DOI:10.13878/j.cnki.jnuist.2019.01.013

龚志[1]　邵曦[1]

# 基于多模态的音乐推荐系统

**摘要**

使用传统协同过滤的方式进行推荐往往会忽视音乐底层特征.通过将音乐的音频特征与歌词信息进行多模态融合,并将融合后的特征信息作为协同过滤推荐的补充,提出了一种基于多模态的音乐推荐系统.主要探讨了音频特征与歌词信息的提取,并在提取歌词信息时利用 LDA 主题模型进行特征降维.针对多模态融合问题,使用一种特征级联早融合法(EFFC)融合方式,并将多模态融合后的结果与单模态结果进行了比较.对于结果的推荐,以多模态特征信息为依据建立用户兴趣模型,并将该模型通过 LSTM 神经网络,以过滤与优化协同推荐的用户组.结果表明,基于多模态的音乐推荐系统将推荐结果的误差项平方和(SSE)由传统的 2.009 降至 0.388 6,验证了该方法的有效性.

**关键词**

音乐推荐;协同过滤;LDA 主题模型;多模态融合;LSTM 神经网络

**中图分类号** TN912

**文献标志码** A

## 0　引言

随着物质生活水平的不断提高,人们对文化产品尤其是高品质音乐产品的需求日渐提高.借由互联网规模的扩大和数字存储技术的进步,音乐产业不断发展,音乐数量也以几何级数激增.一方面,网络中海量音乐资源出现了信息过载现象,这些音乐通过自身携带的标签(如演唱者、年代、音乐流派等)与其他音乐进行区分,但这种分类标准不统一且缺乏开放性;另一方面,用户被这些海量音乐所包围着,传统的检索方式无法满足用户需求,无法从中有效获取自己所需的音乐资源.

推荐系统作为一种“信息推送”模式,是解决信息过载问题的主要手段,它能够在分析预测用户需求的基础上主动推送其可能需要但又无法获取的有用信息,并能够以用户为中心,通过研究用户行为、兴趣和环境等,为用户推荐更具针对性的信息,即实现信息的“按需定制服务”[1].然而,目前绝大部分已成熟商用的推荐系统都采用了传统的基于协同过滤的推荐方式,该方法忽视了音乐内容本身,无法满足用户的实际需求.通过进一步研究发现,将融合音乐的音频特征与歌词信息的多模态特征作为协同推荐方式的补充,可使音乐推荐的结果更加客观与准确.所以,本文提出了一种基于多模态的音乐推荐系统,在进行音乐推荐时使用音乐的多模态特征来提高协同推荐的准确率.

多模态音乐推荐系统框架如图 1 所示.

## 1　协同过滤与用户兴趣模型

协同过滤技术是信息推荐系统中最为成功的技术之一,也是信息推荐和信息服务领域的研究热点[2].该方法主要通过用户之间的相互协同来选择有价值的信息,比如利用用户之间对资源的评分进行推荐,目前比较具有代表性的是基于用户的协同过滤推荐方法.

基于用户的协同过滤推荐方法,首先计算出目标用户与其他用户的相似度,在用户社区中找到与目标用户最相似的部分用户(邻居),再由邻居用户对指定音乐的评分来预测目标用户对该资源的评分,从而产生推荐结果.

基于用户的协同过滤推荐方法的核心就是计算用户之间的相似度.假设:$U=\{U_1,U_2,\cdots,U_A\}$ 是所有用户的集合,$I=\{I_1,I_2,\cdots,I_M\}$

**收稿日期** 2018-04-27

**资助项目** 国家自然科学基金(70573025)

**作者简介**

龚志,男,硕士生,研究方向为多媒体信息系统.783586264@qq.com

邵曦(通信作者),男,博士,副教授,研究方向为多媒体信息系统与基于内容的音乐信息检索.shaoxi@njupt.edu.cn

1　南京邮电大学 通信与信息工程学院,南京,210003

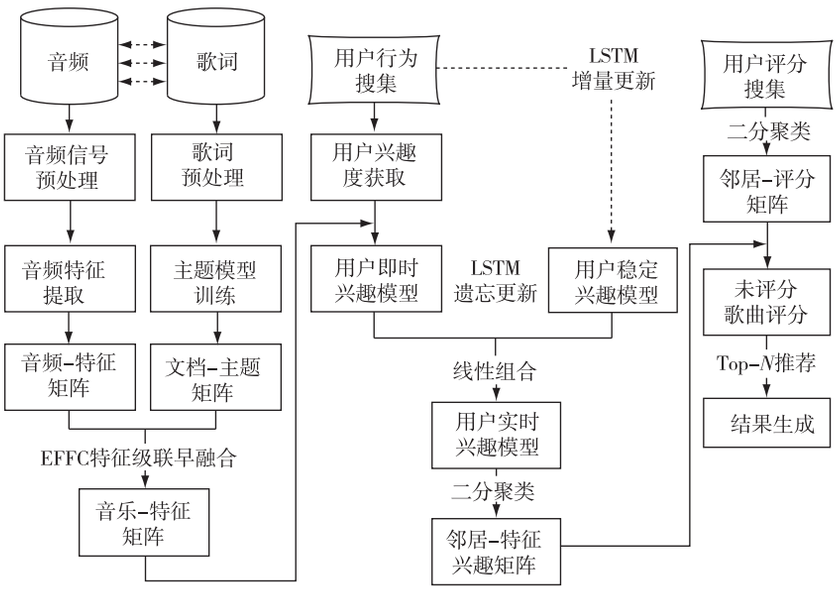


图 1　多模态音乐推荐系统框架

Fig. 1　The framework of recommendation system based on multi-modal fusion

是所有音乐的集合,每个用户对每首歌曲都存在一个评分,用于构建“用户-音乐”评分矩阵并代表用户对该歌曲的喜爱程度.传统的协同过滤推荐在构建评分矩阵后,便直接使用该矩阵来进行相似度计算.

