个性音乐推荐

--基于网易云音乐歌单数据

陈　婷 17210980029

李　赫 17210980037

杨依莹 17210980048

一. 背景

1. 中国数字音乐市场现状

国际唱片业协会（IFPI）报告显示，2017 年全球音乐市场总收入达 173 亿美元，其中数字音乐收入为 94 亿美元，占全部收入比例超过 54%。由于数字音乐具有便捷性和拓展性等优势，借助电子音乐平台随时随地欣赏音乐已成为大多数消费者的选择。近年来全球市场中实体专辑收入占比持续走低，数字音乐收入（包括数字专辑和会员费收入）占比不断提升，可以说音乐数字化已经成为全球趋势。

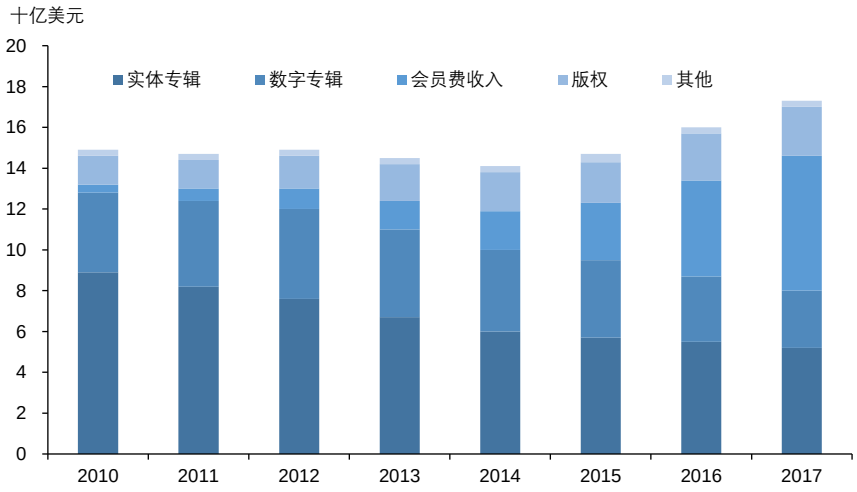

图 1.1 2010-2017 年全球音乐市场收入分布[1]

目前，中国已成为全球第十大音乐市场，2017 年中国音乐市场收入达 5.7 亿美元，世界第一大市场美国收入达 56.5 亿美元。与其他市场略有不同的是，中国音乐市场中超过 90%的收入来自数字音乐，而美国和英国市场这一占比分别为 70%、47%，可见中国音乐市场向数字化转型更为迅速。根据艾瑞咨询的《中国在线音乐用户洞察报告》，2017 年我国在线音乐用户规模达 5.6 亿人，近五年保持 15%以上的年增速，2017 年在线音乐市场规模达 180 亿元，较去年同比增长 26%，我国数字音乐市场未来发展前景广阔。

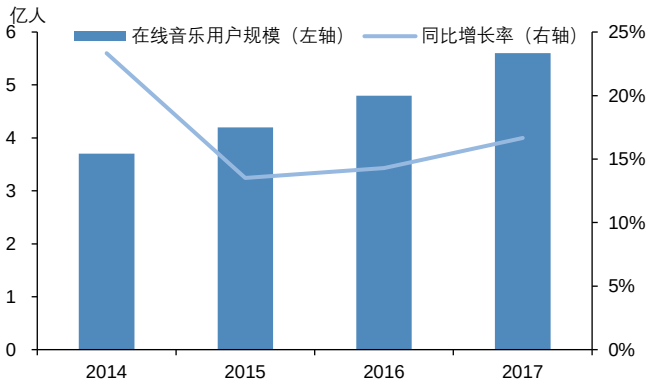

图 1.2 2014-2017 年中国在线音乐用户规模及增长率[2]

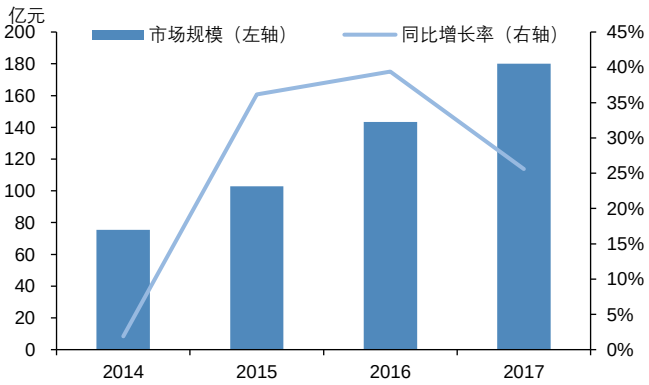

图 1.3 2014-2017 年中国数字音乐市场规模及增长率[3]

