标点引擎源代码概要设计说明

# 概述

## 1.1 简介

### 1.1.1引擎功能介绍

标点引擎利用条件随机场（CRF）算法对文本序列进行处理，对文本序列添加标点符号，主要支持逗号、句号、感叹号和问号四种。

适用场景及性能：语音识别结果后处理，对文本序列进行断句，添加标点符号。

### 1.1.2引擎输入介绍

（1）资源文件，包括：CRF模型文件：model

（2）识别结果：语音识别的转写结果作为输入

### 1.1.3引擎输出介绍

标点引擎会对识别结果进行处理，对识别结果添加标点，然后重写识别结果文件。

## 1.2 读者对象

本文档对标点引擎进行详细设计，包括使用的算法和主要实现代码，主要面向引擎开发和维护人员。

1）有一定的C和C++语言基础；

2）了解相关算法。

## 1.3 术语定义

CRF(Conditional Random Fields)：条件随机场，给定一组输入序列条件下另一组输出序列的条件概率分布模型。

## 1.4 参考资料

Paper：

A CRF Sequence Labeling Approach to Chinese Punctuation Prediction.pdf

Conditional Random Fields Probabilistic Models for Segmenting and Labeling Sequence Data.pdf

# 2. 总体设计

标点引擎主要利用条件随机场算法，并对引擎接口进行设计实现，下面进行简要介绍。

## 2.1 条件随机场

条件随机场（CRF）是给定随机变量X条件下，随机变量Y的马尔可夫随机场。在序列标注问题上，主要用到的是线性链条件随机场，这时概率模型可以表示为P(Y|X)，Y是输出变量，表示标记序列，X是输入变量，表示标注的观测序列，标记序列也称为状态序列。学习时，利用训练数据集通过极大似然估计得到条件概率模型P(Y|X)，预测时，对于给定的输入序列X，求出条件概率P(Y|X)最大的输出序列。

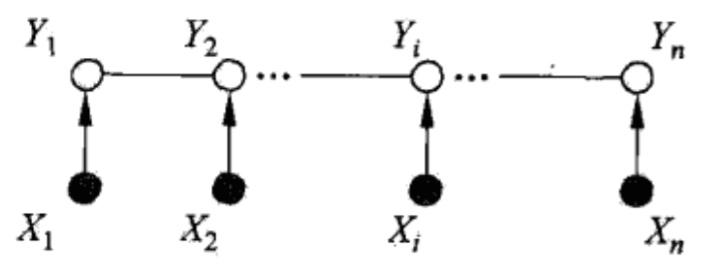


图1.1 X和Y具有相同结构的线性链条件随机场

在本应用中，对识别结果进行分词处理，每个分词作为一个输入label，即，，多个label组成一个输入文本X。为了让模型学习到文本断句信息，这里用较长句子（包含多个标点符号的文本）作为一条训练数据，进行模型训练。

## 2.2 程序设计

标点引擎是以识别引擎输出结果作为输入文件的，对识别结果进行加标点，因为识别引擎本身识别结果为分词形式，所以引擎不再进行重新分词，直接送引擎进行后处理，主要接口包括：

1）模型初始化

2）对识别结果进行处理，添加标点

3）退出引擎

# 3. 可靠性设计

1）函数返回值规范，对应程序里的异常（如：文件打开失败、内存分配、释放异常等），通过函数返回值的形式逐层向上反馈，避免直接退出引擎；

2）支持多线程异步调用，加快处理速度。

# 4. 扩展性设计

## 4.1 支持跨平台

目前引擎支持intel、ARM平台上的编译运行。

# 5. 安全性设计

引擎对各类异常进行判断，有一定的容错机制。

# 6. 易用性设计

引擎接口设计简洁，方便开发者快速熟悉引擎接口调用流程，快速完成接口集成；

引擎同时提供了接口测试程序，可供开发者参考。