聚类引擎数据结构设计

本文档主要对聚类引擎使用的算法及主要函数进行详细说明。

# 特征提取功能开发

## 1.1 fbank特征提取

fbank特征提取模块代码主要位于CreateFeature目录，特征维度60，其特征提取流程如下图所示：

主要代码：CreateFeature

主要流程：

1）预加重，目的是提升高频部分，使信号变得平坦，保持在低频到高频的整个频带中，能用同样的信噪比求频谱，并消除发声过程中的声带和嘴唇效应，突出高频的共振峰；

2）分帧，由于语音具有短时平稳特征，需将语音分成一个个小段，每一小段为一帧，帧长一般为25ms，相邻帧的重叠为10ms。

3）加窗，一般为方窗、汉明窗、汉宁窗等，目的是消除相邻帧两端可能造成的不连续性。

4）快速傅里叶变换FFT，将语音信号转换到频域，分析语音信号在频域的特性。

4）梅尔滤波器组，人耳对不同频率的敏感程度不同，且成非线性关系，因此我们将频谱按人耳敏感程度分为多个Mel滤波器组；在Mel刻度范围内，各个滤波器的中心频率是相等间隔的线性分布，但在频率范围不是相等间隔的，将能量谱通过一组Mel尺度的三角形滤波器组过滤后获得一个与滤波器个数相等的维数的特征。

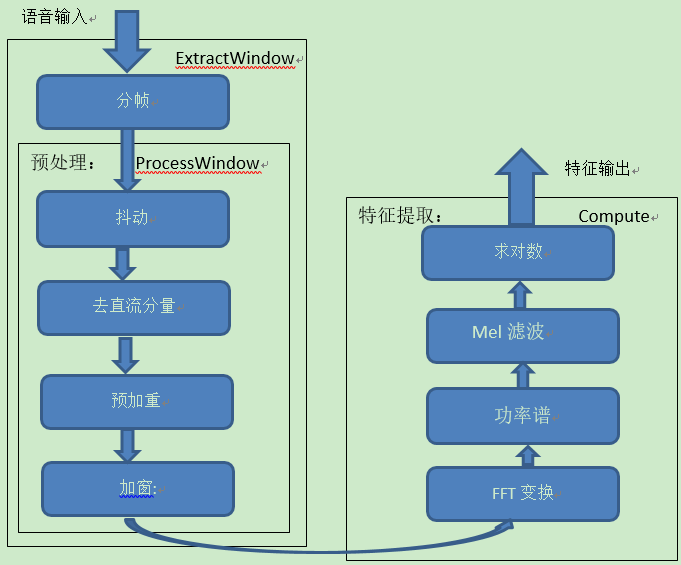


图1.1 fbank特征提取流程图

## 1.2 神经网络说话人特征d-vector

利用神经网络说话人模型，提取隐层特征d-vector，模型输入采用60维fbank特征，利用2000+人，每个人2min以上的数据进行训练，得到说话人模型。



图1.2 TDNN网络提取隐层d-vector特征

主要代码： dvector目录下所有代码

模块接口函数说明：

int process(short \*Feature\_in, int FrameNum, float \*\*dnnResult, int &dnnResultRow, int &dnnResultCol);

输入：

Feature\_in 输入的语音数据

FrameNum 语音帧数

输出：

dnnResult 提取的d-vector特征

dnnResultRow 提取的特征行数

dnnResultCol 提取的特征列数

# 聚类模块

聚类模块主要采用层次聚类和k-means及谱聚类算法，用到的距离公式为离散余弦距离，其计算公式如下：

X和Y分别为音频特征向量，值越大，相似度越高。

下面将对聚类模块进行详细介绍。

## 2.1 层次聚类

层次聚类的合并算法通过计算两类数据点间的相似度，对所有数据点中最为相似的两个数据点进行合并，并反复迭代这一过程。简单的说层次聚类的合并算法是通过计算每一个类别的数据点与所有数据点之间的距离来确定它们之间的相似性，距离越小，相似度越高。并将距离最近的两个数据点或类别进行组合，生成聚类树。