[1] 数据来源：IFPI《Global Music Report 2018》
[2] 数据来源：艾瑞咨询《中国在线音乐用户洞察报告》
[3] 数据来源：艾瑞咨询《中国在线音乐用户洞察报告》

2. 优质的推荐系统是数字音乐平台的核心竞争力

丰富的曲库、良好的音质、多方位的交互和推荐算法是在线音乐平台发展的核心竞争力。曲库覆盖的歌手、曲风范围越广，拥有版权的歌曲越多，越能吸引不同的用户，近年来各大音乐平台的版权争夺战日趋白热化，共享版权又成为新趋势。随着移动端硬件的升级，用户对于音乐品质的要求也越高，对于平台而言，无损音质库的积累也将成为未来竞争的壁垒。同时各大音乐平台尝试泛娱乐的新形式包括 K 歌，直播等，有效增加用户忠诚和粘性。但是这些核心竞争能力的提升需要公司投入大量的人力物力成本，比如曾经因版权纠纷而多次相互起诉的腾讯音乐与网易云音乐。

而精准的推荐系统可以细分用户市场，提高用户黏性，所需的成本较小，应当是各大音乐平台提升竞争力的重要抓手。网易云音乐能在竞争激烈的数字音乐平台市场后来居上，与其优质的推荐系统密不可分。在第一次使用时，个性推荐会引导用户做一个简单的喜好分析，同时再根据用户的收藏音乐进行个性化推荐。“私人 FM”类似于前两者的“猜你喜欢”版块，“每日歌曲推荐”会每天向用户推荐 20 首歌曲。同时还会向用户推荐一些歌单，并且告诉用户是根据用户喜欢的哪首歌推荐的。推荐功能不可能做到每一首都符合用户味口，标签式的推荐及可查看的歌单也给了用户选择的余地。我们选取音乐作为推荐的内容，希望构建有效的推荐系统，提升用户满意度，增强用户黏性，帮助音乐平台进一步拓展市场份额。

二. 数据

我们爬取了网易云音乐平台的数据，共获取 28457 个歌单，764705 首歌曲。由下图的累积分布函数和概率密度函数图可知，曲线头部的热门歌曲只占少数，曲线尾部的冷门歌曲体量不容小觑，推荐系统存在着严重的长尾现象。

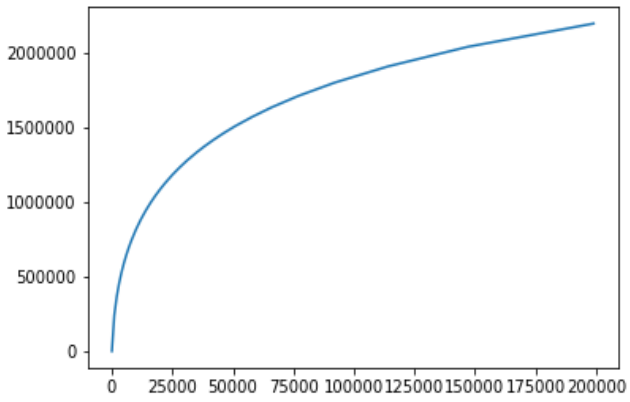

图 2.1 累积分布函数

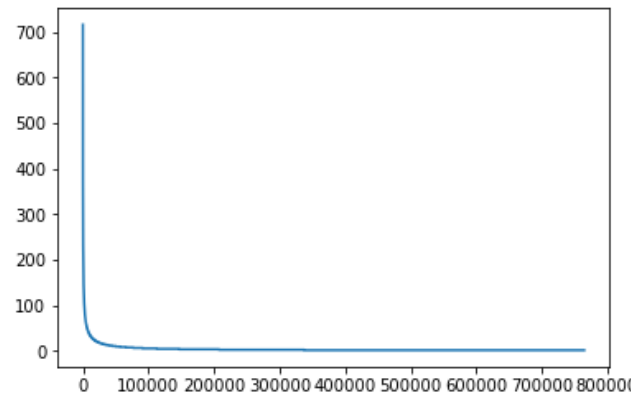

图 2.2 概率密度函数

三. 模型

1. 协同过滤

（1）模型介绍

协同过滤（Collaborative Filtering）是一种基于用户行为的推荐算法。通过分析用户（User）对商品（Item）的评分，进而向用户进行商品推荐。协同过滤根据假设的不同，可以分为两种模型，即基于用户的协同过滤（UserCF）和基于商品的协同过滤（ItemCF）.

