ODESSA/PLUMCOT at Albayzin Multimodal Diarization Challenge 2018

Benjamin MAURICE [1], *Hervé BREDIN* [1], *Ruiqing YIN* [1], *Jose PATINO* [3], *Héctor DELGADO* [3], *Claude BARRAS* [2], *Nicholas EVANS* [3], *Camille GUINAUDEAU* [2]

[1] LIMSI, CNRS, Université Paris-Saclay, Orsay, France
[2] LIMSI, CNRS, Univ. Paris-Sud, Université Paris-Saclay, Orsay, France
[3] EURECOM, Sophia-Antipolis, France

[12]firstname.lastname@limsi.fr, [3]firstname.lastname@eurecom.fr

Abstract

This paper describes ODESSA and PLUMCOT submissions to Albayzin Multimodal Diarization Challenge 2018. Given a list of people to recognize (alongside image and short video samples of those people), the task consists in jointly answering the two questions "who speaks when?" and "who appears when?". Both consortia submitted 3 runs (1 primary and 2 contrastive) based on the same underlying monomodal neural technologies: neural speaker segmentation, neural speaker embeddings, neural face embeddings, and neural talking-face detection. Our submissions aim at showing that face clustering and recognition can (hopefully) help to improve speaker diarization.
Index Terms: multimodal speaker diarization, face clustering

1. Introduction

This paper describes ODESSA[1] and PLUMCOT[2] submissions to Albayzin Multimodal Diarization Challenge 2018 [1]. Given a collection of broadcast news TV recordings and a list of people to recognize (alongside image and short video samples of those people), the task consists in jointly answering the two questions "who speaks when?" and "who appears when?".

While ODESSA submissions are made of the simple juxtaposition of two monomodal system (audio-only speaker diarizaton on one side, visual-only face recognition on the other side), PLUMCOT runs aimed at showing that face clustering and recognition can help to improve speaker diarization (and vice-versa).

Figure 1 provides an overview of PLUMCOT multimodal pipelines. The upper part corresponds to the face recognition pipeline described in details in Section 2. The lower part corresponds to the speaker diarization pipeline further described in Section 3. Section 4 describes the proposed multimodal approaches, depicted by vertical arrows between audio and visual modalities in Figure 1.

Instead of optimizing audio and visual pipelines separately, we propose to tune the whole set of hyper-parameters jointly with respect to the official evaluation metric. This is described in Section 5. Following sections 7 and 8 introduce the experimental protocol and results on the development set[3].

2. Face clustering and recognition

This section describes the building blocks of the "*face*" part of our runs. They all rely on the *pyannote.video* toolkit introduced in [2]. It mostly consists of three sub-modules: face tracking, neural face embedding, and face clustering.

[1] ANR/SNF project ANR-15-CE39-0010
[2] ANR/DFG project ANR-16-CE92-0025
[3] Official results on the test set are not available yet.

2.1. Face tracking

After an initial step of shot boundary detection using optical flow and *displaced frame difference* [3], face tracking-by-detection is applied within each shot using a detector based on histogram of oriented gradients [4] and the correlation tracker proposed in [5]. More precisely, face detection is applied every frame (so every 40ms), and tracking is performed in both forward and backward directions.

2.2. Face embedding

Each face track is then processed using the ResNet network with 29 convolutional layers [6] available in the *dlib* machine learning toolkit [7]. This network was trained on both FaceScrub [8] and VGG-Face [9] datasets to project each face into a 128-dimensional Euclidean space, in which faces from the same person are expected to be close to each other. Each face track is described by its average face embedding x_{face}.

2.3. Face clustering

Face tracks are grouped together using agglomerative clustering. Clustering is initialized with one cluster per face track (described by the average face embedding introduced in the previous section, $x_k = x_{\text{face}}$). Then, the following process is repeated iteratively until a stopping criterion is reached:

- find the two most similar clusters (i and j) according to the Euclidean distance $d_{ij} = d(x_i, x_j)$ between their embedding x_i and x_j;
- compute the embedding x of the newly formed cluster as the weighted average of the embedding x_i and x_j of the two merged clusters i and j:

$$x = \frac{n_i \cdot x_i + n_j \cdot x_j}{n_i + n_j} \quad (1)$$

$$n = n_i + n_j \quad (2)$$

  where n_i and n_j are the total number of face tracks belonging to clusters i and j respectively.

