UNIVERSITY OF CAMBRIDGE

Speaker Diarisation using 2D Self-attentive Combination of Embeddings

Guangzhi Sun, Chao Zhang and Phil Woodland

May 16, 2019

Cambridge University Engineering Department

Contents

- Introduction and motivation.
- Model overview and self-attentive structure
- 2D self-attentive combination approaches
- Modified penalty term
- Experiments and results
- Conclusions

Introduction

Speaker Diarisation: Who Spoke When

- Segmenting audio into speaker-homogeneous intervals.
- Clustering them into groups corresponding to the same speaker

Importance of Speaker Embeddings

- A fixed-length vector representing the speaker of each interval
- Clustering is performed on speaker embeddings
- The use of embeddings helps other speech and language tasks

Types of Speaker Embeddings

- i-vectors: Factor analysis in the total variability space
- d-vectors: Embeddings extracted using deep neural networks

Introduction

Objectives of Model Combination

- Single networks have different strengths and weaknesses
- Take advantage of the complementarity among embeddings

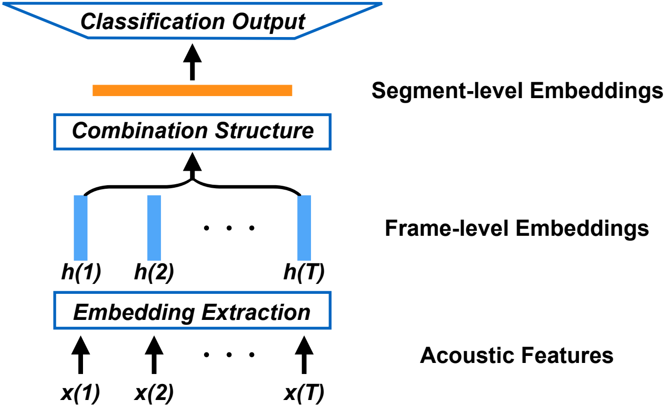
The Advantages of Multi-head Self-attentive Structure

- Dynamic combinations depending on the input
- Multiple annotation vectors to extract diverse characteristics

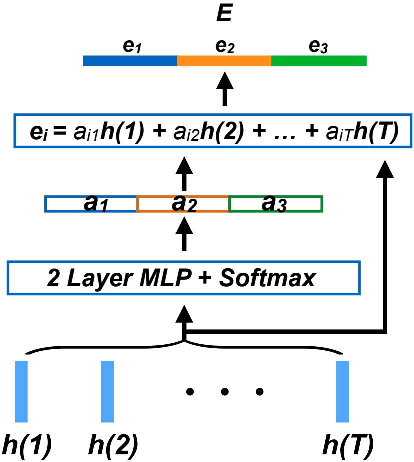
Proposed Methods

- 2D self-attentive combination across time and systems
- Modified penalty term to produce diverse annotation vectors

Model Overview



Self-attentive Layer Structure



Multi-head Output

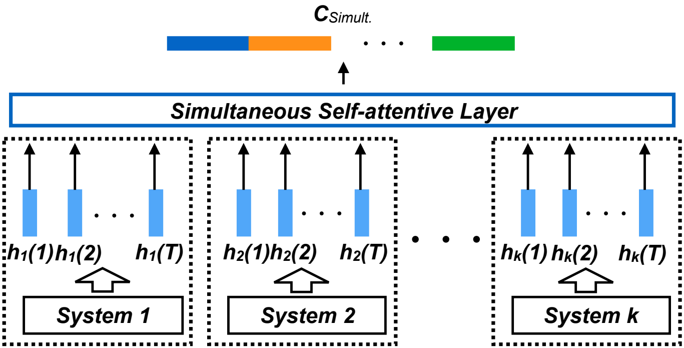
each head produced by a combination with one annotation vector

