# 歌词生成大数据系统

**作者**：徐锡峰，丁一，杨智翔，韩奇志，池培德，冯永

**单位**：重庆大学，计算机学院

**案例版权**：该案例归重庆大学计算机学院所有

**涉及的知识点**：Python数据挖掘、数据预处理、循环神经网络、数据可视化

**案例来源及案例真实性情况**：该案例来源于重庆大学计算机学院专业硕士（电子信息）课程《大数据架构与技术》中的学生精选汇编课程设计。

**摘要** 听音乐是人们休闲娱乐的主要方式之一。歌词是一首歌曲的灵魂，也是影响人们听歌体验的重要因素，对歌曲的传唱度具有举足轻重的意义。但是，近年来歌词中如表达不清、辞藻低俗等问题极大降低了许多听众的收听体验。因此歌词自动生成系统有了其出现的空间。基于以上，本项目是一个歌词生成系统，收集了各个歌手的热门前50名歌词数据，通过训练神经网络来学习歌词中的内涵风格。之后，本案例再将学习到的歌词风格以用户指定的输入主题来生成不同的歌词，为创作者提供歌词写作的辅助手段。在具体实现上，本案例首先基于Python爬虫技术爬取网易云歌词并对获取的数据进行例如生成词云图等相关预处理操作，然后建立深度学习中的循环神经网络模型以学习不同歌词风格，最后由用户指定主题和风格通过系统可视化展示生成的歌词。

**关键词**：歌词生成，数据挖掘，深度学习，数据可视化

## **1 引言**

该教学案例来源于重庆大学计算机学院专业硕士（电子信息）课程《大数据架构与技术》中的学生精选汇编课程设计。该案例的关键问题为歌词生成，涉及歌词的获取、处理、学习和可视化的数据分析处理全阶段。基于以上，需引导学生进行的主要内容有：（1）利用Python爬虫获取网易云音乐中的歌词数据并进行预处理操作；（2）选取适当的深度学习算法对获取的歌词数据进行训练；（3）建立可视化系统以展示训练结果并生成歌词。

## **2 背景介绍**

随着大数据对人们生活的逐渐深入，当今人们对于娱乐生活的追求也在不断提高，听音乐是人们在日常生活中较为普遍的放松方式之一。iiMedia Research的数据显示,2019年手机音乐客户端整体用户规模较上年增长6.6%,达5.80亿人,预计2020年将增至6.18亿人。随着人们精神需求的不断提高，人们对歌曲的质量要求也逐渐提高。但与之相应的却是老一辈作词人年龄增大，产量不足；而年轻作词人积淀不足，歌词质量普遍不高的青黄不接的窘境。其中，歌词是影响人们听歌体验的重要因素，一份好的歌词有助于提高歌曲传唱度，抒发歌曲情感，具有极大的价值。因此，产生了建立于自然语言处理技术之上的歌词生成系统的诞生环境。

基于以上，本案例首先使用Python爬虫获取网易云音乐中的歌词数据，再对获取的数据进行相关预处理操作。之后，本案例使用深度学习中的循环神经网络算法对处理后的数据进行训练，以学得不同歌词的风格。最后建立歌词系统对神经网络学到的歌词风格进行可视化。

本案例是一个歌词生成平台，使用了有关数据挖掘、机器学习算法以及数据可视化等方法，是社会生活中普遍话题，具有一定的实际意义，有利于均衡培养具备各方面素质的人才。

## **3 内容**

该案例的主要内容主要分为三个小节，分别为数据集获取与预处理、循环神经网络训练，以及数据可视化。

3.1 数据集获取与预处理

本案例使用Python编写爬虫代码来实现数据集的收集。爬取的信息主要包括了歌手名、歌曲id、歌曲名以及歌词。

通过对网易云音乐的web页面进行分析，将该爬虫分为三个步骤。第一步，在网页歌手页面获取到歌手的信息和歌手id号；第二步，通过该歌手id号进入歌手详情页，并获取该歌手的前50首热门歌曲id号；第三部，通过歌曲id号爬取该歌曲的歌词信息。具体实现步骤如下。

