

**参赛队号 #2021**  
**第六届“认证杯”数学中国**

**数学建模网络挑战赛**  
**承 诺 书**

我们仔细阅读了第六届“认证杯”数学中国数学建模网络挑战赛的竞赛规则。

我们完全明白，在竞赛开始后参赛队员不能以任何方式（包括电话、电子邮件、网上咨询等）与队外的任何人（包括指导教师）研究、讨论与赛题有关的问题。

我们知道，抄袭别人的成果是违反竞赛规则的，如果引用别人的成果或其他公开的资料（包括网上查到的资料），必须按照规定的参考文献的表述方式在正文引用处和参考文献中明确列出。

我们郑重承诺，严格遵守竞赛规则，以保证竞赛的公正、公平性。如有违反竞赛规则的行为，我们将受到严肃处理。

我们允许数学中国网站([www.madio.net](http://www.madio.net))公布论文，以供网友之间学习交流，数学中国网站以非商业目的的论文交流不需要提前取得我们的同意。

**我们的参赛队号为：2021**

**参赛队员（签名）：**

队员 1：楚彭子

队员 2：李青霞

队员 3：安渊

**参赛队教练员（签名）：** 王辛岩

**参赛队伍组别：** 本科

**参赛队号 #2021**  
**第六届“认证杯”数学中国**  
**数学建模网络挑战赛**  
**编号专用页**

参赛队伍的参赛队号：（请各个参赛队提前填写好）：

2021

竞赛统一编号（由竞赛组委会送至评委团前编号）：

---

竞赛评阅编号（由竞赛评委团评阅前进行编号）：

参赛队号 #2021

# 2013 年第六届“认证杯”数学中国 数学建模网络挑战赛第二阶段

题 目 论流行音乐所处年代及其“成功要素”的变迁

关 键 词 流行音乐；时代风格；支持向量机；情感词

## 摘 要：

据网站音乐库，结合 matlab 软件提取样本音乐的相关数据。本文采用了特征向量提取、AP 聚类法、支持向量机、情感词分析等措施，对不同时代音乐风格进行了探究，成功地由歌曲特征推断出其所处年代，解决了音乐时代推断问题，并建立了用以划分音乐时代风格的一系列模型。

针对题中首要问题，依据音乐的音频特征结合物理学原理，认为音频特性可作为不同时代音乐风格划分的标准：

对于音乐风格这一模糊概念，文中首先对音频文件中音频数据的特征向量进行提取，根据 N 个数据点之间的相似度，提出运用 AP 聚类法对音频数据进行聚类，对音乐的风格进行了描述，建立了音频特征向量空间模型（文中表 1）。

由于该模型在建立上存在局限性，只能对音乐风格进行粗略的分类处理，进而在音频特征向量空间模型、支持向量机、情感词分析的基础上引入了歌词情感分析模型，通过样本歌曲的检验、音频特征向量空间模型与歌词情感分析模型的结合，验证了该整体模型的准确性。

最后，对流行歌曲的“成功要素”的变迁进行探讨，并将中国近半世纪以来流行歌曲的“成功要素”的变迁划分为 3 个阶段，认为一种音乐风格的成功，在其艺术特点和情感表达上一定反映了该时代的时代特征，迎合了该时代人们的精神追求。

参赛队号： 2021

所选题目： B 题

参赛密码 \_\_\_\_\_  
(由组委会填写)

## 参赛队号 #2021

### Abstract

To explore different times music style, this paper uses a feature vector extraction, AP clustering method, support vector machine, emotional words analysis and so on, Successfully by the song features inferred from his age, to solve the inference problem music era, and has established a series of models in order to partition the music style.

For the first question, on the basis of audio characteristics of the music combined with the principles of physics, that can be used as audio characteristics of different times music style classification standards:

For the music style of the fuzzy concept, firstly the feature vector of the audio file data were extracted, according to the similarity between N data points, proposed to cluster the audio data using AP clustering method, the style of music is described, established the audio feature vector space model (text table 1).

Since the model limitations in the establishment, can only classify rough on the style of music, and then in the audio feature vector space model, support vector machine, based on the introduction of emotional words lyrics emotion analysis model, through the test sample songs, audio feature vector space model and the lyrics emotion analysis model, verify the accuracy of the overall model.

Finally, the popular song "success factors" changes are discussed, and the Chinese for over half a century of popular song "success factors" changes divided into 3 stages, as a style of music which is successful, in the artistic characteristics and emotional expression must reflect the features of the times the times, to meet the age people's spiritual pursuit.

**Key words:** Pop music; style; svm; emotional words

## 参赛队号 #2021

### 一、问题重述

随着互联网的发展，流行音乐的主要传播媒介从传统的电台和唱片逐渐过渡到网络下载和网络电台等。网络电台需要根据收听者的已知喜好，自动推荐并播放其它音乐。由于每个人喜好的音乐可能横跨若干种风格，区别甚大，需要分别对待。这就需要探讨如何区分音乐风格的问题。

近几十年来，流行音乐的风格经历了相当复杂的演变过程。每个时代都有其代表性的人物和风格，歌曲的成功要素也随时间不断改变。请你建立合理的数学模型，对曾经走红过的流行歌曲，通过其原唱录音或曲谱来推断其所处时代。并请你写一篇短文<sup>2</sup>，向大众简述（在某个时期内）流行歌曲的“成功要素”是如何变化的。

根据以上描述，对如下问题进行探讨并解决：

- (1) 建立合理的模型分析不同时代的音乐风格；
- (2) 分析歌曲的成功要素；
- (3) 简述（在某个时期内）流行歌曲的“成功要素”是如何变化的。

### 二、问题分析

目前音乐风格的划分还没有一个定性的标准，而且某一个时代的音乐风格的特征也没有一个准确的定义。通过第一阶段问题的解析了解到了以下几点：

- 1、影响音乐风格划分的决策因子为：音频。
- 2、不同时代的音乐的区别主要在于音乐的主旋律和音乐所反映的内容各不相同。

根据所得信息，对该阶段问题做以下分析：

**不同时代的音乐风格方面：**以音频作为区分不同风格音乐的主要因素，并以音频的帧能量为音乐的特征向量，利用 AP 算法进行聚类分析。从图 1 中可以看到不同时代音乐的风格具有良好的聚类簇，但是显示在类群音频特征向量数据上却有很大的交集。所以，单纯的以音频为不同时代音乐划分的标准是不科学的，对此，为提高不同时代音乐的判断准确率，进一步建立情感词分析模型。

歌词是区别不同风格音乐的另一个因素，而最能体现一首歌曲情感基调的是歌词中的情感词。以情感词在一篇歌词中所出现的频率和相对于整片歌词所占的权重为描述情感词的特征向量，对不同时代音乐的风格和感情基调再一次进行划分。

**歌曲的成功要素方面：**通过对不同时代流行歌曲的情感基调进行分类处理，然后结合我国历史文化背景，我们将对歌曲的成功要素进行详细的讨论。

**解析“成功要素”变化方面：**对“成功要素”的分析，结合以上问题中的结论，针对流行歌曲“成功要素”的变迁过程，写出符合题意的短文。

### 三、模型假设

1. 每首歌曲都有自身独特的音频特征；
2. 文中所采用的数据是可靠的；
3. 一定量的样品音乐的特征可以代表这一类音乐的总体特征；
4. 某一种风格音乐的成功不存在偶然性；
5. 能跨越时代的流行音乐是极少的。

