Speaker Recognition

Software project progress report

Abrougui Rim
Balard Srilakshmi
Srivastava Prerak
Yang Ruoxiao

November 06, 2019

UNIVERSITÉ DE LORRAINE

Institut des sciences du Digital
Management & Cognition

Plan

Introduction

I. Pre-Processing

   1. Kaldi tool

   2. Data - VoxForge

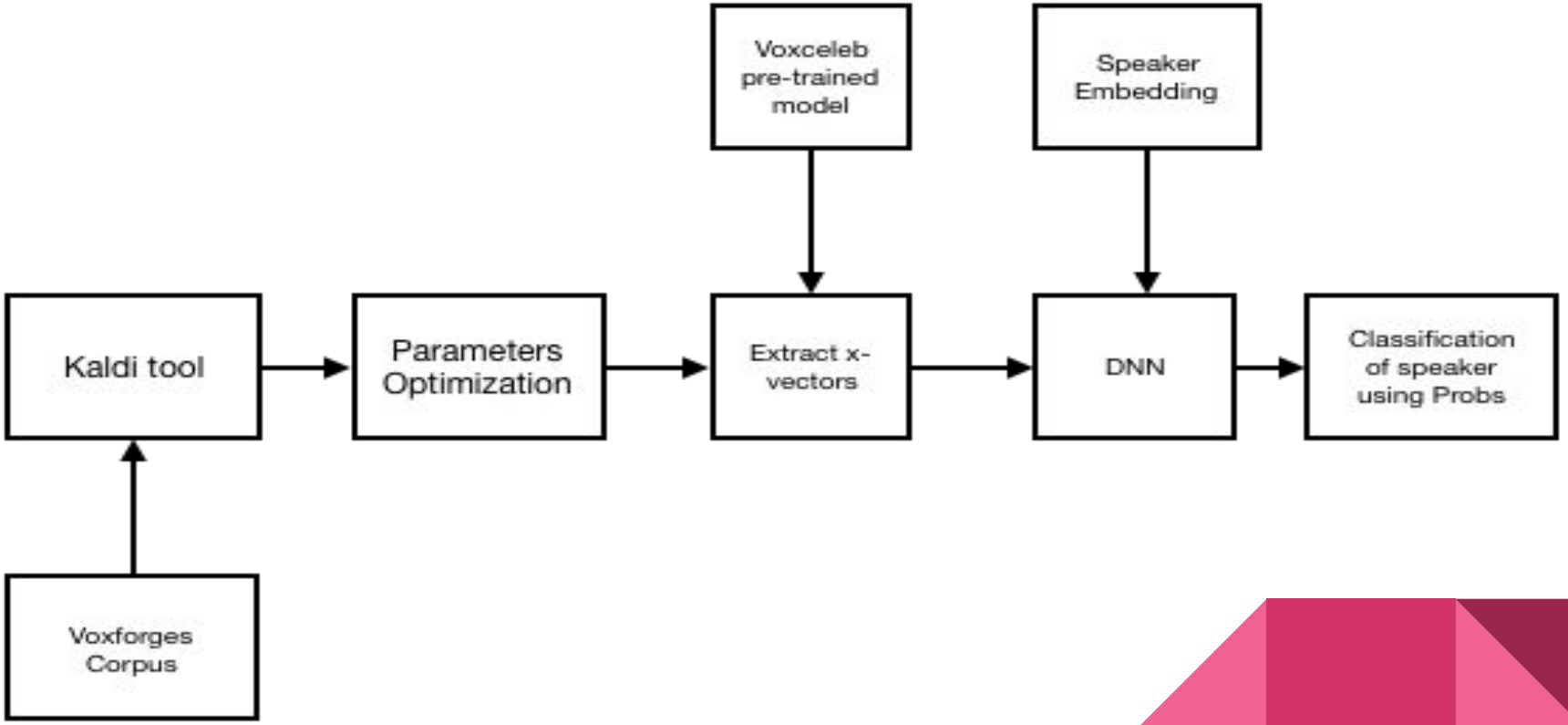