基于用户的协同过滤（UserCF）依据与该用户兴趣相似的其他用户的行为，为该用户提供推荐。实现过程如图 3.2 所示，首先计算该用户与其他所有用户的相似度并找出最为相似的用户，将该相似用户喜欢但是该用户尚未评分的物品推荐给他。

基于商品的协同过滤（ItemCF）依据用户之前喜欢的物品数据，推荐与该物品相似的其他物品。实现过程如图 3.3 所示，首先计算该用户喜欢的商品与其他所有商品的相似度并找出最为相似的商品，将与该用户喜欢的商品相似但是该用户尚未评分的物品推荐给他。

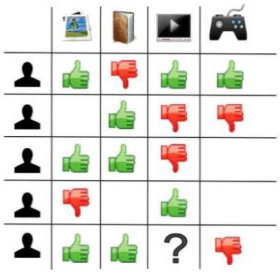

图 3.1 User-Item 表

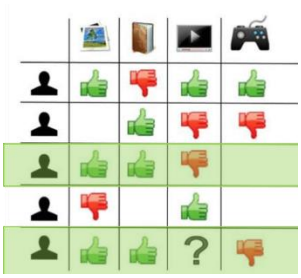

图 3.2 基于用户的协同过滤

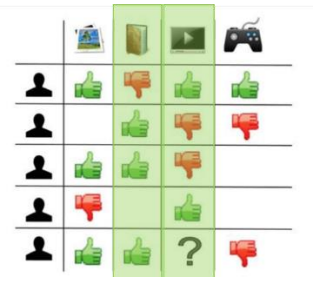

图 3.3 基于商品的协同过滤

（2）模型假设

传统的协同过滤算法基于图 3.1 所示的 User-Item 表进行。这样的模型有一个较强的假设，那就是它假定用户在不同场景、不同心情下，喜好是完全一致的。

然而，用户对歌曲的偏好，实际上是与用户所在情境密切相关的。在开心时，用户偏好欢快的曲风；在悲伤时，用户偏好舒缓的曲风；在热恋时，用户偏好甜蜜的歌曲；在失恋时，用户偏好悲情的歌曲。下表也列示了不同用户在不同场景下，偏好的歌曲。

		不同场景		
		跑步时	写作业时	等车时
不同用户	A：ACG 爱好者	恋爱循环	离人愁	紅楼
	B：影视原声爱好者	超级英雄原声	卡门序曲	星际牛仔主题曲
	C：游戏爱好者	仙剑三主题曲	英雄联盟背景音乐	ICEY 游戏原声

表 3.1 不同用户在不同场景下喜爱歌曲举例

由此，我们可以发现，用户在不同的情境下，将会对歌曲产生不同的偏好，我们称之为用户拥有多个**“音乐子人格”**。我们为这种“音乐子人格”找到了一种最贴近的描述，那就是网易云音乐中的歌单。通常来说，一个歌单反映了非常细粒度的主题，例如歌单《安静看书的背景音乐》，就是一个热爱沉浸式阅读的用户在读书这个特定的场景下的音乐子人格。

（3）模型构建

在传统基于 User-Item 表的推荐当中，由于物品的稀疏性和用户爱好的差异性，一般来说完全一致的用户数量是非常少的。于是在这种场景下，一般不会考虑用户去重的问题。然而，对于细粒度的“音乐子人格”却并非如此，它是一个可以被共享的偏好。一个歌单或许被很多人收藏，而每一次收藏都说明该子人格被某个用户所拥有。

如果沿用 User-Item 基于不去重思想的假设，我们需要将每一个歌单在数据集的数量扩充为 n 倍，其中 n 为该歌单被收藏的次数。这将导致不必要的内存额外开销和低效的算法，因此我们对协同过滤算法进行了一定的修正。修正结果如下表所示。

