Abstract

In this project we investigate Kernel Methods for sequence analysis. We concentrate on two major machine learning tasks for sequence analysis: Supervised Learning and Deep Learning. The first one concentrates on learning hidden structures in the data whereas the second task concentrates on classification. Two crucial problems in speech processing are explored in this context – Speaker Diarization (Unsupervised learning) and Automatic Speech Recognition (ASR). Results show that these alternative processing techniques have significant potential.

Introduction

- Traditionally HMMs are used to handle the temporal variability in sequences.
- In Speech processing HMMs are widely used for both supervised and unsupervised paradigms such as Automatic Speech Recognition (ASR), Speaker Diarization.
- In this work, we investigate alternatives for the traditional HMM based sequence analysis.

Speaker Diarization

- Speaker diarization (who spoke when) involves unsupervised learning and model selection.

Experiments and Results

- Performed all experiments on standard NIST RT06 eval data for “Meeting Recognition Diarization”.

Automatic Speech Recognition (ASR)

- A phoneme recognition task is investigated in the context of ASR.

ASR

- Involves converting input speech into sequence of phonemes.

Figure 2: Phoneme recognition includes recognizing the input speech into corresponding sequence of phonemes.

Figure 3: The Resulting system has comparable performance to the baseline.

Training and decoding times with the proposed system are higher than the baseline.

Employing the entire training data might lead to improved accuracies. This calls for the need to develop highly scalable classification algorithms (e.g., [6]).

Performance of the system largely depends on the applicability and parameters of the employed interpolation kernel. Currently, 8 piece-wise linear basis functions were used in the interpolation kernel. Higher number as well as higher degree polynomial basis interpolation might lead to improved accuracies.

Currently each classifier is trained independently. A better way, would be to build such classifiers in an inter-dependent way (e.g., [7]).

References