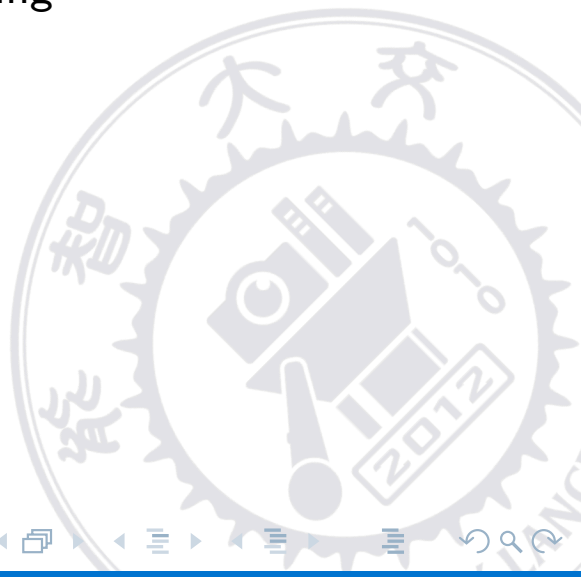
# Project 2: LVCSR 系统搭建

## Building an LVCSR System

Kai Yu and Yanmin Qian

Cross Media Language Intelligence Lab (X-LANCE)
Department of Computer Science & Engineering
Shanghai Jiao Tong University

Spring 2021



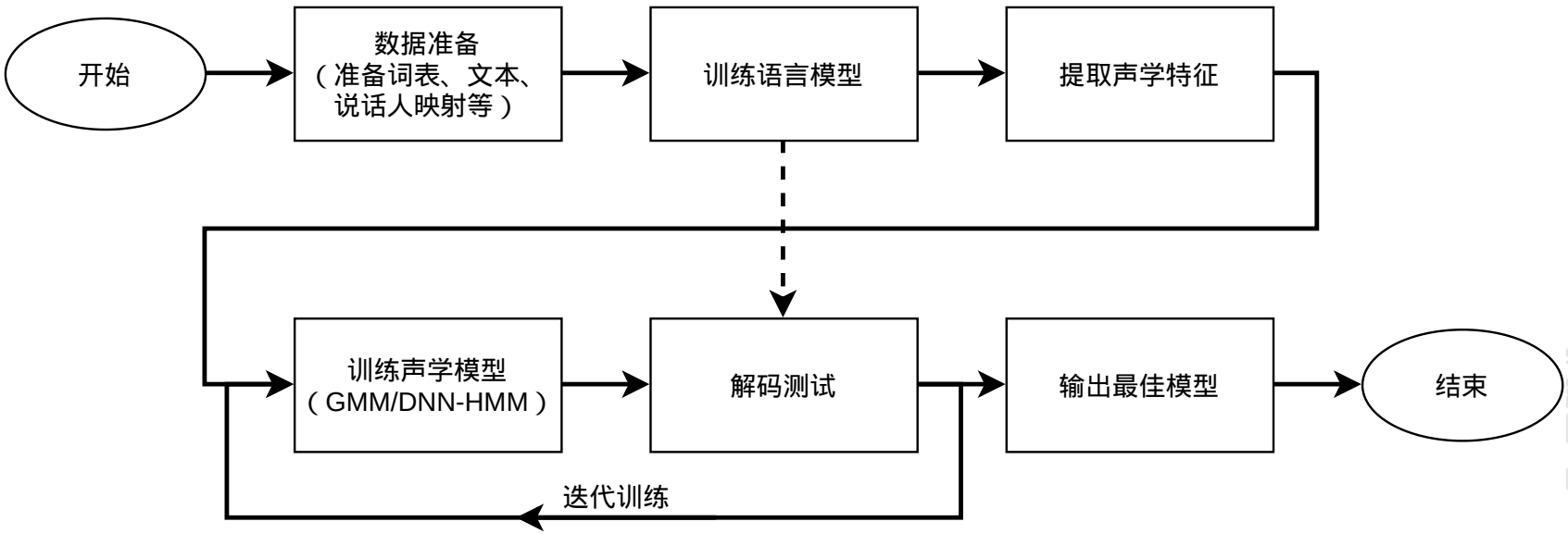
# 任务要求

LVCSR 系统搭建：

- 在给定 10 小时的中文数据集上，利用 Kaldi [和 PyTorch (可选，仅用于 DNN 训练)]，搭建完整的语音识别系统，包括：
  - 数据的处理
  - 特征的提取
  - GMM-HMM 模型的训练
  - 基于 DNN-HMM 的识别系统构建
  - 测试集上的识别解码
- 验收：
  - 所搭建的系统的识别率和标准系统性能相当
  - Technical Report, for all the students.

