第六届“泰迪杯”
数据挖掘挑战赛

优
秀
作
品

作品名称：基于协同过滤的电视产品个性化推荐

荣获奖项：特等奖

作品单位：重庆第二师范学院

作品成员：胡怡 刘凤 柳倩

指导老师：舒巧媛

基于协同过滤的电视产品个性化推荐

摘 要：本文使用基于协同过滤的算法处理用户观看节目的数据，一共解决了两个问题：1、为用户推荐新的电视节目；2、对用户进行画像分析，为用户打上个性化标签。

对于问题 1，首先，本文根据收视和回看的历史信息对数据进行预处理，通过删去观看时间不足 5 分钟的信息，得到每个用户对每个频道的观看时间占比及观看频率。其次，通过将用户观看频率与观看时间按一定权重进行整理，用基于用户的协同过滤算法(userCF)，计算出用户相似度矩阵。接着，根据用户点播信息，计算得到节目点播金额占比、观看时间占比及观看频率，同样按一定权重进行整理，得到点播用户总频率，通过基于物品的协同过滤算法(itemCF)，计算节目相似度矩阵。然后利用节目的相似度和用户的历史收视数据，给点播用户生成推荐列表（见表 8）；根据用户相似度矩阵与点播用户总频率，生成未点播用户推荐列表（见表 9）；整合点播与未点播用户推荐表。运用同样的方法整理附件 2 的电视产品数据。最后，整合附件 1 与附件 2 的推荐表，得到用户推荐节目清单（见表 10）。

对于问题 2，首先，根据节目类型和适宜人群构建附件 2 中的产品标签体系（见表 12）。其次利用入网时间、观看节目及观看时间，构建用户标签体系（见表 17）。然后，建立标签编号，为电视产品信息进行标签编号，得到产品数据标签，进而得到已标签用户推荐表。最后，结合用户相似度矩阵，计算未标签用户的标签推荐列表，最终整合得到用户数据标签及用户推荐标签（见表 22）。

关键词：协同过滤；userCF 算法；itemCF 算法；用户画像；电视产品推荐

1 绪论

1.1 背景

在互联网技术日益发展和进步的时代，各种数据呈现井喷式增长状态，仅2017年“双十一”天猫旗下购买物品所产生的交易额最终定格在1682亿元，其中，无线成交额就占据了90个百分点。这部分数据十分庞大，但对于当今大数据时代所产生的数据总和来说，却只不过是冰山一角。并且互联网的发展还不仅局限于购物，它已经渗透到了生活的各个方面。那么，该如何在这海量的数据中为用户找到并推荐有价值的信息，这一问题已成为当今大数据时代面临的一个重大挑战。

协同过滤（Collaborative Filtering）是现今推荐系统中应协同过滤用最为成熟的一个推荐算法系类，它利用兴趣相投、拥有共同经验之群体的喜好来推荐使用者感兴趣的资讯，个人透过合作的机制给予资讯相当程度的回应（如评分）并记录下来以达到过滤的目的进而帮助别人筛选资讯。

有优点就有缺点，缺点主要体现在:由于依赖用户的行为，捕捉新视频，实时热点视频能力较弱（即 Item 的冷启动问题）；结果由于依赖其他用户的行为，可解释性不强。有时候会发现推荐了一些用户不可理解的内容，著名的例子就是推荐中的“哈利波特”问题。运营编辑人员无法显性的干预推荐结果，对于强媒体属性的公司，这种支持强干预、抗风险能力是必须的。

1.2 问题重述

问题一：产品精准营销推荐

利用附件1中的收视回看信息及点播信息中的用户行为，分析用户的收视偏好。例如喜欢看哪一类的节目，家中成员有哪些，然后为附件2的产品进行分类打包推荐。

问题二：相似用户的电视产品打包推荐

构建用户标签体系和产品标签体系，对相似偏好用户进行分类，为用户贴上

标签；对产品进行分类，为产品推荐标签；为每一位用户生成个性化的产品营销推荐方案。

1.3 问题分析

针对问题一，通过观察发现，附件 1 中的四个子表格中，表 1、表 2 是用户频道数据，表 3、表 4，是用户节目数据，所以可以对其分为两类进行处理。

（1）对于表 1、表 2，首先分别计算用户对于每个频道的观看频率，然后对时间数据进行加权，最后整合得到初始数据。其次，利用协同过滤的 userCF 算法，得到用户间的相似度矩阵。

（2）对于表 3、表 4，先对节目进行预处理。对于点播金额和观看时间分别加权，计算用户观看节目的总频率表。然后采用 itemCF 算法，得到每个节目间的相似度矩阵。

（3）根据节目相似度，计算点播用户的节目推荐列表，再根据用户相似度，计算未点播用户的相似推荐列表

针对问题二，需要为用户推荐节目观看类型标签，进行用户画像，所以要先构建用户及产品标签体系。具体的求解步骤如下：

（1）用附件 2 构建产品标签体系及用户标签体系，得到产品数据标签，并对标签进行编号。

（2）用附件 3 计算入网时长，为用户贴上新老用户标签。

（3）用附件 1 计算每个用户在每个时间段的观看频率，找出频率排名最高的时间段，删除在时间上无明显偏好的用户，为用户贴上时间偏好标签。

（4）结合附件 2 整理得到的数据与用户相似度矩阵，得到用户数据标签列表，以及用户标签推荐列表。

2 模型假设

（1）假设用户观看或回看时长不足 5 分钟的数据，为无效数据

（2）假设时间段观看的最高频率小于 0.5 的用户，为无明显偏好用户

（3）假设用户的偏好不改变

3 符号说明

符号	定义
F_t	时间频率
$T_t^{'}$	该用户观看该节目的总时间
T_t	该用户观看节目的总时间
F_w	观看频率
$F_{\mathrm{r}}^{'}$	回看频率
F_r	回看总频率
F_v	收视频率
$F_d^{'}$	单片点播频率
F_d	单片点播总频率
$D^{'}$	点播频率
D	点播总频率
P_{s}	将收视频率和回看频率整合在一起的频率
$P_{\mathrm{s}}^{'}$	将单片点播频率与单片点播总频率整合在一起的频率
W_c	该用户观看节目的总次数
$W_c^{'}$	该用户观看该节目的总次数
P	点播金额比例
$M^{'}$	该用户观看该节目总金额
M	该用户观看节目总金额
$D_c^{'}$	该用户在该节目点播总次数

