Paper Reading：Slope one predictors for online rating-based collaborative filtering. Daniel Lemire and Anna Maclachlan, 2007.  
  
slope-one是一种item-based的协同过滤算法，核心思想是线性回归f(x) = x+b。根据用户对item的评分信息，得到任意两个item之间的回归直线。然后根据已评分item计算未评分item的分值。最后根据计算出来的item的分值排序做推荐。它的优点是算法简单，容易实现，可扩展性也不错，但需要是基于评分的，如果没有评分，需要构造评分。  
  
**文章提出了几种CF方案：**  
  
a. Per User Average。即根据用户的历史评分记录计算用户的平均评分，然后应用到该用户的所有未评分item上。  
  
b. 基于Pearson相关度的算法，即user-based CF。  
  
c. 基础的SLOPE ONE算法。  
  
d. WEIGHTED SLOPE ONE算法。基础的slope-one算法在计算回归直线时，并没有考虑对同一item的评分用户数。显然假如有2000个用户对J和L共同评分，20个用户对K和L共同评分，这个时候用户对J的评分显然要比对K的评分对L的影响要大。  
  
e. BI-POLAR SLOPE ONE算法。加权的slope-one算法很好地考虑了共同评分用户数的问题，但还有另外一个问题：好评和差评对用户的决策影响是不同的。很多的好评对用户的影响也不如少数的差评。因此这个算法先会计算用户对一个item的平均评分，然后将用户对item的评分集拆成两个：好评集（即>平均评分的）和差评集。对好评集中的item使用带权的slope-one算法预测评分，而对差评集中的item使用基础的slope-one算法（即放大了少量的差评），最后计算item的加权平均值。  
  
**文章测试结果：**  
  
测试使用了EachMovie和MovieLens数据集，MAE评估推荐质量，并比较了Bi-polar slope one, weighted slope one, slope one, bias from mean, adjusted cosine item-based, per user average, pearson这些算法，对于EachMovie数据集，Pearson和Bi-polar slope one的表现最佳。对于MovieLens数据集，三种slope-one算法表现同样好，并且优于所有其他算法。

**思考：**

slope one算法易于实施，动态可更新，在查询时有效，并且对于第一次访问的用户不期望有太多的信息，但相对于其他经常报道的模型依然具有相当的准确性。文章提高准确性 的方法，将评论分成不喜欢和喜欢子集，非常值得借鉴和学习！