## 参赛队号 #2021

## 四、符号说明

符号	符号含义
$E_n$	短时平均能量
$E_i$	帧能量
$ER_i$	帧能量比
$N$	窗口长度
$flen$	每一帧的长度
$G_E$	音频中相对激烈的帧能量
$d_2(x, y)$	欧氏距离
$\alpha_{js}$	词 $j$ 在文本 $s$ 中的权重
$f_{js}$	词 $j$ 在文本 $s$ 中的频率
$P(i)$	第 $i$ 篇歌词，情感词出现的频率
$T$	训练结合
$n_j$	训练文本中包含词 $j$ 的文本数

## 五、模型建立与求解

## 5.1 音频特征向量的提取

由于唯一可以跟踪描述音乐的特征元素为：音频，以音频特征为不同时代的音乐风格划分依据，进而建立数学模型。

## 5.1.1 音频分析

要对不同风格的音乐进行分类，首先对音乐的音频进行分析，假设一段音频信号流  $x$  的离散信号为：

$$x = (x_1 \dots x_n \dots x_k)$$

则在一段连续的音频信号中得到了  $k$  个样本数据，将这  $k$  个样本数据分为  $L$  组，每一组包含  $[k/L]$  个采样点。而这  $[k/L]$  个采样点的特征构成了音频数据流  $x$  的特征，并且这些特征点反映在短时间内的平均能量不同，所以反映在音乐的不同时间段就出现了平缓、高潮等不同的部分。其能量表达式为：

$$E_n = \frac{1}{N} \sum_m [\pi(n) \omega(n-m)]$$

## 参赛队号 #2021

### 5.1.2 音频特征提取

通过音频分析,可以看出音频信号的不同点主要反映在音频信号的能量,因此采用时域分析法对帧能量进行分析。帧能量表达式为:

$$E_i = \sum_{n=1}^{i+flen-1} X_n^2 \quad (\text{其中 } E_i \neq 0)$$

决定某一首音乐风格的主要元素为:这首歌曲高潮部分的时间长短、强度、次数等。对音乐进行特征提取可以看成为一个筛选过程:对不同帧进行分类,将高潮部分或相对激烈的帧作为特征向量。

将所要分类的音乐样本划分为一系列的帧,并对每一个帧的能量进行求解,帧能量求解公式为:

$$G_E = E_{\min} + \alpha \times (E_{\phi} - E_{\min}) \quad (0 \leq \alpha \leq 1)$$

通过计算得到静音阈值,依据阈值对帧能量过滤噪声,得到剩下的声音片段是音乐原声。然后通过帧的能量的计算公式得到帧能量比,帧能量比较大的帧即可判断为特征片段的端点,而端点之间的帧可作为特征片段。帧能量比的计算公式为:

$$ER_i = \max\left(\frac{E_i}{E_{i+1}}, \frac{E_{i+1}}{E_i}\right) \quad (\text{其中 } E_i, E_{i+1} \neq 0)$$

### 5.2 AP 聚类算法

#### 5.2.1 AP 聚类算法的引入

要对音频特征向量进行分类,若采用人为限定类的个数会使得聚类结果在本题中可信度减低,故采用了具有自适应仿射传播聚类特性的 Affinity Propagation (AP) 聚类算法。

该算法根据  $N$  个数据点之间的相似度进行聚类,这些相似度可以是对称的,即两个数据点互相之间的相似度一样(如欧氏距离);也可以是不对称的,即两个数据点互相之间的相似度不等。这些相似度组成  $N \times N$  的相似度矩阵  $S$  (其中  $N$  为有  $N$  个数据点)。AP 算法不需要事先指定聚类数目,相反它将所有的数据点都作为潜在的聚类中心。

针对两个数据点之间的相似度  $S(i, j)$  的问题,引入欧式距离对其进行描述,其函数描述如下式所示:

$$d_2(x, y) = \left[ \sum_{k=1}^p |x_k - y_k|^2 \right]^{\frac{1}{2}}$$

由上式知:相似度值越大,说明点与点之间的距离越近。我们以  $Q$  矩阵的对角线上的数值  $q(k, k)$  作为  $k$  点能否成为聚类中心的判断标准,如果  $s(k, k)$  越大,那么,这个点成为聚类中心的可能性越大。这个点也可以称为参考度  $p$ , 根据参考度  $p$  值的不同,得到不同的聚类数量。

AP 算法中传递两种类型的消息, responsibility 和 availability。  $r(i, k)$  表示从点  $i$  发送到候选聚类中心  $k$  的数值消息,反映  $k$  点是否适合作为  $i$  点的聚类中心。 $a(i, k)$  则从候选聚类中心  $k$  发送到  $i$  的数值消息,反映  $i$  点是否选择  $k$  作为其聚类中心。 $r(i, k)$  与  $a(i, k)$  越强,则  $k$  点作为聚类中心的可能性就越大,并且  $i$  点隶属于以  $k$  点为聚类中心的聚类的可能性也越大。AP 算法通过迭代过程不断更新每一个点的吸引度和归属度值,直到产生  $m$  个高质量的 exemplar,同时将其余的数据点分配到相应的聚类中。

## 参赛队号 #2021

### 5.2.2 AP 聚类算法常用定义

**exemplar:** 聚类中心, **similarity:** 数据点  $i$  和点  $j$  的相似度记为  $S(i, j)$ 。是指点  $j$  作为点  $i$  的聚类中心的相似度。

**Preference:** 数据点  $i$  的参考度称为  $P(i)$  或  $S(i, i)$ 。是指点  $i$  作为聚类中心的参考度。一般取  $S$  相似度值的中值。

**Responsibility:**  $R(i, k)$  用来描述点  $k$  适合作为数据点  $i$  的聚类中心的程度。

**Availability:**  $A(i, k)$  用来描述点  $i$  选择点  $k$  作为其聚类中心的适合程度。

### 5.2.3 AP 算法工作过程

先计算数据中  $N$  个点之间的相似度值, 将值放在  $S$  矩阵中, 再选取  $p$  值(一般取  $S$  的中值)。设置一个最大迭代次数(文中设默认值为 1000), 迭代过程开始后, 计算每一次的  $R$  值和  $A$  值, 根据  $P(k, k) + A(k, k)$  值来判断是否为聚类中心(文中指定当  $P(k, k) + A(k, k) \geq 0$  时认为是一个聚类中心), 当迭代次数超过最大值(即  $\max$  值)或者当聚类中心连续多少次迭代不发生改变(即  $\text{convits}$  值)时终止计算。

通过特征向量的提取, 对采取的样本的特征向量数据运用 AP 算法进行聚类, 得到以下聚类效果图如下所示:

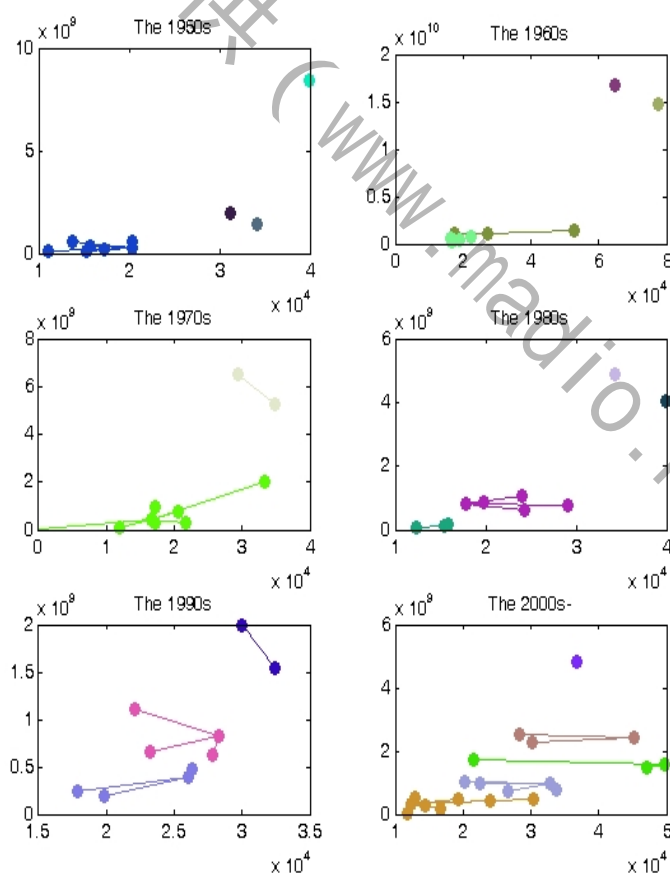


图 1 各时代样本音乐聚类图

Fig 1 Clustering figure music of each time sample



## 参赛队号 #2021

由上图（matlab 程序见附录 1，样本数据见附录 2）可得以下结论：

1. 不同时代流行的歌曲在其聚类图上显示出良好的聚类结果。  
2. 自 50 年代以来，在每个时代盛行的音乐风格在逐渐增多。且自进入 21 世纪以来，流行的音乐风格的种类倍增。

3. 虽然每个时代有大批的歌手和音乐，但是在该时代流行的音乐在其风格上具有相似的特点。因此以同一时代音乐风格所占比重最高的音乐风格来代表该时代的主流风格。将得到各时代的主流音乐风格的类群音频特征向量所处空间列表得表 1：

表 1 各时代音乐风格类群音频特征向量空间表

Tab 1 Audio feature vector space of each times music style group

时代	代表风格	音频特征向量数据
50 年代	风格 1	$11035 \leq x \leq 39936, 1.02\text{E}+08 \leq y \leq 8.43\text{E}+09$
60 年代	风格 2	$77414 \leq x \leq 16268, 1.80\text{E}+08 \leq y \leq 1.68\text{E}+10$
70 年代	风格 3	$11904 \leq x \leq 34790, 9.17\text{E}+07 \leq y \leq 5.28\text{E}+09$
80 年代	风格 4	$12345 \leq x \leq 39979, 9.23\text{E}+07 \leq y \leq 4.02\text{E}+09$
90 年代	风格 5	$17881 \leq x \leq 23259, 1.92\text{E}+08 \leq y \leq 4.82\text{E}+08$
	风格 6	$26038 \leq x \leq 32390, 6.24\text{E}+08 \leq y \leq 2.00\text{E}+09$
21 世纪	风格 7	$11736 \leq x \leq 16576, 1.51\text{E}+08 \leq y \leq 4.03\text{E}+08$
	风格 8	$19257 \leq x \leq 23932, 4.70\text{E}+08 \leq y \leq 7.52\text{E}+08$
	风格 9	$26703 \leq x \leq 33700, 9.72\text{E}+08 \leq y \leq 2.29\text{E}+09$
	风格 10	$45261 \leq x \leq 49792, 2.41\text{E}+09 \leq y \leq 4.80\text{E}+09$

从上表数据可得以下结论：

1. 每一个时代至少有一种具有代表性的音乐风格。  
2. 可以根据不同音乐风格的类群音频特征向量数据对要进行训练的音乐进行粗略的分类处理。  
3. 表 1. 中的音频特征向量数据在一定的区间中有交集，所以单纯依靠音频特征向量数据进行分类不能得到理想的结果。

因此，我们将在音频聚类模型的基础上建立歌词情感分析模型对不同时代音乐的风格进行进一步的探讨研究。

## 5.2 歌词情感分析

音乐风格在其表现形式上主要有音乐的旋律和歌曲所对应的歌词，而前者是感性的情感表达，后者则是理性的具体描述。而且在不同历史文化背景下所创作的音乐，在其歌词的内容上所反映的印象也各不相同，所以我们可以根据歌词所表达的含义、反映的社会现状来判断音乐所创作的时间和其流行的时代。

## 参赛队号 #2021

### 5.2.1 音乐歌词中的情感词研究

音乐是一种源于生活的真实体现，不同时期的音乐主要表现了人民大众在不同历史文化环境下的生活状况、以及心声等，因此在不同的历史环境下音乐所体现的情感也是不相同的，最直观体现歌曲情感的是该首歌的歌词。

在歌词中情感歌词最能反映歌曲的感情基调和主题，对音乐的主流感情基调进行分析，并用一些感情形容词对感情基调进行表征。音乐的主流感情基调和主题如表 2 所示：

表 2 音乐的主流感情基调或主题

Tab2 The mainstream emotional tone or theme of music

主流感情基调 或主题	描述感情基调或主题的关键词
神圣、向往	挚爱、承诺、永恒、坚强、向往、神圣、神奇等形容词
热情、高亢	热情、奔放、真诚、勇气、疯狂、奋斗、激烈、高涨、努力等词
赞扬、敬仰	尊敬、拥护、敬仰、赞扬、歌唱、敬佩、推崇等形容词
快乐、美好	快乐、高兴、美好、喜悦、痛快、得意、欣慰、舒畅、得志、振奋等形容词
希望、憧憬	希望、憧憬、想象、星空、期待、回念等形容词
爱情的痛与苦	温馨、快乐、开心、轻松、享受、爱情、痛苦、悲伤、忧愁、伤感、寂寞、想哭等形容词

从上表可得以下结论：

1. 音乐的感情基调和主题在很大程度上表现出不相同。
2. 关键词在一定程度上反映着音乐的感情基调和主题。

由此，可以根据音乐的感情基调和主题来判定音乐所处的时代，引入支持向量机（SVM）对不同时代音乐的风格进行探究。

### 5.2.2 歌词特征向量的提取

在歌词中往往能反映该首歌曲相应特性的信息为该歌词中的情感词，以情感词在该篇歌词中出现的频率和在整篇歌词中所占的权重进行描述、推断该歌曲所流行的年代和其他等信息。

#### 5.2.2.1 情感词出现的频率

设  $A$  为包含  $a$  篇歌词的文档集合， $A_i$  为第  $i$  篇歌词中的所有词语的集合，那么则有：

$$A = \{A_1, A_2, \dots, A_a\} \text{ 且 } A_i = \{A_{i1}, A_{i2}, \dots, A_{in}\} \quad \text{其中 } i = 1, 2, \dots, a$$

