Improving Speaker Diarization for Low-Resourced Sarawak Malay Language Conversational Speech Corpus

Mohd Zulhafiz Rahim
Faculty of Computer Science and Information Technology
Universiti Malaysia Sarawak, Malaysia
email: mzhafiz1999@gmail.com

Sarah Samson Juan
Faculty of Computer Science and Information Technology
Universiti Malaysia Sarawak, Malaysia
email: sjsflora@unimas.my

Fitri Suraya Mohamad
Institute of Borneo Studies
Universiti Malaysia Sarawak, Malaysia
email:mfitri@unimas.my

2023 International Conference on Asian Language Processing (IALP) | 979-8-3503-3078-6/23/$31.00 ©2023 IEEE | DOI: 10.1109/IALP61005.2023.10337314

***Abstract*: Speaker diarization plays a vital role in speech transcription involving conversations as it improves the transcribed content's accuracy, comprehension, and usability. By having a speech transcription diarized, the conversation data has a more structured presentation, allowing for a variety of applications that rely on accurate speaker attribution. Even so, speaker diarization is a field that has been less explored for low-resourced languages, as current resources that have been optimized and applied in speaker diarization are mostly for more developed and well-resourced languages, such as English, Spanish or French. In this paper, we propose an approach to using pseudo-labelled speech data to perform self-training on the x-vector models to improve diarization accuracy. The proposed method uses almost 13 hours Sarawak Malay unlabeled conversational speech corpus obtained from the *Kalaka: Language Map of Malaysia* website for training, as well as 1 hour and 26 minutes of manually labeled Sarawak Malay speech data for testing and evaluation. We demonstrate how speaker diarization models can be fine-tuned with the pseudo-labeled data.**

Keywords—Speaker diarization, x-vectors, clustering, low-resource, auto-labeling, pseudo-labeling, unsupervised

I. Introduction

A. Speaker Diarization

Speaker diarization is a task to segment conversational speech into corresponding sections based on the identities of the speakers. The speech can be annotated with time boundaries and speaker ids in order to label speaker change. In simpler terms, speaker diarization determines "who spoke when?" in an audio recording. Initially, the research objective of diarization technology was to benefit automatic speech recognition (ASR) by enabling speaker adaptations on the speech transcriptions [1]. Dated in the 1990s, the earliest forms of speaker diarization was for speaker identification for dialogs between an air traffic controller (ATC) and several pilots, as well as speaker adaptation on news broadcasts [2, 3].

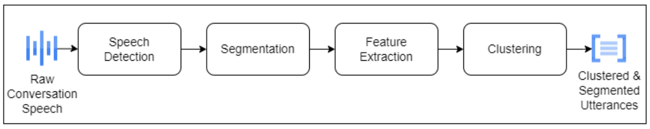

Fig. 1: Speaker Diarization Process

Fig. 1 shows the basic process of speaker diarization. The initial step is speech detection, where regions of audio containing speech are identified. Afterwards, the audio signal is partitioned into small segments based on the speaker boundaries, thereby creating a timeline or "diarization" of who spoke when. Next, feature extraction occurs, where the acoustic features are extracted for clustering. The speech segments are then grouped up into clusters based on the speaker's acoustic features that were extracted prior. The output of the speaker diarization process is a collection of segmented and clustered utterances, with each segment representing its respective speaker.

One of the biggest challenges in optimizing speaker diarization stems from the fact that it requires a large amount of data so that more accurate results can be produced. A large dataset is required as a larger amount of data increases the diversity of the training data itself, allowing the diarization model to learn the different interaction styles and speaker features, leading to a more accurate diarization process. To tackle this issue and optimize diarization, previous works on speaker diarization have implemented popular, large open-source datasets such as MUSAN [4], AMI Corpus [6], and Speaker in the Wild (SITW) [4]. While being large, these datasets also implemented data augmentation to further improve the diversity of data.

