复旦大学计算机科学技术学院

2018-2019 学年第一学期期末论文课程评分表

课程名称：自然语言处理 Natural Language Processing **课程代码：**COMP130141.01

开课院系： 计算机科学技术学院

学生姓名： 祁佳薇 **学号：** 16307130293 **专业：** 计算机科学与技术

论文名称： 中国有说唱——中文 rap 分析与自动生成

（以上由学生填写）

成绩：

中国有说唱——中文 rap 分析与自动生成

摘要：

七月嘻哈音乐席卷中国
揭开 Gold Chain 与 freestyle 的地下王国
蛤蟆镜、肥 T-shirt 和拖地牛仔裤
以前是格格不入现在人人都说 Cool
曾经演出一场几乎要倒贴
现在天天等着要约数不过来钱
Fake, diss, battle , 听着又坏又危险
妹子，跑车，票子，Party 每天从不闲
Wanna know 中国 Hip-hop 到底什么样？
用大数据分析给你中国嘻哈的炫酷报道~

关键词：中文 rap 分析，歌词生成，押韵，马尔可夫模型，RNN

一、缘起

去年七月，某奇艺出品的选秀节目《中国有嘻哈》吸引了无数目光，‘hip-hop’、‘freestyle’、‘rapper’成为那个夏天最热的音乐标签。对大多数人来说，“两个月听的嘻哈比之前半辈子听的都多”一点不夸张。随着中国嘻哈强势闯入主流文化，许多人也意识到：原来除了“药药切克闹”，中国嘻哈还有这么多货色，之前从来没听说过的地下 rapper，居然也有自己的一派粉丝、一方天地。暑假之前还是网易云上一只忧郁民谣狗的舍友，一夜之间，歌单已被黑炮儿占据，微信聊天句子都不自觉押韵还带着旋律，不顾军训的疲惫连夜刷节目，对于豆芽的热爱也一直延续到今天 JonyJ 的每一场演唱会上。

我受舍友的影响，也一直关注着中国说唱圈儿。这学期开始学习 python 自然语言处理以后，就想到用 python 来分析一下：中国嘻哈到底在唱些什么？中国的 rapper 又是一个怎样的群体呢？以及 rap 最明显的、不同于传统歌词的特点就是押韵！酷炫的单押、双押，甚至多押，使得这种音乐非常朗朗上口，又能表达充沛的情感。中华文化博大精深，押韵的词语数不胜数，但毕竟不是多年 rapper，张口就来 freestyle，普通人就需要靠机器来找到押韵的词语，人工拼接或者直接自动生成一段 rap，再也不用担心跟人 battle 会输了！

二、已有相关工作及项目总体思路

- **相关工作**

基于说唱的流量占比，众多媒体也进行了数据分析，例如“嘻哈 x 大数据：300 万字歌词分析告诉你中国 rapper 到底在唱啥”。但这其中还是有一些我感兴趣的信息没有获取到，例如 rapper 的社交网络图、受众情感分析等。

歌词的生成并不是一个冷门的主题，致敬周杰伦、汪峰等人的作词器，效果都还不错；做 rap 生成的也有了先例，但大部分都是英文 rap 生成。这些工作虽然有效但都较为复杂。因此，本项目计划使用 python 及一些简单的工具包来更全面分析、更简单生成说唱歌词。

- **总体思路**

1. 爬取网易云音乐中国 rapper 歌曲，经过数据清洗作为语料库使用
2. 分析中国 rap 情况
   - 词频统计（画词云图）
   - 搭配统计/转移概率（可以看某一个词下一个词会是什么，画柱状图）
   - 国内 rapper 之间的社交网络
   - 情感分析
3. 分别使用马尔可夫模型以及 RNN 生成 rap

- **使用工具**

编程语言版本：python 3.6.3

所需 python 库：jieba, nltk, xpinyin, re, request, numpy, matplotlib, beautifulsoup, wordcloud, snownlp