这里引入用户兴趣模型的概念,相较评分矩阵而言用户兴趣模型反映了用户对某些特征而不是对某些歌曲的喜爱程度,方便协同推荐时引入音乐自身的特征.同时,用户的兴趣会随时间发生变化,评分矩阵无法做到实时更新,而通过建立用户即时(短期)兴趣模型与稳定(长期)兴趣模型,即可实现用户模型的动态维护,从而达到音乐推荐的客观性与实时性.

由于用户兴趣模型可以更好地契合本文的思想,故将用户兴趣模型贯穿整个推荐流程的始终,以实现多模态的音乐推荐.

## 2　音乐特征的提取与多模态融合

音乐特征提取与分析是本音乐推荐系统的基础.特征的提取包括音频特征的提取、歌词信息的提取以及歌词信息的降维 3 个部分,将得到的多个音乐特征经多模态融合后建立音乐数据库并以此为依据构建用户兴趣模型.

### 2.1　音乐特征的提取与多模态融合

音频特征的提取过程主要分为以下 2 个阶段:

1)预处理过程.将所有音频文件转化为统一的 AAC 音乐格式,并从每首歌中取出 20 s(第 50~70 s)转化为单声道信号并进行下一步分析.

2)声学特征提取过程.该过程主要提取一些描述音乐频率、节奏与音色等底层的声学特征.常用的声学特征包括 20 维的 Mel 频率倒谱系数(MFCC)、21 维的感知线性预测系数(PLP)以及 9 维的 PLP 倒谱相关系数.本文选择 MFCC 参数作为声学特征.

MFCC 是基于模仿人耳的听觉特性所提取的短时特征.对人耳而言,1 kHz 以下的声音频率与人的感知能力呈线性关系,1 kHz 以上则呈非线性的对数关系,而 MFCC 正是模拟了这种特性,将其线性频谱映射到基于听觉感知的非线性 Mel 频谱中并最终转换到倒谱上,因此能很好地反映人耳对于音频信息的感知[3].

MFCC 提取过程包括以下步骤:

1)归一化 Mel 滤波器组的系数及倒谱提升窗口,并设置预加重滤波器;

2)对语音信号进行分帧,计算每帧的 MFCC 参数;

3)进行快速傅里叶变换(FFT)将信号从时域转换到频域上,再进行 Mel 滤波并计算倒谱;

4)求取一阶差分系数,合并 MFCC 参数和一阶差分 MFCC 参数;

5)去除一阶差分参数为 0 的首尾两帧.

最终每一首歌曲得到 5 506 帧×$K$ 维的 MFCC 参数,对这 5 506 帧的结果求均值后可得出每一首歌曲的 1×$K$ 维“音频-特征”向量.

### 2.2 歌词信息的提取

由于歌词信息是文本的形式,为了便于计算机的保存与处理,需将歌词进行数字化转换.向量空间模型(VSM)是由 Salto 等[4]提出的一种文本表示方法,该方法将歌词文档表示成高维空间中的向量,每篇文档对应一个向量,该向量中的每一维对应文档的每一个特征项.

VSM 的主要步骤包括:

1)预处理过程.对每一篇歌词文档进行分词,使得句子中的每一个词语分开,并去除掉某些没有意义且浪费空间的词语(停用词).比如一篇文档 $d$ 经过分词、去除停用词后还剩下 $n$ 个特征词,便可建立一个 $1\times n$ 维的"文档 - 词 语" 向量 $\boldsymbol{d}_j=(t_1,t_2,\cdots,t_n)$,其中,$t_i$ 表示特征词,$t_i$ 的值代表该特征词在本篇歌词中出现的次数.

2) 计算特征词的权重.特征词在该歌词中出现的次数能反映出音乐的情感趋势,但不同的歌曲的歌词总数不同,只计算特征词出现次数的方式显得不够"公平".为了能够反映出某特征词是否具有代表性,本文采用词频 - 逆文档频率(TF-IDF) 来计算特征词的权重.

Salton 等[5] 提出了 TF-IDF 算法.该算法主要体现了以下思想:一个词在特定的文档中出现的频率越高,说明它在区分该文档内容属性方面的能力越强(TF);一个词在文档中出现的范围越广,说明它区分文档内容的属性越低(IDF)[6].公式如下:

$$\mathrm{TFIDF}_{i,j}=\frac{N_{ij}}{N_{*j}}\times\log\frac{D}{D_i},$$

其中,$\mathrm{TFIDF}_{i,j}$ 表示特征词 $t_i$ 在文档 $\boldsymbol{d}_j$ 中所占的权重,$N_{ij}$ 表示特征词 $t_i$ 出现在文档 $\boldsymbol{d}_j$ 中的次数,$N_{*j}$ 表示文档 $\boldsymbol{d}_j$ 中所有词的个数,$D$ 表示文档总数,$D_i$ 表示文本集中包含特征词 $t_i$ 的文档数.

### 2.3 歌词信息的降维

在实际操作中,由于表示音频特征的 MFCC 参数只有几十维,而一首歌的歌词中可能出现上百个特征词(几百维),多模态融合后的特征信息一定会偏向反映歌词信息而疏远音频特征.为了解决音频与歌词之间的不平衡,还需要对歌词信息进行降维,这里将"文档-词语"向量的维度降至与 MFCC 参数相同的 $K$ 维即可.

降维的方法有很多,传统的有奇异值分解(SVD)、非负矩阵分解(NMF)等.1990 年 Deer wester 等[7]提出采用奇异值分解 SVD 方法来过滤文档中的噪声,即潜在语义分析(LSA),将文档从稀疏的高维特征词空间映射到一个低维的向量空间上.LSA 采用基于数学的方式进行矩阵分解以达到降维的目的,故分解出的矩阵缺乏解释性.随着对 LSA 的深入优化,主题模型逐渐发展起来.