This agglomerative process stops when d_{ij} is greater than a tunable threshold θ_{fc}.

2.4. Face recognition

While a perfect face clustering should lead to a perfect (visual) diarization error rate, the actual metric used in the Albayzin Multimodal Diarization Challenge assumes that only a limited list of T target persons should be returned by the system. Enrolment data is provided for each target, containing approximately 10 pictures and one short video sample showing their face.

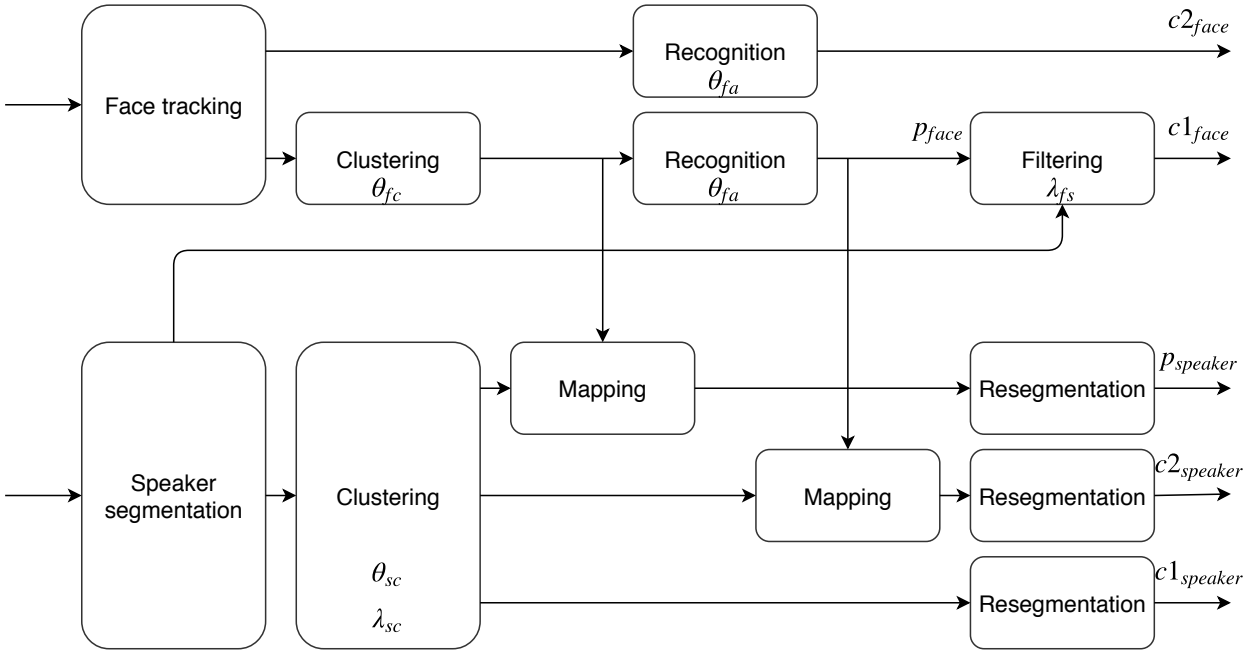

Figure 1: *Pipeline used for PLUMCOT submissions (p = primary, c = contrastive).* $\theta_\bullet$ *and* $\lambda_\bullet$ *are jointly-optimized hyper-parameters.*

One can then assign each cluster to the closest target t^* by comparing the cluster embedding x to each target embedding x_t computed as the average embedding extract from all their enrolment pictures:

$$t^* = \text{argmin}_{t \in \{1 \dots T\}} d(x, x_t) \quad (3)$$

In case $d(x, x_{t^*})$ is greater than a tunable threshold θ_{fa}, the cluster is decided to be a non-target person and therefore not returned by the system. This approach is denoted by p_{face} in Figure 1 and constitutes the "*face*" part of PLUMCOT primary submission and of all ODESSA submissions.