However, this problem is even more intensified while trying to fine-tune speaker diarization for low-resourced languages, such as the Sarawak Malay language, as there is a lack of labeled data that are needed for the purpose of a more accurate diarization. This is due to the sensitivity of diarization to the style of interaction as audio data collected, despite diarization itself being language-agnostic. For example, broadcast news, courtroom discussions and dinner-party conversations would have different styles and intonations of speeches, and be uniquely different for every language [21]. Furthermore, even with a sufficient amount of available raw speech data for the Sarawak Malay language, the task of manually adding the speaker change tags to the dataset to be used for model training and finetuning will prove to be laborious and costly.

This paper aims to present a methodology for leveraging raw audio speech to enhance speaker diarization in a language with limited resources. The chosen diarization approach involves implementing x-vector models through the PyAnnote toolkit. This method not only streamlines the process but also promotes efficiency by offering a mostly automated and user-friendly approach. It enables fine-tuning and diarization tasks to be conducted with minimal user supervision.

II. Related Works

A. Speaker Diarization Modelling

There are several types of speaker diarization models such as x-vector models [4], i-vector models [16], and end-to-end neural diarization models (EEND) [17]. Currently, the x-vector model is considered the state-of-the-art model as it consistently achieves the best performances for diarization compared to the other models. X-vectors are embeddings extracted from Deep Neural Networks (DNN), which consists of several layers of neural networks including convolutional layers, time delay neural network (TDNN) layers, and dense layers [19]. These embeddings were introduced by Snyder et al. as DNN embeddings as a way to provide an alternative approach for learning representations via DNNs to remove the i-vector extraction process from the pipeline entirely [16]. The approach was proposed due to the task of employing i-vector clustering for short segments of speech could be considered too cumbersome and costly for the front-end role [18]. There have been several implementations of x-vectors in studies throughout the years. The tools frequently used for the utilization of x-vector models will also be discussed for their advantages in the later parts of this section.

A recent study showed that to improve the performance of deep neural network (DNN) embeddings of x-vectors for speaker recognition, is by using data augmentation. Snyder et. al found out that data augmentation, which increases the amount and diversity of existing training data, is a lot more helpful for x-vector DNN compared to i-vector DNN because of the supervised training of the x-vector DNN [4].

The strategy implemented for data augmentation was by employing additive noises as well as reverberation. For additive noises, the MUSAN dataset was used, which contains over 900 noises, 42 hours of music from various genres and 60 hours of speech from twelve languages. For reverberation on the other hand, room impulse responses (RIR) are combined with audio. Both MUSAN and RIR datasets are available for free at http://openslr.org.

The performance evaluation of the DNN embeddings were conducted on two distinct datasets: Speakers in the Wild (SITW) and the Cantonese portion of the NIST SRE 2016 (SRE16). The end results show the x-vector system significantly outperforms i-vector baselines on SRE16 Cantonese as not only the x-vectors achieved much lower error-rates, they also only require speaker labels to train, thus ideal for domains with little transcribed speech [4]. The key takeaways from this experiment is that the augmentation strategy enhanced diversity, making x-vectors more robust for speaker recognition tasks, thus could potentially involve multi-speaker conversations.

Snyder et al. combined their previous work on DNN embeddings with x-vectors [4] and applied it to the problem of speaker recognition on multi-speaker conversations. They found out that the diarization performance substantially increases when there are multiple speakers involved, without diminishing the performance on single-speaker recordings. The authors applied the same data augmentation strategy that they used in their prior work, which is implementing additive noises and reverberations. This time, however, the interest in the topic of speaker recognition on multi-speaker conversations was taken into account. This leads to speaker diarization to be encouraged to be performed in conjunction with speaker recognition [5].

The dataset used for evaluation was SITW, which is the same dataset they used in their prior work [4]. Experiments conducted resulted in a major performance boost after the diarization backend was trained on randomly extracted three second segments from the full-length augmented recordings [6].

The same data augmentation techniques proved effective in enhancing performance for speaker diarization even in these complex contexts. This demonstrates the adaptability and scalability of applying the x-vector approach across single-speaker and multi-speaker scenarios, emphasizing the need for combined diarization and recognition in such contexts.