II. Processing

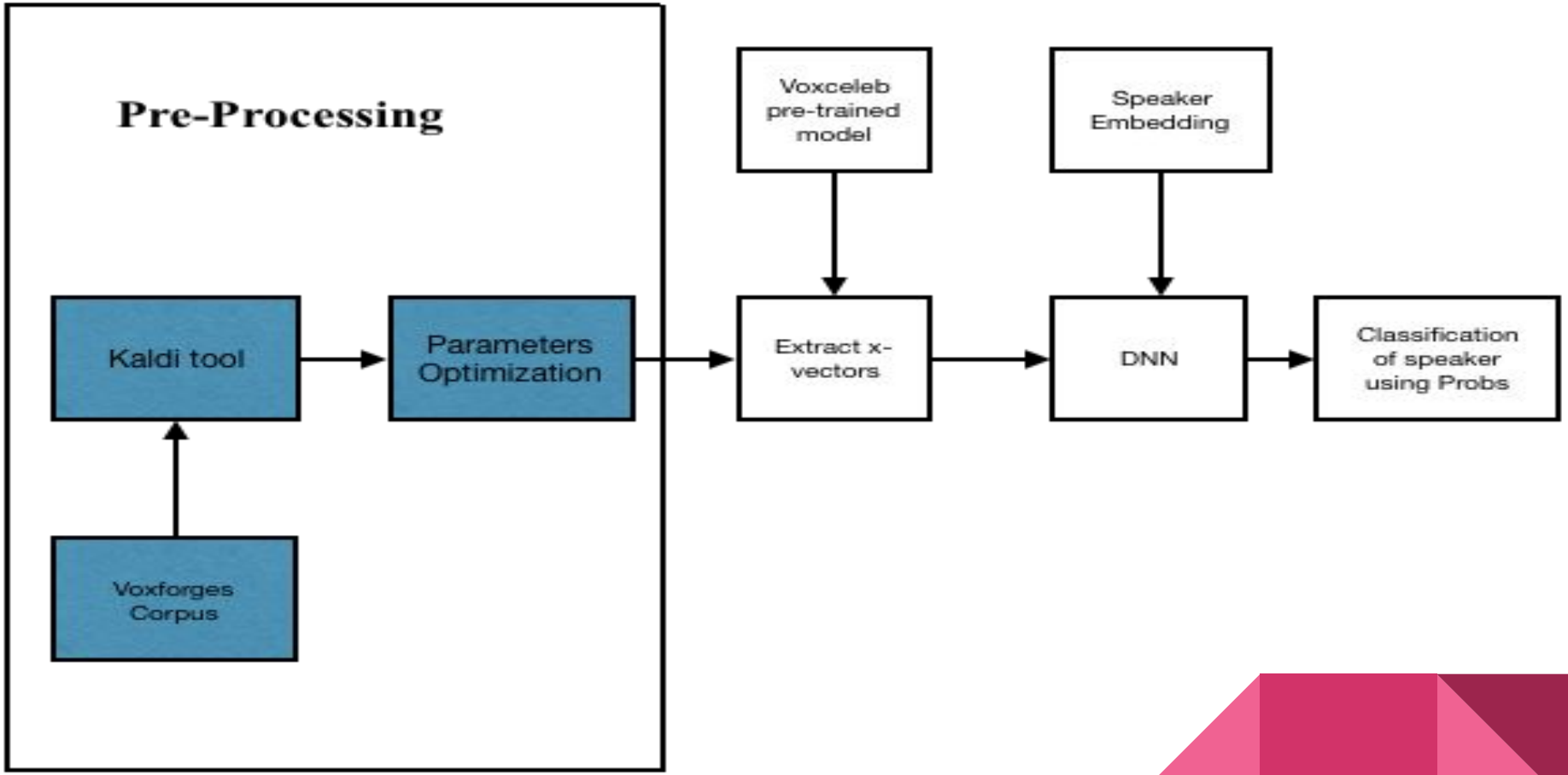
   1. X-vectors

   2. Voxceleb model

   3. Speaker Embedding

Introduction




Pre-Processing
Kaldi tool
Parameters Optimization
Voxforges Corpus
Voxceleb pre-trained model
Extract x-vectors
Speaker Embedding
DNN
Classification of speaker using Probs