# 任务框图

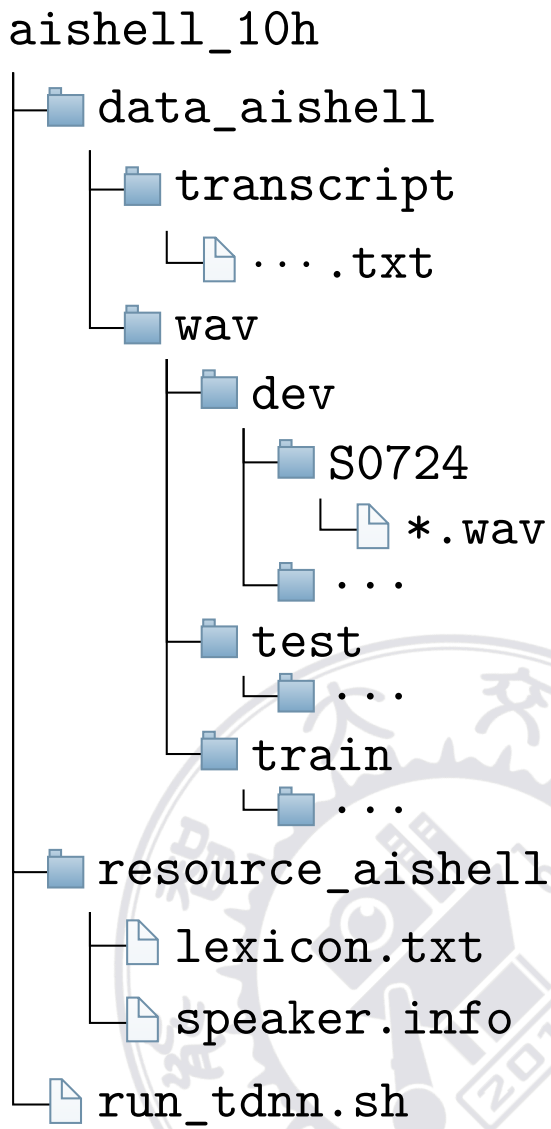
传统 LVCSR 系统的搭建流程如图所示：



# 数据介绍

- 音频与文本（3.2GB）：取自中文普通话开源语音数据库 AIShell-1。其中仅取原训练集中约 10 小时的数据作为训练集，而开发集与测试集同原数据集。所有音频长度通常小于 10 秒，均有文本标注。
  - 训练集 (train)：8000 条音频
  - 开发集 (dev)：14326 条音频
  - 测试集 (test)：7176 条音频
  - 所有音频数据的采样率均为 16 kHz
  - 音频中无背景噪声
- 资源文件：词表和说话人性别信息。
- `run_tdnn.sh`：用法详见报告模板。
- 以上数据及文件下载地址：

  `https://jbox.sjtu.edu.cn/l/Jns2kQ`



# 数据格式介绍

`data_aishell/transcript` 目录中，`aishell_transcript_v0.8.txt` 为数据集全部标注文本，`train_large.txt` 为 AIShell 原训练集文本（用于训练更大的语言模型）。

以 `aishell_transcript_v0.8.txt` 的第一行为例：

BAC009S0002W0122 而 对 楼市 成交 抑制 作用 最 大 的 限 购

- 标签文件的每一行有多列，以空格分隔
- 第一列为每条音频的唯一 ID，与音频文件名相同
- 后面为该条音频对应的文本，已进行分词并以空格隔开。

# 数据格式介绍

- `data_aishell/wav` 目录中为音频数据，分为 train, dev, test 三个子集，分别用于训练、调整超参数、测试。
- 每个子集中分为若干说话人目录，每个说话人目录中包含若干该说话人录制的音频，每条音频具有唯一的 ID，与标注文本一一对应。
  - 例如 `data_aishell/wav/dev/S0724/BAC009S0724W0121.wav` 为说话人 `S0724` 录制的音频，ID 为 `BAC009S0724W0121`，属于 dev 子集，用于调整超参数。其对应的文本在 `aishell_transcript_v0.8.txt` 的第 120099 行。