其它工具：tensorflow, Gephi

程序列表：

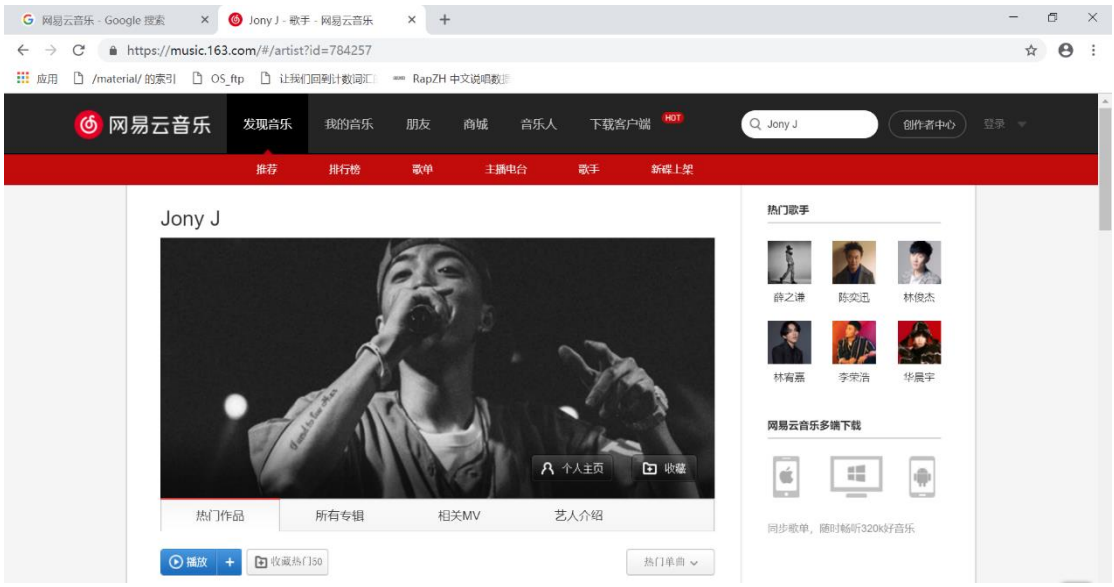
```
rap.py          # 生成歌词
spider.py       # 爬取歌词
ciyuntu.py      # 根据出现频率生成词云图
tran_prob.py    # 根据转移概率生成水平柱状图
comments.py     # 爬取评论并进行分析
emotion.py      # 歌词情感分析
```