	公式	ItemCF	UserCF
余弦相似度	$\cos(x,y)=\frac{x\cdot y}{\|\|x\|\|\|\|y\|\|}$	每个元素都乘$\sqrt{n}$	无需特别处理
欧式距离	$d(x,y)=\sqrt{\sum_{i=1}^{n}(x_i-y_i)^2}$	1. 使用向量 w 表示每个 user 的权重 2. 重新定义带权重的相似度计算公式	
Jaccard 相似度	$d_j(A,B)=\frac{\|A\cup B\|-\|A\cap B\|}{\|A\cup B\|}$		
Pearson 相似度	$\text{Pearson}(x,y)=\frac{\sum(x,y)}{\sigma_x\times\sigma_y}$		

表 3.2 协同过滤算法修正结果

2. Song2vec

（1）模型介绍

Word2Vec 也叫 word embedding，中文名“词向量”，作用是将自然语言中的字词转为计算机可以理解的稠密向量。

在 Word2Vec 出现之前，自然语言处理经常把字词转为离散的单独的符号，也就是 One-Hot。One-Hot 表示法用来表示词向量非常简单，但是却有很多问题。最大的问题是向量维度的大小取决于语料库中字词的多少，当词汇表为百万级别时，每个词都用百万维的向量来表示，会造成维度灾难，且矩阵可能过于稀疏，表达销量不高。另一个问题是，由于 One-Hot 编码是完全随机的，向量之间相互独立，无法表达字词之间可能存在的关联关系。

为了解决上述问题，distributed representation 被构建出来了。Word2Vec 可以将 One-Hot 向量转化为低维度的连续值，也就是稠密向量，并且其中意思相近的词将被映射到向量空间中相近的位置，进而可以用普通的统计学的方法来研究词与词之间的关系。

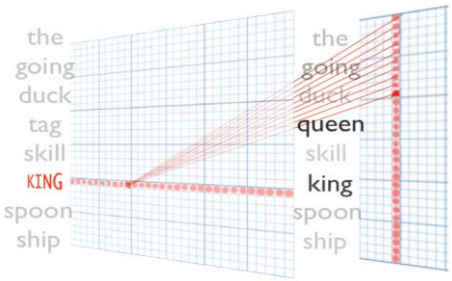

图 3.4 distributed representation

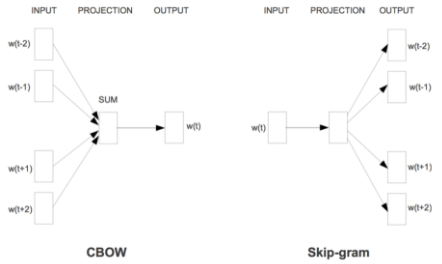

图 3.5 CBOW 和 skip-gram

Word2Vec 采用的是 n 元语法模型(n-gram model)，即假设一个词只与周围 n 个词有关，而与文本中的其他词无关。最常用的是 CBOW 模型和 skip-gram 模型：这两个模型非常相似。CBOW 模型能够根据输入周围 n-1 个词来预测出这个词本身，而 skip-gram 模型能够根据词本身来预测周围有哪些词。本模型中默认使用 skip-gram 模型。

（2）模型构建

基于假设“**同一个歌单中的歌曲，互相之间有关联性**”，我们将 Word2Vec 的思想运用在

歌曲建模上，试图构建每首歌曲的稠密向量表示，并将其映射到高维空间中。高维空间中向量的远近可以在一定程度上表达歌曲间的相似度。

具体构建过程使用 python 的 gensim 扩展库，可以看作是一个简化版的神经网络，将一首歌与同歌单中的歌作为输入和输出，把中间层得到的维数较低的隐向量作为歌曲的稠密表达。通过同歌单歌曲间的关联性，我们认为相似的歌将在高维空间距离接近，由此得到 song2vec 模型。例如，歌手林宥嘉的歌曲《成全》，总是与五月天的《后来的我们》、李荣浩的《戒烟》、赵紫骅的《可乐》等歌曲出现在同一个歌单中，那么在 song2vec 模型中，它们之间的向量距离也会很接近。通过该模型，我们就可以为每首歌找出最相似的其他歌曲，将《后来的我们》、《戒烟》等歌推荐给喜欢《成全》的用户。

在歌曲的冷启动场景中，由于新歌本身不在 song2vec 模型中，可以考虑推荐该歌手受欢迎的旧歌。但这样的推荐缺乏新颖性且较为死板，因此，我们考虑引入歌手的 singer2vec 模型。通过将歌曲上升为更粗粒度的歌手，我们构建了反映歌手间相似度的模型，每个歌手可以对应到高维空间中的某个向量。例如，前例中的林宥嘉，他总是与田馥甄、周兴哲、陈势安等歌手出现在相同歌单中，我们可以将林宥嘉的新歌推荐给喜欢田馥甄的用户。

