Disentangled dimensionality reduction for noise-robust speaker diarisation

You Jin Kim\textsuperscript{1,}\textsuperscript{*}, Hee-Soo Heo\textsuperscript{1,}\textsuperscript{*}, Jee-woon Jung\textsuperscript{1,}\textsuperscript{*}, Youngki Kwon\textsuperscript{1}, Bong-Jin Lee\textsuperscript{1}, Joon Son Chung\textsuperscript{2}

\textsuperscript{1}Naver Corporation, South Korea
\textsuperscript{2}Korea Advanced Institute of Science and Technology, South Korea

youjin.kim117@navercorp.com

Abstract

The objective of this work is to train noise-robust speaker embeddings adapted for speaker diarisation. Speaker embeddings play a crucial role in the performance of diarisation systems, but they often capture spurious information such as noise and reverberation, adversely affecting performance. Our previous work has proposed an auto-encoder-based dimensionality reduction module to help remove the redundant information. However, they do not explicitly separate such information and have also been found to be sensitive to hyper-parameter values. To this end, we propose two contributions to overcome these issues: (i) a novel dimensionality reduction framework that can disentangle spurious information from the speaker embeddings; (ii) the use of a speech/non-speech indicator to prevent the speaker code from representing the background noise. Through a range of experiments conducted on four different datasets, our approach consistently demonstrates the state-of-the-art performance among models without system fusion.

Index Terms: speaker diarisation, speaker embeddings, noise-robust

1. Introduction

Speaker diarisation is an interesting but challenging problem. The ability to determine “who spoke when” provides important context in speech transcription tasks, such as meeting transcription and video subtitling. One of the main challenges in speaker diarisation involves the task of clustering speech into an unknown number of speakers. The difficulty is augmented by the challenging environmental characteristics, such as background noise.

There are two main approaches to solve this challenging problem in previous literature: conventional module-based [1–3] and end-to-end [4–10]. The former “divides-and-conquers” speaker diarisation into several sub-tasks. The exact configuration differs from system to system, but in general they consist of speech activity detection (SAD), embedding extraction and clustering [1–13]. The latter directly segments audio recordings into homogeneous speaker regions using deep neural networks [8,14–16]. However, current end-to-end approaches have been reported to be strongly overfitted to the environments that they are trained for, not generalised to diverse real-world conditions. Therefore, the winning entries to recent diarisation challenges [17–19] either exploits the former approach or fuses both approaches.

The performance of the conventional speaker diarisation system which consists of multiple modules, is highly dependent on the ability to cluster the speaker embedding. Our recent work has proposed a number of methods to adapt the speaker embedding for speaker diarisation [20]. Among such proposals, the dimensionality reduction (DR) module utilised an auto-encoder (AE) trained in an unsupervised manner, and projected speaker embeddings to a low-dimensional code (e.g., 256 to 20), adapting towards each session. Speaker embeddings in diarisation tasks are only required to discriminate a small number of speakers, compared to thousands in case of verification. Therefore, finding a low-dimensional latent space effectively reduced unnecessary background noise and showed a potential in this line of researches.

However, we empirically found that the effectiveness of our DR module varies from session to session. When the AE is trained independently for each session, we adopt a fixed code dimensionality, whereas we assume that the optimal code dimensionality may differ in each session depending on two factors: (i) the number of speakers and (ii) the duration. If the dimensionality is too small, the information required for speaker diarisation in the code becomes insufficient, resulting in performance degradation. In contrast, the excessive dimensionality may cause unnecessary information (e.g., background noise) to reside in the code [21]. Furthermore, the existing DR module trains the AE without distinction of speech or non-speech, potentially enforcing the projected embedding to also represent background noise as well as speaker identity [22]. The focus of this work will therefore be on mitigating the limitations of the existing DR module, and improving to be less hyper-parameter-dependent.