三、具体过程

1. 搜集语料和原始数据清洗

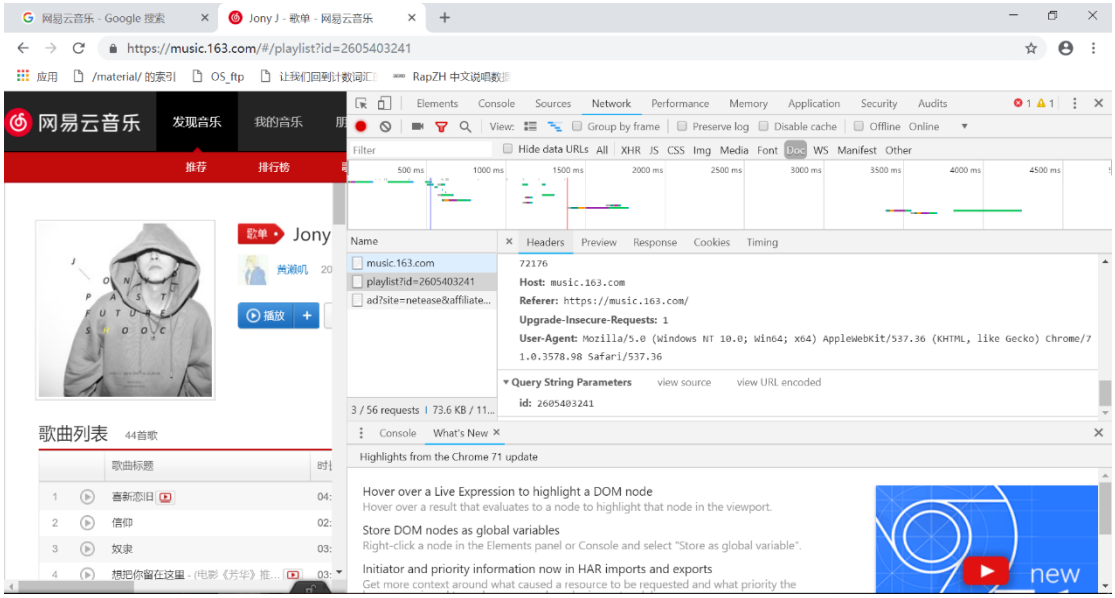
1.1 爬取网易云音乐上中国 rapper 的歌曲作为语料库

首先进入网易云音乐界面，搜索一位中国 rap 歌手（以 JonyJ 为例），进入其主页

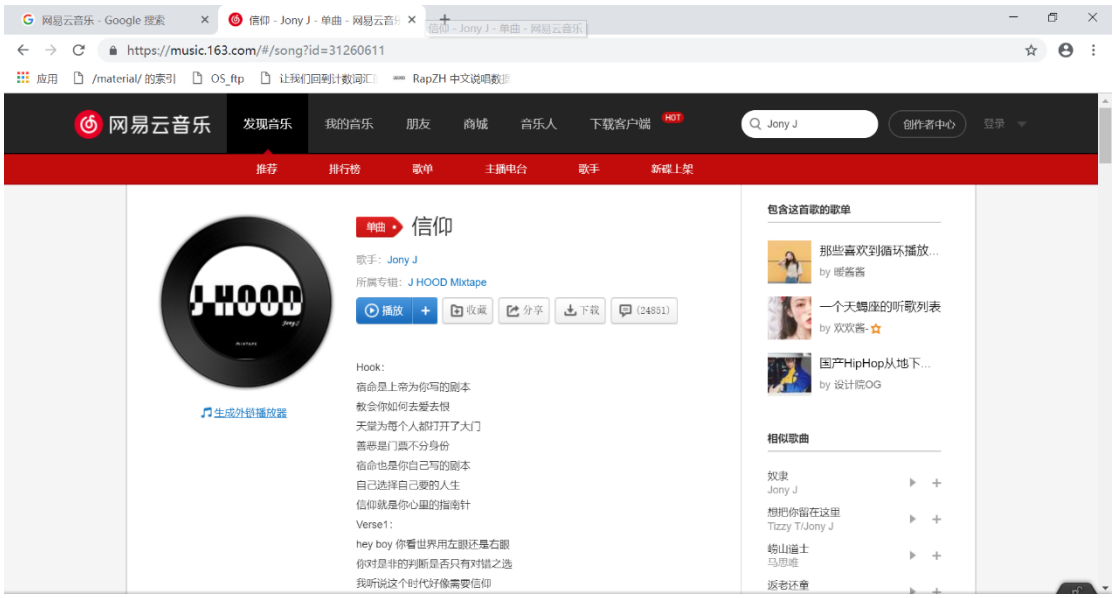


可以看出每个歌手都有自己唯一的 ID，通过这个 ID 就能找到歌手的页面，但是考虑到一个歌手下面列出的单曲中会有很多重复（各种 live 版、与他人合唱版等），我们换一种思路，搜索 JonyJ 的歌单，是因为考虑到这些歌单由用户创建，为便于自己听歌，一般不会收

录很多现场版重复版，以及平衡数量多、创建时间新、收藏人数多等因素进入一个歌单



可以看到每个歌单也有自己的一个 ID，进一步每首歌也有自己的一个 ID



所以整个爬虫的大体思路很简单：歌单→歌词
手动确定感兴趣的歌手名单，获取歌单 ID，请求歌单 ID，从上一步请求歌单的结果中可以提取歌单内所有歌曲的 ID，继续请求获取歌词，使用正则表达式匹配方括号中的内容进行过滤（过滤掉每一句歌词对应的时间）