D_c	该用户点播节目总次数
u	表示某个用户
$\|N(i)\|$	喜欢物品i的用户数
$\|N(j)\|$	是喜欢物品j的用户数
$\|N(i)\&N(j)\|$	同时喜欢物品i和物品j的用户数
P_{uj}	用户u对物品j的兴趣
$N(u)$	用户喜欢的物品集合
$S(i,k)$	和物品i最相似的k个物品集合
W_{ji}	物品j和i的相似度
R_{ui}	用户u对物品i的兴趣
G	推荐系数
S	相似度
T	点播用户观看得节目的最高的频率

4 基于 userCF 算法的用户节目推荐

4.1 思路分析

通过观察发现，附件 1 中的四个表中，分别有用户和频道数据、用户和节目数据，两类数据。对于表 1、表 2，先分别计算各自的观看频率，然后进行整合，利用协同过滤的 userCF 算法，计算用户间的相似度矩阵。对于表 3、表 4，先对节目进行预处理，计算用户观看节目的总频率表，运用 itemCF 算法，计算节目相似度。根据节目相似度，计算点播用户的节目推荐列表，再根据用户相似度，计算未点播用户的相似推荐列表，然后同理计算附件 2，得到总的节目推荐表，其思维导图如下图 4-1：

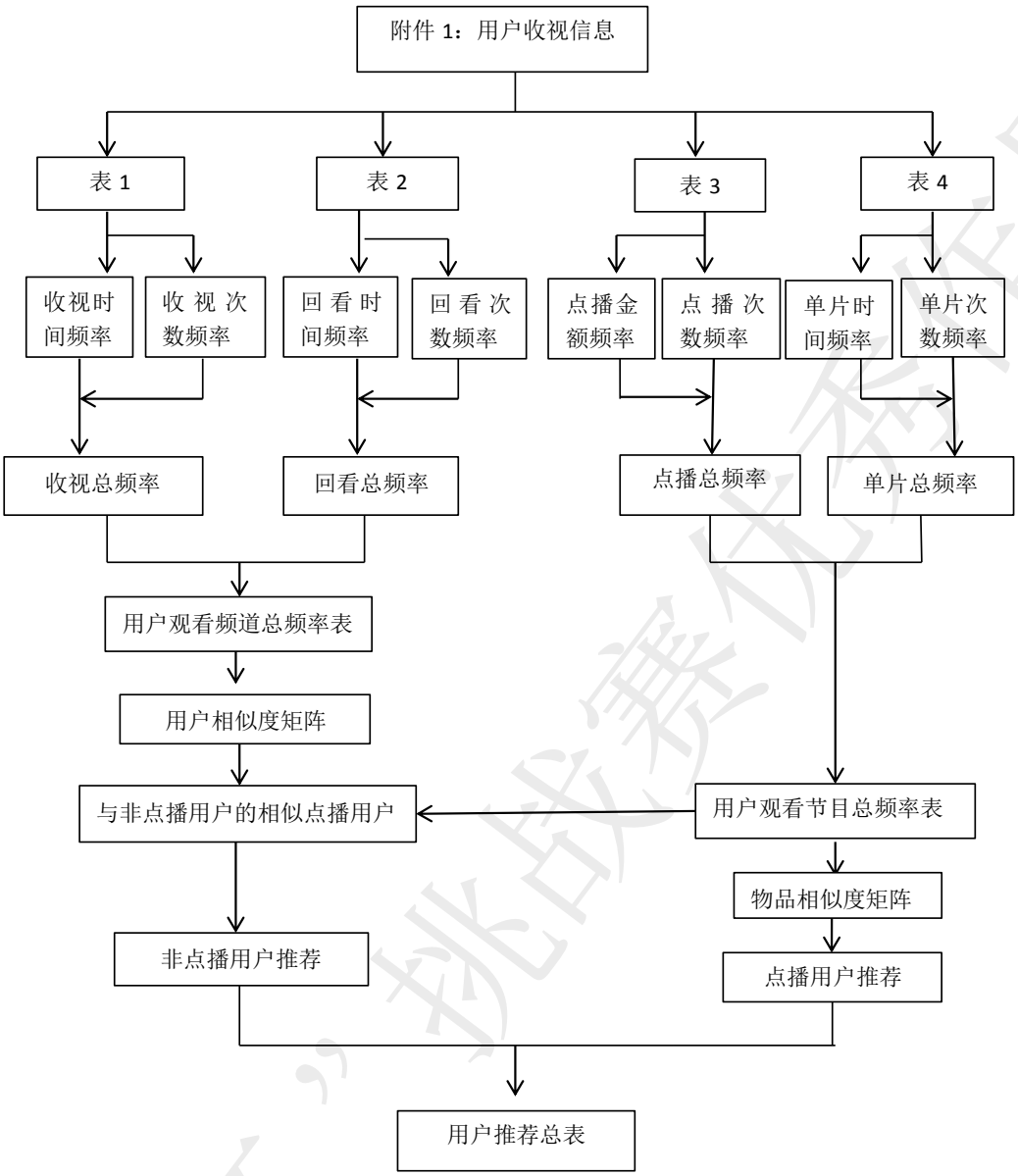

图 4-1 思维导图

注：表 1 指用户收视信息表，表 2 指用户回看信息表，表 3 指用户点播信息表，表 4 指用户单片点播信息表。

4.2 整理数据

4.2.1 用户和频道信息数据处理

观察原始数据发现附件 1 给出的频道号有断层，所以本文对其进行了重新编号（详表见附件 1）。计算出用户收看节目的时长，计算出时间频率，并删除掉收看时长小于或等于 5 分钟的数据，其公式如下：