A variant of this cluster-wise face recognition approach is to perform recognition directly at the level of face tracks (i.e. without prior face clustering). This variant is denoted by $c2_{\text{face}}$ in Figure 1 and constitutes the "*face*" part of PLUMCOT contrastive submission #2.

3. Speaker diarization

This section describes the building blocks of the "*speaker*" part of PLUMCOT runs. They all rely on the speaker diarization approach introduced in [10].

3.1. Speech turn segmentation

All submission share a same speech activity detection (SAD) system proposed in [11]. SAD is modeled as a supervised binary classification task (speech vs. non-speech), and addressed as frame-wise sequence labeling tasks using bi-directional LSTM on top of MFCC features. For SCD, two systems were explored: the first one named uniform segmentation which splits the speech parts into 1s segments, the second one using the system proposed by [12]. Similar to SAD, SCD is modeled as a supervised binary sequence labeling task (change vs. non-change).

3.2. Speaker embedding

The embedding architecture used is the one introduced in [2] and further improved in [13]. In the embedding space, using the triplet loss paradigm, two sequences $\mathbf{x}_i$ and $\mathbf{x}_j$ of the same speaker (resp. two different speakers) are expected to be close to (resp. far from) each other according to their angular distance. The embeddings are trained on the Voxceleb corpus.

3.3. Speech turn clustering

As proposed in [10], we use Affinity propagation (AP) algorithm [14] to perform clustering of speech turns. AP does not require a prior choice of the number of clusters contrary to other clustering methods. All speech segments are potential cluster centers (exemplars). Taking as input the pair-wise similarities between all pairs of speech segments, AP will select the exemplars and associate all other speech segments to an exemplar. In our case, the similarity between i^{th} and j^{th} speech segments is the negative angular distance between their embeddings. AP has two hyper-parameters: preference θ_{sc} and damping factor λ_{sc}.

3.4. Re-segmentation

A final re-segmentation step is performed to refine time boundaries of the segments generated in the clustering step. It uses Gaussian mixture models (GMM) to model the clusters, and maximum likelihood scoring at feature level. Since the log-likelihoods at frame level are noisy, an average smoothing within a 1s sliding window is applied to the log-likelihood curves obtained with each cluster GMM. Then, each frame is assigned to the cluster which provides the highest smoothed log-likelihood.

4. Multimodal fusion

This section describes our attempts at improving speaker diarization with face clustering, and vice versa. Those two approaches were respectively submitted as PLUMCOT primary run (4.1) and PLUMCOT first contrastive run (4.2).

4.1. Improving speaker diarization with face clustering

Let us assume that there are N speakers according to speaker diarization, and M persons according to face clustering (or recognition). Let $K \in \mathbb{R}^{N \times M}$ be the co-occurrence matrix of the output of both pipelines: K_{ij} is the overall duration in which speaker $i \in \{1 \dots N\}$ is speaking and person $j \in \{1 \dots M\}$ is visible.

The main intuition motivating this approach arises from the following observation about broadcast news videos: most of the time, the camera is pointing at the current speaker. Therefore, the proposed approach simply updates each speaker cluster by assigning them to the most co-occurring face cluster:

$$i \leftarrow \text{argmax}_{j \in \{1 \dots M\}} K_{ij} \tag{4}$$

Thanks to the joint optimization (described in Section 5) of stopping criteria for both face clustering and speaker diarization, we anticipate that this approach will *"choose"* to favour smaller (but purer) speaker clusters than the purely monomodal speaker diarization pipeline. A speaker divided into several small clusters may then be merged back together thanks to (a hopefully better) face clustering and Equation 4.