B. Speaker Diarization System based on X-vector

In their recent studies in 2023, Khoma et al. found out x-vector models could be used to develop a speaker identification system, as they proposed two architectures of speaker identification systems based on a combination of diarization and identification methods. Both systems were developed using the pyannote framework. The systems that were developed are identified as Architecture A, a segment-level approach that was designed by substituting the clustering module with the classification module in the basic architecture of an unsupervised diarization system, and Architecture B, a group-level approach that was designed by adding the classification module to the basic architecture of an unsupervised diarization system [6].

Four experiments were conducted to determine the supervised pyannote diarization algorithm that has the highest performance by measuring the accuracy of speaker utterance identification. The first experiment looks into the selection of distance function between vector embedding, the second experiment aims to find the best clustering and classification methods, the third experiment investigates different segmentation algorithms, and the fourth experiment examines embedding window sizes [6].

The speaker identification performance was evaluated by applying the AMI Corpus open-source audio data, which contains 100 hours of annotated and transcribed audio and video data [6]. The results achieved by Khoma et al. showcased that the group-level approach offered better identification results, but the segment-level approach provided the advantage of real-time processing.

To summarize, these two architectures highlight the adaptability, scalability as well as robustness of the x-vector model. This is due to x-vectors providing its users the options to customize the implementation of the model to be optimized for different kinds of scenarios (speaker identification, real-time processing) while still maintaining a state-of-the-art performance.

C. Tool: PyAnnote.Audio, Neural Building Blocks for Speaker Diarization

Pyannote.audio, or just pyannote for short, is an open-source speaker diarization toolkit that utilizes clustering using x-vector models and written in Python [7]. This toolkit was developed by Bradin et al. as a way to provide an alternative approach for speaker diarization as it provides a set of trainable end-to-end neural building blocks and pre-trained models that can be optimized to build speaker diarization pipelines. Before the development of pyannote, there were already a few existing open-source toolkits that also addressed the field of speaker diarization such as S4D, Kaldi, and ALIZE [8]. What separates pyannote from all these other toolkits is its ease of use as it is written in Python, as well as reliability to achieve state of the art performances as it implements x-vector models.

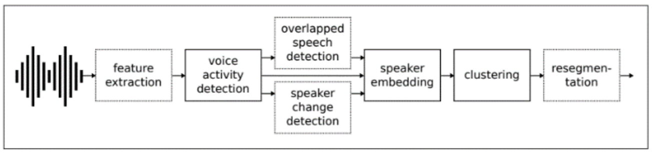

Fig. 2: PyAnnote speaker diarization pipeline [7]

The pyannote follows the traditional speaker diarization pipeline as shown in Fig. 1 but it has more steps in between the processes of feature extraction and clustering as presented in Fig. 2. For instance, voice activity detection does not only detect the speech, but also detects overlapping speech and speaker changes, before generating speaker embeddings. After the clustering process, the pyannote speaker diarization system runs the process of re-segmentation, which is the task of refining speech, boundaries of each speaker turn, and labels coming out of a diarization pipeline [8].

Performance evaluation was conducted by Bradin et al. to compare the results pyannote compared to the baseline, which are densely-connected LSTM architecture trained to predict ideal ratio masks of speech from log-power spectra (LPS) features [23]. According to Bradin et al., the baseline already corresponds to the best result that can be found in the literature as of October 2019 [7], but pyannote still managed to achieve a better accuracy compared to the baseline as this approach yields the higher values of precision and recall. Pyannote also serves to be a very reliable tool for x-vector clustering as this system has consistently achieved state-of-the-art performances in not only speaker diarization, but also other fields in speech processing such as speech activity detection [6, 11, 12].

D. Tool: VBx, VBHMM X-Vectors Diarization

Variational Bayes Hidden Markov Models (VBHMM) x-vectors diarization, or VBx for short, is a diarization system based on a Bayesian HMM model for clustering x-vectors, combined with a ResNet101 x-vector extractor. This diarization method was introduced by Landini et al. to propose an alternative method for speaker diarization without having to rely on end-to-end approaches, and focus towards the clustering of x-vectors [10]. The VBx recipe consists of computing x-vectors, performing agglomerative hierarchical clustering on x-vectors as a first step to produce an initialization, applying variational Bayesian HMM over x-vectors to produce the diarization output, then finally scoring that diarization output [9].