四．应用

1.协同过滤和 Song2vec 对比

我们选择为喜爱歌曲《成全》的用户推荐十首歌曲，使用协同过滤和 Song2vec 方法推荐的歌曲结果如下表所示。可以看到两种方法推荐的十首歌曲中有七首是相同的，分别为《我们》、《可乐》、《你就不要想起我》、《戒烟》、《哑巴》、《你，好不好?》、《后来的我们》，推荐结果较为相似。

	协同过滤	Song2vec
推荐歌曲 1	说谎	后来的我们
推荐歌曲 2	我们	哑巴
推荐歌曲 3	可乐	戒烟
推荐歌曲 4	你就不要想起我	肆无忌惮
推荐歌曲 5	戒烟	可乐
推荐歌曲 6	哑巴	你，好不好?
推荐歌曲 7	你，好不好?	你就不要想起我
推荐歌曲 8	浪费	无问
推荐歌曲 9	后来的我们	我们
推荐歌曲 10	空空如也	爱了很久的朋友

表 4.1 歌曲《成全》推荐结果

2.真实用户调查

我们随机选取了 9 名同学作为测试者，使用 Song2vec 方法，根据他们喜爱的 5 首歌曲为其推荐包含 10 首歌曲的“猜你喜欢”歌单。表 4.2 展示了 3 个典型用户的推荐歌单结果。用户 A 偏好的曲风具有明显的二次元风格，推荐给他的歌单具有相同的风格；用户 B 喜爱欧美歌曲和日文歌曲，推荐歌单同样涵盖了日文和英文歌；用户 C 日常所听的歌曲大多为小清新的民谣，推荐结果基本为民谣风。9 名用户对推荐歌单都较为满意，我们的推荐系统基本满足了不同用户的需求，完成了个性化音乐推荐。

用户 A		用户 B		用户 C	
喜爱歌曲	推荐歌单	喜爱歌曲	推荐歌单	喜爱歌曲	推荐歌单
银临 裁梦为魂	玄觞 东风第一枝	米津玄師 Lemon	DAOKO 打上花火	Jam 七月上	赵雷 南方姑娘
银临 牵丝戏	玄觞 画诗	Birdy Shadow	majiko アイロニ	陈鸿宇 理想三旬	赵雷 成都
林俊杰 醉赤壁	泠鸢 相思赋	ラムジ PLANET	4 円 アイロニ	马頔 南山南	宋冬野 斑马，斑马
双笙 世末歌者	双笙 少女净妖师	Lia 夏影	高橋優 ヤキモチ	田馥甄 小幸运	宋冬野 安和桥
KBShinya 霜雪千年	银临 笔底知交	Eminem The Monster	Eminem Love The Way You Lie	陈绮贞 旅行的意义	陈鸿宇 理想三旬
	银临 湘桥月		Bruno Mars Lighters		谢春花 借我
	银临 春笺		まじ娘 心做し		Jam 七月上
	银临 狐言		秦基博 Rain		刘昊霖 儿时
	玄觞 不朽之罪		JAY'ED また君と		陈粒 奇妙能力歌
	流仙 谓风		茅野愛衣 secret base ~君 がくれたもの~		李志 天空之城

表 4.2 部分典型用户推荐歌单

3.冷启动

（1）新用户进入系统

对于新进入系统的用户，由于没有历史数据，无法根据其过去喜爱的歌曲进行个性化推荐。这时，我们选择呈现给他们一组歌曲风格差异较大同时比较热门的歌曲，具体做法为选择丰富度最高的 20 首歌曲，即使得包含这些歌曲的歌单的数量最大。我们的初始歌单包含《小半》、《Shape of You》等 20 首热门歌曲，包含这 20 首歌曲的歌单共 6213 个，占全部歌单总数的 22%。通过引导用户点击初始歌单中自己感兴趣的歌曲，我们可以立即显示这些歌曲的相似歌曲，从而完成推荐。比如用户选择听初始歌单中陈粒的《小半》，我们会进一步推荐歌曲《岁月神偷》、《房间》、《凉城》等。收集到的用户信息可以用于协同过滤推荐算法，这种推荐应该是比较低频的，比如网易云日推，因为对服务器的要求较高。

歌手	歌曲
陈粒	《小半》
Ed Sheeran	《Shape of You》
陈百强	《偏偏喜欢你》
Pianoboy 高至豪	《The truth that you leave》
Dragon Pig	《全部都是你》