Annotation Vectors

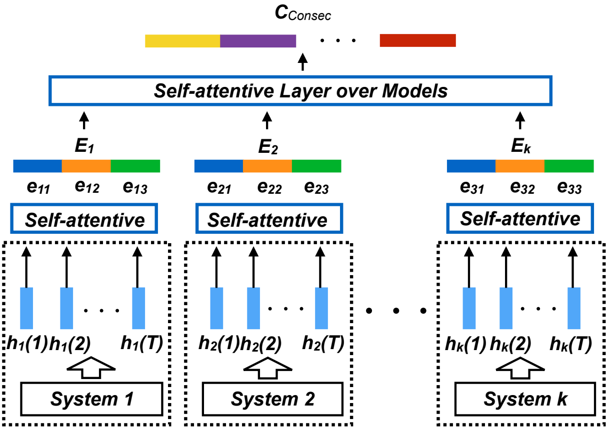
A set of linear combination weights

Frame-level Embeddings

Simultaneous Combination Architecture



Consecutive Combination Architecture



Two Types of the Second Combination Stage

Type 1 combination

- Weighted average of the segment-level embeddings, $\mathbf{E}_i$.
- Multiple output heads from the same system share the same weight in each annotation vector.

Type 2 combination

- Weighted average of the heads in the embeddings, $\mathbf{e}_{ij}$.
- Different heads from the same system may have different weight in the annotation vector.

The Modified Penalty Term

Original Definition

$$P = \mu\left(\sum_{i=1}^{h}(\boldsymbol{a_i}^T\boldsymbol{a_i} - 1)^2 + \sum_{i,j,i\neq j}^{h}(\boldsymbol{a_i}^T\boldsymbol{a_j})^2\right),$$

Penalty Term Functionality

- It is to be minimised together with the cross-entropy loss function.
- The first term forces each annotation vector to be one-hot.
- The second forces different annotation vectors to be orthogonal.

The Modified Penalty Term

Why to Adopt the Modification

- The penalty term was originally designed for sentence embedding extraction. Focusing on as few words as possible.
- Unweighted mean of frame-level embeddings showed its ability to capture speaker characteristics.

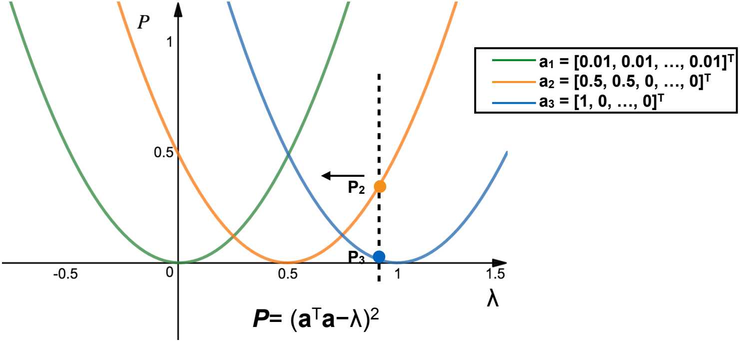
Modified Term

$$P = \mu \left(\sum_{i=1}^{h} (\mathbf{a_i}^T \mathbf{a_i} - \lambda)^2 + \sum_{i,j,i \neq j}^{h} (\mathbf{a_i}^T \mathbf{a_j})^2 \right),$$

where λ's are a set of hyper-parameters that controls the smoothness of the annotation vectors.

Penalty Term Modification

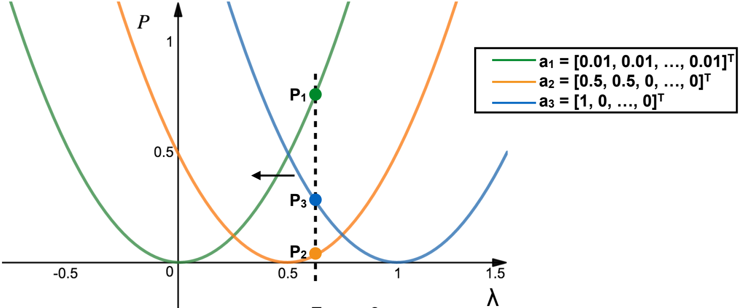
Shift of the Optimal Point with Different Diagonal Value λ



