Interspeech 2018
2-6 September 2018, Hyderabad

Diarization is Hard: Some Experiences and Lessons Learned for the JHU Team in the Inaugural DIHARD Challenge

Gregory Sell, David Snyder, Alan McCree, Daniel Garcia-Romero, Jesús Villalba, Matthew Maciejewski, Vimal Manohar, Najim Dehak, Daniel Povey, Shinji Watanabe, Sanjeev Khudanpur

Center for Language and Speech Processing & Human Language Technology Center of Excellence
Johns Hopkins University, USA

gsell@jhu.edu

Abstract

We describe in this paper the experiences of the Johns Hopkins University team during the inaugural DIHARD diarization evaluation. This new task provided microphone recordings in a variety of difficult conditions and challenged researchers to fully consider all speaker activity, without the currently typical practices of unscored collars or ignored overlapping speaker segments. This paper explores several key aspects of currently state-of-the-art diarization methods, such as training data selection, signal bandwidth for feature extraction, representations of speech segments (i-vector versus x-vector), and domain-adaptive processing. In the end, our best system clustered x-vector embeddings trained on wideband microphone data followed by Variational-Bayesian refinement, and a speech activity detector specifically trained for this task with in-domain data was found to be the best performing. After presenting these decisions and their final result, we discuss lessons learned and remaining challenges within the lens of this new approach to diarization performance measurement.

Index Terms: speaker diarization

1. Introduction

Speaker diarization is the problem of organizing a conversation into the segments spoken by the same speaker (often referred to as "who spoke when"). While diarization performance continued to improve, in recent years, individual research projects have tended to focus on specific datasets or domains (such as Callhome, AMI, or broadcast). This, at the very least, makes it difficult to compare performance, and, more problematic, could lead to divergent solutions that overfit to particular characteristics.

In response to this research rut, the inaugural DIHARD challenge was intended to provide a standard set of data drawn from diverse and challenging conditions to evaluate current system performance and provide a standard set for future diarization research. The development (dev) data released with labels during the challenge included data from ten diverse domains ranging from monologues to interviews with children to meetings to internet videos. An additional three domains were included in the evaluation (eval) data as well, though truth marks for this set were not available at time of writing, and so the effects of those additional domains remain unknown. This resulted in a highly diverse and challenging dataset for diarization.

Additionally, teams were invited to participate on two tracks. The first track followed a standard often utilized for Callhome diarization research in which oracle speech marks are known. In Track 2, however, teams were required to automatically estimate marks with a speech activity detection (SAD) algorithm.

This paper describes the submissions for the inaugural DIHARD challenge from the Johns Hopkins University (JHU) team, as well as our experiments on the path from an initial system built for Callhome diarization to our final microphone diarization system. The discussion also includes possible directions for future work, as the limited time of the challenge meant many paths were necessarily left unexplored.

2. DIHARD Challenge Experiments

Through the course of this challenge, we explored a number of system configurations. This section outlines our initial system along with the set of experiments that guided our final system design. For simplicity, the results are discussed using diarization error rate (DER) with no unscored collars and including overlapping speech, which was one of the two official metrics of the evaluation. The second metric, mutual information (MI), is not included here because the high-level conclusions are essentially the same as with DER.

2.1. Initial System

As a starting point for the challenge, we began with our existing system built for performance on Callhome [4]. In that previous work, oracle speech marks were used, and so a SAD system was needed in order to build a baseline for Track 2. For this purpose, we used a bidirectional long short-term memory (BLSTM) DNN that processed all data at 8kHz (more details on this system in Section 2.3).

In order to map speech marks to speaker marks, the initial baseline system divided the speech signal into 1.5-2 second segments with a one second hop. I-vectors [1] were extracted for each of the these segments based on 20 mel-frequency cepstral coefficients (MFCCs) at 10ms hops, scored with probabilistic linear discriminant analysis (PLDA) [2, 3] and clustered using agglomerative hierarchical clustering (AHC) with average linking of scores (instead of rescoring at every merge). Stopping criteria for the AHC was determined with unsupervised calibration and confirmed with a parameter sweep on dev. System components were trained for this initial system with data from NIST SRE '04,'05,'06, and '08, and the speech was processed at 8kHz.

Scores for the initial system for the dev and eval set on both tracks are shown in Table 2. These numbers served as our starting point (or "out of the box" solution) for the challenge.