We propose two additional improvements upon the existing DR module to accomplish the goal. First, we extend the AE architecture by adding another code whereby the two codes each stand to represent speaker identity (“speaker code”) and other irrelevant information (“noise code”), respectively (Section 3.1). Employing two codes, the proposed method excludes noise-relevant factors from the speaker code. Second, we introduce an “indicator” to the DR module which represents whether the input is extracted from a speech or a non-speech segments (Section 3.2). Training with this indicator would ideally force the speaker code to be empty for speaker embeddings from non-speech segments, and therefore prevent the speaker code from representing the background noise.

We evaluate the effectiveness of the proposed methods on a range of datasets, on which we show the state-of-the-art performance (Section 4). In addition, we present additional analysis that our proposed approaches result in a less hyper-parameter-dependent module (Section 5.2).

2. Speaker Diarisation Pipeline

In this section, we introduce the overall pipeline of our speaker diarisation system, which consists of SAD, speaker embedding extraction, feature enhancement, and clustering modules. We omit explanation of SAD because the scope of this work only includes the scenario with a reference SAD. However, our frame-
Speaker labels are derived using a k-means clustering algorithm on the spectral embeddings.

3. Disentangled dimensionality reduction with indicator

We propose a new model referred to as DDRI (Figure 2-(b)), extending the original DR (Figure 2-(a)) with two proposals: (i) we present another code with the existing code in parallel, resulting in different codes; (ii) we adopt an indicator denoting whether the speaker embedding includes speakers’ voice.

3.1. Embedding disentanglement

In the original DR module, an input is projected into a low-dimensional code and then reconstructed. During this process, the noise factor is inevitably entangled in the code because noise is also required to reconstruct the original input [29, 30]. The noise factor entangled in the code may disturb speaker clustering, as noise may be consistent across different speakers’ identities. To mitigate this potential threat, we propose to disentangle the noise factor. We divide the latent space into two, and force them to represent speaker-relevant (speaker code) and irrelevant information (noise code) respectively. We apply dropout only to the noise code, making the neglectful information flow to it. This is a frequently used technique for disentanglement [31], where it has been reported that it makes essential information for reconstruction to be gathered in the code where no dropout exists. We concatenate the two codes and feed it to the decoder. After the training is complete, only the speaker code is used for subsequent clustering step, discarding the noise code.

3.2. Indicator

Using two kinds of codes opens a new potential by discarding speaker-irrelevant information from the speaker code, however, the behaviour of the AE becomes more complicated. The speaker code should primarily represent input embeddings extracted from speech segments. On the other hand, the noise code should mainly represent input embeddings from non-speech segments. To enable this ideal scenario, the AE is required to distinguish whether an input is from speech.

We further introduce an indicator to the proposed DDRI module, assuming that the DDRI can leverage this information to decide which code to be more utilised. The indicator takes the form of a vector which has a dimensionality identical to the input embedding and is added element-wisely to the input embedding, similar to the positional encoding [32]. Concretely, we adopt two indicator vectors: one for the speech embedding and the other for the non-speech embedding. Note that since the SAD is already included in the speaker diarisation pipeline (either system or reference) and precedes the speaker embedding extraction step, we can utilise SAD results at no additional cost.

4. Experiments

We evaluate the effectiveness of the proposed methods on DIHARD and VoxConverse datasets. The datasets and the experimental details are described in the following paragraphs.

4.1. Datasets

DIHARD datasets. The DIHARD challenges publish evaluation datasets which include sessions recorded in restaurant, clin-
HARD datasets, whereas we set a do not use forgiveness collar for experiments involving the DI- 
the DR or the proposed DDRI.

SAD to precisely compare the impact of SC caused by either experiments conducted on four datasets, we use the reference primary metric. FA and MS are related to the SAD module, missed speech (MS), and speaker confusion (SC), is used as the Diarisation error rate (DER), the summation of false alarm (FA), and 58.3 seconds forgiveness collar for experiments involving the DI-

Architecture comparison between DR [20] and the proposed DDRI.

Figure 2: Architecture of DDRI. It disentangles the code into speaker code and the noise code, applying dropout and the indicator. The orange part illustrates speaker code, and the dotted part depicts noise code.

