR SVM算法，按照排序方法的类型分类，IR SVM算法属于文档对方法，文档对方法考虑了任意两个文档间的相对位置关系，相对于单文档方法效果更好，而文档列表方法在文档数量较大时会明显降低处理效率。

由于IR SVM算法是在ranking SVM算法基础上进行改进的结果，因此首先介绍Ranking SVM，然后介绍IR SVM算法的改进。

Ranking SVM

Ranking SVM方法由Herbrich等人提出，该方法的基本思想是通过训练得到能对文档对进行分类的分类器，然后利用分类器完成排序任务。

如图（）所示，假设在特征空间中有两组与查询关联的文档集合。每个文档集合中的文档被划分成了三个等级。 第一组中的对象x1、x2和x3的级别分别为 rank3、rank2、rank1。

此时，可以使用第二章的公式（2.2）来进行排序，此处、和对应 2.2中的得到的对象向量，向量仍代表每个特征向量的权重矩阵。根据该公式来执行排序操作，实际上是将文档的特征向量投影到各自权重向量中，并根据特征向量的投影来对文档进行排序。

2.2

对于以上的排序问题，可以通过线性SVM分类来解决。首先，将同一组不同等级特征向量之间的差异作为新的特征向量处理，如将、和作为新的特征向量，同时，每个新的特征向量的级别也被标签所代替，如根据原向量的等级，将、和标记为负相关，将、和标记为正相关，这样的好处是可以把每个特征向量的权重 进行组合，形成新的适合所有特征向量的权重。需要注意的是，只需要对同一组中不同等级的特征向量创建新的特征向量。

图4中，支持向量机模型中的两条边界线，从几何意义上讲，代表两个等级的对象的投影之间的最近距离。在在SVM的超平面中，通过原实例来表示的正、负实例是成对相关的。例如，和分别是正的和负的实例。我们在训练时可以只选择正实例，负实例与正实例代表的意义是一样的，可以不重复训练。此时，SVM分类器的权重向量，其实就是排序函数。

因此，此时的训练数据的形式为 ，每个实例由两个特征向量来表示，用来表示哪个向量应该排在前面。此时可以将将SVM的学习问题形式化的表示为以下的QP（Quadratic Programming，二次规划）问题。

和分别表示一对特征向量中的第一个和第二个特征向量，||·||表示第二范式，m表示训练向量的代号，c为大于0的系数。它等价于下面的无约束优化问题，即，使损失函数最小化。

(3.13)

公式中代表，。

下面开始是IR SVM

Ranking SVM将排序转化为文档对分类，在学习过程中，实际上使用的是0-1损失函数。而此损失函数的优化目标与IR的评价指标之间存在一定差距。

Ranking SVM的第一个问题是，该方法对不同等级的文档对一视同仁。例如，三个等级的正确排序为等级1、等级2、等级3，则等级3>等级2和等级3>等级1都是错误的相对顺序，这两个错误Ranking SVM训练过程造成的影响程度是同等的。但是，对于一个IR系统来说，排名越靠前的文档，其准确性就更加重要，显然此算法对影响程度判断与IR系统的需求存在落差。

Ranking SVM的另一个问题是，它对不同查询中的每个文档对结果同等对待。但是不同查询相关的文档数量是不同的，Ranking SVM这样处理会导致相关文档数量越多的查询对于训练过程造成的影响就越大，相关文档数越少的查询对于训练过程造成的影响就越小。这种效果与信息检索要求的“每个查询的重要性等同”[23]是不相符的。

由Cao等人提出的IR SVM方法，基于信息检索方向的应用对ranking SVM方法进行了改进。

IR SVM通过将0-1对分类转化为代价敏感的文档对分类问题，解决了上述两个问题。该方法通过修改SVM的合页损失函数(hinge loss)来实现，它为不同级别和不同查询的文档对设置了不同的损失。

1）损失函数严重惩罚与排序列表顶部相关的错误。为了强调正确的排名在顶部的重要性，对于排序靠前的文档，即相似度等级较高的文档所在的文档对，赋予较大的损失权重；

2）根据相关文档数给予不同的损失权重。为了解决由于相关文档数的不同对训练过程的影响，对于相关文档数目较多的查询，赋予每个文档较小的损失权重；对于相关文档数目较少的查询，赋予每个文档较大的损失权重。

IR SVM的最优化问题可以表示成如公式(3.14)所示。

(3.14)

公式中表示第个文档对的等级，表示等级的权重值，表示与第个文档对相关的查询文档，表示相关文档对的参数。相较于公式（），加入了和作为惩罚因子，用来弥补Ranking SVM应用在IR领域的两个问题。

参数值的通过：选定一个评价指标，对每个查询，找到能够使得评价指标最优的排序序列，然后随机交换任意两文档的位置，查看评价指标的降低值，重复该过程，最后对降低值取平均值作为分类下对查询文档对所加的权重。

参数代表属于第档的实例的损失权重值。确定该值的方法：对属于这一档文档对的两个文档，随机交换它们的排序位置后，查看评价指标的降低值，对所有减少值的平均数，即得到该损失权重值。参数对应了查询的归一化系数。可以表示为，即该查询下的相关文档数目的倒数。

## 特征选择

研究证明，以某些文本特征作为输入进行学习排序在文本检索和需求追溯领域有着很好的效果，本文从常用的特征中选择以下五个特征，作为学习排序模型中构建文档向量的特征值。

（1）语义相似度，通过提出的语义相似度算法计算；

（2）Tanimoto系数，Tanimoto系数可以用来判断两个数据的相关程度，此处通过文档向量计算两个文档的上下文相似度，如公式(3.10)所示。

(3.10)

公式中，向量和向量代表文档向量，文档向量通过计算文档中所有词向量的平均值表示。和分别表示向量和向量的模，表示向量乘积。

（3）IDF之和，在候选文档中计算查询中的词或短语的IDF值的总和，用来表示候选文档对于查询文档的重要程度，如公式(3.11)所示。

(3.11)

其中为查询文档中的词，为在候选文档中的idf值。

（4）关键词数量，表示候选文档中关键词的数量，关键词根据词的TFIDF值与设置的阈值确定。

（5）文本中不重复词的数量，去除停用词后，文档中出现过的词的数量，重复出现的不计算在内，一定程度上体现该结果的有效程度。