最后将结果输出到以各个歌手名字命名的 txt 文件中，同时将所有歌手的歌曲输出到 init_lyric.txt 文件中

1.2 爬取每个歌手热门歌曲评论

思路也很简单：歌手→歌词

之前的爬虫进行得很顺利，到这里第一次失败是因为找不到歌手清单中某些歌手（红花会和 pgone 下架…）；第二次失败是因为数据量加大，遭遇反爬虫，因为来不及学习解决方案，所以只能分析一下爬取到的 JonyJ 的前几首歌曲的评论信息

1.3 数据清洗

虽然歌词格式较为整齐，但还是需要进行处理才能应用于本项目。

- 用于生成歌词：过滤掉所有非中文字符和空字符（因为是中文 rap 生成，并且英语的押韵不好考虑，所以一并过滤）；过滤掉作词作曲等信息
- 用于生成词云图和转移频率条形图：过滤掉所有除英文字母和中文字符之外所有的字符（在总体分析时需要保留英文单词，中国歌手的 rap 还是用了很多英语，大概是因为 rap 起源是美国，这种音乐形式更加适合用英文表达）
- 生成社交网络图时需要把评论中歌手昵称改成统一格式
- 后续大部分操作都需要分词，使用 jieba 和 nltk，有时需要去除停用词（根据 rap 的特殊性会对停用词列表做出调整）

2. 中国 rap 情况分析

- **词频统计（画词云图）**

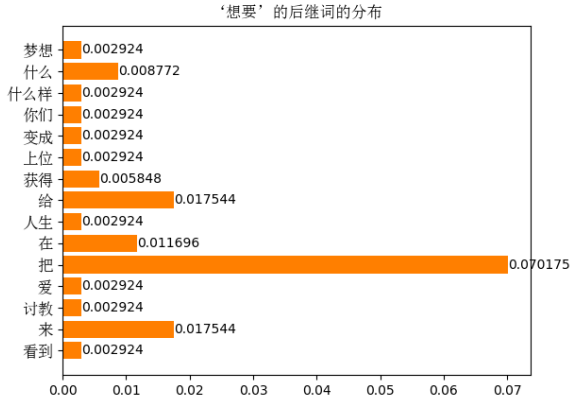
这里数据清洗时保留了中文和英文，分词之后需要进行过滤，中文的 stop_words 选了网络上找到的资料，英文的 stop_words 使用 nltk 内置工具，另外结合特殊性，保留了例如不，没有这些词；添加了 I 等词。（后来觉得其实还可以改进，这些词反应出来的信息太少，远不如 love, baby 等）

（这真的是 JonyJ…）

分析上图可以看出来（忽略掉上面提到的本来特意保留，后来发现效果不好的虚词，以及后来发现没有过滤作词作曲），英文也占据了中文说唱的半壁江山，rapper 们很关心时间、现在，想要和 wanna 出现频率很高，like,love,baby,oh 的占比非常符合形象了。

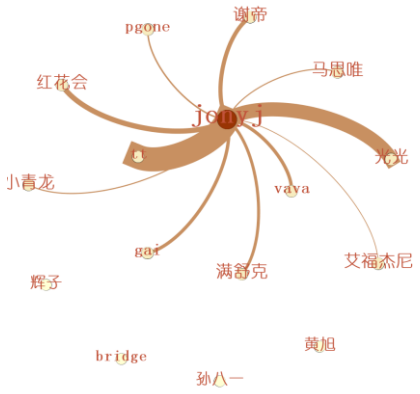
- **搭配统计/转移概率**

这里也是先要分词，考虑的是相邻关系，即一个词推出紧接着的下一个词是什么。统计 bigram 出现频率，写入字典，这里我们看一下上面提到的高频词“想要”，看看这些嘻哈歌手到底想要什么：看起来 rapper 也不是想象中的那么消极，他们也想要爱，想要人生、梦想，想要+讨教组合可能多出现在两人 battle 中。