所谓主题模型,就是通过引入一个统计模型,用来抽离出隐含在文档中的主题(Topic).假设一篇文档可以由多个 Topic 混合而成,而每个 Topic 都是词汇上的概率分布,且文章中的每个词都是由一个固定的 Topic 生成的,那么可以通过:

$$p(\text{从}\mid\text{复})=\sum_{\text{主题}}p(\text{词语}\mid\text{主题})\times p(\text{主题}\mid\text{文档})$$

将一篇歌词文档的"文档-词语"向量映射为低维的"文档-主题"向量.

1999 年,Thomas[8] 在 LSA 的基础上提出了概率潜在语义分析(pLSA),pLSA 引入概率模型的方式来表达 LSA 问题,使得每个变量以及相应的概率分布和条件概率分布都有了明确的物理解释.pLSA 的主题概率分布是一个确定的概率分布,虽然主题本身不确定,但主题符合的概率分布是确定的,若符合高斯分布,那这个高斯分布的各个参数就都是确定的,也正因如此,pLSA 存在着过拟合问题.

针对 pLSA 的缺点,Blei 等[9] 于 2003 年提出了 LDA 主题模型.对 LDA 而言,文档中每个主题出现的概率是不确定的,在选取之前需要利用 Dirichlet 先验随机参数确定出主题和词分布,再进行进一步分析.由于 LDA 比 pLSA 更具鲁棒性,本文采用 LDA 主题模型提取出歌词文档的主题,以达到降维的目的.LDA 主题模型的结构如图 2 所示.

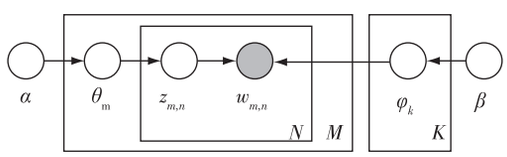


图 2 LDA 主题模型

Fig. 2 LDA topic model

图 2 中,$M$ 表示歌词文档 $m$ 的集合,$N_m$ 表示第 $m$ 篇文档中特征词 $n$ 的个数,令文档隐含的主题数 $k$ 共有 $K$ 个,$\alpha$ 和 $\beta$ 是 Dirichlet 分布先验参数,这 2 个参数对每篇文档都一样,用于控制每篇文档的概率分布和条件概率分布,$\theta$ 对每篇文档都一样,$\theta_m$ 表示第 $m$ 篇文档的主题分布,$w_{m,n}$ 表示第 $m$ 篇文档中的第 $n$ 个特征词,$z$ 用来表征文档中特征词的主题分布,$z_{m,n}$ 表示第 $m$ 篇文档中的第 $n$ 个特征词对应的主题[10].

使用 LDA 主题模型进行降维的过程就是求出 $\theta_m$ 后验分布的过程,具体步骤如下:

1)$\theta_m$ 服从概率分布 $p(\theta_m)$,称为参数 $\theta_m$ 的先验分布;$z$ 满足多项式分布 $z_m \sim \mathrm{Mult}(z_m \mid \theta_m)$,所以选择 Dirichlet 分布为先验分布,得到:

$$\mathrm{Dir}(\theta_m \mid \alpha) = \frac{1}{\Delta(\alpha)} \prod_{k=1}^{K} p_k^{\alpha_k - 1}, \quad \alpha = \alpha_1, \alpha_2, \cdots, \alpha_k, \tag{1}$$

其中,$\Delta(\alpha)$ 为归一化因子 $\mathrm{Dir}(\alpha)$,

$$\Delta(\alpha) = \int \prod_{k=1}^{K} p_k^{\alpha_k - 1} \mathrm{d}\theta_m.$$

2) 由参数 $\theta_m$ 的先验分布 $\mathrm{Dir}(\theta_m \mid \alpha)$,以及各主题出现的次数 $n_m \sim \mathrm{Mult}(n_m \mid \theta_m, N_m)$,其中 $n_m = (n_m^{(1)}, \cdots, n_m^{(k)})$,$n_m^{(k)}$ 表示第 $m$ 篇文档中第 $k$ 个主题产生词语的个数,可得到 $\theta_m$ 的后验分布为

$$p(\theta_m \mid z_m, \alpha) = \mathrm{Dir}(\theta_m \mid n_m + \alpha) = \frac{1}{\Delta(n_m + \alpha)} \prod_{k=1}^{K} p_k^{n_k + \alpha_k - 1}, \tag{2}$$

则可推算出第 $m$ 篇文档的主题分布为

$$p(z_m \mid \alpha) = \int p(z_m \mid \theta_m) p(\theta_m \mid \alpha) \mathrm{d}\theta_m = \frac{1}{\Delta(\alpha)} \int \prod_{k=1}^{K} p_k^{n_k + \alpha_k - 1} \mathrm{d}\theta_m = \frac{\Delta(n_m + \alpha)}{\Delta(\alpha)}. \tag{3}$$

完成 $\theta_m$ 后验分布 $p(z_m \mid \alpha)$ 的求解后,即可将几百维的“文档-词语”向量降至 $K$ 维的“文档-主题”向量.

### 2.4 多模态融合

通过音频特征的提取,一首歌可以表示成 $1 \times K$ 维的“音频-特征”向量;通过歌词信息的提取与降维,一首歌也可以表示成 $1 \times K$ 维的“文档-主题”向量.接下来就需要将音频特征与歌词信息进行多模态融合,建立音乐数据库并以此为依据构建用户兴趣模型.