当  $A_i$  中包含有  $b$  个  $c$  类的情感词且这  $c$  类情感词满足离散分布，则有：

$$A_{ib} = \{b_1, b_2, \dots, b_\beta\} \quad \text{其中 } b = \beta \times c$$

那么第  $i$  篇歌词中情感词出现的频率为：

$$P(i) = \frac{b_i}{A_i} \quad \text{其中 } 0 < P(i) \leq 1$$

#### 5.2.2.2 不同情感词在整篇文档中所占的权重

考虑到情感词在整篇歌词中出现的频率以及在训练集中出现的频率，本文采取 TF\*IDF 权重<sup>[8]</sup>来描述歌词中情感词的比重。在该方法中，如果一个情感词在训练集中

## 参赛队号 #2021

出现的频率越小, 则对于分类处理来说, 它越重要。

在 TF\*IDF 权重算法中, 情感词的权重与该词在文本中的频率成正比, 与其在整个训练集中的权重成反比, 其公式表达为:

$$\alpha_{js} = f_{js} \times \log\left(\frac{N}{n_j}\right)$$

其中  $\alpha_{js}$  是词  $j$  在文本  $s$  中的权重,  $f_{js}$  是词  $j$  在文本  $s$  中的频率,  $N$  是训练集中总的文本数,  $n_j$  是训练文本中包含词  $j$  的文本数。

### 5.2.3 支持向量机探讨歌词并对音乐进行分类

#### 5.2.3.1 训练集的建立

根据训练数据建立训练集合, 其公示表达如下:

$$T = \{(x_1, y_1), (x_2, y_2), \dots, (x_t, y_t)\} \in (X \times Y)^t$$

因此我们在训练集合范围内寻找一个实值函数  $g(x)$ , 并利用分类函数:  $f(x) = \text{sgn}(g(x))$  对任意  $X$  和  $Y$  进行分类讨论。

#### 5.2.3.2 支持向量机进行求解

归一化超平面方程, 使得所有样本集满足如下约束条件:

$$y_i[W^T X_i - b] \geq 1, i = 1, 2, \dots, l$$

两类模式间隔的距离为:  $\frac{2}{\|W\|}$  即需要最大化  $\phi(W) = \frac{1}{2}(W^T W)$

根据约束条件建立目标函数, 其表达式为:

$$\Phi(w, \xi) = \frac{1}{2}\|w\|^2 + c \sum_{i=1}^n \xi_i$$

根据以上叙述, 其次构造拉格朗日函数, 其表达式为:

$$L(W, b, \alpha) = \frac{1}{2}(W^T W) - \sum_{i=1}^l \alpha_i \{[W^T X_i - b]y_i - 1\}$$

最后构造最优判别函数, 其表达式为:

$$f(x) = \text{sgn}([w^*, x] + b^*) = \text{sgn}\left(\sum_{X \in SV} \alpha_i^* y_i [x \bullet x_i] + b^*\right)$$

以情感词的词频及情感词在文档中所占的比重为特征向量并根据支持向量机 (SVM) 对近 50 年各时代的代表性音乐歌词进行训练, 将所得结果与 AP 聚类结果数据相结合, 既得较优的分类结果。

### 5.3 成功要素的分析

#### 5.3.1 近 50 年中国历史文化背景和人民价值观变化研究

对近 50 年中国所发生的一些较大的历史事件 (具体见附录二: 表 4) 进行分析得到人们的价值观和思想观念也在随这时间的变迁而不断产生变化, 并且歌曲的主题也在随人们价值观的变化而产生变化, 其具体变化如下表所示:

## 参赛队号 #2021

表 3 不同时代人们的价值观和歌曲主题  
Tab3 People's values and theme song in different times

时代	人们价值观	成功歌曲的主题
40、50 年代	以社会主义建设和奋斗为目标，饱含高亢的建设热情	歌唱人民劳作学习，歌颂解放军，庆祝国家解放。
60 年代	以毛主席为榜样	致力于歌颂毛主席
70 年代	赞美毛主席、向往北京	赞美毛主席，向往北京。
80 年代	对未来的憧憬和展望，	生活所发生的巨大的变化
90 年代	爱情开始	爱情自由化
21 世纪-	思想和价值观开始多元化发展	以爱情为主题并多方向进行辐射发展

据表 3 及前文总结，可认为在不同时代音乐成功的要素主要有：

- 1、歌曲在其主旋律或歌词上表现有很强的情感，能和人们的内心产生共鸣。
- 2、歌曲的主旋律或歌词能反映时代的特征，并且能够在其艺术特色方面迎合该时代大多数人的认知情感，诉说了大众的心声。
- 3、歌曲中有将强的艺术特色和文化特色。

### 5.4 短文

据以上分析，结合自身看法写成短文一篇：《流行歌曲的“成功要素”的变迁》，见附录 1.1。

## 六、模型检验

通过歌曲特征向量所在空间和歌词所体现的人生价值观来判定歌曲所处年代，这里用小虎队在 1992 年发行的歌曲《爱》作为检测样品。

首先，用 matlab 提取其特征向量为：

距离均值	距离方差
27140.3636	1.31E+09

通过与表格 1 对比可知该歌曲风格位于风格 1、风格 3、风格 4、风格 6、风格 9 中，由此可见该首歌曲的个性并不强烈。

其次，观察该首歌的歌词(附录 2)，可以发现该首歌最明显的特征词为“爱”，而据表 3 显示，以爱情为歌曲主题的时代为 90 年代和 21 世纪以来，这与表 1 基本吻合，而在这首歌的歌词中有一句：“让我们自由自在的恋爱”，这与表 3 中成功歌曲主体项的“爱情自由化”相吻合，由此，我们可以断定该首歌曲为 90 年代诞生的流行歌曲，从而验证了模型的正确性。

由以上分析可进一步了解到，单一的 AP 聚类模型不能满足对歌曲所处时代进行准确判别，而歌词与聚类算法结合的模型可以很好地对歌曲所处年代进行判别，此时，歌词情感词的确定起着重要作用。

## 参赛队号 #2021

### 七、模型评价与改进

#### 7.1 模型优点

1. 两种模型的建立，降低了不同时代所流行音乐风格划分中的盲点，增加了结果的准确度。

2. 对歌词进行了情感词的分析，并结合近 50 年中国的历史文化背景，进行对不同时代所流行的音乐的风格进行划分。

3. 运用支持向量机（SVM）对歌词中的情感词进行分类讨论，两种算法的结合相对于其他算法可得到更优解。

#### 7.2 模型缺点

1. 文中所选取数据量较少，因此对模型的求解和结果有一定的影响，并且小数据建立的模型可信度较低。

2. 对音乐的主流感情基调分析较少，可能对模型的结果有所影响。

#### 7.3 模型改进

##### 7.3.1. 音频特征向量提取的改进

在计算短时间区域内音频信号流的能量时需要对其进行标准化处理，因为处理的歌曲来源不同，不同歌曲之间的平均能量有较大的差异，用片段平均短时能量作为特征向量会对模型所求的结果有一定的误差。由此，对音频片段的平均短时能量做出标准化处理：

$$\bar{E}_{ave} = \frac{E_{al}}{E_{sg}} \times E_{ave}$$

其中， $E_{al}$  为所有训练目标的平均短时能量， $E_{sg}$  为一首歌曲的平均能量， $\bar{E}_{ave}$  为

标准化后的样本音频片段的平均短时能量值，进而以标准化后的  $\bar{E}_{ave}$  作为音乐的特征向量寻找流行歌曲中的关键音频段并进行分类处理。

##### 7.3.2 TF\*IDF 权重的改进

由于 TF\*IDF 权重没有考虑情感词长度对其权重的影响，对 TF\*IDF 权重进行算法改进，实现 TF\*IDF 权重对情感词长度进行归一化处理：

$$\alpha_{js} = \frac{f_{js} \times \log(\frac{N}{n_j})}{\sqrt{\sum_i^M [f_{is} \times \log(\frac{N}{n_i})]^2}}$$