An extensive experiment was conducted in order to compare the performance of the VBx diarization with other methods in the literature such as Kaldi and AHC. Landini et al. found out that VBx achieves superior performance when evaluated on three popular datasets, which are CALLHOME, AMI, and DIHARDII datasets [10]. VBx diarization depends on a pre-trained x-vector extractor and a PLDA model, thus relying on the training of both models. The x-vector training involves data augmentation to obtain additional copies of the data with artificially added noise, music, or reverberation. The PLDA training, on the other hand, involves the training of 8kHz and 16kHz PLDA models on the same data as corresponding x-vector extractors. The PLDA trained on those x-vectors is later utilized in VBx to operate on x-vectors extracted from much shorter 1.5s segments [10].

The results presented a state-of-the-art performance on the CALLHOME, AMI and DIHARDII datasets, without having to perform any specific model adaptations. Furthermore, most approaches for x-vector clustering and embeddings are complementary with VBx [13, 14, 15], thus showcasing more potential of this diarization method.

E. Speaker Diarization for Low-Resourced Language

Recent studies show that the process of speaker diarization has indeed been carried out on low-resourced languages. Kizitskyi et al. conducted a study on improving speaker verification models for low-resourced languages, specifically the Ukrainian language in their study [20]. Speaker verification, an essential task in speech processing, is the process focusing on verifying the identity of the speaker. The speaker diarization was performed on the speaker verification models after the experiment. The validation results on Ukrainian language of the speaker diarization shows that, even for low-resourced languages, the models can achieve state-of-the-art performance metrics by transferring skills from other languages to the low-resourced language.

The performance of the multiple speaker diarization tools has also been tested on multiple different low-resourced languages recently. One particular investigation was conducted by Gina-Anne Levow on low-resourced languages which are the Cicipu, Effutu, Mocho', Northern Prinmi, Sakun, Upper Napo Kichwa, Toratan, and Ulwa [21]. The speaker diarization tools that were chosen on the other hand are the baseline LIUM diarization model, the Kaldi speaker diarization recipe [22], pyannote [7], and VBx [9]. The results obtained, which are the mean diarization error rates (DER) of each speaker diarization approach, showcases pyannote outperforming other approaches, thus the reason its pre-trained x-vector model was chosen as the approach to be used for obtaining speaker segments and investigating the DERs on the Sarawak Malay conversational speech corpus, which is discussed in greater detail in the next section.

III. Sarawak Malay Language & Data

The dataset used during this research was obtained from the *Kalaka: Language Map of Malaysia* website which can be accessed at http://kalakamap.unimas.my/kalaka, where the Sarawak Malay conversations were collected through a crowdsourcing strategy which was implemented during course assignments with students from Universiti Malaysia Sarawak. The students were given several topics for discussion, such as traditional stories and games, to conduct interviews with their speakers. The conversation audio files, along with details, are then uploaded to the *Kalaka* website for the purpose of preserving the data.

The Sarawak Malay language is a variant of the Malay language native to the Malaysian state of Sarawak. This language is the mother tongue of Sarawakians, as it is spoken not only by the Sarawakian Malay people but also by other races, both native and non-native as well such as the Ibans, Bidayuhs, and even the Chinese and the Indians. Thus, this language is relatively popular in Sarawak as it is spoken by approximately over 1,000,000 people.

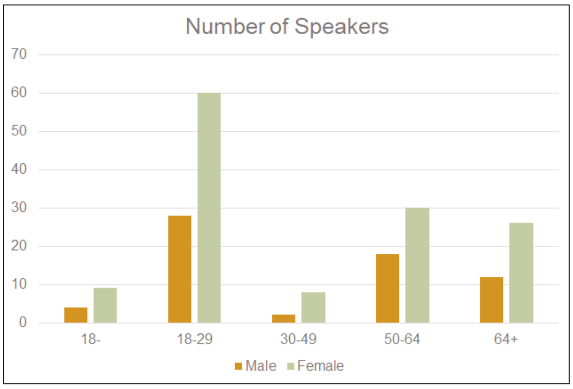

Fig. 3: Age distribution of Sarawak Malay speakers

For training, there are a total of 237 Sarawak Malay conversation audio files with a total duration of 12 hours and 46 minutes in duration, averaging 3 minutes and 14 seconds per file. There are a total of 146 speakers with 58 being male and 88 being female.