常用的多模态融合方法主要分为特征级融合法和决策级融合法[11].特征级融合法的主要思路是通过将音频特征与歌词信息以串联的方式结合起来并归一化后作为音乐的整体特征,并对整体特征进行下一步操作.决策级融合法的主要思路是先对音频特征与歌词信息进行分类,并赋予分类结果某种既定规则,再以该规则作为音乐的整体特征.对本文提出的方法,显然决策级融合法的可操作性更强,这里选用特征级联早融合法(EFFC)作为两种特征信息多模态融合的方法.

EFFC 方法的优点是在音频特征空间与歌词信息空间的基础上将二者映射到了一个统一的多模态特征空间,对于特征信息的后续处理,只需要对该多模态特征空间进行操作即可,不再需要对音频特征与歌词信息进行训练,大幅提高了操作效率.

EFFC 方法的具体步骤如下:

1)构造映射矩阵.这里构造 2 个 $K \times 2K$ 维的矩阵($\boldsymbol{I} \mid \boldsymbol{0}$) 与($\boldsymbol{0} \mid \boldsymbol{I}$),它们都由一个 $K \times K$ 维的单位矩阵 $\boldsymbol{I}$ 与一个 $K \times K$ 维的零矩阵组成;

2) 映射“音频-特征”矩阵.将 $M$ 首音乐的 $M \times K$ 维“音频-特征”矩阵 $\boldsymbol{A}$ 与映射矩阵($\boldsymbol{I} \mid \boldsymbol{0}$) 相乘,得到一个 $M \times 2K$ 维的矩阵($\boldsymbol{A} \mid \boldsymbol{0}$);

3) 映射“文档 - 主题”矩阵.将 $M$ 首音乐歌词的 $M \times K$ 维“文档-主题”矩阵 $\boldsymbol{B}$ 与映射矩阵($\boldsymbol{0} \mid \boldsymbol{I}$) 相乘,得到一个 $M \times 2K$ 维的矩阵($\boldsymbol{0} \mid \boldsymbol{B}$);

4) 矩阵融合.串联 $M \times 2K$ 维的矩阵($\boldsymbol{A} \mid \boldsymbol{0}$) 与 $M \times 2K$ 维的矩阵($\boldsymbol{0} \mid \boldsymbol{B}$),即可得到包含 2 种音乐特征信息的 $M \times 2K$ 维“音乐-特征”矩阵($\boldsymbol{A} \mid \boldsymbol{B}$).

通过上述过程,可对所有音乐构建一个包含多模态特征的音乐数据库,为后续建立用户兴趣模型打下基础.

## 3 兴趣模型的建立与用户聚类

### 3.1 获取用户个性化信息

个性化信息收集方式分为 2 种:一种是显式收集方式,主要通过用户与系统的交互实现,用户通过系统提供的选项直接告诉系统其对所有项目的评分;另一种是隐式收集方式,主要通过挖掘用户的访问和浏览历史去推算出用户对各个项目的兴趣度[12].

#### 3.1.1 显式获取评分矩阵

在用户第一次登录时,请求用户对歌曲进行评分(不必对所有歌曲评分,只需用户评价感兴趣的歌曲),得到如表 1 所示的“用户-音乐”评分矩阵.

**表 1 “用户-音乐”评分矩阵**

Table 1 “User-Music” score matrix

| | $I_1$ | $I_2$ | … | $I_q$ | … | $I_M$ |
|---|---|---|---|---|---|---|
| $U_1$ | 10 | 8 | … | 9 | … | \ |
| $U_2$ | \ | \ | … | 10 | … | 7 |
| ⋮ | ⋮ | ⋮ | | ⋮ | | ⋮ |
| $U_j$ | 7 | 9 | … | \ | … | 8 |
| ⋮ | ⋮ | ⋮ | | ⋮ | | ⋮ |
| $U_A$ | \ | \ | … | \ | … | \ |

注:\表示用户未对歌曲评分.

#### 3.1.2 隐式用户行为分析

为了隐式收集用户对歌曲的兴趣度,需建立一

个兴趣函数 Interest($I_i$),该函数反映了用户对某一首歌曲 $I_i$ 的兴趣度.这里认为用户对一首歌曲的行为包括:下载、评论、分享与收听,可相应地建立下载函数 Download($I_i$)、评论函数 Comment($I_i$)、分享函数 Share($I_i$)、收听次数函数 Times($I_i$)以及收听时长函数 Duration($I_i$,$t$),则有:

$$\text{Interest}(I_i)=f(\text{Download}(I_i),\text{Comment}(I_i),\text{Share}(I_i),\text{Times}(I_i),\text{Duration}(I_i,t)).$$

获取用户对歌曲兴趣度的过程如下所示:

1) 收听次数函数 Times($I_i$).针对一次收听,用户对歌曲 $I_i$ 的兴趣度为

$$\text{perInterest}(I_i,n)=f(\text{Download}(I_i),\text{Comment}(I_i),\text{Share}(I_i),\text{Duration}(I_i,t)),$$

只需将单次兴趣度 perInterest($I_i$,$n$) 根据收听次数 $n$ 进行叠加即可得到 Interest($I_i$).

2) 下载函数 Download($I_i$)、评论函数 Comment($I_i$)、分享函数 Share($I_i$).只要下载、评论、分享这 3 个行为中有任何一个发生就代表用户对这首歌有最高兴趣度(兴趣度为 1),故 3 个函数的计入原则为:若用户首先下载了歌曲 $I_i$,则 perInterest($I_i$,$n$) = Download($I_i$) = 1,无论评论、分享行为是否发生,都不计入这 2 个行为产生的兴趣度,同时收听时长函数 Duration($I_i$,$t$) 也不计入 perInterest($I_i$,$n$);同理若用户首先分享了歌曲 $I_i$,那么 perInterest($I_i$,$n$) = Share($I_i$) = 1,其他行为也不再计入;以此类推.