Previous Callhome diarization research had found refining the clustering marks with a frame-level diarization system

10.21437/Interspeech.2018-1893

utilizing subspace models via Variational Bayes (VB) [1] to be highly effective [5]. However, in our initial experiments, it was found to be detrimental to performance, and several modifications were required. As a result, the VB refinement is left out of our initial system, and the modifications necessary for the improvements seen in the final submissions are described in Section 2.9.

2.2. Speaker Representation

The initial system utilized i-vectors as the speaker representation, but recent work has shown that DNN-based representations called x-vectors [6] can also be effective for diarization [7]. Since that initial report, x-vectors have improved performance significantly with multiclass discriminative training [8] and augmented training data [9].

All JHU x-vector systems for this challenge utilized Kaldi [10]. 8kHz systems used features of 20 MFCCs drawn from 23 mel bins, while 16kHz systems used 24 MFCCs drawn from 30 mel bins. For i-vector systems after the initial system described above, first-order differences (deltas) were also appended as feature inputs, and for all systems after the initial system, a sliding mean normalization was applied over a 3 second window. X-vector systems were built in accordance with previous speaker recognition work [9], except the embedding layer was limited to 128 or 256 dimension. The initial i-vector system used a 1024-component UBM and 64-dimensional subspace, while the final i-vector system increased to a 2048-component UBM and 128-dimensional subspace.

Performance for x-vectors and i-vectors was measured for a number of different training lists built from combinations of VoxCeleb [11], various broadcast news corpora distributed by LDC (primarily in English, Arabic, or Chinese), audio from European Parliament videos, Librispeech, or Mixer 4/5.

As was shown in prior work [9], x-vectors are more able to capitalize on larger quantities of training data, as their performance continues to improve with additional data, while i-vectors, on the other hand, did not improve in performance with additional data beyond VoxCeleb. In all cases, PLDA parameters were also learned from the VoxCeleb dataset. Additionally, unlike in previous Callhome work [4], we found PLDA to be more effective when trained with all same-speaker audio in the same class, as opposed to training to the combination of speaker and channel. For whitening lists, however, the best data combination was found to include VoxCeleb, Mixer 4/5, and the provided DIHARD data, and this combination was best for all systems.

2.3. Speech Activity Detection

Track 2 of the DIHARD challenge required systems to estimate their own SAD marks. Our initial SAD system utilized a BLSTM-DNN trained with the CURRENNT[2] toolkit on a subset of 8kHz telephony from Switchboard with synthetic variations such as added noise, reverberation, and low-bitrate speech coding. Input features were 13 MFCCs computed every 10ms with deltas and double-deltas appended. This system was found to perform at a miss rate of 10.2% at a false alarm of 4.6% on the DIHARD dev data. The threshold for separating speech and non-speech can be modified in order to shift the balance of misses and false alarms, but we found that misses are a preferable error in SAD systems for segment-clustering diarization, presumably because including segments of non-speech can corrupt the clustering, creating additional errors beyond the SAD mistakes. Furthermore, every second of corrected false alarms results in a second of overall error reduction, while corrected misses still need to be clustered properly to reduce final error, so it may simply be that lower false alarm rates more reliably map to improved final error.

In order to improve SAD performance on this data, we first trained a 5-layer time-delay neural network (TDNN) [12, 13] with 16kHz microphone data of audio from European Parliament videos, again with various synthetic variations to add more diversity to the training data. The benefit of a TDNN for this task is that it can incorporate a wider input context without exploding the number of parameters. In this case, each layer doubles the width, resulting in +/-320ms of input context for each frame-wise speech/nonspeech decision. However, this system performed resulted in worse performance than the initial SAD system, with a much higher miss rate of 17.4% at a similar false alarm rate of 4.8%.

But, if we instead retrained only the final sigmoid classifier layer with the provided dev data (or, when testing on dev, with a two-fold split of the dev data), the performance improved beyond the baseline (7.3% miss and 4.1% false alarm). A similar strategy using MFCC input features was also effective (7.3% miss and 6.0% false alarm), but retraining the final layer of the TDNN yielded the best performance. Taking this strategy an extra step and training separate final layers for each domain was able to provide small additional gains for dev (6.1% miss and 4.2% false alarm), but reduced performance on eval, as will be discussed in more detail in Section 2.7.