$$F_t = \frac{T_t^{'}}{T_t} \quad (1)$$

注： F_t 表示时间频率， $T_t^{'}$ 表示该用户观看该节的总时间， T_t 表示该用户观看节目的总时间。

运用 Matlab 分别计算用户收视频率表 1（代码见附录 1）、用户回看频率表 2（代码见附录 2）：

$$F_w(F_r^{'}) = \frac{W_c^{'}}{W_c} \quad (2)$$

$$F_v(F_r) = F_w(F_r^{'}) + F_t \quad (3)$$

注： W_c 表示该用户观看节目的总次数， $W_c^{'}$ 表示该用户观看该节目的总次数， F_v 表示收视频率， F_r 表示回看总频率， F_w 表示观看频率， $F_r^{'}$ 表示回看频率， F_t 表示时间频率。

表 1 收视频率

用户	频道 1	频道 2	频道 3	……
10003	0.7252776	0	0	……
10004	0.2276548	0	0	……
10005	0.3021896	0	0	……

表 2 回看频率表

用户	频道 1	频道 2	频道 3	……
10003	0.4261306	0	0	……
10005	0.3109954	0	0	……
10006	0	0	0	……

根据经验可以知道，当一个人认为某个频道好看时，就会多次返回观看，所以，本文将表 1 和表 2 整合为一个表，其计算公式如下：

$$P_s = \frac{F_v + aF_r^{'}}{a+1} \quad (4)$$

注： F_v 表示收视频率，$F_t^{'}$ 表示回看频率，P_s 表示用户观看频道总频率，a 表示权重。

在本文中令 a=1，整理得到下表 3：

表 3 用户观看频道总频率表

用户	频道 1	频道 2	频道 3	……
10003	0.575704	0	0	……
10004	0.113827	0	0	……
10005	0.806592	0	0	……

4.2.2 计算用户相似度

在协同过滤中两个用户产生相似度是因为他们共同喜欢同一个物品，两个用户相似度越高，说明这两个用户共同喜欢的物品很多。假设每个用户的兴趣都局限在某几个方面，因此如果两个用户都喜欢某一个物品，那么这两个用户可能就很相似，而如果两个用户喜欢的物品大多都相同，那么他们就可能属于同一类，因而有很大的相似度。其计算公式如下：

$$W_{uv} = \frac{|N(u) \cap N(v)|}{\sqrt{|N(u)||N(v)|}} \quad (5)$$

注：$|N(u)|$ 是用户 u 喜欢的物品集合，$|N(v)|$ 是用户 v 喜欢的物品集合，$|N(u) \& N(v)|$ 是用户 u 和 v 同时喜欢的物品集合。

通过运行 Matlab（代码见附录 3），得到用户相似度矩阵，见下表 4。

表 4 用户相似度矩阵

用户	10003	10004	10005	……
10003	0	0.000914	0.000910	……
10004	0.000914	0	0.001082	……
10005	0.000910	0.001082	0	……

4.2.3 用户和点播信息数据处理

首先，将同一个用户观看的多个相同的节目整合为一个节目，待整合完毕后，对节目依次进行编号（详表见附件）。其次，计算出用户收看节目的时长，计算

出时间频率，并删除掉收看时长不足 5 分钟的数据（注：计算时间频率公式如（1）所示）。然后，对于用户点播信息中的数据，计算点播金额比例：

$$P=\frac{M^{'}}{M} \tag{6}$$

注：P 表示点播金额比例，M 表示该用户观看节目总金额，$M^{'}$ 表示该用户观看该节目总金额。

运用 matlab，分别计算单片点播总频率表 5 与单片点播总频率表 6，其中

$$F_d^{'}(D^{'})=\frac{D_c^{'}}{D_c} \tag{7}$$

$$D(F_d)=D^{'}(F_d^{'})+P(F_t) \tag{8}$$

注：$F_d^{'}$ 表示单片点播频率，$D^{'}$ 表示点播频率，$D_c^{'}$ 表示该用户在该节目点播总次数，D_c 表示该用户点播节目总次数，F_d 表示单片点播总频率，$F_d^{'}$ 表示单片点播频率，P 表示点播金额比例，$D^{'}$ 表示点播频率，D 表示点播总频率。

表 5 单片点播总频率

用户	节目 3	节目 4	节目 5	……
10085	0.1301447	0	0	……
10088	0	0	0	……
10089	0.0088306	0	0	……