3) 收听时长函数 Duration($I_i$,$t$).上文介绍了当下载、评论、分享中任何一个行为发生时,Duration($I_i$,$t$) 不计入兴趣度,而这 3 个行为均未发生时,则有 perInterest($I_i$,$n$) = Duration($I_i$,$t$).这里给出 2 个临界时间参数:$t_1$ 与 $t_2$.若用户的收听时长 $t < t_1$ 时,表示用户可能不喜欢这首歌就快速进行切歌状态,perInterest($I_i$,$n$) = Duration($I_i$,$t$) = 0;若收听时长 $t > t_2$ 时,表示用户可能已离开播放器或者忘记关闭音乐,perInterest($I_i$,$n$) = Duration($I_i$,$t$) = 0;若收听时长 $t \in (t_1,t_2)$,则 Duration($I_i$,$t$) 符合线性函数,有 Duration($I_i$,$t$) = $kt+b$($k>0$,$b<0$),且仍满足最高兴趣度 max[perInterest($I_i$,$n$)] = 1 的条件.此时,可以将单次兴趣度函数归纳为

$$\text{perInterest}(I_i,n)=f(\text{Download}(I_i),\text{Comment}(I_i),\text{Share}(I_i),\text{Duration}(I_i,t))=$$
$$[1-\text{Action}(I_i)]\times\text{Duration}(I_i,t)+\text{Action}(I_i),$$

其中,当下载、评论、分享中任何一个行为发生时,Action($I_i$) = 1,若 3 个行为均未发生时,Action($I_i$) = 0.

综上,通过用户行为分析得到用户对某一首歌曲的兴趣度:

$$\text{Interest}(I_i)=\sum_{n=1}^{\text{Times}(I_i)}\text{perInterest}(I_i,n).$$

### 3.2 建立用户兴趣模型

#### 3.2.1 即时兴趣模型

通过第 2 章的特征提取,得到了多模态特征信息的 $M\times 2K$ 维“音乐-特征” 矩阵($\boldsymbol{A}\mid\boldsymbol{B}$),对于第 $i$ 首歌而言,$1\times 2K$ 维“音乐-特征” 向量可以表示为

$$I_i=\{(f_1,w_{i,1}),(f_2,w_{i,2}),\cdots,(f_{40},w_{i,40})\},$$

其中,$f_j$ 表示第 $j$ 个特征,$w_{i,j}$ 表示第 $i$ 首歌中第 $j$ 个特征的权值.此时结合 3.1 节中给出的用户对某一首歌曲的兴趣度 Interest($I_i$),可计算出用户对歌曲 $I_i$ 的即时兴趣:

$$\left\{\left(f_1,\frac{\text{Interest}(I_i)}{\max[\text{Interest}(I_x)]}\times w_{i,1}\right),\left(f_2,\frac{\text{Interest}(I_i)}{\max[\text{Interest}(I_x)]}\times w_{i,2}\right),\cdots\right\},$$

$\max[\text{Interest}(I_x)]$ 表示在音乐数据库全部的 $M$ 首歌中,用户对歌曲 $I_x$ 的兴趣度最大,故用作分母以进行归一化.

通过上述分析,可得出用户对 $M$ 首歌的即时兴趣模型为

$$\{(f_1,W_1^s),(f_2,W_2^s),\cdots,(f_{40},W_{40}^s)\},$$

其中,

$$W_i^s=\frac{1}{M}\sum_{m=1}^{M}\left[\frac{\text{Interest}(I_m)}{\max[\text{Interest}(I_x)]}\times w_{m,i}\right].$$

#### 3.2.2 稳定兴趣模型与 LSTM 神经网络

当用户的即时兴趣模型建立后,可以充分代表用户在一段时间内对不同音乐特征的兴趣度,但随着用户不断地收听新的音乐或是很长一段时间不听某种音乐,其喜好程度也会随之改变,故即时兴趣模型无法反映出用户兴趣度的变化趋势.

长短时记忆模型(LSTM)于 1997 年由 Sepp 等[13]提出,它可以模拟出人类的记忆曲线,实现新信息的输入、旧信息的遗忘以及最终信息的输出等功能.本文采用 LSTM 神经网络,将用户的即时兴趣模型作为输入,通过神经网络内部各节点的计算,输出用户的稳定兴趣模型.

图 3 是一个基本的 LSTM 神经网络模型,并结合本文要求进行了相应的修改.LSTM 有 3 种特殊的门结构,分别是输入门、输出门与遗忘门,是 LSTM 实现记忆与遗忘的关键.

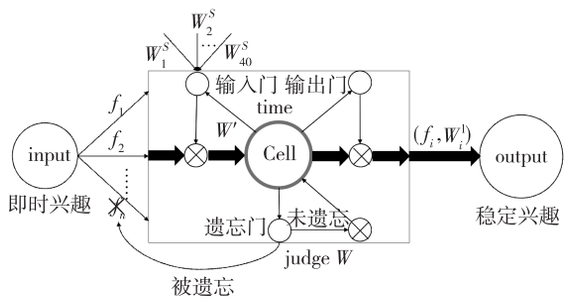


图 3 LSTM 神经网络的结构

Fig. 3 The structure of LSTM network

LSTM 训练稳定兴趣模型的步骤如下:

1)LSTM 第 1 次训练时,以用户的即时兴趣模型作为输入,在神经网络内部进行相应计算后通过输出门将当前状态进行输出.

2)第 2 次训练时,以第 1 次训练的输出状态作为输入,在输入门中进行处理,同时将用户对各个特征的权值(兴趣度)$W$ 送入遗忘门,由遗忘门根据该特征当前的权值判断其是否应该被遗忘.若被遗忘则将该特征的权值置零;若未被遗忘则通过输出门将当前状态进行保存输出.