Table 1: *DER scores on the dev data comparing performance for both x-vectors and i-vectors with 8kHz or 16kHz speech. The additional bandwidth in 16kHz processing is clearly helpful in the task*

Type	Sample Rate	Track 1 DER	Track 2 DER
i-vector	8kHz	24.81	35.84
i-vector	16kHz	21.74	33.72
x-vector	8kHz	23.42	34.69
x-vector	16kHz	21.42	33.17

2.4. Signal Bandwidth

The initial diarization system utilized by the JHU team was built for the telephone recordings of Callhome, and therefore was only trained for 8kHz data. However, the DIHARD challenge data is sampled at 16kHz, and so half of the bandwidth would be ignored without retraining with 16kHz data. Results comparing performance on both data at each sampling rate can be seen in Table 1 for both x-vectors and i-vectors. In both cases, the systems were only trained with VoxCeleb, though for the x-vector training, data augmentation was also employed.

The results in Table 1 show clearly that using the full signal bandwidth provides an advantage for these systems, especially for Track 1. Based on these experiments, our research focused on utilizing 16kHz processing in both i-vector and x-vector systems.

[1] Code available at http://speech.fit.vutbr.cz/software/vb-diarization-eigenvoice-and-hmm-priors

[2] https://sourceforge.net/projects/currennt/

2.5. Signal Enhancement

Given the presence of microphone recordings from multiple noisy environments, we tested the effects of signal enhancement via mask-based speech separation. The specific algorithm used to estimate the spectral masks utilized a BLSTM-DNN trained with CHiME-3 data [14]. Several insertion points for the masking were explored: in front of test (dev) data only; in front of test and PLDA training data; in front of test, PLDA, and i-vector training data.

Overall the effects of mask-based separation were detrimental to diarization, dropping performance in one system from 22.8% DER to 28.7% when only enhancing test data, and to 24.1% when enhancing all training data as well. However, for the SEEDLINGS domain, the enhancement led to consistent gains (from 44.7% DER to 38.7% in the matched case). More analysis is required to fully understand this effect, but it might indicate that some noisier files benefitted from the estimated masks, and that a mask estimation algorithm trained with a more diverse set of data sources could be more broadly effective as a diarization front-end.

2.6. AHC Threshold

An important parameter for speaker diarization with AHC determines when the AHC merges should stop. In past work, unsupervised estimation of this threshold proved to be effective for Callhome diarization [4], but it can also be determined in a supervised fashion when labeled in-domain data is available, typically by sweeping the threshold for optimal DER performance.

For the DIHARD challenge, we found supervised thresholding to be a consistently effective approach. However, in the case of wideband i-vectors, a small gain of roughly 0.2% DER was attainable with unsupervised threshold estimation on the eval data. However, due to time constraints, this method was not used for the final submissions, which instead used supervised estimation learned on the dev set.

2.7. Domain-specific Processing

The gains found from utilizing dev data for SAD retraining or representation whitening suggested that specialized systems could yield improvements for this challenge. Continuing that logic, we tried domain-specific processing for both our SAD system and for learning the AHC threshold.

For SAD, separate final DNN layers were learned for every domain using the known labels of the dev data. A domain estimation was also learned with a simple logistic regression classifier using the mean of the features across the file. Running the SAD for the estimated domain resulted in an improvement on SAD performance for held-out dev data (improving miss/false alarm from 7.3%/4.1% to 6.1%/4.2%), but these gains did not map to eval, resulting in an overall degradation in DER performance on the baseline i-vector system (43.9% to 45.0%).

Similarly, an attempt was made to use domain-specific thresholds to stop AHC merges. In this case, the thresholds were learned on known dev labels, and a domain estimator was again trained, this time using the segment i-vectors. In this case, the threshold was defined by a linear combination of domain-specific thresholds weighted by the posterior probability for each domain in the estimation. While performance improved in held-out dev experiments, the metrics again degraded for eval.

These two domain-specific experiments modified different modules of the pipeline, varied in their domain estimation (though both performed at over 80% accuracy on held-out dev data), and differed in making hard or soft domain assignments. And yet, in both cases, performance improved on dev but degraded on eval. This outcome suggests that either the domain-specific processes are overfitting the dev data, or the presence of unseen domains in the eval data is problematic enough to overcome the improvements seen in dev. Without eval truth labels, it is difficult to know at this time, but, either way, we were unable to capitalize on domain-specific processing for the challenge.

2.8. System Fusion

Finally, we explored fusing the PLDA scores of multiple systems prior to AHC clustering. The scores themselves were fused via a weighted sum with coefficients learned to optimize performance on the dev set. After fusion, the scores were clustered in the same fashion as for an individual system. Early in the challenge, we saw worthwhile improvements from fusion, but, by the end, the system fusion prior to clustering was only yielding a small gain over our best x-vector system. Furthermore, as can be seen in Table 2, the VB refinement (discussed in the next section) essentially wiped out those gains, resulting in essentially equivalent performance between our best x-vector system and best fusion.