表 6 点播总频率

用户	节目 12	节目 13	频道 14	……
10133	0	0	0	……
10138	0	0.066293	0	……
10148	0	0	0	……

最后，将单片点播总频率与单片点播总频率整合为一体，其公式如下：

$$P_s^{'}=\frac{F_d+aF_d^{'}}{a+1} \tag{9}$$

注：F_d 表示单片点播总频率，$F_d^{'}$ 表示单片点播频率，a 表示权重，$P_s^{'}$ 表示

用户观看节目总频。由于用户在点播时是需要购买观看的，所以在本文中 a=2 。

4.3 协同过滤推荐算法

4.3.1 点播用户推荐

（1）计算物品相似度

在协同过滤中两个物品产生相似度是因为它们共同被很多用户喜欢，两个物品相似度越高，说明这两个物品共同被很多人喜欢。假设每个用户的兴趣都局限在某几个方面，因此如果两个物品属于一个用户的兴趣列表，那么这两个物品可能就属于有限的几个领域，而如果两个物品属于很多用户的兴趣列表，那么它们就可能属于同一个领域，因而有很大的相似度。其计算公式如下：

$$W_{ij} = \frac{|N(i) \cap N(j)|}{\sqrt{|N(i)||N(j)|}} \quad (10)$$

注：$|N(i)|$是喜欢物品 i 的用户数，$|N(j)|$是喜欢物品 j 的用户数，$|N(i) \& N(j)|$是同时喜欢物品 i 和物品的 j 用户数。

得到节目相似度矩阵，见下表 7：

表 7 节目相似度矩阵

节目	节目 1	节目 2	节目 3	……
节目 1	0	0.07182	0.02450	……
节目 2	0.07182	0	0.00938	……
节目 3	0.02450	0.00938	0	……

（2）生成推荐列表

ItemCF 通过如下公式计算用户 u 对一个物品 j 的兴趣：

$$p_{uj} = \sum_{i \in N(u) \cap S(i,k)} W_{ji} R_{ui} \quad (11)$$

注：其中 P_{uj} 表示用户 u 对物品 j 的兴趣， $N(u)$ 表示用户喜欢的物品集合， $S(i,k)$ 表示和物品 i 最相似的 k 个物品集合， W_{ji} 表示物品 j 和 i 的相似度， P_{ui} 表示用户 u 对物品 i 的兴趣。

由上式可以看出，和用户历史上感兴趣的物品越相似的物品，越有可能在用户的推荐列表中获得比较高的排名，运行 Matlab（见附录 4），根据上表 7，利用了协同过滤算法原理生成了每个用户的相邻矩阵，即可以得到与每个用户喜好程度最为接近的前 20 位用户的信息，以此生成点播用户推荐表，见下表 8：

表 8 点播用户推荐列表

用户	产品名称	推荐指数
10004	艾菲格雷	0.45708691
10004	桑德森实验	0.45708691
10004	亡命地中海	0.45708691
……	……	……
10064	煎饼侠	0.680328547
10064	哈尔的移动城堡	0.680181108
10064	农民宇航员	0.681045533
……	……	……

4.3.2 非点播用户推荐

由于在附件 1 的用户收视信息、用户回看信息表中的收视用户、回看用户并不全是点播用户。所以，本文考虑，利用用户收视信息、用户回看信息表得出用户相似度矩阵表 4，来为未点播用户推荐节目。

（1）计算点播用户观看频率最高的节目。

（2）计算未点播用户推荐频率，其公式如下：

$$G=\frac{S+T}{2} \tag{12}$$

注：G 表示推荐系数，S 表示相似度，T 表示点播用户观看得节目的最高的频率。

通过计算得到未点播用户推荐列表，见下表 9：

表 9 未点播用户推荐列表

用户	产品名称	推荐指数
10005	建国大业	0.252197045
10005	极速前进第四季	0.244901045
……	……	……

4.3.3 附件 2 节目整理

运用同样的方法，将附件 2 中用户看过的节目推荐给附件 1 中用户看过的节目，通过整理得到附件 2 节目推荐表，见下表 10：

表 10 附件 2 节目推荐表

用户	产品标签	推荐指数
10004	小猪佩奇	0.245516314
10004	怒放	0.246715317
10004	星光大道	0.506091167
……	……	……
10025	超级飞侠一	0.246384009
10025	幻城	0.23773982
10025	上古情歌	0.25428519
……	……	……

4.4 用户节目个性化推荐

通过整理计算得到用户推荐总表，见下表 11，其中有一部分用户的推荐节目相同，是因为与未点播用户相似的点播用户观看的节目相同导致的结果。

表 11 部分结果示意图

用户	产品标签	推荐指数
10003	谋杀似水年华	0.441344
10003	乡村爱情故事 6	0.441331
10003	通天狄仁杰	0.441184
……	……	……
10006	超时空男臣	0.487183
10006	那年花开月正圆	0.678444
10006	风之谷	0.688122
10006	非你莫属	0.470455
10006	使徒行者 2	0.233333
……	……	……

5　基于 itemCF 算法的用户画像

5.1 思路分析

针对问题二，需要计算用户推荐观看标签，进行用户画像，所以要先构建用户及产品标签体系。用附件 2 构建产品标签体系及用户标签体系，得到产品数据标签，并对标签进行编号，用附件 3 计算入网时长，为用户贴上新老用户标签，用附件 1 计算每个用户在每个时间段的观看频率，找出频率排名最高的时间段，删除在时间上无明显偏好的用户，为用户贴上时间偏好标签，结合附件 2 整理得到的数据与用户相似度矩阵，得到用户数据标签列表，以及用户标签推荐列表，其思维导图如下图 5-1：