其中  $\alpha_{js}$  是情感词  $j$  在歌词  $s$  中权重， $f_{js}$  是情感词  $j$  在歌词  $s$  中频率， $N$  是训练集合中总的文本数， $n_j$  是训练集合中包含情感词  $j$  的文本数， $M$  是训练集合中总词数。

通过音频特征向量提取方法的改进，以精确关键音频的选取，通过 TF\*IDF 权重的改进以满足权重确定的需求，改进后的模型所得数据将更为准确。

**参赛队号 #2021****八、参考文献**

- [1] 徐艺萍, 邓辉文, 李阳旭. 一种新的最近聚类算法[J]. 西南师范大学学报(自然科学版) 2006, 31(6):114-116.
- [2] 王开军, 张军英等. 自适应放射传播聚类[J]. 自动化学报, 2007, 12, 3(12).
- [3] 钟将, 程一峰. 音乐和歌词融合的歌曲情感分类研究[J]. 计算机工程, 2012, 4.
- [4] 李静, 林鸿飞, 李瑞敏. 基于情感向量空间的歌曲标签预测模型[J]. 中文信息学报, 2012, 12.
- [5] 徐利民, 舒君, 谢优忠. 基于MATLAB的信号与系统实验教程. 北京: 清华大学出版社, 2010, 2.
- [6] 司守奎, 孙玺菁. 数学建模算法与应用[M]. 北京: 国防工业出版社, 2012. 5. 重印
- [7] 李学文, 李炳照, 王洪洲. 数学建模论文精选与点评[M]: 2005—2010. 北京: 清华大学出版社, 2011. 9.
- [8] 孙向琨, 邓伟. 结合TF\_IDF的歌曲情感多标记分类[J]. 计算机工程, 2011, 10, 37: 189-191.

## 参赛队号 #2021

## 九、附录

## 附录 1 短文与程序

## 附录 1.1 短文欣赏

## 流行歌曲的“成功要素”的变迁

随着时代的发展和进步，人们的价值观、人生观和世界观也在随时间的变化而变化。在本文中我们结合中国近 50 年的历史文化背景，对流行歌曲的“成功要素”的变迁过程做简单的叙述。

自建国初期以来，人民大众的精神风貌发生了翻天覆地的变化，并且音乐艺术形态也由以前的单一形态逐渐转向多元化发展。虽然音乐的风格在随时代的变迁而不断地发生变化，但是不论哪一个时代，一首成功的歌曲或一种流行于当时的音乐风格一定蕴含着某种强烈的情感并在其艺术特色等方面迎合了该时代大多数人的情感认知，诉说了大众的心声。

从我国历史文化发展趋势和特征来看，我国流行歌曲“成功要素”的变迁主要经过了 3 个阶段：

第一个阶段（1949—1976）：该时代音乐不论在其风格上还是在其艺术特色上都具有明显的时代特征。对该阶段的音乐我们进行典型数据的研究得到：该阶段的歌曲主要表现了对毛主席的歌颂之情和赞扬了广大人民群众高亢的建设热情。

所以，在该阶段比较成功歌曲的艺术特色为：高亢而奋进的旋律。歌曲大多歌颂的内容主要有：对毛主席的敬仰、对北京的向往和对劳动者的歌颂。

第二个阶段（1978—1999）：经过 10 年文革后的广大人民沐浴到改革开放的春风时，重新燃起了新生活的希望。因此该阶段比较成功的歌曲在其歌词内容上主要反映了改革开放所带来的经济文化生活的巨变以及广大人民对美好明天的憧憬。

第三个阶段（进入 21 世纪以来）：进入 21 世纪以来，我国文化呈多元化、国际化发展，人们的价值观和文化观也出现了多元化发展。因此在此阶段的音乐事业的发展可以说是“百家争鸣、百花齐放”，从这个阶段脱颖而出的音乐在其风格上逐渐趋向于娱乐化、国际化和民族化。

综上所述，虽然在不同的时期，音乐事业的发展状况各不相同，人们对不同风格的音乐偏好有所不同。但是一种音乐风格的成功，在其艺术特点和情感表达上一定反映了该时代的时代特征，迎合了该时代人们的精神追求。

## 附录 1.2 matlab 程序

Matlab 提取歌曲音频主程序

```
y=sigspec(action);  
% SIGNAL SPECTRUM ANALYSER      THEE 2001/10  
% y=sigspec(action);  
  
% Usage:y=sigspec;  
%=====
```

```
clear;  
if nargin<1;action='initialized';  
end;  
[fname,pname]=uigetfile('*.wav','Open Wave File');  
file=[pname,fname];
```

## 参赛队号 #2021

```
[x,fs,bits]=wavread('finename.wav');          % 读入声音文件 (*.wav)
sound(x,fs,bits);                             % 数据通过声卡转换为声音
%=====选择声道=====
%y=x(:);                                     % 双声道信号波形数据
y=x(:,1);                                     % 左声道信号波形数据
%y=x(:,2);                                   % 右声道信号波形数据
%=====
disp(' 按任意键显示左声道声音波形!! ');
disp(' -----');
pause
plot(y);
set(gca,'FontName','Arial','FontSize',10);
title(['左声道声音波形']);
xlabel(['采样点数']);
disp(' 按任意键听左声道信号三次!! ');
disp(' -----');
pause
sound(y,fs,bits);                             % 数据通过声卡转换为声音
pause
sound(y,fs,bits);
pause
sound(y,fs,bits);
[k]=fft(y,32768);
disp(' 按任意键显示左声道声音频域的幅值!! ');
disp(' -----');
disp(' -----');
pause
plot(abs(k));
set(gca,'FontName','Arial','FontSize',10);
title(['左声道声音频域的幅值']);
xlabel(['采样点数']);
[m1,i1]=max(abs(k));                          % 找出频域最大值
F1=i1/32768*fs;
F1t=[num2str(F1) ' Hz'];
text(1500,450,F1t);
[m2,i2]=max(abs(k(4000:5000)));               % 找出频域次大值
F2=(4000+i2)/32768*fs;
F2t=[num2str(F2) ' Hz'];
text(3000,40,F2t);
[m3,i3]=max(abs(k(10000:12000)));             % 找出频域次次大值
F3=(10000+i3)/32768*fs;
F3t=[num2str(F3) ' Hz'];
text(8000,20,F3t);
%=====
```