4.2. Filtering face detection with speech activity detection

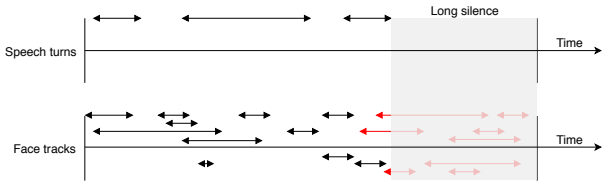

Figure 2: *Face tracks within long non-speech regions (red) are removed.*

Our face detection and tracking module tends to detect lots of non-target faces, leading to a huge amount of false alarms (e.g. in crowds, in credits at the end of TV shows, etc.). As depicted in Figure 2, we propose a very simple solution to this problem: filtering face tracks in long non-speech regions.

5. Hyper-parameters joint optimization

As mentioned in Section 4.1, the various modules of PLUMCOT runs are jointly optimized. For instance, the "*speaker*" part of PLUMCOT primary run is the combination of two modules with their own set of hyper-parameters: face clustering (θ_{fc}) and speaker diarization (θ_{sc} and λ_{fc}). Instead of tuning the former for optimal face clustering performance and the latter for optimal speaker diarization separately, the whole pipeline (including the assignment step described in Equation 4) is jointly optimized.

Practically, we use the Covariance Matrix Adaptation Evolution Strategy minimization method [15] available in the chocolate library[4] to automatically select the set of hyper-parameters that minimizes the speaker diarization error rate for "*speaker*" part and the face diarization error rate for "*face*" part.

[4] `chocolate.readthedocs.io`

6. Submissions

Figure 1 summarizes the primary and two contrastive runs of the PLUMCOT consortium. All of them have been introduced in the previous sections of this paper.

The ODESSA consortium mostly focused on the monomodal speaker diarization aspects of the task. Therefore, ODESSA submissions to the "*speaker*" part of the multimodal diarization challenge rely on the same systems used for its open-set submissions to the speaker diarization challenge: the fusion at similarity-level of various speech turn representation (such as neural embeddings and binary keys). More information can be found in [16]). All three ODESSA submissions use the same "*face*" part as PLUMCOT primary submission.

7. Experimental protocol

7.1. RTVE2018 corpus

The RTVE2018 dataset is a collection of diverse TV shows aired between 2015 and 2018 on the public Spanish National Television (RTVE). The development subset of the RTVE2018 database contains one single 2 hours show "La noche en 24H" labeled with speaker and face timestamps. It also contains 11 additional files (for a total duration of 14 hours) labeled with speaker timestamps only. Enrollment files for the target persons are also provided: they consist of a few pictures and one short video for each target.

The evaluation set contains 3 videos files of almost 2 hours each of TV shows labeled with speaker and face timestamps. However, at the time of the submission of this paper, we have no result on the test set so we are not reporting results on this test set.

7.2. Evaluation metric

The evaluation metric used for this task is the diarization error rate (DER) defined as follows:

$$DER = \frac{\text{false alarm} + \text{missed detection} + \text{confusion}}{\text{total}} \tag{5}$$

where false alarm is the duration of non-speech incorrectly classified as speech, missed detection is the duration of speech incorrectly classified as non-speech, confusion is the duration of speaker confusion, and total is the total duration of speech in the reference. Note that this metric does take overlapping speech into account, potentially leading to increased missed detection in case the speaker diarization system does not include an overlapping speech detection module. DER is a standard metric for evaluating and comparing speaker diarization systems but it can also be applied for face clustering by replacing speech turns by face tracks.

7.3. Implementation details

7.3.1. Face clustering and recognition

As already stated in Section 2, we use the pre-trained face detector and face embedding available in *dlib* library [7], wrapped in our *pyannote.video* toolkit[5]. All hyper-parameters of the face

[5] `github.com/pyannote/pyannote-video`

clustering and recognition pipeline are jointly optimized in order to minimize the (face) diarization error rate on the only annotated video of the RTVE2018 development set provided by the organizers of the challenge.

7.3.2. Speaker diarization

Feature extraction. All modules in the speaker diarization pipeline share the same feature extraction step: 19 MFCC coefficients (with their first and second derivatives, and the first and second derivatives of the energy) are extracted every 10ms on a 25ms windows. The only exception is the re-segmentation step that does not use any derivative.