# Kaldi 介绍

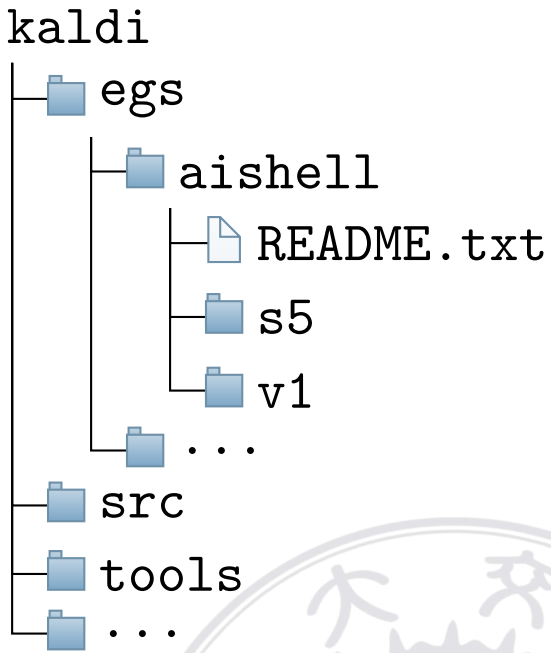
Kaldi 提供了语音识别系统中常用的工具，并为常见数据集的常见任务（语音识别、说话人识别等）提供了 recipe（从数据准备到模型训练与评估的完整流程）。

- 运行环境：Kaldi 可在 Windows/Linux/Mac OS 下运行，但**原则上本项目要求在 Linux/Mac OS 环境下完成**。实际上，在 Windows 下安装 Kaldi 通常会遇到更多的问题。
  - 对于使用 Windows 的同学，你可以使用开源虚拟机软件 Virtual Box，并在其中安装 Linux 系统。推荐使用 Ubuntu 等 Linux 发行版。
  - **尽早着手安装 Kaldi！** Kaldi 的安装过程通常并不一帆风顺，初学者往往会在此花费几天的时间！
  - 本次项目（包括数据集）预计需要占用约 20GB 的硬盘空间，推荐使用至少 4GB 内存的机器（或虚拟机）。
- 安装步骤：参考 Kaldi 源代码根目录下的 INSTALL 文件（文本格式）。若无 GPU，需在 configure 时指定`--no-cuda`。
- **搜索引擎（尤其是 Google）是你最好的伴侣**

# Kaldi 目录结构介绍

以下简要介绍 Kaldi 中的目录结构，以便快速上手。

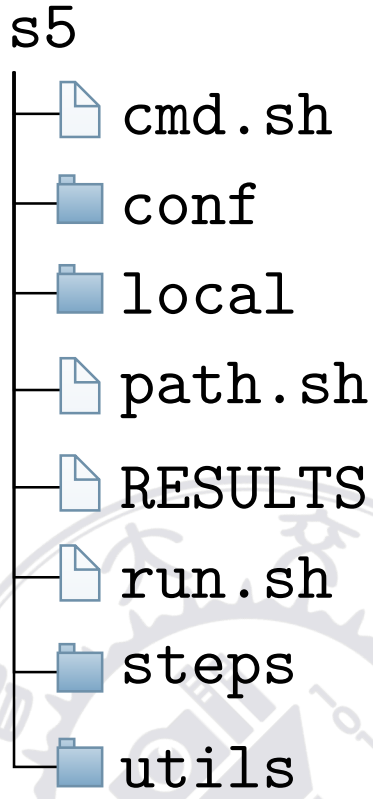
- egs：各个数据集上的 recipe
  - aishell
    - README.txt：该数据集的基本情况介绍
    - s5：语音识别 recipe 目录
    - v1：说话人识别 recipe 目录
- src：Kaldi 工具包的 C++ 源码，编译后生成的二进制程序也存储于此
- tools：第三方工具




除安装过程外，原则上你只应修改 `egs/aishell/s5` 目录内的文件（不包括 `steps` 和 `utils` 两个公共目录），最终仅需提交该目录下的所有代码文件作为你的作业源码。如确有需要修改其它文件，请另附 `README.txt` 注明修改内容，且说明其必要性。