For testing, there are a total of 37 Sarawak Malay conversation audio files with a total duration of 1 hour and 26 minutes, averaging 2 minutes and 19 seconds per file. There are a total of 52 speakers with 6 being male and 46 being female. The test dataset also includes Rich Transcription Time Marked (RTTM) files, which contain all the speaker turns and durations, as well as UEM files, which contain the start time and total duration of each audio file. As for the recording environment, most of the interviews were conducted in the same room using the same microphone, some were conducted through a voice call, and thus having different environments. Even when the recordings were done in the same room, some of the conversations were recorded in a quiet place while some were recorded with background noises such as vehicular sounds, music, and unrelated speech from others in the general area.

For both datasets, the age of the speakers is also taken into account to ensure diversity among the speakers. Fig. 3 shows the age distribution among the speakers.

IV. Experiments and Setups

For this research, the diarization process was conducted using the pyannote.audio toolkit. Pyannote.audio is an open source toolkit for speech processing written in Python and based on the PyTorch machine learning framework [7]. Other than speaker diarization, this toolkit also provides models for segmentation, embedding, overlapped speech detection, and speaker change detection. Pyannote.audio uses x-vector Time Delay Neural Network (TDNN) based architecture, thus using x-vector models.

The aim of this research is to answer the research questions: (1) How to improve speaker diarization on low-resourced languages such as the Sarawak Malay Language? (2) How to utilize raw conversation data to fine-tune a speaker diarization system without manual labeling? Thus, the Sarawak Malay conversation data obtained from the Kalaka: Language Map of Malaysia website was used to answer the first research question. To answer the second research question, the raw Sarawak Malay conversation data was auto-labeled to be used for training the x-vector models used to do speaker diarization on the test set, which is the manually labeled Sarawak Malay dataset. The methodology for this research involves obtaining the baseline of the performance of the pre-trained x-vector model to be used for performance evaluation, the processes of auto-labeling the unlabeled data to be used as the training set, and fine-tuning the pre-trained x-vector model using the training data to evaluate the effect of Sarawak Malay auto-labeled data on the diarization performance of fine-tuned x-vector model.

A. Obtaining Speaker Diarization Baseline Results for Sarawak Malay test data

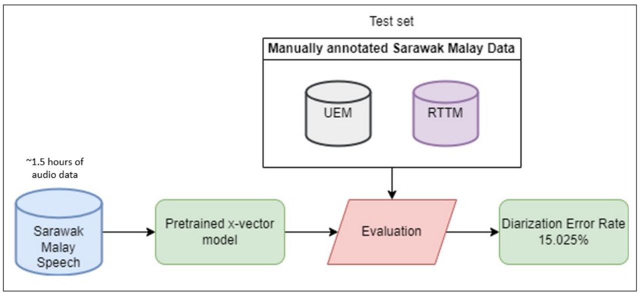
Test set
Manually annotated Sarawak Malay Data
UEM
RTTM
~1.5 hours of audio data
Sarawak Malay Speech
Pretrained x-vector model
Evaluation
Diarization Error Rate 15.025%

Fig. 5: Obtaining baseline using a pre-trained x-vector Model

In order to gain baseline results on our test data, we used the pre-trained x-vector model (x-vector model). The pre-trained model diarized 37 conversational speech files and we evaluated the outputs with the manually annotated ground truth UEMs and RTTMs, yielding the diarization error rate (DER) of 15.03%. Diagram in Fig. 5 shows the speaker diarization process on the Sarawak Malay test data.