$$P = (\mathbf{a}^{\mathsf{T}}\mathbf{a} - \lambda)^2$$

Penalty Term Modification

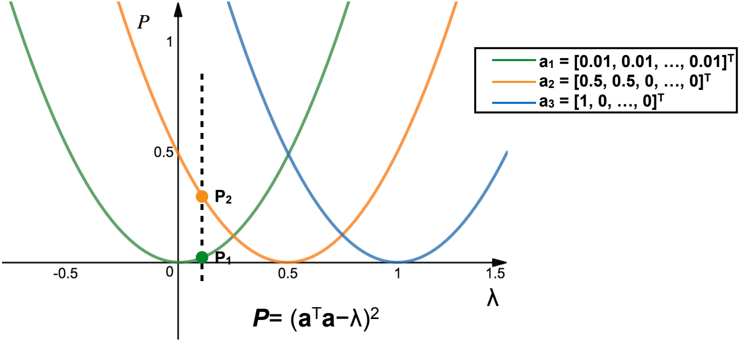
Shift of the Optimal Point with Different Diagonal Value λ



$$P = (\mathbf{a}^T\mathbf{a} - \lambda)^2$$

Penalty Term Modification

Shift of the Optimal Point with Different Diagonal Value λ



$$P = (\mathbf{a}^T\mathbf{a} - \lambda)^2$$

Experimental Setup

Data

- The Augmented Multiparty Interaction (AMI) meeting corpus.

	Meetings	Speakers
Train	135	149
Dev	14	17 (4 seen in Train)
Eval	12	12 (0 seen in Train)

Systems for Combination (k=2)

- Time-delay Neural Network (TDNN).
- High-order Recurrent Neural Network (HORNN).

Experimental Setup

Diarisation Pipeline

- Implemented with HTK 3.5.1 and PyHTK
- 40d filter bank features.
- 2s sliding segment with 1s overlap is used.
- Segment-level embeddings clustered using spectral clustering.
- Choose the mode among the segments in each utterance.
- Report Speaker Error Rate (SER) on dev and eval sets.

Baseline Systems

- Statistical pooling layer which calculates the mean and standard deviation across frame-level embeddings.

Experimental Results

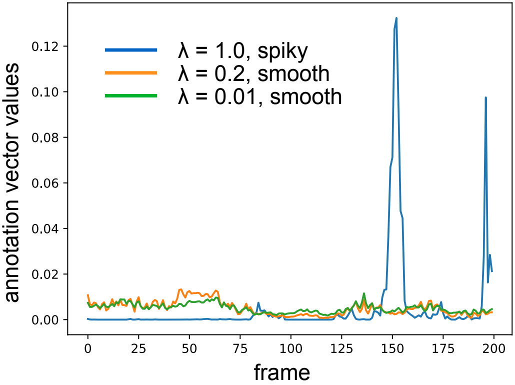
Speaker Error Rate for Separate Systems

	Dataset	Mean+std. deviation	Attention (original)	Attention (modified)
HORNN	Dev	21.0%	16.7%	**13.4%**
	Eval	23.7%	20.6%	**16.0%**
TDNN	Dev	17.5%	15.0%	**13.4%**
	Eval	19.2%	15.0%	**14.8%**

- 21% relative SER reduction in HORNN and 6% relative SER reduction in TDNN by introducing the modified penalty term.

Experimental Results

Effects of the Modified Penalty Term



Experimental Results

Comparisons of Different Combination Methods

Systems	#Params.	Dev	Eval
d-vector TDNN	1.8M	13.4%	14.8%
d-vector HORNN	0.3M	13.4%	16.0%
c-vector Simult.	2.0M	12.7%	16.3%
c-vector Consec. 1	2.5M	13.2%	13.5%
c-vector Consec. 2	2.0M	**12.2%**	**13.0%**

- A further 10% relative SER reduction was found using the second type of the consecutive combination.

Conclusions

Main Contributions Include

- A novel embedding extraction approach using a multi-head 2D self-attentive structure.
- A modified penalisation term to increase the diversity among the multi-head d-vectors.
- The modified penalty term achieved a 21% rel. SER reduction for HORNN system and a 6% rel. SER reduction for TDNN system.
- A further 10% rel. SER reduction was achieved by using 2D consecutive combination method.

Thanks for listening!