2.9. VB Refinement

After segment clustering (which is the outcome of all modules discussed to this point), it is often helpful to refine the mark boundaries, as the initial segmentation was likely too quantized. This step would seem to be especially important in the case of the DIHARD challenge, where the practice of unscored collars around speaker transitions was abandoned, requiring more precise labeling. However, the VB refinement previously found to be highly effective for Callhome [5] was initially detrimental to system performance.

Improving the VB refinement for this data took a few steps. First, parameters were re-learned with 16kHz microphone data (in this case, VoxCeleb). Then, the stopping criteria and optimization criteria were de-coupled and early stopping was employed. In previous work, the VB refinement was permitted to iterate until convergence. However, for this data, that leniency was found to degrade performance, and stopping earlier was found to be a better practice. In fact, permitting only one pass yielded the best results.

This development not only helped VB refinement more consistently improve performance, but it also raised interesting questions about the underlying models for microphone diarization. The fact that the optimization criteria is less connected to improved diarization suggests that there is an incorrect assumption, a possibility that warrants further analysis. The changing dynamics of far-field microphone recordings (non-stationary noise, moving sources, etc.) may violate the total variability assumption that diarization is finding a speaker in the same channel. If the channel is indeed varying throughout the recording, this would not only explain the degradation in VB refinement as compared to Callhome, but also the improvements found here by training PLDA to recognize speakers across channels (as is done for speaker recognition) instead of speakers within channels. Again, this requires more analysis, but the possibility is quite interesting and could have important implications for future diarization with microphone/far-field recordings.

Table 2 shows the performance of the final clustering systems with and without VB refinement (system details in the next section). Comparing each system with and without VB refine-

Table 2: *Dev and Eval performance measured in DER for both tracks for several JHU systems, including the final submitted systems (marked with *). Details of the initial system and final submissions are described in Sections 2.1 and 2.10, respectively*

	Track 1		Track 2	
System	Dev DER	Eval DER	Dev DER	Eval DER
All same speaker	35.97	39.01	48.69	55.93
Initial System	26.58	31.56	40.89	50.78
i-vector, no VB	21.74	28.06	33.72	40.42
x-vector, no VB	20.03	25.94	31.80	39.43
Fusion, no VB	19.54	25.50	31.79	39.00
i-vector, with VB*	19.69	25.06	31.29	37.41
x-vector, with VB*	18.20	23.73	29.84	37.29
Fusion, with VB*	18.17	23.99	30.31	37.19

ment shows that the addition is quite advantageous to performance in all conditions. The gains diminish with better clustering, but the process is clearly valuable for all cases.

2.10. Final Submissions

The aggregation of all these factors can be seen in Table 2, where performance results for the final JHU submissions are shown for both dev and eval.

The i-vector system was trained on 16kHz VoxCeleb data, while the x-vector system (which resulted in a 256-dimensional embedding) was trained on an aggregation of 16kHz data from VoxCeleb, Mixer 4/5, Librispeech, European Pariliament videos, and various broadcast news corpora. In both cases, PLDA parameters were trained with VoxCeleb, and thresholds were learned in a supervised fashion on the dev data. VB refinement followed, as described above, on all systems, and all submission systems used the TDNN SAD with the final layer retrained with dev data.

The fusion is then the combination of the i-vector system and two x-vector systems (128 dimensions and 256 dimensions), fused as described above. Interestingly, there are gains from fusion without VB refinement, but with VB refinement included, the single x-vector system is essentially equivalent to the fusion.

With all factors combined, the improvements from the initial system to the final submissions are clear in both tracks. In Track 1, our performance improved by 7.83% DER, while Track 2 performance improved by 13.59% DER, an improvement of approximately 25% relative in both cases. This process was a valuable experience in generalizing diarization to simultaneously perform in a variety of environments, and these gains show that the process was a success. At the same time, the error rates are still high (especially for Track 2) and so it is clear that there is still plenty of opportunity for the future.

3. Future Work

As the descriptions above should demonstrate, the initial experience of the JHU team for the inaugural DIHARD challenge was mostly devoted to updating existing systems to work in the challenging microphone conditions of the evaluation. And while this was an important and worthwhile process, there were many longer-term research directions that were largely left for future work.