首先在https://music.163.com/#/discover/artist/cat?id=？&initial=？（）中获取歌手的歌手名和id号，通过对该url进行分析，url中的id的值为该歌手的类型分类。例如1001代表华语男歌手，1002代表华语女歌手，而initial中的值为该歌手类型的姓名首字母排序。例如initial=65表示该歌手姓名首字母为A，initial=66表示该歌手姓名为B。其爬取算法如下：

|  |
| --- |
| 算法1：爬取歌手id |
| 获取网站headers信息 |
| For id |
| For initial |
| 拼接url |
| r = requests.get(url, headers=headers) |
| soup = BeautifulSoup(r.text, 'html5lib') |
| Find artist\_name, artist\_id |
| End for |
| End for |

在实现算法时先需要获取网站的headers信息，利用headers信息来模拟浏览器访问网页。然后通过request包来调用该url和头部。得到一个从服务器返回的Response 响应对象。该响应对象包含服务器返回的所有信息。然后通过BeautifulSoup来提取HTML中的数据。最后在HTML查找对应的属性标签，获取artist\_name和artist\_id。爬取的部分歌手名字和id如下：

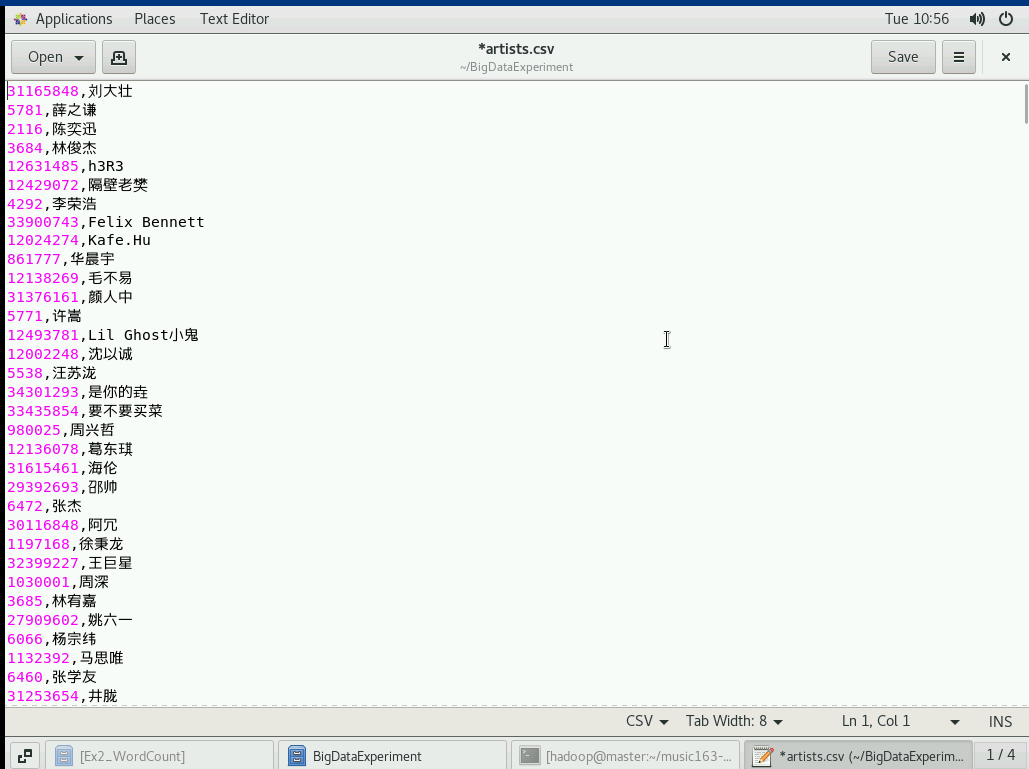


图3.1 歌手名字和id

随后，通过歌手id号来进行入该歌手的详情页。例如https://music.163.com/#/artist?id=？该url地址为相应id号歌手的详情页，可通过该页面获取该歌手的50首歌曲的歌歌曲id号，并通过歌曲id号来获取歌词。其爬取算法和算法一基本相同爬取到的歌曲id号部分信息如下：