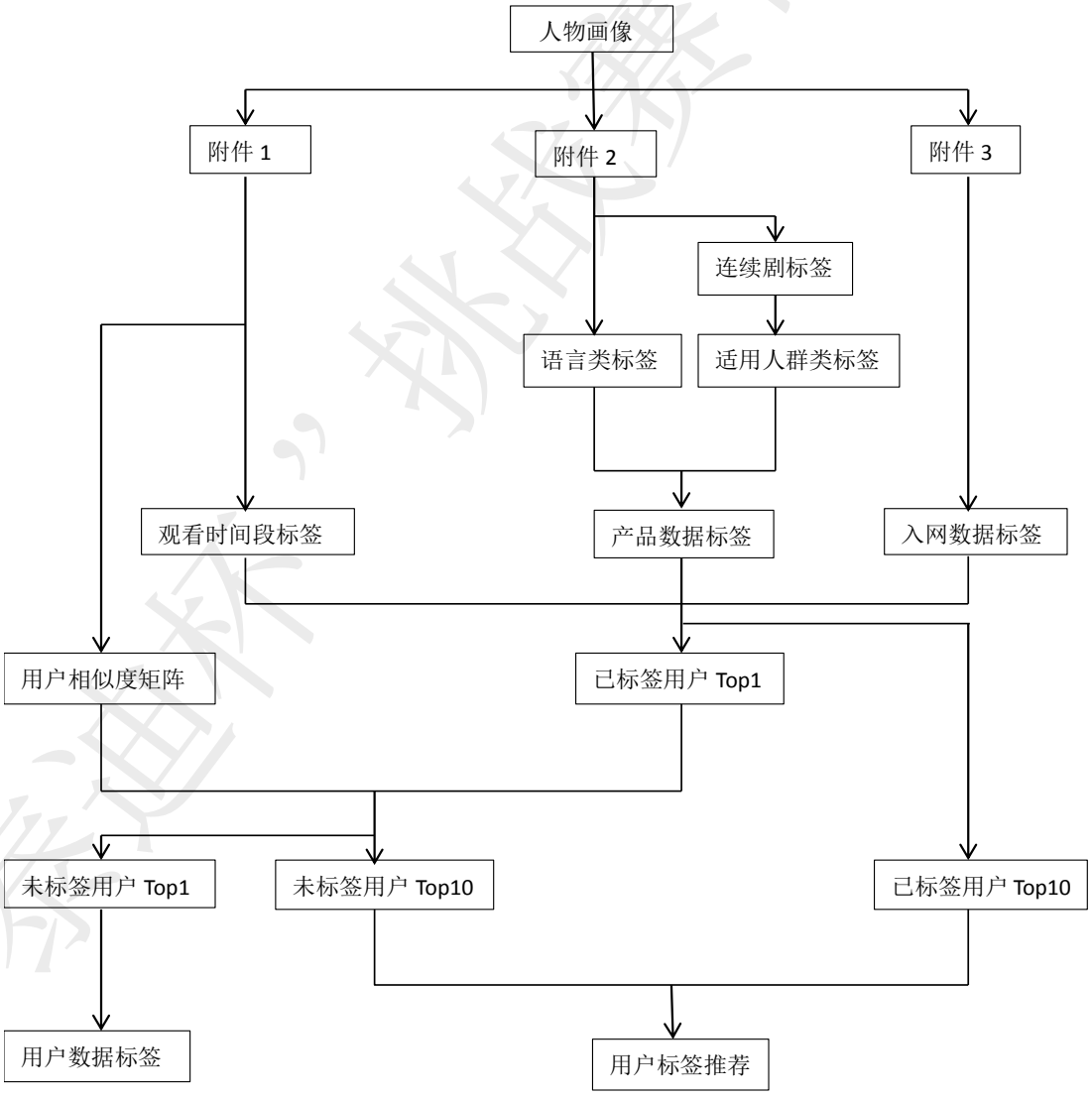

图 5-1 思维导图

5.2 处理数据

5.3.2 产品数据标签整理

（1）利用附件 2，建立产品标签体系。

1）对于基本特征的语言类，利用附件 2 进行语种划分

2）对于基本特征的电视剧等类别，利用“分类名称”对数据进行预处理

3）对于使用人群，本文做以下分类：

$$\begin{cases} \text{儿童更加偏好动画、动漫类节目} \\ \text{老人更加偏好养生类节目} \\ \text{女性更加偏好情感、爱情、文艺类节目} \\ \text{男性更加偏好体育、新闻类节目} \end{cases}$$

将以上标签进行整合，对标签进行编号（如表 12 所示）：

表 12 产品标签体系表

编号	一级标签	二级标签	三级标签
1	基本特征	电视剧	大陆
2	基本特征	电视剧	港台
3	基本特征	电视剧	欧美
……	……	……	……
56	适用人群	家庭成员	男性
57	适用人群	家庭成员	女性

5.3.3 用户数据标签整理

利用附件 1、附件 2、附件 3 建立用户标签体系，共分为观看时间段、收视偏好、入网时间。

（1）观看时间段

划分观看时间段，其划分结果如表 13 所示；

表 13 观看时间段

早上	上午	中午	下午	晚上	半夜
6:00~8:00	8:00~12:00	12:00~14:00	14:00~20:00	20:00~23:00	23:00~次日 6:00

整理附件 1 中各个用户观看时间段，将其整理如下表 14 所示：

表 14 用户观看时间段整理表

用户	时间段	编号
10001	上午	2
10001	下午	4
……	……	……
10001	上午	2
10002	中午	3
…….	……	……

计算各个用户在不同时间段观看节目的频率，其结果如表 15 所示；

表 15 用户观看频率

用户	早上	上午	中午	下午	晚上	半夜
10003	0	0.125000	0.062500	0.312500	0.500000	0
10004	0.00349	0.097902	0.055944	0.527900	0.244755	0.06993
……	……	……	……	……	……	……