## 参赛队号 #2021

```
%=====
t=(0:0.0001:1);
y1=(sin(2*pi*788*t)+sin(2*pi*3174*t)*(22.65/490)+sin(2*pi*6936*t)*(2.996/490))*0.18;
disp(' 按任意键显示左声道简单合成声音波形!! ');
disp(' -----');
pause
plot(t,y1);
set(gca,'FontName','Arial','FontSize',10);
title([' 左声道简单合成声音波形']);
disp(' 按任意键听简单合成左声道信号三次!! ');
disp(' -----');
pause
sound(y1,fs,bits);           % 数据通过声卡转换为声音
pause
sound(y1,fs,bits);
pause
sound(y1,fs,bits);
[k1]=fft(y1,32768);
disp(' 按任意键显示左声道简单合成声音的频域幅值!! ');
disp(' -----');
disp(' -----');
pause
plot(abs(k1));
set(gca,'FontName','Arial','FontSize',10);
title([' 左声道简单合成声音的频域幅值']);
xlabel([' 采样点数']);
[m1_r,i1_r]=max(abs(k1));      % 找出简单合成声音频域最大值
F1_r=i1_r/32768*10000;
F1_rt=[num2str(F1_r) ' Hz'];
text(3000,850,F1_rt);
[m2_r,i2_r]=max(abs(k1(10000:12000)));
F2_r=(10000+i2_r)/32768*10000; % 找出简单合成声音频域次大值
F2_rt=[num2str(F2_r) ' Hz'];
text(9000,60,F2_rt);

%=====
```

### AP 算法的主程序

```
function [idx,netsim,dpsim,expref]=apcluster(s,p,varargin)
if nargin==0, % display demo
    fprintf(' Affinity Propagation (APCLUSTER) sample/demo code\n\n');
    fprintf(' N=100; x=rand(N,2); % Create N, 2-D data points\n');
    fprintf(' M=N*N-N; s=zeros(M,3); % Make ALL N^2-N similarities\n');
```

## 参赛队号 #2021

```

fprintf(' j=1;\n');
fprintf(' for i=1:N\n');
fprintf('   for k=[1:i-1,i+1:N]\n');
fprintf('     s(j,1)=i; s(j,2)=k; s(j,3)=-sum((x(i,:)-x(k,:)).^2);\n');
fprintf('     j=j+1;\n');
fprintf('   end;\n');
fprintf(' end;\n');
fprintf(' p=median(s(:,3)); % Set preference to median similarity\n');
fprintf(' [idx, netsim, dpsim, expref]=apcluster(s, p, ''plot'');\n');
fprintf(' fprintf('' Number           of           clusters:
%%d\\n'', length(unique(idx)));\\n');
fprintf(' fprintf('' Fitness (net similarity): %%g\\n'', netsim);\n');
fprintf(' figure; % Make a figures showing the data and the clusters\n');
fprintf(' for i=unique(idx)''\n');
fprintf('   ii=find(idx==i); h=plot(x(ii,1),x(ii,2),''o''); hold on;\n');
fprintf('                                     col=rand(1,3);
set(h, ''Color'', col, ''MarkerFaceColor'', col);\n');
fprintf('   xil=x(i,1)*ones(size(ii)); xi2=x(i,2)*ones(size(ii)); \n');
fprintf('   line([x(ii,1),xil]'', [x(ii,2),xi2]'', ''Color'', col);\n');
fprintf(' end;\n');
fprintf(' axis equal tight;\n');
return;
end;
start = clock;
% Handle arguments to function
if nargin<2 error('Too few input arguments');
else
    maxits=1000; convits=100; lam=0.9; plt=0; details=0; nonoise=0;
    i=1;
    while i<=length(varargin)
        if strcmp(varargin{i}, 'plot')
            plt=1; i=i+1;
        elseif strcmp(varargin{i}, 'details')
            details=1; i=i+1;
        elseif strcmp(varargin{i}, 'sparse')
            % [idx, netsim, dpsim, expref]=apcluster_sparse(s, p, varargin{:});
            fprintf('''sparse'' argument no longer supported; see website for
additional software\n\n');
            return;
        elseif strcmp(varargin{i}, 'nonoise')
            nonoise=1; i=i+1;
        elseif strcmp(varargin{i}, 'maxits')
            maxits=varargin{i+1};
            i=i+2;

```

**参赛队号 #2021**

```

        if maxits<=0 error('maxits must be a positive integer'); end;
    elseif strcmp(varargin{i},'convits')
        convits=varargin{i+1};
        i=i+2;
        if convits<=0 error('convits must be a positive integer'); end;
    elseif strcmp(varargin{i},'dampfact')
        lam=varargin{i+1};
        i=i+2;
        if (lam<0.5) || (lam>=1)
            error('dampfact must be >= 0.5 and < 1');
        end;
    else i=i+1;
    end;
end;
end;
if lam>0.9
    fprintf('\n*** Warning: Large damping factor in use. Turn on plotting\n');
    fprintf('        to monitor the net similarity. The algorithm will\n');
    fprintf('        change decisions slowly, so consider using a larger value\n');
    fprintf('        of convits.\n\n');
end;

% Check that standard arguments are consistent in size
if length(size(s))~=2 error('s should be a 2D matrix');
elseif length(size(p))>2 error('p should be a vector or a scalar');
elseif size(s,2)==3
    tmp=max(max(s(:,1)),max(s(:,2)));
    if length(p)==1 N=tmp; else N=length(p); end;
    if tmp>N
        error('data point index exceeds number of data points');
    elseif min(min(s(:,1)),min(s(:,2)))<=0
        error('data point indices must be >= 1');
    end;
elseif size(s,1)==size(s,2)
    N=size(s,1);
    if (length(p)~=N)&&(length(p)~=1)
        error('p should be scalar or a vector of size N');
    end;
else error('s must have 3 columns or be square'); end;

% Construct similarity matrix
if N>3000
    fprintf('\n*** Warning: Large memory request. Consider activating\n');
    fprintf('        the sparse version of APCLUSTER.\n\n');

```

## 参赛队号 #2021

```

end;
if size(s,2)==3 && size(s,1)~=3,
    S=-Inf*ones(N,N,class(s));
    for j=1:size(s,1), S(s(j,1),s(j,2))=s(j,3); end;
else S=s;
end;

if S==S', symmetric=true; else symmetric=false; end;
realmin_=realmin(class(s)); realmax_=realmax(class(s));

% In case user did not remove degeneracies from the input similarities,
% avoid degenerate solutions by adding a small amount of noise to the
% input similarities
if ~nonoise
    rns=randn('state'); randn('state',0);
    S=S+(eps*S+realmin_*100).*rand(N,N);
    randn('state',rns);
end;
% Place preferences on the diagonal of S
if length(p)==1 for i=1:N S(i,i)=p; end;
else for i=1:N S(i,i)=p(i); end;
end;

% Numerical stability -- replace -INF with -realmax
n=find(S<-realmax_); if ~isempty(n), warning('-INF similarities detected;
changing to -REALMAX to ensure numerical stability'); S(n)=-realmax_; end;
clear('n');
if ~isempty(find(S>realmax_,1)), error('+INF similarities detected; change to
a large positive value (but smaller than +REALMAX)'); end;

% Allocate space for messages, etc
dS=diag(S); A=zeros(N,N,class(s)); R=zeros(N,N,class(s)); t=1;
if plt, netsim=zeros(1,maxits+1); end;
if details
    idx=zeros(N,maxits+1);
    netsim=zeros(1,maxits+1);
    dpsim=zeros(1,maxits+1);
    expref=zeros(1,maxits+1);
end;

% Execute parallel affinity propagation updates
e=zeros(N,convits); dn=0; i=0;
if symmetric, ST=S; else ST=S'; end; % saves memory if it's symmetric