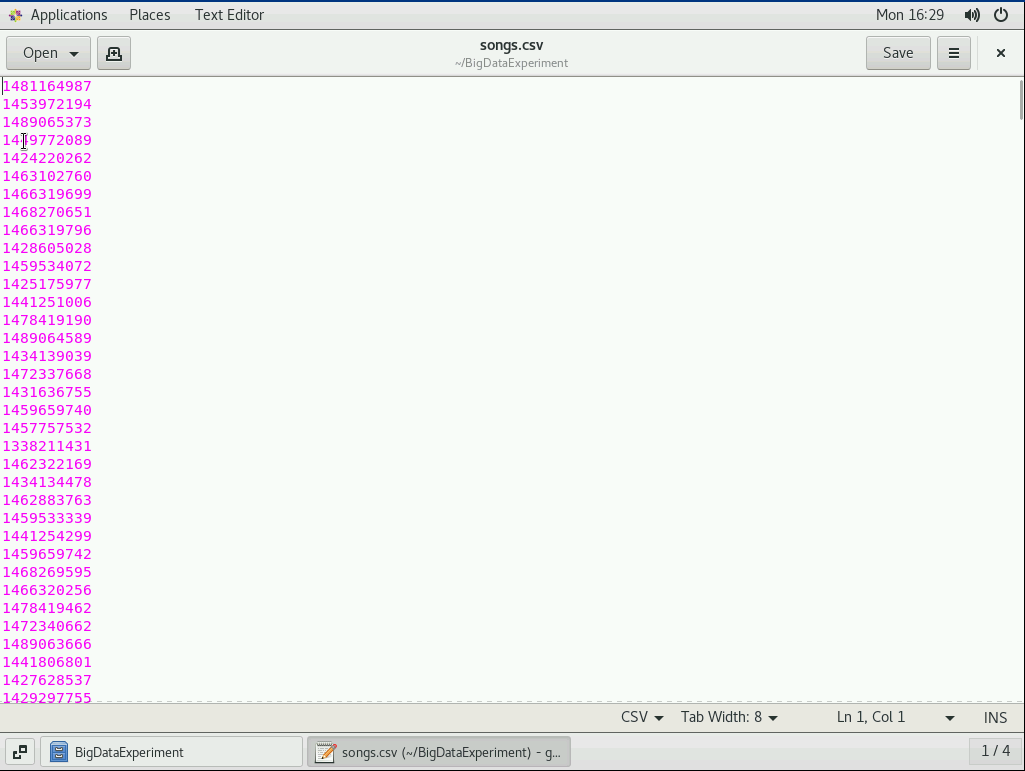


图3.2 歌曲名字

考虑到可能一首歌曲会有多个歌手，按照歌手id-— id来爬取歌曲的时候，可能存在重复的歌曲id。本案例使用分布式爬虫策略scrapy-redis，利用其自带的去重机制来进行歌曲的去重。在实际工作中，可以看到虽然scrapy-redis自带的去重机制，能够有效的对重复的Request进行指纹检查，但是对于不同URL中的重复歌曲还是会产生一定的重复，于是，本案例又对爬取的数据进行遍历，从而执行进一步去重，最终去除了重复数据。最后将获取的歌词信息保存为csv文件，以方便后续的歌词清洗。爬取的歌词信息部分如下:

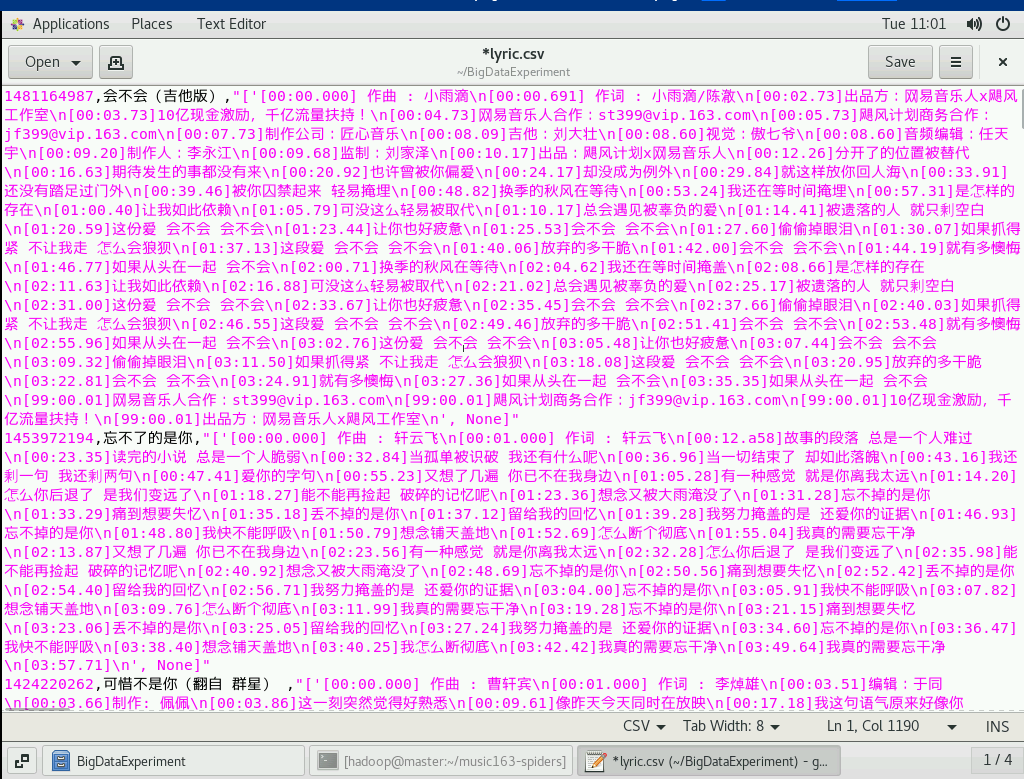


图3.3 获取的歌词信息

通过爬虫获取到的数据分析，获取到的主要分成了3个部分，第一部分是该歌曲的id号，第二部分是该歌曲的歌名，第三部分是歌曲的歌词信息。

通过遍历读取歌曲文件的中csv第三列，可获取歌词的信息。歌词中包含的信息包括了三部分的信息

表3.1歌词信息表

|  |  |
| --- | --- |
| 内容 | 展示方式 |
| 时间戳 | [00:00.000] |
| 创作信息 | 作曲 : 小雨滴\n |
| 歌词信息 | 分开了的位置被替代\n |

其中，需要去除掉时间戳，和创作人信息。只保留歌词信息方便后续的操作。实现算法如下:

|  |
| --- |
| 算法2歌词清洗 |
| 输入：lyric.csv |
| For line in lyric.scv |
| Read lyric |
| 去除时间戳信息 |
| 歌词分行 |
| 去除创作人信息 |
| End for |

去除时间戳信息，可以通过正则表达式中r'\[.\*?\]'来定位到时间戳标签。然后通过sub函数去掉时间戳信息。Sub为正则表达式中的替代函数。采用了一个空字符串来替代时间戳信息，歌词分行采用了replace函数，其作用一样，通过该方法将歌词进行分行来方便表示。再去除已分行歌词中的创作信息，因为创作信息中每一行都带有冒号。可以通过正则表达式中的compile方法来定位到带冒号的一行，然后去除该行。

最后清洗后的部分数据如下：

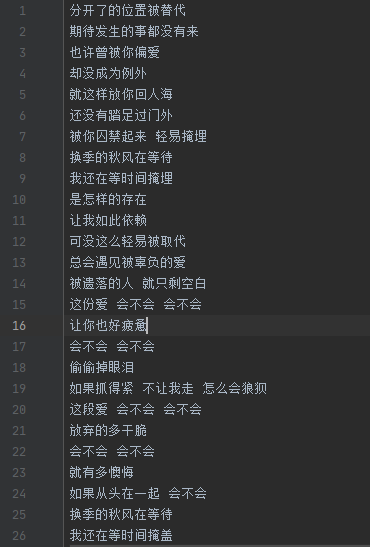


图3.4 清洗后部分歌词

3.2 循环神经网络训练

本案例使用循环神经网络来训练爬取的歌词，以从中学得歌词风格。在进行网络构建之前，需要对数据进行预处理，其实大体的思路为建立字符的数字表示，因为字符没有办法直接输入到网络中，所以需要用不同的数字去代表不同的字符。同时可以设定一个最大字符数，如果文本中读取到的所有不重复的字符数超过了这个最大字符数，就按照字符出现的频率截取掉最后的部分。