Segmentation. Both speech activity and speaker change detection modules are trained with the Catalan broadcast news database from the 3/24 TV channel proposed for the 2010 Albayzin Audio Segmentation Evaluation [17]. We use the exact same configuration as the one described in [10]: stacked bi-directional LSTMs and multi-layer perceptron on 3.2s sliding windows.

Speaker embedding. Speaker embeddings are trained using VoxCeleb1 dataset [18]. We use the exact same architecture as the one used in [13] (stacked bi-directional LSTMs on a 3s window) and the training process introduced in [19] (triplet loss with angular distance).

Speaker diarization pipeline. Once every module is trained, hyper-parameters of the speaker diarization pipeline are jointly optimized in order to minimize the diarization error rate on the development set (dev2) of RTVE2018 corpus provided by the organizers of the challenge.

8. Results and discussion

Table 1 summarizes the performance of each submission on the development set. Official results on the test set were not available at the time of writing the paper.

Consortium	Run	Speaker	Face
PLUMCOT	primary	6.86	28.15
	contrastive 1	10.59	28.15
	contrastive 2	10.68	31.01
ODESSA	primary	7.21	28.15
	contrastive 1	9.29	28.15
	contrastive 2	11.46	28.15

Table 1: *Diarization error rate on the development set*

Comparing "*speaker*" parts of PLUMCOT primary run (DER = 6.86%) and constrative run #1 (DER = 10.59%) shows that speaker diarization can be greatly improved when guided by face clustering: this amounts to a relative improvement of 35%. Face clustering also helps significantly for face recognition: it is improved from DER = 31.01% for track-wise face recognition ($c2_{\text{face}}$) to DER = 28.15% for cluster-wise face recognition (p_{face}).

There are no difference between face primary run and constrative run #1 maybe because during the long silence founded faces were already deleted with the recognition threshold θ_{fa} with the enrollment data.

While cluster-wise face recognition (DER = 28.15%, p_{face}) is better than raw face clustering (DER = 46.02%, not shown in Table 1) for the "*face*" part, the latter does lead to better "*speaker*" performance than the former when jointly optimized with the speaker diarization pipeline (p_{speaker} gets DER = 6.86% while $c2_{\text{speaker}}$ only gets DER = 10.68%). This shows the benefit of the joint optimization of hyper-parameters: a better "*face*" system does not necessarily lead to a better multimodal "*speaker*" pipeline.

As described in details in [16], ODESSA "*speaker*" primary run is the combination at similarity level of three different representations (x-vector trained on NIST SRE data, triplet loss embedding trained on VoxCeleb and binary key). This complex system reaches a performance of DER = 7.21% which is still below the simpler multimodal PLUMCOT primary run (that combines triplet loss speaker embedding and neural face embedding) with DER = 6.86%. One could hope that combining both approaches would help us get even closer to perfect diarization.

9. Conclusion and future work

We have conducted experiments on monomodal face clustering and speaker diarization and shown an improvement of the results when we combine them into a multimodal approach. It has also been shown that combining two monomodal approaches tuned separately does not automatically lead to the best results: one should rather tune them jointly using a global optimization process.

While results of the multimodal approaches are promising, there is still room for improvement. In particular, we plan to investigate the use of the talking-face detection approach introduced in [20] to improve the module in charge of mapping face clusters with speaker clusters.

Finally, we would like to highlight the fact that the code for most monomodal building blocks is available for other researchers to use[6][7].

10. Acknowledgements

This work was partly supported by ANR through the ODESSA (ANR-15-CE39-0010) and PLUMCOT (ANR-16-CE92-0025) projects.