```

## 参赛队号 #2021

```

while ~dn
    i=i+1;

    % Compute responsibilities
    A=A'; R=R';
    for ii=1:N,
        old = R(:, ii);
        AS = A(:, ii) + ST(:, ii); [Y, I]=max(AS); AS(I)=-Inf;
        [Y2, I2]=max(AS);
        R(:, ii)=ST(:, ii)-Y;
        R(I, ii)=ST(I, ii)-Y2;
        R(:, ii)=(1-lam)*R(:, ii)+lam*old; % Damping
        R(R(:, ii)>realmax_, ii)=realmax_;
    end;
    A=A'; R=R';

    % Compute availabilities
    for jj=1:N,
        old = A(:, jj);
        Rp = max(R(:, jj), 0); Rp(jj)=R(jj, jj);
        A(:, jj) = sum(Rp)-Rp;
        dA = A(jj, jj); A(:, jj) = min(A(:, jj), 0); A(jj, jj) = dA;
        A(:, jj) = (1-lam)*A(:, jj) + lam*old; % Damping
    end;

    % Check for convergence
    E=((diag(A)+diag(R))>0); e(:, mod(i-1, convits)+1)=E; K=sum(E);
    if i>=convits || i>=maxits,
        se=sum(e, 2);
        unconverged=(sum((se==convits)+(se==0))~N);
        if (~unconverged&&(K>0)) || (i==maxits) dn=1; end;
    end;

    % Handle plotting and storage of details, if requested
    if plt || details
        if K==0
            tmpnetsim=nan; tmpdpsim=nan; tmpexpref=nan; tmpidx=nan;
        else
            I=find(E); notI=find(~E); [tmp c]=max(S(:, I), [], 2); c(I)=1:K;
            tmpidx=I(c);
            tmpdpsim=sum(S(sub2ind([N N], notI, tmpidx(notI))));
            tmpexpref=sum(dS(I));
            tmpnetsim=tmpdpsim+tmpexpref;
        end;
    end;
end;

```

## 参赛队号 #2021

```

end;
if details
    netsim(i)=tmpnetsim; dpsim(i)=tmpdpsim; expref(i)=tmpexpref;
    idx(:,i)=tmpidx;
end;
if plt,
    netsim(i)=tmpnetsim;
    figure(234);
    plot(((netsim(1:i)/10)*100)/10,'r-');    xlim([0    i]);    %    plot
barely-finite stuff as infinite
    xlabel('# Iterations');
    ylabel('Fitness (net similarity) of quantized intermediate solution');
%    drawnow;
end;
end; % iterations
I=find((diag(A)+diag(R))>0); K=length(I); % Identify exemplars
if K>0
    [tmp c]=max(S(:,I),[],2); c(I)=1:K; % Identify clusters
    % Refine the final set of exemplars and clusters and return results
    for k=1:K ii=find(c==k); [y j]=max(sum(S(ii,ii),1)); I(k)=ii(j(1)); end;
notI=reshape(setdiff(1:N,I),[],1);
    [tmp c]=max(S(:,I),[],2); c(I)=1:K; tmpidx=I(c);
    tmpdpsim=sum(S(sub2ind([N N],notI,tmpidx(notI))));
    tmpexpref=sum(dS(I));
    tmpnetsim=tmpdpsim+tmpexpref;
else
    tmpidx=nan*ones(N,1); tmpnetsim=nan; tmpexpref=nan;
end;
if details
    netsim(i+1)=tmpnetsim; netsim=netsim(1:i+1);
    dpsim(i+1)=tmpdpsim; dpsim=dpsim(1:i+1);
    expref(i+1)=tmpexpref; expref=expref(1:i+1);
    idx(:,i+1)=tmpidx; idx=idx(:,1:i+1);
else
    netsim=tmpnetsim; dpsim=tmpdpsim; expref=tmpexpref; idx=tmpidx;
end;
if plt||details
    fprintf('\nNumber of exemplars identified: %d    (for %d data
points)\n',K,N);
    fprintf('Net similarity: %g\n',tmpnetsim);
    fprintf('Similarities of data points to exemplars: %g\n',dpsim(end));
    fprintf('Preferences of selected exemplars: %g\n',tmpexpref);
    fprintf('Number of iterations: %d\n\n',i);
    fprintf('Elapsed time: %g sec\n',etime(clock,start));

```

## 参赛队号 #2021

```
end;
if unconverged
    fprintf('\n*** Warning: Algorithm did not converge. Activate plotting\n');
    fprintf('    so that you can monitor the net similarity. Consider\n');
    fprintf('    increasing maxits and convits, and, if oscillations occur\n');
    fprintf('    also increasing dampfact.\n\n');
end;
```

### 附录 2：各类数据

表 4 中国自建国以来历史文化发展背景

时代	年份	历史、文化背景
40—50 年代	1949 年	新中国成立
50 年代	1951 年	西藏和平解放
	1952 年	彻底废除封建剥削制度
	1953 年	第一个五年计划开始
60 年代	1966 年	文化大革命开始
70 年代	1976 年	文化大革命结束
	1978 年	开始改革开放
80 年代	1980 年—1990 年	文革修整，经济恢复时期
90 年代	1991 年—2000 年	经济复苏时期，全面实行改革开放
21 世纪以来	2001 年—2013 年	全面建设小康社会，文化多元化发展

表 5 歌曲特性汇总表

时代	曲名	距离均值	距离方差
20 世纪以来	海鸣威 - 老人与海.wav	12982.8571	538979540.4
	黄立行、刘若英 - 分开旅行.wav	14497.6842	251161176.5
	李克勤 - 红日.wav	21524.7568	1749368389
	黄小琥 - 放心不下.wav	30432	470362201
	刘德华 - 练习.wav	23931.871	403010883.7
	莫文蔚 - 电台情歌.wav	19256.8889	479614293.3
	裘海正 - 爱我的人和我爱的人.wav	32736	972481936.7
	孙燕姿 - 遇见.wav	33699.84	751598960.6
	王菲 - 夜会.wav	28330.6667	2515614484
	温岚 - 夏天的风.wav	22575.1579	998892114.3
	谢霆锋 - 前前后后左左右右.wav	12356.7761	322039102.2
	徐良、孙羽幽 - 情话.wav	47136	1459382565
	徐若萱 - 爱笑的眼睛.wav	30144	2290248363
	杨幂 - 一定要幸福.wav	49792	1577287680
	张娜拉 - 双鱼座.wav	36754.2857	4802186913
	张志林 - 我输了.wav	45260.8	2414863038
	郑智化 - 水手.wav	11736	48936082.29
	周冰倩 - 真的好想你.wav	20323.9024	1031074212
	朱铭捷 - 曾经最美.wav	26702.7692	742201854.4