处理好数据之后进行网络的定义，网络需要定义三层，第一层是word embedding，即词嵌入层，第二层是RNN层，第三层是线性映射，因为最后是一个分类问题，所以将结果的位数隐射到类别数目。在向前传播的时候，可以指定传入的隐藏状态，虽然训练中可以不用特别指定，但是在生成文本的时候是需要指定的。另外在最后网络输出的时候，网络会将输出进行out.view(-1, out.size(1))这个操作，这个操作是为了将所有的序列拼起来，例如现在的输出是(batch, length)，通过这个操作之后就变成了(batch x length, 1)，这样做是为了方便loss的计算。训练过程为把序列扔到网络中即可。这里有两个小细节，第一个是将label y进行y.view(-1)，这对应于前面网络输出结构的操作，第二个细节是通nn.utils.clip\_grad\_norm()对网络进行梯度裁剪，因为RNN中容易出现梯度爆炸的问题，本案例采用梯度剪裁的方式而非采用新的算法LSTM。

在生成文本中，为了增加随机性，本案例会将预测概率最高的前五个依概率进行选择，并不是每次都选择概率最大的。在生成文本的时候，先通过一句话对网络进行预热，主要是为了得到预热后的隐藏状态，然后将这句话的最后一个词和预热之后的隐藏状态作为网络的第一个输入，得到结果，然后将结果作为下一步的输入，不断循环，直到达到最后的要求的长度。