3)接下来的训练同理,通过输出门保存上一次训练的状态并作为当前训练的输入.若用户收听了新的歌曲,则需要通过输入门更新对应特征的权值,再重新进行训练.

4)选用梯度最速下降法来限定 LSTM 训练的迭代次数.在迭代过程中,一旦损失值(loss)收敛到 1 以下就可终止训练过程.此时输出门的结果就是用户的稳定兴趣模型:

$$\{(f_1,W_1^l),(f_2,W_2^l),\cdots,(f_{40},W_{40}^l)\}.$$

3.2.3 实时兴趣模型

在 3.2.1 节与 3.2.2 节中已分别得到了用户的即时兴趣模型与稳定兴趣模型,这里还需结合这 2 种兴趣模型来构建用户的实时兴趣模型.一般采用加权平均的方法来计算某个特征在实时兴趣模型中的权重,对于特征 $f_i$,其权重 $W_i$ 可以表示为

$$W_i=\alpha\times W_i^s+\beta\times W_i^l,$$

其中,$W_i^s$ 为即时兴趣模型中特征 $f_i$ 所对应的权值,$W_i^l$ 为稳定兴趣模型中该特征所对应的权值,$\alpha+\beta=1$,$W_i\in(0,1]$.由于即时兴趣更能体现用户当下的兴趣爱好,再经过多次实验后选定 $\alpha=0.6$,$\beta=0.4$.

用户实时兴趣模型如表 2 所示.其中,$W_{j,p}$ 表示第 $j$ 个用户对第 $p$ 个特征的实时兴趣度.

表 2 "用户-特征"实时兴趣模型

Table 2 "User-Feature" real-time interest model

| | $f_1$ | $f_2$ | … | $f_p$ | … | $f_K$ |
|---|---|---|---|---|---|---|
| $U_1$ | $W_{1,1}$ | $W_{1,2}$ | … | $W_{1,p}$ | … | $W_{1,K}$ |
| $U_2$ | $W_{2,1}$ | $W_{2,2}$ | … | $W_{2,p}$ | … | $W_{2,K}$ |
| ⋮ | ⋮ | ⋮ | | ⋮ | | ⋮ |
| $U_j$ | $W_{j,1}$ | $W_{j,2}$ | … | $W_{j,p}$ | … | $W_{j,K}$ |
| ⋮ | ⋮ | ⋮ | | ⋮ | | ⋮ |
| $U_A$ | $W_{A,1}$ | $W_{A,2}$ | … | $W_{A,p}$ | … | $W_{A,K}$ |

通过上述过程,可对所有用户构建一个实时兴趣模型,为后续用户聚类和个性化推荐打下基础.

### 3.3 用户聚类

3.1.1 节中通过显式方式获取了"用户-评分"矩阵.由于用户并非对所有歌曲都进行评分,使得该矩阵的稀疏性很大,此时可通过用户聚类将目标用户与其邻居放入同个用户组内,令"用户-评分"矩阵转变为"邻居-评分"矩阵,在某种程度上降低矩阵的稀疏性.

常用的聚类方法有 $k$-means 算法,它采用距离作为判断相似性的标准,即认为 2 个对象的距离越近,其相似度就越大.该算法存在一些缺点:若选取的初始聚类中心点不当,就会对聚类结果产生较大的影响.这里使用一种改进算法——基于二分的聚类算法(bi-section)来进行用户聚类.

二分聚类算法的过程如下:

1)初始情况下,将所有用户作为一个簇,然后随机选取 2 个初始聚类中心点(seed),使用余弦相似度依照用户实时兴趣模型计算用户之间的相似度 $\cos(U_i,U_j)$,并以此将该类分为 2 个簇;

2)从 2 个簇中选择用户数量较多的簇,再从该簇中随机选出 2 个 seed,同样将该簇中的用户分配到 2 个子簇中;

3)此时产生了 3 个簇,但这 3 个簇并非最优,需要进行调整,即将任意一个簇中的任意一个用户移动到其他簇中,若结果变优,则执行本次操作,否则不执行.

判断结果是否变优包括:

1)簇内优化:所有簇的内部平均相似度 $l$ 达到最大;

2)簇间优化:该簇与所有簇之间的相似度期望值 $E$ 最小.

故可建立一个优化函数 fun(),使得:

$$\text{fun}()=\min\left(\frac{\min E}{\max l}\right).$$

以用户 $j$ 为例,通过用户聚类后得到了用户 $j$ 的邻居用户 $U_1$、$U_2$ 和 $U_5$,通过邻居用户的评分即可进行个性化推荐.聚类后的"邻居-特征"实时兴趣模型如表 3 所示,"邻居-评分"矩阵如表 4 所示.

**表 3　聚类后的"邻居-特征"实时兴趣模型**

Table 3　Clustered "Neighbor-Character" real-time interest model

| | $f_1$ | $f_2$ | … | $f_p$ | … | $f_K$ |
|---|---|---|---|---|---|---|
| $U_1$ | $W_{1,1}$ | $W_{1,2}$ | … | $W_{1,p}$ | … | $W_{1,K}$ |
| $U_2$ | $W_{2,1}$ | $W_{2,2}$ | … | $W_{2,p}$ | … | $W_{2,K}$ |
| $U_5$ | $W_{5,1}$ | $W_{5,2}$ | … | $W_{5,p}$ | … | $W_{5,K}$ |
| $U_j$ | $W_{j,1}$ | $W_{j,2}$ | … | $W_{j,p}$ | … | $W_{j,K}$ |

**表 4　聚类后的"邻居-评分"矩阵**

Table 4　Clustered "Neighbor-Score" matrix

| | $I_1$ | $I_2$ | … | $I_q$ | … | $I_M$ |
|---|---|---|---|---|---|---|
| $U_1$ | 10 | 8 | … | 9 | … | \ |
| $U_2$ | \ | \ | … | 10 | … | 7 |
| $U_5$ | 8 | 10 | … | 7 | … | \ |
| $U_j$ | 7 | 9 | … | ? | … | 8 |

注:\表示用户未对歌曲评分.