11. References

[1] E. Lleida, A. Ortega, A. Miguel, V. Bazán, C. Pérez, M. Zotano, and A. de Prada, "Albayzin evaluation: Iberspeech-rtve 2018 multimodal diarization challenge," 06 2018. [Online]. Available: http://catedrartve.unizar.es/reto2018/EvalPlan-Multimodal-v1.3.pdf

[2] H. Bredin and G. Gelly, "Improving speaker diarization of tv series using talking-face detection and clustering," in *Proceedings of the 2016 ACM on Multimedia Conference*. ACM, 2016, pp. 157–161.

[3] Y. Yusoff, W. Christmas, and J. Kittler, "A study on automatic shot change detection," in *European Conference on Multimedia Applications, Services, and Techniques*. Springer, 1998, pp. 177–189.

[4] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 1. IEEE, 2005, pp. 886–893.

[5] M. Danelljan, G. Häger, F. Khan, and M. Felsberg, "Accurate scale estimation for robust visual tracking," in *British Machine Vision Conference, Nottingham, September 1-5, 2014*. BMVA Press, 2014.

[6] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[6] github.com/pyannote/pyannote-audio

[7] github.com/pyannote/pyannote-video

[7] D. E. King, "Dlib-ml: A machine learning toolkit," *Journal of Machine Learning Research*, vol. 10, no. Jul, pp. 1755–1758, 2009.

[8] H.-W. Ng and S. Winkler, "A data-driven approach to cleaning large face datasets," in *Image Processing (ICIP), 2014 IEEE International Conference on*. IEEE, 2014, pp. 343–347.

[9] O. M. Parkhi, A. Vedaldi, A. Zisserman *et al.*, "Deep face recognition." in *BMVC*, vol. 1, no. 3, 2015, p. 6.

[10] R. Yin, H. Bredin, and C. Barras, "Neural Speech Turn Segmentation and Affinity Propagation for Speaker Diarization," in *19th Annual Conference of the International Speech Communication Association, Interspeech 2018*, Hyderabad, India, September 2018.

[11] G. Gelly and J.-L. Gauvain, "Minimum Word Error Training of RNN-based Voice Activity Detection." in *186th Annual Conference of the International Speech Communication Association, Interspeech 2015*, Dresden, Germany, September 2015, pp. 2650–2654.

[12] R. Yin, H. Bredin, and C. Barras, "Speaker Change Detection in Broadcast TV using Bidirectional Long Short-Term Memory Networks," in *18th Annual Conference of the International Speech Communication Association, Interspeech 2017*, Stockholm, Sweden, August 2017.

[13] G. Wisniewksi, H. Bredin, G. Gelly, and C. Barras, "Combining speaker turn embedding and incremental structure prediction for low-latency speaker diarization," in *18th Annual Conference of the International Speech Communication Association, Interspeech 2017*, Stockholm, Sweden, September 2017.

[14] B. J. Frey and D. Dueck, "Clustering by Passing Messages Between Data Points," *science*, vol. 315, no. 5814, pp. 972–976, 2007.

[15] C. Igel, N. Hansen, and S. Roth, "Covariance matrix adaptation for multi-objective optimization," *Evolutionary Computation*, vol. 15, no. 1, pp. 1–28, 2007.

[16] J. Patino, H. Delgado, R. Yin, H. Bredin, C. Barras, and N. Evans, "ODESSA at Albayzin Speaker Diarization Challenge 2018," in *IberSPEECH*, Barcelona, Spain, November 2018.

[17] A. Ortega, D. Castan, A. Miguel, and E. Lleida, "The albayzin 2012 audio segmentation evaluation." Iberspeech 2012, 2012.

[18] A. Nagrani, J. S. Chung, and A. Zisserman, "Voxceleb: a large-scale speaker identification dataset," in *18th Annual Conference of the International Speech Communication Association, Interspeech 2017*, Stockholm, Sweden, September 2017.

[19] H. Bredin, "TristouNet: Triplet Loss for Speaker Turn Embedding," in *ICASSP 2017, IEEE International Conference on Acoustics, Speech, and Signal Processing*, New Orleans, USA, March 2017.

[20] J. S. Chung and A. Zisserman, "Out of time: automated lip sync in the wild," in *Workshop on Multi-view Lip-reading, ACCV*, 2016.