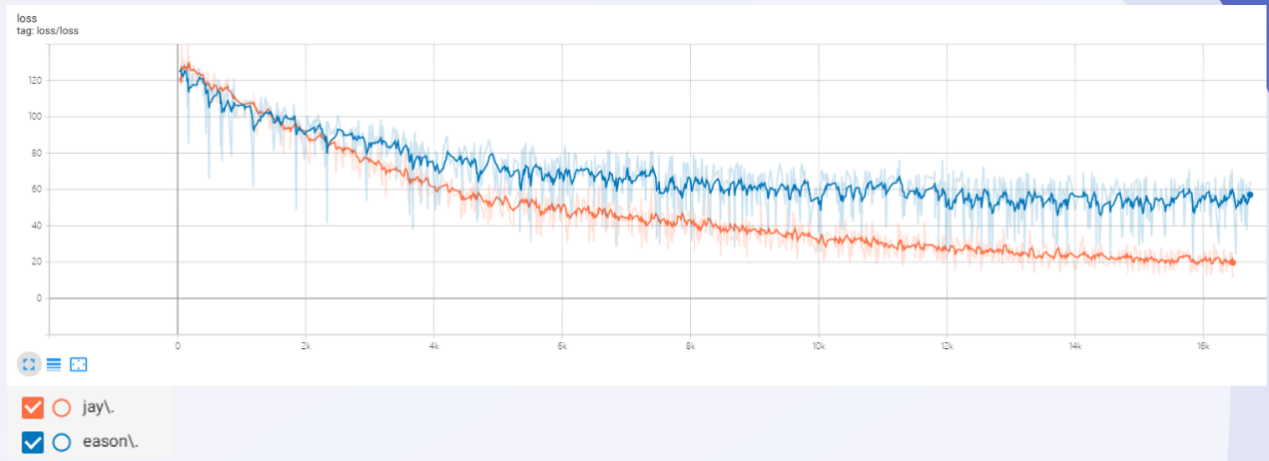
本案例将保存训练后得到模型，并分别对周杰伦和陈奕迅训练过程展示如下：

图3.5 歌词训练过程

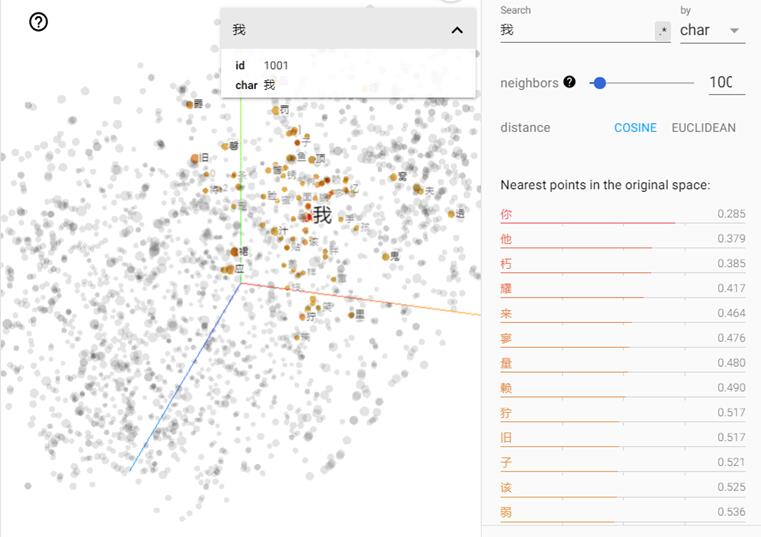


图3.6 词嵌入的可视化图

3.3 数据可视化

本案例采用FLASK-WTF作为框架设计出本案例的前端界面，使用Echarts的工具包来生成词云图，对词云图格式进行了一些调整融入到前端界面中去，其中对于词频数据读取采用的是json的形式。得到的词云图如下所示。



图3.7 歌词词云图

从上面的词云图的结果中，可以看出“爱”这个字出现次数的是最多的，其次是“说”、“想”、“心”。

对于歌词生成系统，其界面如下所示。

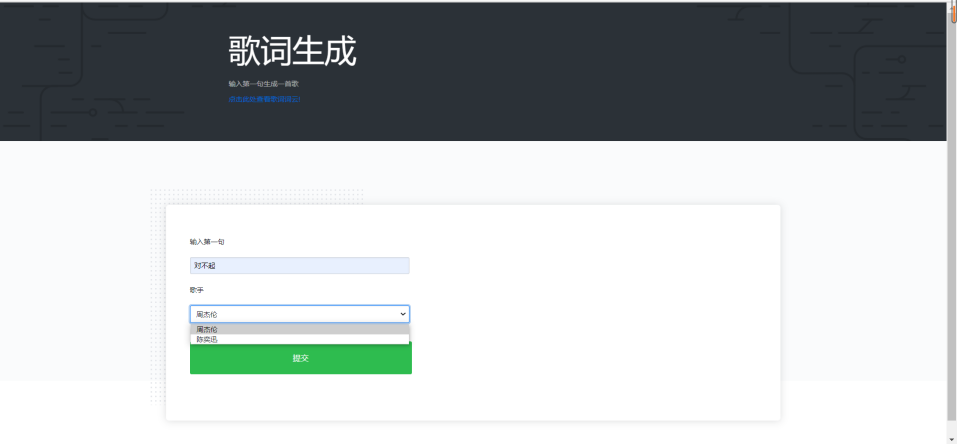


图3.8 歌词生成系统

该页面可以选择需要生成歌词的第一句或是词语，来决定整首歌生成出来的一个感情基调。也可以选择生成的风格，可选的有陈奕迅风格和周杰伦风格。用户在前端界面上输入第一句并选择歌手后，点击提交按钮从而提交表单。后台根据打包的数据自动生成一首对应的歌。图中生成的歌词结果如下所示。