VoxConverse. It is an audio-visual speaker diarisation dataset, which consists of speech clips extracted from YouTube videos. The corpus contains overlapped speech, a large speaker pool, and diverse background conditions, including talk-shows, panel discussions, political debates, and celebrity interviews [35]. Test set version 0.0.2 is used for experiments.

4.2. Evaluation protocol

Diarisation error rate (DER), the summation of false alarm (FA), missed speech (MS), and speaker confusion (SC), is used as the primary metric. FA and MS are related to the SAD module, whereas SC to the DR or the proposed DDRI modules. For all experiments conducted on four datasets, we use the reference SAD to precisely compare the impact of SC caused by either the DR or the proposed DDRI.

We use the d-score toolkit for measuring the DER. We do not use forgiveness collar for experiments involving the DI-

HARD datasets, whereas we set a 0.25 seconds forgiveness collar for VoxConverse experiments to match the scenario with corresponding challenges.

4.3. Results

Table 1 presents the performances of the proposed methods on the four datasets compared with the baselines. We also conduct ablation studies where we exclude each proposed component to verify the effect of each component on the overall performance. Note that, since we utilise reference SAD results, FA is zero in all cases and MS corresponds to the proportion of the overlapped speech included in each dataset.

Comparison with the baselines. In all datasets, DDRI outperforms the baselines without DR module by a large margin. In the case of the DIHARD datasets, the SC error is more than halved, and in VoxConverse SC reduced by more than 30%. In

all four datasets, DDRI performs even better than the DR consistently.

Comparison with state-of-the-art systems. Experimental results on DIHARD I and II show that the proposed DDRI outperforms the winning systems of the challenges. DDRI also outperforms the best single system in DIHARD III challenge. In case of VoxConverse, the test set used in VoxSRC challenge [19] has been recently updated. Also, the majority of recent re-

searches apply a system SAD in place of a reference SAD; the VoxSRC challenge which uses VoxConverse only has scenarios that use a system SAD. Therefore, we did not compare DDRI’s performance with the systems submitted to the challenge.

Ablation studies. DDRI has two components on top of the baseline with DR, that are the noise code and the indicator. We perform ablation studies by excluding each component from the DDRI, and show how each proposal affects the performance. In all four datasets, removing each component from the DDRI, and show how each proposal affects the performance. In all four datasets, removing the noise code have a greater im-

Table 1: Results on DIHARD I, II, III, and VoxConverse datasets (DER: diarisation error rate, FA: false alarm, MS: miss, SC: speaker confusion). DR stands for dimensionality reduction, and DDRI for the proposed method with two improvements (noise code and indicator).

| Configuration | DER | FA | MS | SC |
|---------------|-----|----|----|----|
| DIHARD I      |     |    |    |    |
| Track 1 winner [36] | 23.73 | -  | -  | -  |
| Baseline w/o DR | 25.85 | 0.00 | 8.71 | 17.14 |
| Baseline w/ DR | 17.70 | 0.00 | 8.71 | 8.98 |
| DDRI w/o indicator | 17.04 | 0.00 | 8.71 | 8.33 |
| DDRI w/o noise code | 17.75 | 0.00 | 8.71 | 8.54 |
| DDRI          | 16.75 | 0.00 | 8.71 | 8.04 |
| DIHARD II     |     |    |    |    |
| Track 1 winner [37] | 18.42 | -  | -  | -  |
| Baseline w/o DR | 27.39 | 0.00 | 9.69 | 17.70 |
| Baseline w/ DR | 18.40 | 0.00 | 9.69 | 8.71 |
| DDRI w/o indicator | 17.76 | 0.00 | 9.69 | 8.08 |
| DDRI w/o noise code | 18.21 | 0.00 | 9.69 | 8.52 |
| DDRI          | 17.44 | 0.00 | 9.69 | 7.75 |
| DIHARD III    |     |    |    |    |
| Track 1 best single system [38] | 15.50 | -  | -  | -  |
| Baseline w/o DR | 20.99 | 0.00 | 9.52 | 11.47 |
| Baseline w/ DR | 15.49 | 0.00 | 9.52 | 5.97 |
| DDRI w/o indicator | 15.28 | 0.00 | 9.52 | 5.76 |
| DDRI w/o noise code | 15.32 | 0.00 | 9.52 | 5.80 |
| DDRI          | 15.05 | 0.00 | 9.52 | 5.33 |
| VoxConverse   |     |    |    |    |
| Baseline w/o DR | 5.83 | 0.00 | 1.60 | 4.23 |
| Baseline w/ DR | 4.58 | 0.00 | 1.60 | 2.98 |
| DDRI w/o indicator | 4.51 | 0.00 | 1.60 | 2.91 |
| DDRI w/o noise code | 4.55 | 0.00 | 1.60 | 2.95 |
| DDRI          | 4.45 | 0.00 | 1.60 | 2.85 |