计算每个用户在各个时间段观看频率，并且筛选出观看频率对应最高的时间段，其中，本文剔除了最高频率小于 0.5 的用户，因为用户观看时间段的的最高频率小于 0.5，则有理由认为该用户在时间上无偏好，其他用户信息整合如下表 16 所示:

表 16 时间段观看频率表

用户	时间段	编号	频率
10003	晚上	5	0.5
10004	下午	4	0.527972
……	……	……	……

（2）收视偏好

根据附件 2 的节目内容，整理得到用户数据标签

（3）入网时长

根据附件 3 的“入网时间”计算入网时长

$$\begin{cases} \text{入网时长}>2 \ , & \text{老用户} \\ \text{入网时长}<2 \ , & \text{新用户} \end{cases}$$

整理得到入网时长的标签

（4）用户数据标签体系

整理以上信息，得到下表 17：

表 17 用户标签体系表

编号	一级标签	二级标签	三级标签
1	基本特征	观看时间段	早上
2	基本特征	观看时间段	上午
……	……	……	……
7	收视偏好	电影	喜剧
……	……	……	……
64	基本特征	家庭成员	男性
65	基本特征	家庭成员	女性

5.4 营销推荐

（1）根据产品标签体系，对附件 2 中的节目进行标签，得到产品数据标签表（详表见附件）。通过附件 2“来源”项的用户号，建立已标签用户表，并计算每个用户对每个标签的频率，见下表 18：

表 18 已标签用户频率

用户	标签 1	标签 2	标签 3	……
10002	0	0	0	……
10139	0.301376849	0.007734149	0	……
10177	0	0	0.2	……
……	……	……	……	……

（2）找出已标签的每个用户标签频率排名第一的标签，见下表 19：

表 19 已标签用户频率 TOP1

用户	一级标签	二级标签	三级标签	编号
10050	收视偏好	生活	社会百态	25
10063	收视偏好	电视剧	港台	2
10069	收视偏好	电影	爱情	8
10139	适用人群	家庭成员	男性	46
10177	适用人群	家庭成员	男性	46
……	……	……	……	……

（3）运用公式（10），计算得到标签相似度矩阵，根据 itemCF 算法，进一步得到已标签用户推荐排名 TOP10，见下表 20：

表 20 已标签用户推荐 TOP10

用户	一级标签	二级标签	三级标签	推荐指数
10002	收视偏好	电视剧	日韩	0.216987022
10002	收视偏好	电视剧	动漫	0.209638931
10002	收视偏好	电影	动画	0.179510475
10002	收视偏好	生活	社会百态	0.184172479
……	……			……
10009	收视偏好	生活	社会百态	0.182047607
10009	收视偏好	新闻	时事	0.209279013
10009	收视偏好	娱乐	综艺	0.220465514
10009	收视偏好	科教	纪录片	0.22591614
……	……	……	……	……

（4）利用用户相似度矩阵表 4，计算并得出与未标签用户相似度排名第一和排名前十的标签用户，排名第一的表是为了方便为附录 3 的用户贴上收视偏好的标签，排名前十是为了给与已标签用户相似的未标签用户推荐标签产品，下表 21 仅是排名第一的整理表。

表 21 未标签用户 TOP1

未标签用户	相似已标签用户	相似度
10018	10006	1.03609196
10018	10011	1.011795742
10018	10014	1.017266824
10018	10016	1.040362459
10018	10019	1.013212597
……	……	……

注：此处相似度最高为 2。

（5）根据用户相似度矩阵表 4、已标签用户频率表 19、未标签用户 TOP1 表 21，计算得到用户数据标签（详表见附件）。

5.5 结果

运行 Matlab（见附录 6），计算得出推荐结果，其部分数据如表 22 所示：

表 22 推荐结果数据表

用户	一级标签	二级标签	三级标签	推荐指数
10001	收视偏好	电影	动画	0.36054725
10001	收视偏好	娱乐	歌曲	0.36487143
10001	收视偏好	电视剧	大陆	0.54184699
10001	收视偏好	语言	国语	0.31767789
……	……	……	……	……
10623	基本特征	语言	粤语	0.47080261
10623	收视偏好	电影	动作	0.53419203
10624	基本特征	语言	国语	0.54122817
10624	收视偏好	电影	动画	0.37192743
……	……	……	……	……

6 模型评价

6.1 优点

本文很好的利用用户的历史行为，避免共用其他人的经验，避免了内容分析的不完全或不精确，并且能够基于一些复杂的，难以表述的概念（如资讯品质、个人品味）进行过滤，直接后天间接性继承前辈经验。

本文建立了比较完整的用户标签体系。

6.2 缺点

由于过分依赖用户的行为，捕捉新视频，实时热点视频能力较弱（即 Item 的冷启动问题）。

结果由于依赖其他用户的行为，可解释性不强。有时候会发现推荐了一些用户不可理解的内容，著名的例子就是推荐中的“哈利波特”问题。

7 模型的应用与推广

该模型着重于对用户信息与物品信息的处理，最终归纳总结出用户对物品的偏好，从而对不同的用户推荐不同的物品，不仅如此，还能得到用户与用户兴趣特征的相似度，将兴趣特征相似度较高的用户整合在一起，便于物品的推荐，供

应商便可以根据此种方式来规划自己的营销策略。

并且本文在频道与节目数据整合的同时加入了权重，提高了为用户推荐节目的精确度，该思想也有一定的借鉴之处。

参考文献

[1]何佳知. 基于内容和协同过滤的混合算法在推荐系统中的应用研究[D]. 东华大学, 2016.

[2]刘青文. 基于协同过滤的推荐算法研究[D]. 中国科学技术大学, 2013.

[3]任品. 基于置信用户偏好模型的电视推荐系统[J]. 现代电子技术, 2014(16):30-33.