- 画 rapper 之间的关系图（画那个）

因为爬虫数据获取困难，我们仅分析 jonyj 和其它 rapper 的共现次数，从而推知关系。对于列表中每一位歌手，从所有评论中查找，找到一次共现次数+1，这样的统计只是十分粗略的。最终使用 Gephi 得到如下的共现图:



jonyj 与其它 rapper 共现图

- 情感分析

对每一个歌手的歌词进行情感分析（需要进一步了解）

```
红花会 1.0
PGone 1.6564527527407336e-13
VaVa 1.0
艾福杰尼 1.0
BooM黄旭 1.0
Bridge 1.0
GAI爷 1.0
TizzyT 1.0
JonyJ 1.0
小青龙 1.0
辉子 1.0
孙八一 1.0
谢帝 1.3322676295501878e-15
马思维 1.0
Mc光光 1.0
满舒克 1.0
```

3. 实现马尔可夫模型

前面已经统计了词频，并计算了词之间的转移频率，现在实现马尔可夫模型

3.1 根据概率转移矩阵生成下一个词

运行 rap.py，命令行请求用户输入，作为 rap 的第一个词（后面的结果演示都以**“我”**作为 rap 的第一个词）。如果从这个 start 开始，每次根据上一个词由转移矩阵选择出现概率最高的下一个词，则必然会出现循环：

我 万岁 钞票 万岁 钞票 万岁 钞票…

考虑两种解决方法：一是考虑词性的转换，用 jieba 标注词性，建立词性转移矩阵，两个概率相乘选择概率最大的一组词，即使用维特比算法；二是其它要求确定句子可能的结构（例如下面的字数要求、押韵要求、首尾词要求）

3.2 使用马尔可夫模型生成符合字数要求的 rap

首先读入模板，默认是中文说唱经典作品《差不多先生》的第一小段（后面生成的 rap 都是以该段为模板，即 8 行，字数如下），用户也可以根据自己的需求更改模板。模板主要规定了 rap 行数（句数）、每行字数，便于模仿创作。

我抽着差不多的的烟
又过了差不多的一天
时间差不多的闲
我花着差不多的的钱
口味要差不多的咸
做人要差不多的贱
活在差不多的边缘
又是差不多的一年

采用最简单直接的递归生成，从前往后填词，参数中包括当前词，当前句数，当前词在句中的位置。当前词 curr 生成的下一个词 succ 是一句的末尾时，要检查是否满足该句字数要求，如果找遍 curr 的后续词都不满足要求，就回退，重新选择 curr。为避免上文出现的循环问题（如果继续选择最大概率的下一个词，只限制字数，仍有很大可能出现循环），我们随机选择，即先打乱顺序，然后按顺序遍历，当遍历到链表末尾仍不满足时才回退。
结果：

我很天真也差不多想
发达差不多的药又干
了八十八个贱人
这差不多的姿势看着
差不多都像乌龟但
乌龟乌龟但乌龟但
乌龟翘吧差不多想
发达差不多的边缘

3.3 使用马尔可夫模型生成押韵的 rap

接下来是最重要，最能体现一段 rap 是不是够 rap 的地方了——押韵。平时所说的押韵一般指古诗中，连续的句子最后一个字用韵母相同的字“窗前明月[光]，疑是地上[霜]”，最后一个字都押韵母[ang]。说唱里的押韵有所不同，或许是受英语多音节押韵的影响，今年中国 rapper 越来越追求多字押韵，简称多押，作为评判歌手实力的标准。也是说唱小白认识一段 rap 是不是够 rap 的标准。

所谓多押，就是在连续的句子里面，不仅仅是最后一个字押韵，是最后几个字都押韵。例如：

| 我现在 freestyle 给你们示范[双押]

| 你们不适合玩说唱你们适合种[庄稼]

两句的韵脚是[ang,a]