## 4　实验及结果

### 4.1　实验数据集

用于实验的数据集中,音频文件为 AAC 格式,截取时长为 20 s 且已转化成单声道信号的 500 首英文歌曲,这些音频文件为 iTunes 美区商店售卖的正版音源,基本涵盖了 Pop、EDM、Classical、Country、Rock、R&B 以及 Hip-Hop 等各种流派的音乐.为了保证截取的音乐片段具有代表性,经多人讨论后选择出一首歌最具代表性的第 50~70 s.歌词文件为 txt 格式,且与音频文件一一对应,用户总数为 40 人,其中 1 人为目标用户.通过问卷调查的方式收集这 40 人对 500 首歌中任意 100 首歌曲的评分.

### 4.2　结果分析

#### 4.2.1　个性化推荐结果

上文已完成了基于多模态的音乐推荐系统的前期工作,包括由音乐的多模态特征建立了用户的兴趣模型,又由兴趣模型找到了目标用户的邻居.这里同样指定 $U_j$ 为目标用户.首先选定音频特征与歌词文档主题的维数 $K=20$,接下来根据其邻居用户 $U_1$、$U_2$ 和 $U_5$ 来预测 $U_j$ 对未评分资源的得分,从而进行个性化推荐.

个性化推荐的步骤如下:

1)3.3 节的二分聚类过程中,已得到 $U_j$ 与其邻居用户的余弦相似度:$\cos(U_j,U_1)$、$\cos(U_j,U_2)$ 与 $\cos(U_j,U_5)$,分别简记为 $c_1$、$c_2$ 与 $c_5$.

2) 计算 $U_j$ 所有已评分歌曲的平均分 $\bar{r}_j$.

3) 针对某一首 $U_j$ 未评分的歌曲 $q$,由邻居用户分别对歌曲 $q$ 的评分 9、10、7 来预测 $U_j$ 对该歌曲的评分 $r_{j,q}$:

$$r_{j,q}=\bar{r}_j+\frac{c_1\times(9-\bar{r}_j)+c_2\times(10-\bar{r}_j)+c_5\times(7-\bar{r}_j)}{c_1+c_2+c_5}.$$

同理,可计算出 $U_j$ 所有未评分歌曲的得分.

4) 采用 Top-$N$ 推荐方法,罗列出 $U_j$ 所有未评分歌曲的得分,并降序排列,如:$\{(I_4,10),(I_5,9),(I_q,9),\cdots\}$,并将得分最高的前 $N$ 首未评分歌曲推荐给用户.

将数据集中的 500 首音乐、500 篇歌词文档以及 40 位用户的评分作为输入,输出个性化推荐的 3 首歌曲如图 4 所示.

```
为您推荐以下曲目:
    "03 i did something bad"
    "09 getaway car"
    "15 new year_s day"
```

图 4　个性化推荐结果

Fig. 4　The result of personalized recommendation

#### 4.2.2　验证结果的准确性

除了验证系统的有效性之外,还需进行对比实验以验证实验结果的准确性.这里给出 3 种对比环境,这 3 种环境也能充分代表目前常见的几种音乐推荐系统:

1)环境 1:本系统,即基于音频特征与歌词信息的多模态音乐推荐系统;

2)环境 2:只基于音频特征的单模态音乐推荐系统;

3)环境 3:不采用任何音乐内容特征,传统的协同过滤推荐系统.

选用误差平方和(SSE)作为结果准确性的判断标准.SSE 计算的是拟合数据和原始数据对应点的误差的平方和,SSE 越接近于 0,说明模型选择和拟合更好,数据预测也越成功.

选取 40 位用户中任意 1 位作为目标用户,将该用户的"用户-评分"向量中部分真实评分删除,并分别在 3 种不同对比环境下预测该用户的得分,计算被删除的真实评分与预测得分之间的 SSE,结果如图 5 所示.

| | |
|---|---|
| 采用多模态的预测得分与真实评分的差异SSE: | 0.3886 |
| 采用单模态的预测得分与真实评分的差异SSE: | 1.1111 |
| 采用传统的协同过滤的预测得分与真实评分的差异SSE: | 2.0090 |

图 5　预测得分与真实评分的差异 SSE

Fig. 5　SSE between predicted score and real score

通过图 5 可以证实多模态音乐推荐系统与真实数据的拟合度最高,其次是单模态系统,而传统的协同过滤拟合度最低,这一结论也充分验证了多模态音乐推荐系统准确性较高.

图 6 为不同环境下用户的真实评分与预测得分曲线.为使折线图更加直观易读,这里随机挑选 500 首歌中的 15 首来作图.

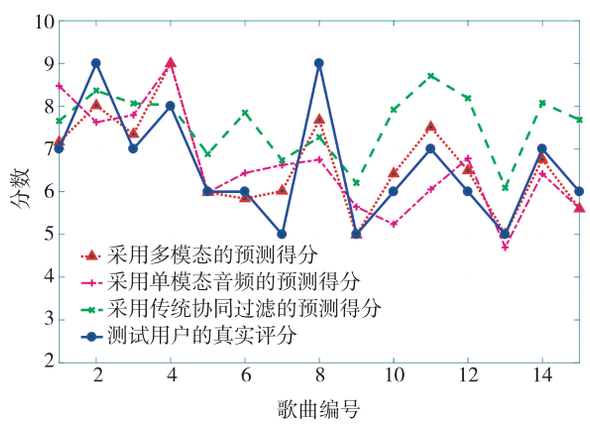


图 6　不同环境下用户的真实评分与预测得分

Fig. 6　Real score and predicted score under different environments

图 7 则分别计算了 3 种环境下,真实评分与预测得分的偏差.