图3.9 歌词生成结果

下两图给出了周杰伦和陈奕迅风格的歌词示例。

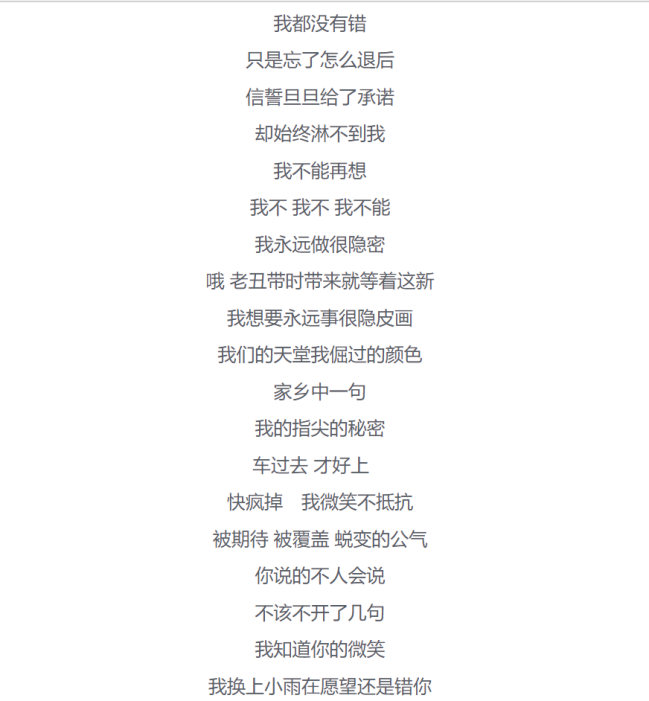


图3.10 周杰伦风格歌词生成结果

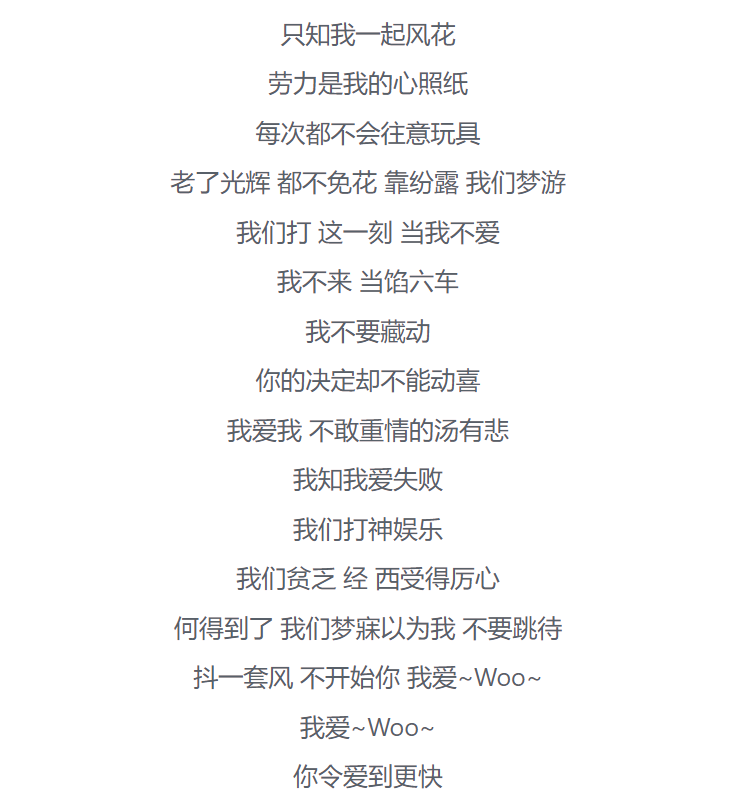


图3.11 陈奕迅风格歌词生成结果

## **4 小结**

本案例的主要研究内容是大数据背景下文本生成问题中的歌词生成子问题，研究的目的是掌握数据挖掘和预处理方法、神经网络训练方法和数据可视化方法，其中使用了Python爬虫技术，并以深度学习中的循环神经网络算法作为歌词学习的主要工具。该案例将现实生活中的问题与多种大数据技术相结合，可以加深学生对大数据理论和技术的理解，能够锻炼学生在解决问题中的思考和动手能力。同时本案例主题贴合实际，具有一定的经济和社会前景，可以引起学生的学习兴趣，有利于学生学习和掌握。

## **附录**

1. 本案例提供配套的PPT、视频、数据集与代码等，发布于Github，链接为：https://github.com/Wanghui-Huang/CQU\_bigdata。

2. 本案例涉及到数据预处理以及多种机器学习算法，建议使用python语言进行编写，推荐的工具包有pandas（数据读取与预处理库），scikit-learn（深度学习算法库），Flask和ECharts（可视化绘图库）。

3. 本案例参考文献如下：

[1]IDC. The Digitization of the WorldFrom Edge to Core [EB/OL]. [2018].

https://www.seagate.com/cn/zh/our-story/data-age-2025

[2] 艾媒大文娱产业研究中心.艾媒报告|2019-2020中国手机音乐客户端市场研究报告[ EB/OL].[2020-02-06]. https://www.iimedia.cn/c400/68645.html

[3] CodingDict.Flask教程. [EB/OL]. http://codingdict.com/article/4862

[4] Jun-Li L . Automatic Generation Method of Poetry Based on Char-RNN[J]. Modern Computer, 2019.

[5] Flagel Lex,Brandvain Yaniv,Schrider Daniel R. The Unreasonable Effectiveness of Convolutional Neural Networks in Population Genetic Inference.[J]. Molecular biology and evolution,2019,36(2).

[6]吴宇鹏.分布式网络爬虫技术的研究与实现[J].电脑编程技巧与维护,2020(11):9-10+19.

[7]子雨大数据之Spark入门教程(Python版). [EB/OL]. http://dblab.xmu.edu.cn/blog/1709-2/