有时还可以押近音，即韵母不同，但读音相近，例如[ei]和[ui]。经典的酷炫多押有红花会贝贝的：

| 心境高雅韵如风

| 英俊潇洒令狐冲

当然，实际的说唱中，也不能片面地追求每句都押，越多越好。其它两个维度 flow（节奏）和内容也是十分重要的。

本次项目中采用*8 的押韵方式（前面提过用差不多先生的 8 句作为模板），由用户来指定韵脚（yunjiao 作为类的成员），当生成函数检测到下一个词符合字数要求的时候就判断其是否押韵，如果一直找不到，就回退。

提取一个词的韵母需要先用 xpinyin 库中函数 get_pinyin()得到例如’an-eng’格式的双联词的拼音，以‘-’划分以后可以使用。匹配韵母可以在一个字的拼音中以元音’aeiou’作为分界，取当前以及后面的部分作为韵脚，或者直接用正则表达式进行匹配

结果：

[an]

我唱了命令中被我慢
我做着他们笑我吐痰
你生活有时我慢
自己摊我长处你我赞
想要的帅着笑我唱
了该我选择自己摊
我的咔差不多是玩
当然希望可以慢慢

[en]

我除了我说不是我很
了解我们走的对不分
身份宿命也不分
身份宿命也就被我很
了解我们也是我很
了解我们走对不分
身份宿命是好的人
要怎么样我和他们

可以看到到目前为止，生成的 rap 已经基本符合要求，包括字数行数、句内较为连贯、但是开头和结尾会卡在奇怪的地方…比如以“的”、“了”开头，以“很”结尾。考虑下面的进一步改进方案。

3.4 使用马尔可夫模型生成限制首尾词的 rap

对出现在每一句第一个词、最后一个词两个特殊位置的词进行统计，得到 start_words 和 last_words 两个类。当上一步发现下一个词是末尾且符合字数要求时，需要判断是否在是否在 last_words 中，如果遍历完以后还是没有，就从所有押韵的词中随机选择一个；而在选择开头词时略有不同，如果都不在 start_words 中，就从开头词中随机选择一个。

结果：

[en]

我捧起四川的歌都跟
老子杀敌以前不管狠
情形自己那这根
他住还有就拎着部分
我定定没有太认真
听好梦幻想红他问
老师温柔是否你笨（好了我知道了…）
最美这个是酷的分

[o]，输入他

他信不信我拍着录我
现在是狐狸我哪怕我
完全驾驭当你噢
真的旋律配吃多加我
下句么谈恋爱的馍
求神拜佛在玩过我
尽情自由怀抱为博
上井盖里面陪好我

[ou],输入爱

爱明明琳琅满目的又
或者不会变自由午后
我自作多情多抽
你们轰动时刻现在有
一次嘛嘛笑到把后
兄弟要朝着梦想又
玩这些歌速度和狗
看电影夜场对只狗

```
What do you want to start your rap with?
 > 爱
Alright, here's your rap:
爱叨叨琳琅满目的又
或者不会变自由午后
我自作多情多抽
你们轰动时刻现在有
一次嘛嘛笑到把后
兄弟要朝着梦想又
玩这些歌速度和狗
看电影夜场对只狗
```

4.实现 RNN 生成 rap

这里模仿已有的“赵雷曲风歌词生成器”进行设计。机器作词最常用的方法就是序列建模（seq2seq），序列 A 到序列 B 之间的映射建立模型，工作流程如下：

- A 中每个单词通过 embedding 操作后，输入到编码器（多层 RNN 结构），输出一个向量；
- 训练时，解码器输入同编码器，输出与 B 的交叉熵作为模型的目标函数；
- 生成时，给定种子序列作为输入，解码器上一时刻输出作为下一时刻输入，循环直到生成给定数量的序列