1https://github.com/nryant/dscore
5. Further analysis

In this section, we present further analyses to show the role of each code and the strength of DDRI.

5.1. Visualisation

Figure 3 depicts the code representation of the DR and the DDRI module. Figure 3 (a) shows the code from the DR. Figure 3 (b) represents the speaker code and (c) shows the noise code of the proposed DDRI. We randomly select an audio recording with nine speakers from the DIHARD II dataset, extract codes from the audio, and visualise them using t-SNE technique [39, 40].

As shown in the figure, the proposed speaker code (b) represents nine clusters corresponding to nine speakers. On the other hand, the original code (a) shows more than nine clusters, with the codes of the most dominant speakers divided into multiple clusters. We interpret that this unexpected result is due to the change of noise information within the same speaker, and in the case of the proposed method, this additional information is represented by noise code in (c). This role of the noise code makes the speaker code in (b) have more suitable distribution for speaker diarisation.

5.2. Analysis based on the number of speakers

We present Figure 4 to show the limitation of DR module and the effectiveness of our DDRI using the three DIHARD datasets. We evaluate the performance of our baseline (DR module of [20]) and the proposed DDRI using diverse code dimensionalities. (a) shows SC of the entire sessions, (b) indicates SC of the sessions where the number of speakers is four or fewer, and (c) shows SC of the session with more than four speakers. As argued, the baseline requires low dimensionality for sessions with fewer speakers, and high dimensionality for sessions with more speakers. Performance degradation is observed, especially in (b), when the dimensionality is not ideal. In contrary, our proposed DDRI module demonstrates the stable and optimal performance regardless of the number of speakers, when dimensionality is 30 or more. As a result, this stability leads relatively higher performance improvements in the entire dataset, even though the optimal performances of the two systems in each subset do not show a significant difference.

6. Conclusion

This paper addresses a novel unsupervised disentanglement framework, which generates noise-robust speaker embeddings for speaker diarisation. Speaker embeddings are the crucial component of diarisation systems, but they often contain the unnecessary information that degrades the performance, such as background noise and reverberation. Recently proposed DR module reduces the dimensionality of the embeddings, in order to remove the spurious information. However, the effect of DR is limited, being sensitive to the code dimensionality.

To this end, we propose DDRI introducing two more techniques on top of the DR module: (i) explicit disentanglement of the spurious information from the original code; (ii) the introduction of a speech/non-speech indicator. DDRI show the state-of-the-art performance as a single system on four benchmark datasets, and ablation studies on DDRI demonstrate that both of the proposals lead to performance gains. In addition, visualising the disentangled code confirms that DDRI performs as intended.
7. References

[1] D. Garcia-Romero, D. Snyder, G. Sell, D. Povey, and A. McCree, “Speaker diarization using deep neural network embeddings,” in Proc. ICASSP, 2017.

[2] K. J. Han, S. Kim, and S. S. Narayanan, “Strategies to improve the robustness of agglomerative hierarchical clustering under data source variation for speaker diarization,” IEEE Transactions on Audio, Speech, and Language Processing, vol. 16, no. 8, 2008.

[3] Z. Huang, S. Watanabe, Y. Fujita, P. García, Y. Shao, D. Povey, and S. Khdanpur, “Speaker diarization with region proposal network,” in Proc. ICASSP, 2020.