[4]万敏. 数据挖掘算法在卫星直播广播电视用户收视行为分析中的应用[C]// 中国新闻技术工作者联合会 2016 年学术年会论文集. 2016.

[5]刘鹏, 王超. 计算广告:互联网商业变现的市场与技术[M]. 人民邮电出版社, 2015.

[6]邓爱林, 朱扬勇,施伯乐. (2003). 基于项目评分预测的协同过滤推荐算法. 软件学报, 14(9).

[7]赵亮, 胡乃静,张守志. (2002). 个性化推荐算法设计. 计算机研究与发展, 39(8), 986-991.

[8]邓爱林, 左子叶,朱扬勇. (2004). 基于项目聚类的协同过滤推荐算法. 小型微型计算机系统, 25(9), 1665-1670.

[9]张光卫, 李德毅, 李鹏, 康建初,陈桂生. (2007). 基于云模型的协同过滤推荐算法 (Doctoral dissertation).

[10]刘建国, 周涛, 汪秉宏. (2009). 个性化推荐系统的研究进展. 自然科学进展, 19(1), 1-15.

[11]赵亮, 胡乃静, 张守志. (2002). 个性化推荐算法设计. 计算机研究与发展, 39(8), 986-991.

[12]王国霞, 刘贺平. (2012). 个性化推荐系统综述. 计算机工程与应用, 48(7).

[13]余力, 刘鲁, 李雪峰. (2004). 用户多兴趣下的个性化推荐算法研究. 计算机集成制造系统, 10(12), 1610-1615.

[14]朱岩,林泽楠. (2009). 电子商务中的个性化推荐方法评述. 中国软科学, (2), 183-192.

[15]吴丽花, 刘鲁. (2006). 个性化推荐系统用户建模技术综述. 情报学报, 25(1), 55-62.

[16]闫莺, 王大玲, 于戈. (2005). 支持个性化推荐的 Web 面关联规则挖掘算法. 计算机工程, 31(1), 79-81.

附录 1

```
%计算用户收视频率
%!!!! 导入整理矩阵 A
pin=146;%频道数
yong=1032;%用户数
jian=1;%时间系数
Z=zeros(yong,pin);%初始一个 0 矩阵，用于写入频率
F=A(:,1);%A 的第一列是序号，第二列是用户，第三列是频道序号，第四列是频道号，第五列是时间
[m,n]=hist(F,unique(F));%F 是用户列
G=[n,m'];%第一列用户，第二列总数
%!!!! 先运行前 6 行，得到 G
G1=[0,0;G];%在 G 前加 0 一行，然后在 G1 后加一列总数，得到 G2，导入 G2
yong1=yong+1;
for j=2:yong1
    d=G2(j-1,3)+1;
    D=A(d:G2(j,3),1:3);%选出单个用户的观看矩阵
    h=G1(j,2);
    f=D(:,2);%f 是频道列
    [p,q]=hist(f,unique(f));%计算单个用户观看不同频道的频数
    if size(p)==size(q)%判断该用户是否只观看了一个频道
        P=[q;p];
    else
        P=[q,p'];
    end
    K=size(P);
    a=size(q);
    a1=sort(f);%对 f 进行排序
    if a(1,2)==a1(end)%判断该用户是否只观看了一个频道
        Z(j-1,a(1,2))=2;%如果是，则该项等于 1
    else
        for i=1:K(1,1)%如果不是，计算频率
        g(i,1)=P(i,2)/h;
```

```
        %计算时间
        shi=0;
        Shi=sum(D);
        for ii=1:h
            if D(ii,2)==P(i,1)
                shi=shi+D(ii,3);
            else
            end
            g1(i,1)=shi/Shi(1,3);
        end
        Z(j-1,P(i,1))=g(i,1)+jian*g1(i,1);%计入 Z 矩阵相应位置
        end
    end
end

Z=[G(:,1),Z];%加入用户列
Z=[0:pin;Z];%加入频道行
```

附录 2

```
%计算用户回看频率
%!!!! 导入 H 整理矩阵,第一列用户，第二列频道序号,第三列是时间
pin=57;
yong=46;
jian=1;
z=zeros(yong,pin);%初始一个 0 矩阵，用于写入频率
F=H(:,1);%F 是 H 用户列
[m,n]=hist(F,unique(F));
H1=[n,m'];%各个用户的总数



%!!!! 运行前 6 行，得到 H1，利用 H1 加入一列总数，导入 H2
H3=[0,0,0;H2];%H3 的第 1 列是用户列，第 2 列是总数列，第三列是求和列
yong1=yong+1;
for j=2:yong1
    d=H3(j-1,3)+1;
    D=H(d:H3(j,3),:);
    h=H3(j,2);
    f=D(:,2);
    [p,q]=hist(f,unique(f));
    if size(p)==size(q)
        P=[q;p];
    else
        P=[q,p'];
    end
    K=size(P);
    a=size(q);
    a1=sort(f);
    if a(1,2)==a1(end)
        z(j-1,a(1,2))=2;
    else
        for i=1:K(1,1)
```

```
        g(i,1)=P(i,2)/h;

    %计算时间
        shi=0;
        Shi=sum(D);
        for ii=1:h
            if D(ii,2)==P(i,1)
                shi=shi+D(ii,3);
            else
            end
            g1(i,1)=shi/Shi(1,3);
        end
        z(j-1,P(i,1))=g(i,1)+jian*g1(i,1);
        end
    end
end
```