Kaldi Tool

- **Kaldi** is an automatic speech recognition (ASR) toolkit written in C++. It contains almost any algorithm currently used in ASR systems.
- We all installed Kaldi successfully in our computers by following the instructions at http://www.kaldi-asr.org/doc/install.html.
  - 2 of us installed Kaldi in Windows Subsystem for Linux with WindowsOS
  - Another 2 of us installed Kaldi in Ubuntu through VirtualBox with MacOS
- We also tested Kaldi successfully by running the demo in the voxforge folder

Data – VoxForge

- At the current stage, we use **VoxForge** as our database.
- **VoxForge** was set up to collect transcribed speech for use with Free and Open Source Speech Recognition Engines (on Linux, Windows and Mac). [ref. http://www.voxforge.org/]
- We downloaded the raw **VoxForge** language data (English) by running the *getdata.sh* in the folder kaldi/egs/voxforge/s5
  - The raw data downloaded is about 27 G, in two folders ~/s5/tgz/ and ~/s5/extracted/, each of which contains 6247 files.
- The raw data from VoxForge is quite large – reduce the data

Data – VoxForge: reducing the data

- We used the following criterion to reduce the data:
  - ignore the folders/speakers starting with “number(s), anonymous, not formal name
  - ignore the folders/speakers which does not have files in “.wav” format
  - 1200 speakers
- We realize this by writing a Python file.
- After this manipulation, the data we are now having 400 speakers

Data preparation for Kaldi

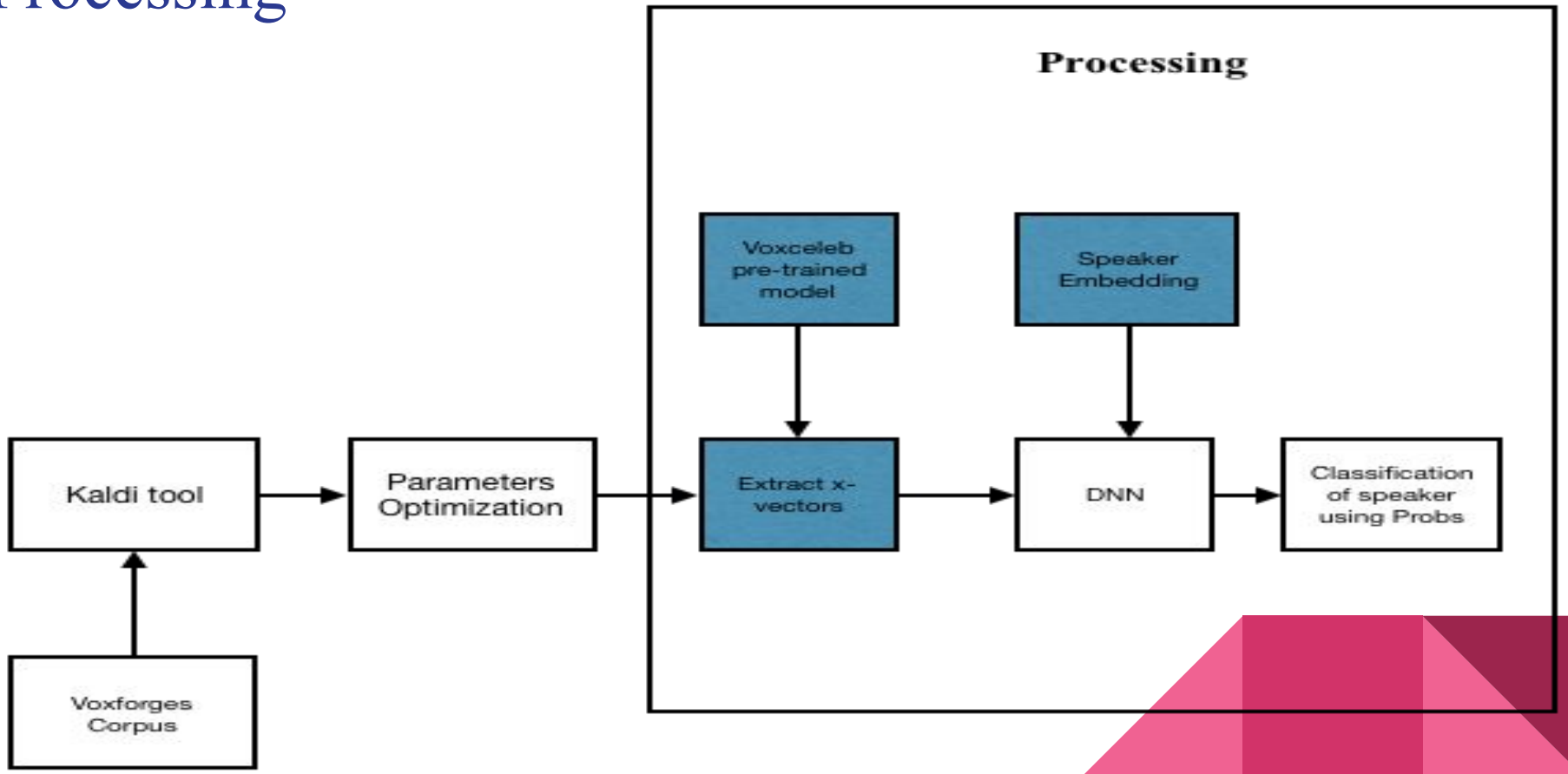
wav.scp: <speaker-id>_<utterance-id> <path>