B. Diarization of Unlabeled Sarawak Malay Conversational Speech Data (Auto-labeling)

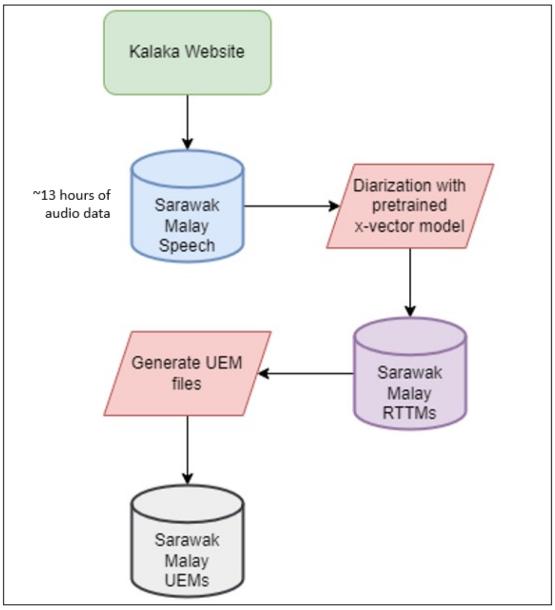

Fig. 4: Generating pseudo-labels for unlabeled speech data using x-vector model

The next approach is to auto-label the unlabeled data obtained from the *Kalaka* website with speaker segments. All audio files are in mono audio WAV format with 16khz sample rates. Fig. 4 shows how we produce the pseudo-labels for the ~ 13 hours conversational speech data. Using a pre-trained x-vector model [8] available in pyannote, we diarized the conversation data to get the speaker segments in RTTM formats. These pseudo-labels are then used to generate labels in Unpartitioned Evaluation Map (UEM) formats for evaluation.

C. Fine-tuning Pre-trained Diarization Model using Pseudo-labels

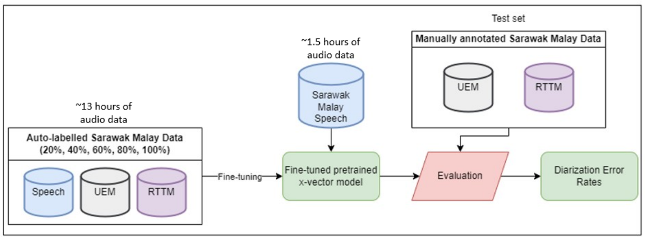

Fig. 6: Fine-tuning x-vector model

After obtaining the pseudo-labels, we used the labels and the speech data to fine-tune the pre-trained x-vector model. In this fine-tuning process, we experimented with five different training data sizes to study how much the size affects the diarization performance.

First, the pre-trained x-vector model was fine-tuned using only 20% of the training data, using only 48 files out of 237 for at most 20 epochs. After the training, the model is then further fine-tuned using segmentation threshold at 0.871 and clustering threshold at 0.822, both are default hyperparameter values in the pyannote training pipeline. Afterwards, the x-vector model is evaluated on the same test data for getting the DER.

The same process is then repeated with 40%, 60%, 80%, and 100% of the training data resulting in another four speaker diarization models. Fig. 6 presents the fine-tuning strategy for improving the speaker diarization performance on the Sarawak Malay test data. The DERs achieved using each variation of training data size are shown in Table 1.

V. Results & Discussions

TABLE I. DERs of Fine-tuned X-vector Model Against Portions of Training Set

	BASELINE	20%	40%	60%	80%	100%
DER (%)	15.03	14.83	14.60	14.16	13.85	13.59

As the size of the training data is being increased, the DERs of the x-vector models gradually decrease. These results prove that changing the size of the dataset used for training affects the performance of the diarization performance of the diarization model.

With the decrement of DER, the diarization performance increases over time. Despite already achieving a significant low DER as the baseline result, it is shown that the baseline can be further reduced by fine-tuning the pre-trained x-vector model. Moreover, it is interesting to observe that using auto-labeled Sarawak Malay conversational speech in training data can obtain fine-tuned x-vector models that outperformed the baseline result.

A detailed analysis on the effect of gender on diarization performance shows a general balance among the DERs for the audio files involving either gender (13.66% average DER for male audios and 13.57% average DER for female audios) due to the more relatively balanced dataset for both genders is being used as the training set (58 males and 88 females) to fine-tune the x-vector model. Thus, the skewing of DER in the x-vector model was relatively minor, despite the test set being used involving only 6 males and 46 females.