附录 3

```
%用户相似度      计算相似用户排名
%计算用户相似度备注以下
a=1;
pin=146;
yong=1032;
pin1=pin+1;
yong1=yong+1;
n=20;
%S1=(S66+a*S44)/(1+a);
S=S1';
for i=2:pin1
    for j=1:yong
        if S(i,j)>0
            S(i,j)=1;
        else

        end
    end
end

yonghu=zeros(yong,yong);%用户的相似度矩阵
for i=2:pin1
    C=S(i,1:yong);
    for j=1:yong
        for p=1:yong
        if C(1,j)==C(1,p)
            yonghu(j,p)=yonghu(j,p)+1;
            yonghu(j,j)=0;
        else
        end
        end
    end
end
```

```
N=sum(yonghu);
for i=1:yong
    for j=1:yong
        yh(i,j)=yonghu(i,j)/sqrt(N(1,i)*N(1,j));
    end
end
```

附录 4

```
%物品相似度

a=2;%a 是回频率的系数，控制权重
S2=(S6+a*S4)/(1+a);%S6 是收看整理，整理所有用户看其他频道的频率，S4 是回看整理
S=S2;%整理权重之后的矩阵
Z1=zeros(size(S2));
pin=1464;
yong=402;
n=1;%TOP 的个数
pin1=pin+1;
yong1=yong+1;
for i=1:yong
    for j=2:pin1
        if S(i,j)>0
            S(i,j)=1;
        else

        end
    end
end

wupin=zeros(pin,pin);%物品的相似度矩阵
for i=1:yong
    C=S(i,2:pin1);
    for j=1:pin
        for p=1:pin
            if C(1,j)==C(1,p)
                wupin(j,p)=wupin(j,p)+1;
                wupin(j,j)=0;
            else
            end
        end
```

```
    end
end

N=sum(wupin);
for i=1:pin
    for j=1:pin
        wp(i,j)=(wupin(i,j)/sqrt(N(1,i)*N(1,j)));
    end
end


for i=1:yong
    hu=S2(i,2:pin1);
    for j=1:pin
        if hu(1,j)==0
            wu=wp(:,j);
            Z1(i,j)=sum(hu*wu);
        end
    end
end


Z1=[S(:,1),Z1];


%计算每个用户的 TOP20          sort(A,'descend')对 A 各列降序排列
TOP=[];
for i=1:yong
    TOP1=Z1(i,:);
    iii=[i*n,3];
    TOP2=sort(TOP1,'descend');
    for j=2:pin1
        if TOP1(1,j)>=TOP2(1,n+1)
            TOP=[TOP;TOP1(1,1),j,TOP1(1,j)*1000];
```

```
                if size(TOP)==iii
                    break
                else
                end
            else
            end
        end
end
```

附录 5

```
yh1=[S1(:,1),yh];
yh1=[0,S(1,:);yh1];
yh2=yh1;
bbbb=size(OO);
for i=1:bbbb(1,1)
    for j=2:yong1
        if yh1(j,1)==OO(i,1)%OO 表示点播用户
            yh2(j,:)=zeros(1,yong1);
        else
        end
    end
end
yh2(all(yh2==0,2),:)=[];%删除行
bb1=size(yh2);

OOO=yh2(2:bb1(1,1),1);
yh3=yh2';
yh4=yh3;
for i=1:bb1(1,1)-1
    for j=2:yong1
        if yh4(j,1)==OOO(i,1)
            yh3(j,:)=zeros(1,bb1(1,1));
        else
        end
    end
end
yh3(all(yh3==0,2),:)=[];
yh3=yh3';


%计算用户相似度最高的，查看该用户的前 top1

top=[];
```

```
tt=size(yh3);%tt(1,1)
tt1=tt(1,2);
for i=2:tt(1,1)
    top1=yh3(i,:);
    iii=[i*n,3];
    top2=sort(top1,'descend');
    for j=2:tt1
        if top1(1,j)>=top2(1,n+1)
            top=[top;top1(1,1),yh3(1,j),top1(1,j)*1000];
            if size(top)==iii
                break
            else
            end
        else
        end
    end
end
```

附录 6

```
%导入 O1=附件 2 的用户列
%导入 ZZ，用户相似度表
%计算附件 1 与附件 2 用户的相似度表
yong=1032;%这里指附件一的用户数
yong1=yong+1;
ZZ1=ZZ;%用户相似度矩阵
bb=size(O1);

for i=1:bb(1,1)%点播用户的数量
    for j=2:yong1
        if ZZ(j,1)==O1(i,1)%O1 表示点播用户
            ZZ1(j,:)=zeros(1,yong1);
        else
        end
    end
end
ZZ1(all(ZZ1==0,2),:)=[];%删除行
b=size(ZZ1);

O2=ZZ1(2:b(1,1),1);%新的用户列
ZZ2=ZZ1';
ZZ3=ZZ2;
for i=1:b-1
    for j=2:yong1
        if ZZ3(j,1)==O2(i,1)
            ZZ2(j,:)=zeros(1,b(1,1));
        else
        end
    end
end
ZZ2(all(ZZ2==0,2),:)=[];
ZZ2=ZZ2';
bb4=size(ZZ2);
```

```
yong=bb4(1,1)-1;
pin=bb4(1,2)-1;
pin1=pin+1;
n=10;
topp=[];
bbb=size(ZZ2);
Z1=ZZ2(2:bbb(1,1),:);
for i=1:yong
    topp1=Z1(i,:);
    iii=[i*n,3];
    topp2=sort(topp1,'descend');
    for j=2:pin1
        if topp1(1,j)>=topp2(1,n+1)
            topp=[topp;topp1(1,1),ZZ2(1,j),topp1(1,j)*1000];
            if size(topp)==iii
                break
            else
            end
        else
        end
    end
end
```