# Kaldi 目录结构介绍——egs/aishell/s5

- cmd.sh：本项目中一般不须关心
- conf：一些步骤中使用的配置文件，如 `mfcc.conf` 指定了提取 MFCC 时的超参数
- local：仅在当前 recipe 内适用的脚本等文件，通常不与其它 recipe 共用
- path.sh：被大多数 Kaldi 脚本在开头调用，用于设置环境变量（通常是 `$PATH`）
- RESULTS：当前任务的参考性能数据（由官方提供）
- **run.sh**：recipe 的入口脚本，原则上在配置适当的机器上，仅需在 recipe 目录中执行 `./run.sh` 即可完成训练–测试的全流程
- steps：包含用于进行训练流程中某些步骤的脚本，封装了 Kaldi 二进制程序
- utils：包含训练流程中随时可能使用的工具脚本，封装了 Kaldi 二进制程序



# 常见问题汇总

- 由于本项目配置难度较大，配置环境及运行脚本时可能出现各种问题，我们将维护一个在线文档，随时将同学们遇到的问题及其解决方案更新在上面。文档地址：
  https://docs.qq.com/doc/DR2JVeXlXZ3BLQXhP
- 同学们在实验过程中遇到问题可先在文档中查阅，若无则自行搜索，搜索不到时再在课程群中提问。
- 目前文档中已有一些内容，请同学们进行实验之前仔细阅读。

# 模型性能的评估指标

字错误率：

$$\text{CER} = \frac{\text{所有样本中预测文本与标注文本的按字编辑距离之和}}{\text{标注文本总字数}}$$

词错误率：

$$\text{WER} = \frac{\text{所有样本中预测文本与标注文本的按词编辑距离之和}}{\text{标注文本总词数}}$$

本项目中以字错误率（CER)作为最终评分标准，词错误率仅用作参考。

在 recipe 目录下的 `steps/scoring` 目录中有相应的计算脚本。

# 提交要求

- 在测试集数据上生成标注文件，格式应与标注文本一致
- 你撰写的 recipe 也应该能仅需执行 `./run.sh` 即可完成训练 – 测试全流程（提交时请打包性能最好的完整 recipe 代码）
  - 请在 recipe 的 run.sh 同目录下添加 README.txt 文件，介绍所作修改、recipe 使用方法和预期的最终模型（一般名为 `final.mdl`）所在目录
- 报告（中文）采用 LaTeX 编写，提交 PDF 格式
  - 模板下载地址：
    `https://latex.sjtu.edu.cn/read/gtstcptfvhhd`
  - 报告命名为“学号-姓名.pdf”
  - 需给出测试集上的测试结果，包括字错误率与词错误率

# 提交要求

- 报告、recipe 最终代码（即 `cmd.sh`、`conf`、`local`、`path.sh`、`run.sh`）和两种预测结果 `test_filt.txt` 与 `test_filt.chars.txt` 文件（格式要求同 P5）一起打包提交，压缩包命名格式为“Project2-学号-姓名.zip”
- 其中 `test_filt.txt` 与 `test_filt.chars.txt` 文件请选取你的字错误率最低的模型生成的预测结果，分别按词与按字分割。这些文件一般位于 recipe 目录下 `exp/.../decode_test/scoring_kaldi` 中。
- 提交方式：Canvas
- 截止时间：2021 年 6 月 24 日 23:59:59

# 评分标准

- 最终得分由系统性能得分、报告得分、报告重复率惩罚三部分构成。计算公式为：

$$\begin{aligned} \text{得分} = & 0.4 \times \text{系统性能得分} \\ & + 0.6 \times \text{报告得分} \\ & - \text{报告重复率惩罚} \end{aligned}$$

其中：

- 字错误率大于 30% 时，系统性能得分为 0；小于 11% 时，系统性能得分为 100；在 30%~11% 之间时，系统性能得分线性增长。
- 报告得分由报告质量决定，有关报告的详细要求请参考报告模板。
- 报告重复率惩罚在报告查重率小于某阈值时为 0，否则线性增长。具体阈值及增长速度在最终评分时确定。**查重率过高时将人工复核，确认为抄袭的以 0 分计**。