For one, we were unable to devote resources to handling the overlapping speech that is frequently present in real conversational dynamics. Roughly 8% of the absolute error in our systems was from overlapping speech, which accounted for at least a fifth of our error in Track 2, and a third in Track 1. However, the challenge of overlapping speech is not trivial, as it will likely require a complete rethinking of the diarization process, since our current system simply does not allow for multiple speakers to be responsible for the same frame of speech. This is an important direction, but could not be addressed during the limited duration of the challenge.

Although significant gains were made during the challenge in SAD performance, this remained another source of significant error. It is also somewhat disappointing that improvements in SAD required retraining on truth marks provided with the dev data. Ideally, SAD systems should work reasonably well without requiring in-domain supervision, and this is a bar that our systems were unable to clear. A robust and widely-applicable SAD is another area of continuing research.

It is also the case that, given the nature of this evaluation, we do not yet know the details of the successes and failures of our systems beyond the overall performance, since eval truth marks were not provided to teams. As a result, an important step for future work will be to better understand the sources of error in the eval set. This is especially important for understanding the effect of the unseen domains, as well as measuring the degree of overfitting to the specific nuances of the dev data. So, understanding directions for future work is also a goal of future work, once the receipt of the truth marks allows for a deeper understanding of the shortcomings of the submitted systems.

4. Conclusions

The inaugural DIHARD challenge provided an opportunity to measure diarization performance in challenging conditions without the benefit of knowing the answers. This process was a valuable experience in rethinking systems to work in more general conditions, but also in confirming the effectiveness of our general pipeline. By the completion of the evaluation, we had trained more effective diarization systems with wideband data, learned how to make VB refinement more effective in microphone conditions, and built a single x-vector system that was essentially our best performing diarization system.

5. Acknowledgements

The authors would like to thank the organizers of the DIHARD challenge for an interesting dataset and format, and we look forward to future iterations.

6. References

[1] N. Dehak, P. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-End Factor Analysis for Speaker Verification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 4, pp. 788–98, May 2011.

[2] S. Ioffe, "Probabilistic Linear Discriminant Analysis," in *ECCV*, A. Leonardis, H. Bischof, and A. Pinz, Eds. Springer-Verlag, 2006, pp. 531–42.

[3] S. J. D. Prince and J. H. Elder, "Probabilistic Linear Discriminant Analysis for Inferences About Identity," in *IEEE International Conference on Computer Vision*, 2007, pp. 1–8.

[4] G. Sell and D. Garcia-Romero, "Speaker Diarization with PLDA I-Vector Scoring and Unsupervised Calibration," in *Proceedings of the IEEE Spoken Language Technology Workshop*, 2014.

[5] ——, "Diarization Resegmentation in the Factor Analysis Subspace," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 2015.

[6] D. Snyder, P. Ghahremani, D. Povey, D. Garcia-Romero, Y. Carmiel, and S. Khudanpur, "Deep Neural Network-based Speaker Embeddings for End-to-End Speaker Verification," in *Proceedings of the IEEE Spoken Language Technology Workshop*, 2016.

[7] D. Garcia-Romero, D. Snyder, G. Sell, D. Povey, and A. McCree, "Speaker Diarization Using Deep Neural Network Embeddings," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 2017.

[8] D. Snyder, D. Garcia-Romero, D. Povey, and S. Khudanpur, "Deep neural network embeddings for text-independent speaker verification," in *Proceedings of Interspeech*, 2017.

[9] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-Vectors: Robust DNN Embeddings for Speaker Recognition Using Data Augmentation," submitted to ICASSP 2018.

[10] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannermann, P. Motlíček, Y. Qian, P. Schwarz, J. Silovský, G. Stemmer, and K. Veselý, "The Kaldi Speech Recognition Toolkit," in *Proceedings of the IEEE Workshop on Automatic Speech Recognition and Understanding*, 2011.

[11] A. Nagrani, J. S. Chung, and A. Zisserman, "VoxCeleb: a large-scale speaker identification dataset," in *Proceedings of Interspeech*, 2017.

[12] A. Waibel, T. Hanazawa, G. Hinton, K. Shikano, and K. J. Lang, "Phoneme Recognition Using Time-Delay Neural Networks," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, no. 3, pp. 328–39, March 1989.

[13] V. Peddinti, D. Povey, and S. Khudanpur, "A time delay neural network architecture for efficient modeling of long temporal contexts," in *Proceedings of Interspeech*, 2015.

[14] H. Erdogan, J. R. Hershey, S. Watanabe, and J. Le Roux, "Phase-Sensitive and Recognition-Boosted Speech Separation Using Deep Recurrent Neural Networks," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 2015.