[4] T. J. Park, N. Kanda, D. Dimitriadis, K. J. Han, S. Watanabe, and S. Narayanan, “A review of speaker diarization: Recent advances with deep learning,” arXiv preprint arXiv:2101.09624, 2021.

[5] S. Horiguchi, N. Yalta, P. Garcia, Y. Takashima, Y. Xue, D. Raj, Z. Huang, Y. Fujita, S. Watanabe, and S. Khdanpur, “The hitachi-chu dihard ii system: Competitive end-to-end neural diarization and x-vector clustering systems combined by dover-lap,” arXiv preprint arXiv:2102.01363, 2021.

[6] Y. Xue, S. Horiguchi, Y. Fujita, Y. Takashima, S. Watanabe, P. Garcia, and K. Nagamatsu, “Online end-to-end neural diarization handling overlapping speech and flexible numbers of speakers,” arXiv e-prints, 2021.

[7] S. Horiguchi, P. Garcia, Y. Fujita, S. Watanabe, and K. Nagamatsu, “End-to-end speaker diarization as post-processing,” in Proc. ICASSP, 2021.

[8] Y. Fujita, N. Kanda, S. Horiguchi, K. Nagamatsu, and S. Watanabe, “End-to-end neural speaker diarization with permutation-free objectives,” arXiv preprint arXiv:1909.03952, 2019.

[9] Y. Fujita, S. Watanabe, S. Horiguchi, Y. Xue, and K. Nagamatsu, “End-to-end neural diarization: Reformulating speaker diarization as simple multi-label classification,” arXiv preprint arXiv:2003.02966, 2020.

[10] Y. Takashima, Y. Fujita, S. Watanabe, S. Horiguchi, P. Garcia, and K. Nagamatsu, “End-to-end speaker diarization conditioned on speech activity and overlap detection,” in 2021 IEEE Spoken Language Technology Workshop (SLT), 2021.

[11] F. Landini, J. Profant, M. Diez, and L. Burget, “Bayesian hmm clustering of x-vector sequences (vbx) in speaker diarization: theory, implementation and analysis on standard tasks,” Computer Speech & Language, 2022.

[12] F. Landini, S. Wang, M. Diez, L. Burget, P. Matajka, K. Zmolkilová, L. Mošner, A. Silnova, O. Plchot, O. Novotný et al., “But system for the second dihard speech diarization challenge,” in Proc. ICASSP, 2020.

[13] Y. Kwon, H. S. Heo, J. Huh, B.-J. Lee, and J. S. Chung, “Look who’s not talking,” in Proc. SLT, 2021.

[14] S. Horiguchi, Y. Fujita, S. Watanabe, Y. Xue, and K. Nagamatsu, “End-to-end speaker diarization for an unknown number of speakers with encoder-decoder based attackers,” in Proc. Interspeech, 2020.

[15] K. Kinoshita, M. Delcroix, and N. Tawara, “Advances in integration of end-to-end neural and clustering-based diarization for real conversational speech,” in Proc. ICASSP, 2021.

[16] A. Zhang, Q. Wang, Z. Zhu, J. Paisley, and C. Wang, “Fully supervised speaker diarization,” in Proc. ICASSP, 2019.

[17] N. Ryan, P. Singh, V. Krishnamohan, R. Varma, K. Church, C. Cieri, J. Du, S. Ganapathy, and M. Liberman, “But system description for dihard speech diarization challenge 2019,” arXiv preprint arXiv:2012.01477, 2020.

[18] S. Watanabe, M. Mandel, J. Barker, E. Vincent, A. Arora, X. Chang, S. Khdanpur, V. Manohar, D. Povey, D. Raj et al., “Chime-6 challenge: Tackling multispeaker speech recognition for unsegmented recordings,” arXiv preprint arXiv:2004.09249, 2020.

[19] A. Nagrani, J. S. Chung, J. Huh, A. Brown, E. Coto, W. Xie, M. McLaren, D. A. Reynolds, and A. Zisserman, “Voxceleb 2020: The second voxceleb speaker recognition challenge,” arXiv preprint arXiv:2012.06867, 2020.