## 参赛队号 #2021

	庄心妍 - 一万个舍不得.wav	16576	150589178.6
平均数		26837.5128	1198494648
90 年代	Beyond、黄家驹 - 海阔天空.wav	26342.4	481871970.9
	阿桑 - 叶子阿桑.wav	28268.3077	824103000.2
	陈慧娴 - 千千阙歌.wav	23259.4286	659859404.4
	陈琳 - 花样年华.wav	32389.5652	1533684611
	陈琳 - 雨夜.wav	19840	191877419.7
	陈瑞 - 夜来香.wav	30006.8571	1996550999
	陈淑桦 - 笑红尘.wav	22090.1053	1103526724
	邓丽君 - 天涯歌女.wav	27827.2	623532780.6
	邓丽君 - 夜上海.wav	26037.6774	394606208.8
	凤飞飞 - 追梦人.wav	17881.0435	241137432.5
平均数		25394.25848	805075055.1
80 年代	邓丽君 - 阿里山的姑娘.wav	39978.6667	4022701839
	邓丽君 - 万水千山总是情.wav	15896.1509	185490231.4
	费翔 - 冬天里的一把火.wav	17814.2609	793499915.1
	费玉清 - 一剪梅.wav	24000	1090669945
	关贵敏 - 浪花里飞出欢乐的歌.wav	29065.8462	769399092.8
	蒋大为 - 祝酒歌.wav	24280.6154	597188633.2
	经典老歌 - 康定情歌.wav	15419.0769	99226545.23
	罗大佑 - 童年.wav	12345.3134	92304121.4
	平安 - 我爱你中国.wav	34350.5455	4901897043
	张明敏 - 我的中国心.wav	19735.814	876044097.5
平均数		23288.62899	1342842146
70 年代	才旦卓玛 - 太阳最红毛主席最亲.wav	29319.5294	6524724164
	才旦卓玛 - 在北京的金山上.wav	17171.6923	309827940.5
	蒋大为 - 乌苏里船歌.wav	16571.7333	449056842.5
	经典老歌 - 军港之夜.wav	21701.4857	329550034
	金磊鑫 - 赞歌.wav	34790.4	5279062426
	李双江 - 北京颂歌.wav	20594.5263	774896591.9
	李谷一 - 大海啊故乡.wav	0	0
	李双江 - 红星照我去战斗.wav	11904	91717632
	李双江 - 我爱五指山我爱万泉河.wav	33408	2047911058
	宋祖英 - 渔家姑娘在海边.wav	17137.3714	: 935591715.71
平均数		20259.87384	1756305188
60 年代	才旦卓玛 - 唱支山歌给党听.wav	27315.2	1142171125
	革命歌曲 - 不忘阶级苦.wav	16267.6364	610337140.4
	郭兰英 - 翻身道情.wav	17589.0732	1132072940
	何纪光 - 挑担茶叶上北京.wav	16704	180266850.5
	经典老歌 - 送别.wav	52800	1484546793
	李谷一 - 谁不说俺家乡好.wav	17024	457340716.1
	李双江 - 走上这高高的兴安岭.wav	64731.4286	16763921554
	彭丽媛 - 我的家乡沂蒙山.wav	22528	724724206.3



## 参赛队号 #2021

	宋祖英 - 毛主席的话儿记心上.wav	77414.4	14816090522
	童丽 - 荷花颂.wav	19148.8	368114520.4
平均数		33152.25382	3767958637
50 年代	刀郎 - 敖包相会.wav	31245.4737	1972374905
	邓丽君 - 玫瑰玫瑰我爱你.wav	17232	175815396.8
	革命歌曲 - 我骑着马儿过草原.wav	15222.8571	129270270.7
	郭兰英 - 妇女自由歌.wav	34092.5217	1406126428
	胡松华 - 草原上升起不落的太阳.wav	20331.7895	580442190.7
	李双江 - 弹起我心爱的土琵琶.wav	11034.9474	101526227.7
	李双江 - 太阳出来喜洋洋.wav	20381.5385	302761196.3
	宋祖英 - 远方的客人请你留下来.wav	13725.9574	540910075.7
	彭丽媛 - 我的祖国.wav	39936	8431730688
	阎维文 - 歌唱二郎山.wav	15693.283	366519116.2
平均数		21889.63683	1400747650

注：以上歌曲来自《酷狗音乐》。2013-5-17. 15:22-5:35。

## 歌词

## 《爱》

ti:爱]  
[ar:小虎队]  
[al:BEST]  
[offset:0]  
[00:38.30]把你的心我的心串一串  
[00:41.23]串一株幸运草、串一个同心圆  
[00:45.80]让所有期待未来的呼唤  
[00:48.97]趁青春做個伴  
[00:53.35]别让年轻越长大越孤单  
[00:56.47]把我的幸运草种在你的梦田  
[01:00.97]让地球随我们的同心圆  
[01:04.22]永远的不停转  
[01:08.47]向天空大声的呼唤說声我爱你  
[01:12.41]向那流浪的白云說声我想你  
[01:16.28]让那天空听得见、让那白云看得见  
[01:20.47]谁也擦不掉我们许下的诺言  
[01:24.54]想带你一起看大海說声我爱你  
[01:28.58]给你最亮的星星說声我想你  
[01:32.39]听听大海的誓言、看看执著的蓝天  
[01:36.27]让我们自由自在的恋爱  
[01:53.52]把你的心、我的心串一串  
[01:56.47]串一株幸运草、串一个同心圆  
[02:00.72]让所有期待未来的呼唤  
[02:04.04]趁青春做個伴  
[02:08.22]别让年轻越长大越孤单  
[02:11.66]把我的幸运草种在你的梦田  
[02:16.04]让地球随我们的同心圆  
[02:19.35]永远的不停转  
[02:23.66]向天空大声的呼唤說声我爱你  
[02:27.66]向那流浪的白云說声我想你  
[02:31.41]让那天空听得见、让那白云看得见  
[02:35.66]谁也擦不掉我们许下的诺言  
[02:39.79]想带你一起看大海說声我爱你  
[02:43.73]给你最亮的星星說声我想你  
[02:47.55]听听大海的誓言、看看执著的蓝天  
[02:51.49]让我们自由自在的恋爱  
[03:00.99]别让年轻越长大越孤单  
[03:04.05]把我的幸运草种在你的梦田  
[03:08.37]让地球随我们的同心圆  
[03:11.62]永远的不停转  
[03:15.93]向天空大声的呼唤說声我爱你  
[03:19.93]向那流浪的白云說声我想你  
[03:23.74]让那天空听得见、让那白云看得见  
[03:27.99]谁也擦不掉我们许下的诺言  
[03:32.12]想带你一起看大海說声我爱你  
[03:36.07]给你最亮的星星說声我想你  
[03:39.88]听听大海的誓言、看看执著的蓝天  
[03:43.84]让我们自由自在的恋爱