凝聚层次聚类流程如下图所示：

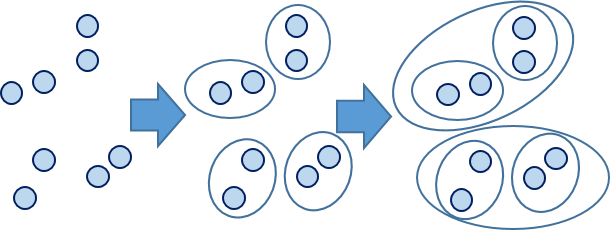


图2.1 凝聚层次聚类(AHC)流程图

层次聚类主要代码：cluster.cc

主要函数：void HCClustering() 层次聚类函数入口

## 2.2 K-means

K-means聚类在层次聚类之后，通过层次聚类确定中心点，对聚类中心进行修正。采用聚类中心作为初始聚类中心点的原因，是因为K-means方法对聚类中心的选取比较敏感，通过层次聚类可以得到相对比较准确的聚类中心。

K-means流程如下：

1）计算每个分段到聚类中心的距离，将该分段标记为对应类别；

2）计算完所有分段后，利用对应类别的特征向量更新聚类中心，即重新确定聚类中心点；

3）重复1）、2）步骤，直到K-means聚类迭代次数完毕。

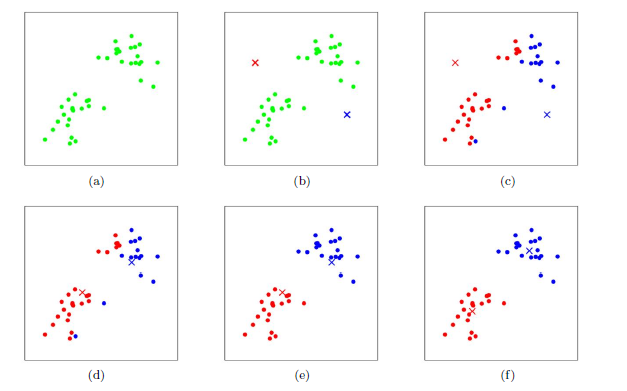


图2.2 K-means聚类中心点更新过程

K-means主要代码：cluster.cc

主要函数：

void WCluster::KMeanClustering(int iItTimes)

函数功能：K-means聚类迭代更新

参数说明：

iItTimes：K-means聚类迭代更新的次数

void WCluster::KMeanClustering(int iStart, int iEnd)

函数功能：K-means聚类，更新聚类中心

参数说明：

iStart：分段起始id

iEnd：分段结束id

## 2.3 谱聚类

谱聚类（Spectral Cluster），这里的谱指的某个矩阵的特征值。谱聚类的思想来源于图论，它把聚类的数据集中的每一个样本看做是图中的一个顶点，这些顶点连接在一起，连接的这些边上有权重，权重的大小表示这些样本之间的相似度。同一类的顶点他们的相似度很高，在图论中体现为同一类的顶点中连接他们的边的权重很大，不在同一类的顶点连接他们的边的权重很小。因此，谱聚类的最终目标就是找到一种切割图的方法，使得切割之后的各个子图内的权重很大，子图间的权重很小。

谱聚类流程：

1）构建图的拉普拉斯矩阵，这里主要由每个分段直接的离线余弦距离组成，并进行对称化和Diffusion操作：

Diffusion(X)=XX^T

该操作主要让不同说话人直接的边界更加清晰。

2）对矩阵进行分解，计算矩阵的特征值和特征向量；

3）取特征值最大的N个特征向量，将说话人特征映射为N维特征；

4）利用得到的新特征做K-means，进一步提高聚类准确性。

主要代码：cluster.cc 谱聚类主要操作位于该文件中。

# 数据结构说明

## 聚类结果

聚类结果以链表形式返回，数据结构说明如下：

typedef struct \_SpkTimeInfo

{

//char\* timeInfo;

int startTime;

int endTime;

char spkTypeABC;

struct \_SpkTimeInfo\* next;

}SpkTimeInfo;

说明：

typedef struct \_SpkTimeInfo 定义结构体

结构体成员：

startTime 分段起始时间

endTime 分段结束时间

spkTypeABC 分段所属聚类类别

struct \_SpkTimeInfo \* next 指向下一个链表指针