utt2spk: <utterance-id>_<speaker-id>

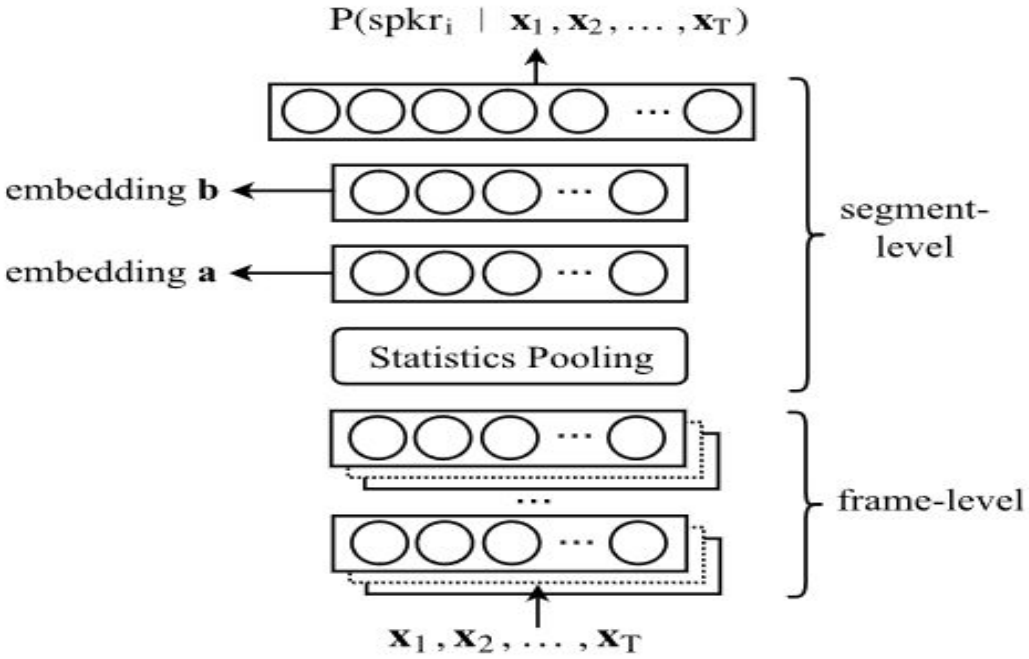
spk2utt: <speaker-id> <utterance-id> <utterance-id2>...