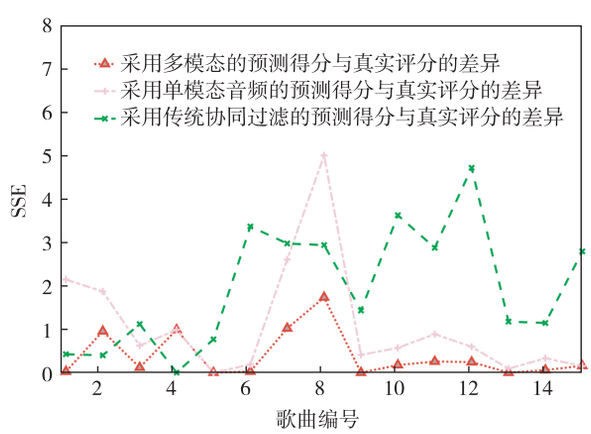


图 7　不同环境下预测得分与真实评分的偏差

Fig. 7　Differences between predicted score and real score under different environments

从图 6 中可以看出虚线代表的多模态推荐系统下的预测得分与真实评分最为拟合,单模态系统次之,传统协同过滤效果最差;从图 7 可以看出多模态推荐系统下的预测得分偏差最小,也最接近 0.

综上所述,基于音频特征与歌词信息的多模态音乐推荐系统无论从有效性方面还是从准确性方面来说,都要比目前常见的几种音乐推荐系统效果更好.

## 5　结论与未来工作展望

文中描述了一个将音乐的音频特征和歌词信息经过多模态融合的音乐推荐系统,应用 LDA 模型来处理歌词信息.对于多模态融合方法的问题,使用了 EFFC 融合方法,并建立了多模态音乐数据库.将用户兴趣模型通过 LSTM 神经网络进一步得到用户的实时兴趣模型,以此作为用户聚类的基础.在与传统协同过滤推荐方法相对比后发现,该方法的准确率确有一定程度的提高,并在最后充分验证了该方法的有效性和准确性.今后,将把研究重点放在如何把用户的情感信息作为推荐系统的另一极,根据用户的情感进一步筛选个性化推荐的结果.

## 参考文献

References

[1] 曾子明.信息推荐系统[M].北京:科学出版社,2015:65-66
ZENG Ziming. Information recommendation system[M]. Beijing: China Science Publishing & Media Ltd, 2015: 65-66

[2] 唐沁钦.多媒体系统中个性化推荐的研究和设计[D].北京:北京交通大学,2011
TANG Qinqin. Design of personalized information delivery in multimedia system[D]. Beijing: Beijing Jiaotong University, 2011

[3] 薛昊.基于多模态融合的音乐情感分类方法[D].南京:南京大学,2016
XUE Hao. The research on music mood classification methods based on multi-modal fusion [D]. Nanjing: Nanjing University, 2016

[4] Salton G, Wong A, Yang C S. A vector space model for automatic indexing [J]. Communications of the ACM, 1975, 18 (11): 613-620

[5] Salton G, Yu C T. On the construction of effective vocabularies for information retrieval [C] // Proceedings of the 1973 Meeting on Programming Languages and Information Retrieval. New York: ACM, 1973: 11

[6] 施聪莺,徐朝军,杨晓江.TFIDF 算法研究综述[J].计算机应用,2009,29(b06):167-170
SHI Congying, XU Chaojun, YANG Xiaojiang. Study of TFIDF algorithm[J]. Journal of Computer Applications,

2009,29(b06):167-170
[7] Deerwester S,Dumais S T,Furnas G W,et al.Indexing by latent semantic analysis[J].Journal of the American Society For Information Science,1990,41:391-407
[8] Thomas H.Probabilistic latent semantic indexing[C]// Proceedings of the Twenty-Second Annual International SIGIR Conference on Research and Development in Information Retrieval (SIGIR-99),1999
[9] Blei D M,Ng A Y,Jordan M I.Latent dirichlet allocation [J].Journal of Machine Learning Research,2003,3 (4/5):993-1022
[10] Morris C. Parametric empirical bayes inference: theory and applications[J].Journal of the American Statistical Association,1983,78(381):47-65
[11] Zeng Z,Hu Y,Liu M,et al.Training combination strategy of multi-stream fused hidden markov model for audio-visual affect recognition[C]// Process of 14th ACM International conference of multimedia,2006:65-68
[12] Nichols D M. Implicit rating and filtering [C] // Proceedings of the Fifth DELOS Workshop on Filtering and Collaborative Filtering,1997
[13] Hochreiter S, Schmidhuber J. Long short-term memory [J].Neural Computation,1997,9(8):1735-1780

# A music recommendation system based on multi-modal fusion

GONG Zhi[1] SHAO Xi[1]

1 College of Telecommunications and Information Engineering,Nanjing University of Posts and Telecommunications,Nanjing 210003

**Abstract** Despite the continuous enrichment of music,the underlying music features are often overlooked when using traditional collaborative filtering.By multi-modal fusion of audio features and lyric information and supplementing the fusion information feature as a collaborative filtering recommendation,a multi-modal music recommendation system is proposed.This studyprimarily discusses the extraction of audio features and lyrics information and uses the LDA topic model to reduce the character dimension of the lyrics information.For the multi-model fusion problem, this study proposes an EFFC fusion method,and compares the results of multi-modal fusion with the results using single-mode.For result recommendations,the user interest model is established based on the multi-modal information feature with the input of LSTM networks to filter and optimize the user group.The results show that the multi-modal music recommendation system reduces the SSE of the result from 2. 009 to 0. 388 6,verifying the effectiveness of the method.

**Key words** music recommendation;collaborative filtering;LDA topic model;multi-modal fusion;LSTM networks