鹿先森乐队	《春风十里》
Vicetone	《Nevada》
谢安琪	《钟无艳》
孙燕姿	《遇见》
Two Steps From Hell	《Victory》
Justin Timberlake	《Five Hundred Miles》
Maroon 5	《Sugar》
朴树	《平凡之路》
高胜美	《千年等一回》
MKJ	《Time》
陈奕迅	《好久不见》
OmenXIII	《Black Sheep》
茅野愛衣	《secret base ~君がくれたもの~ (10 years after Ver.)》
房东的猫	《云烟成雨》
The Chainsmokers	《Something Just Like This》

图 4.1 新用户进入系统推荐的初始歌单

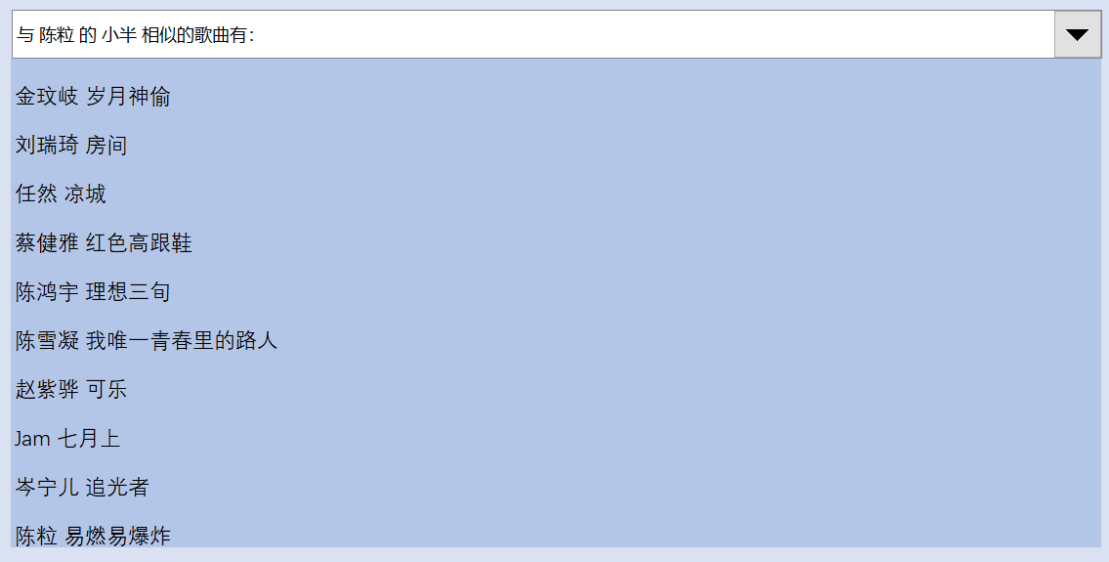

图 4.2 根据用户对初始歌单的选择进行后续推荐

（2）新歌曲进入系统

对于一首新歌曲进入系统，共有两种方法处理歌曲冷启动问题。其一是推荐该歌手的其他歌曲进行推荐，该方法的局限在于推荐歌单新颖性不足，缺乏对小众歌曲的推荐能力，用户歌单可能仅覆盖小部分歌手；其二是通过构建反映歌手间相似度的 singer2vec 模型推荐相似歌手的歌曲，该方法拓展了推荐歌曲的范围。比如，林宥嘉的新歌曲进入系统时，我们可以把新歌曲推荐给喜爱林宥嘉《全世界谁倾听你》、《残酷月光》、《成全》、《想自由》、《说谎》、《我爱的人》、《你是我的眼》、《心酸》、《浪费》、《心有林夕》、《早开的晚霞》等曲库已有歌曲的用户。同时我们也可以推荐与林宥嘉相似歌手的歌曲，包括：田馥甄、周兴哲、陈势安、郭顶、李代沫、赵紫骅、张禄粂、陈嘉桦、光泽、AllenRock 等。

方法	**基于统计**	singer2vec
推荐	全世界谁倾听你	田馥甄
	残酷月光	周兴哲
	成全	陈势安
	想自由	郭顶
	说谎	李代沫
	我爱的人	赵紫骅
	你是我的眼	张禄桑
	心酸	陈嘉桦
	浪费	光泽
	心有林夕	AllenRock

图 4.3 以林宥嘉新歌为例的歌曲冷启动