Rest of the files are created on the go when we run the pipeline for extracting x-vectors

Processing


Processing
Voxceleb pre-trained model
Speaker Embedding
Kaldi tool
Parameters Optimization
Extract x-vectors
DNN
Classification of speaker using Probs
Voxforges Corpus

DNN Based X-vector

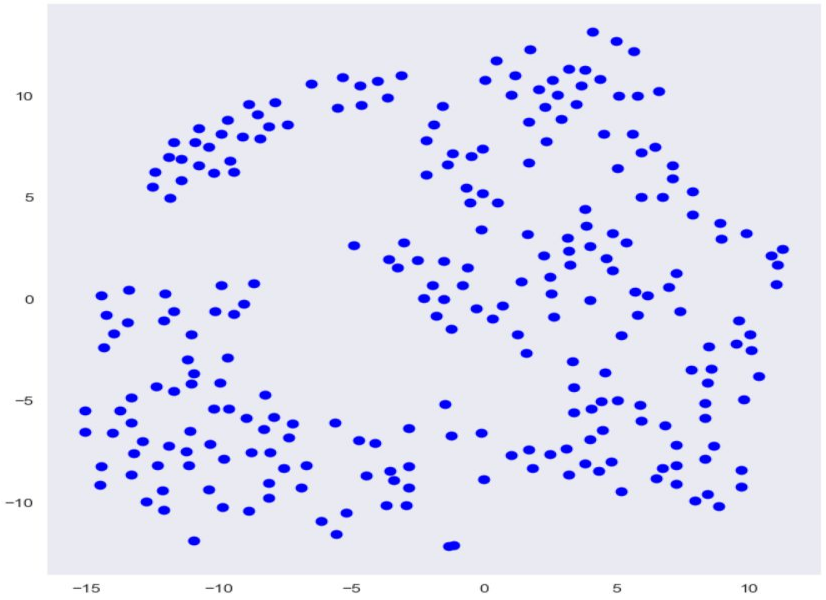


Voxceleb pre-trained model

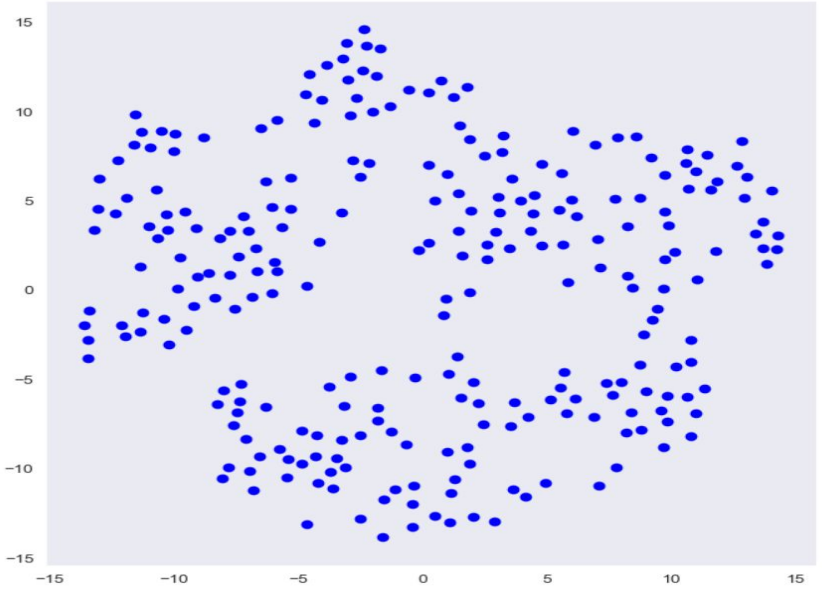
- Created python scripts to prepare data for kaldi
- Created MFCC and VAD for every utterance (Part of pipeline)
- Used pre-defined shell script to extract x-vector on already trained model on voxceleb dataset (Transfer Learning)
- Understood working of various important utils shell scripts in kaldi
- Experimented with 3 speakers and extracted their x-vectors

Speaker Embeddings

Speaker-1



Speaker-2



Schedule Reminder

- X-vector from kaldi
- DNN First Model
- Draft of the report

- Evaluation
- Complete report

- Report
- Defense

https://gitlab.com/prerakshrivastava/asr-sv

References

- Snyder, David, Daniel Garcia-Romero, Gregory Sell, Daniel Povey and Sanjeev Khudanpur. “X-Vectors: Robust DNN Embeddings for Speaker Recognition.” *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2018): 5329-5333.
- Snyder, David, Daniel Garcia-Romero, Daniel Povey and Sanjeev Khudanpur. “Deep Neural Network Embeddings for Text-Independent Speaker Verification.” INTERSPEECH (2017).
- https://kaldi-asr.org/doc/

Thank you!

Merci pour votre attention!