[20] Y. Kwon, J. Jung, H. Heo, Y. J. Kim, B. Lee, and J. S. Chung, “Adapting speaker embeddings for speaker diarisation,” in Proc. Interspeech, 2021.

[21] W.-C. Huang, H. Luo, H.-T. Hwang, C.-C. Lo, Y.-H. Peng, Y. Tsao, and H.-M. Wang, “Unsupervised representation disentanglement using cross domain features and adversarial learning in variational autoencoder based voice conversion,” IEEE Transactions on Emerging Topics in Computational Intelligence, vol. 4, no. 4, 2020.

[22] A. Polvak and L. Wolf, “Attention-based wavenet autoencoder for universal voice conversion,” in Proc. ICASSP, 2019.

[23] A. Nagrani, J. S. Chung, and A. Zisserman, “Voxceleb: a large-scale speaker identification dataset,” in Proc. Interspeech, 2017.

[24] J. S. Chung, A. Nagrani, and A. Zisserman, “Voxceleb2: Deep speaker recognition,” in Proc. Interspeech, 2018.

[25] V. Pratap, Q. Xu, A. Srivam, G. Synnaeve, and R. Collobert, “Mic: A large-scale multilingual dataset for speech research,” in Proc. Interspeech, 2020.

[26] X. Wu, R. He, Z. Sun, and T. Tan, “A light cnn for deep face representation with noisy labels,” IEEE Transactions on Information Forensics and Security, vol. 13, no. 11, 2018.

[27] H. Ning, M. Liu, H. Tang, and T. S. Huang, “A spectral clustering approach to speaker diarization,” in Proc. ICSP, 2006.

[28] Q. Wang, C. Downey, L. Wan, P. A. Mansfield, and I. L. Moreno, “Speaker diarization with lstm,” in Proc. ICASSP, 2018.

[29] J. S. Chung, J. Huh, and S. Mun, “Delving into voxceleb: environment invariant speaker recognition,” in Proc. Odyssey, 2019.

[30] J. Huh, H. S. Heo, J. Kang, S. Watanabe, and J. S. Chung, “Augmentation adversarial training for self-supervised speaker recognition,” in Workshop on Self-Supervised Learning for Speech and Audio Processing, NeuPS, 2020.

[31] A. Jaiswal, R. Y. Wu, W. Abd-Almageed, and P. Natarajan, “Unsupervised adversarial invariance,” Advances in neural information processing systems, 2018.

[32] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, “Attention is all you need,” Advances in neural information processing systems, 2017.

[33] N. Ryan, K. Church, C. Cieri, A. Cristia, J. Du, S. Ganapathy, and M. Liberman, “First dihard challenge evaluation plan,” 2018, tech. Rep., 2018.

[34] ——, “The second dihard diarization challenge: Dataset, task, and baselines,” arXiv preprint arXiv:1906.07839, 2019.

[35] J. S. Chung, J. Huh, A. Nagrani, T. Afouras, and A. Zisserman, “Spot the conversation: speaker diarisation in the wild,” in Proc. Interspeech, 2020.

[36] G. Sell, D. Snyder, A. McCree, D. Garcia-Romero, J. Villaïba, M. Maciejewski, V. Manohar, N. Debak, D. Povey, S. Watanabe et al., “Diarization is hard: Some experiences and lessons learned for the jhu team in the inaugural dihard challenge,” in Proc. ICASSP, 2018.

[37] F. Landini, S. Wang, M. Diez, L. Burget, P. Matějka, K. Zmolkilová, L. Mošner, O. Plchot, O. Novotný et al., “But system description for the third edition of dihard challenge 2020,” The Third DIHARD Speech Diarization Challenge, 2020.

[38] S. Liu, D. Maljovec, B. Wang, P.-T. Bremer, and V. Pasucci, “Visualizing high-dimensional data: Advances in the past decade,” IEEE transactions on visualization and computer graphics, vol. 23, no. 3, 2016.

[39] L. Van der Maaten and G. Hinton, “Visualizing data using t-sne,” Journal of machine learning research, vol. 9, no. 11, 2008.