具体实现流程：

- 确定好 encoder-decoder 中 cell 的结构，这里使用 tensorflow 提供的 LSTM
- 将输入数据转化成 tensorflow 中 decoder 需要的格式，得到输出和最后一个隐含状态
- 将输出数据经过 softmax 层得到概率分布，得到误差函数，确定梯度下降优化器
  仿照“赵雷歌词生成器”的经验，选择两层 LSTM，每层包含 128 个 cell，vector 大小为 100，序列长度 16，学习率设置为 0.001，进行训练

由于第一次完整实现深度学习的代码，实在是不熟悉，写完模型，再经过 8 个小时的训练，没出来结果…然后就再参考其它代码修改，最后时间关系只能放弃。

四、结果分析

根据 hmm 最后生成的歌词来看，只能说勉强通顺和押韵，而且由于是递归生成，需要大量的回溯，看其他人通过深度学习出来的中英文 rap 结果也不够好。但其实同样的算法，可能换成短文本例如对联可能表现会好很多，或者其它较长文本，例如宋词会因为堆砌很华丽的辞藻而看不出通顺程度，或者其它主题的歌词可能不像 rap 需要很多押韵、歌词过于口语化（导致一些口语连在一起无意义，而一些文艺的词，即使是相同词性连起来，读着也不觉得不知所云）。所以 rap 生成器仅供参考和娱乐。

通过本次课程设计，我收获了很多。首先是第一次尝试并完成爬虫获取批量数据，然后数据得清洗整理花费了很多时间；其次对于 hmm，课上其实讲过最简单的情形，当时作业生成的文本表现很差，这次加上了字数、押韵等的限制，有很大改进。然后可能是因为期末考完身心疲惫做 pj，写基础代码就花费了不少的时间（各种 bug），但是对 python 的使用可以说熟练了很多；用 python 作图十分方便（但是改格式改了好久），画词云图、柱状图、共现图等，数据可视化非常便于分析；然后这学期没有选修 CV 课，没在这门课的课程项目中独

立完成一次深度学习模型的训练，也没有自己想着学过，就导致临时狂学一整天，最后写的代码也有很多漏洞…就做完以后回头看自己怎么花了这么多时间在这么简单的事情上，或者以前觉得很难不想主动去学的东西现在被迫静下来去看觉得还好，为什么没有早点学，最后结果还不好，还是拒绝拖延，要再努力吧，“差不多”到底不是现在应该有的态度，“Life is a struggle”还是真理。

前面谈到了不足，还有很多改进方向：一是扩大语料库的范围，这次的歌曲只是来源于十几个大家较为熟悉的 rapper，中国的 rapper 群体比这要庞大得多，学习更多的数据应该会带来更好的结果；二是关于词性标注，其实写了相应的代码以后才发现词性不是特别必要，因为句内的关联由转移矩阵计算出来的结果还算合理，rap 本来就很口语化，很随意，中国人说话逻辑不像英语那么严谨，所以就没有使用，改进的话，可以生成一句后使用维特比算法找出最大概率的结果，可能会有一定程度的改进；三是不同的说唱作品表达的主题是不同的，表现出来的情感也是不同的，可以通过指定主题生成 rap，考虑从语料库中提取主题词；四是刚才说到的递归耗费大量时间的问题，可以考虑优化递归方式，以及运行程序可以看出来时间耗费其实更多的是在前面建立数据库上，可以考虑把字典存入数据库中，这样不用每次都计算状态转移矩阵等。

最后关于中文说唱，有人说以说唱街舞等为代表的嘻哈文化是黑人文化的糟粕，充斥着肮脏的话语和绝望的情绪，应该坚决抵制；也有人说中文根本就不适合唱 rap，中文讲究字正腔圆不便于饶舌。但是我们看到周杰伦的音乐不曾有所谓的糟粕，而是充满了积极，jonyj 被誉为说唱界的诗人；vava《我的新衣》以及 gai 等人将中国的方言融入到说唱中，独特又好听。中国的 rapper 应该继续发扬创新这一外来音乐形式，而不是仅仅学到 diss 的精髓。

论文评语（教师填写）：

任课教师签名：

日　　期：