One limitation of this research is that the recording environments were not taken into account as there are many different environments where the interviews were conducted. The only way to specifically determine the specific number of interviews conducted in each environment is to manually check the audio files one by one, which will prove to be too time consuming.

VI. Conclusions & Future Work

In conclusion, not only can the goal of improving diarization in low-resourced language be achieved, this research also opens the windows of opportunity around the study of auto-labeled data, which involves unsupervised machine learning. This is due to the diarization performance of the pre-trained x-vector model that utilizes x-vectors, which was already trained using hundreds of hours of data with five different languages and can still be further improved without the usage of manually labeled data.

The future steps in our work would involve exploring various avenues for expanding the dataset's scope. These include techniques such as augmenting existing data through simulated data generation within the same low-resourced language and categorizing the training datasets according to the environments of the recordings for better analysis. Additionally, merging multiple conversation audio files together can introduce a broader range of speakers, presenting an opportunity to delve into diverse speaker diarization attributes. This investigation could illuminate the roles these attributes play in enhancing the training process and subsequently elevating the diarization performance of the model.

Acknowledgment

This work has been funded by the European Union's Horizon 2020 research and innovation program under the Marie Skłodowska-Curie grant agreement No 101007666, the Agency is not responsible for this results or use that may be made of the information. The research reported here was conducted at the 2023 Frederick Jelinek Memorial Summer Workshop on Speech and Language Technologies, hosted at Le Mans University (France) and sponsored by Johns Hopkins University. The authors would like to thank Prof Sanjeev Khudanpur and Hervé Bredin for their input and feedback during the workshop.

References

[1] Park, T. J., Kanda, N., Dimitriadis, D., Han, K. J., Watanabe, S., & Narayanan, S. (2022). A review of speaker diarization: Recent advances with deep learning. *Computer Speech &amp; Language*, *72*, 101317. https://doi.org/10.1016/j.csl.2021.101317

[2] Gish, H., Siu, M.-H., & Rohlicek, R. (1991). Segregation of speakers for speech recognition and speaker identification. *[Proceedings] ICASSP 91: 1991 International Conference on Acoustics, Speech, and Signal Processing*. https://doi.org/10.1109/icassp.1991.150477

[3] U. Jain, M. a. Siegler, S.-J. Doh, E. Gouvea, J. Huerta, P. J. Moreno, B. Raj, R. M. Stern, Recognition of continuous broadcast news with

multiple unknown speakers and environments in: Proceedings of ARPA Spoken Language Technology Workshop, 1996, pp. 61-66

[4] Snyder, D., Garcia-Romero, D., Sell, G., Povey, D., & Khudanpur, S. (2018). X-vectors: Robust DNN embeddings for speaker recognition. *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. https://doi.org/10.1109/icassp.2018.8461375

[5] Snyder, D., Garcia-Romero, D., Sell, G., McCree, A., Povey, D., & Khudanpur, S. (2019). Speaker recognition for multi-speaker conversations using X-vectors. *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. https://doi.org/10.1109/icassp.2019.8683760

[6] Khoma, V., Khoma, Y., Brydinskyi, V., & Konovalov, A. (2023). Development of supervised speaker diarization system based on the PyAnnote Audio Processing Library. *Sensors*, *23*(4), 2082. https://doi.org/10.3390/s23042082

[7] Bradin, H. (2019). *Pyannote/pyannote-audio: Neural building blocks for speaker diarization: Speech activity detection, speaker change detection, overlapped speech detection, speaker embedding*. GitHub. https://github.com/pyannote/pyannote-audio

[8] Bredin, H., Yin, R., Coria, J. M., Gelly, G., Korshunov, P., Lavechin, M., Fustes, D., Titeux, H., Bouaziz, W., & Gill, M.-P. (2020). Pyannote.audio: Neural building blocks for speaker diarization. *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. https://doi.org/10.1109/icassp40776.2020.9052974

[9] Landini, F. (2020, December 24). *BUTSpeechFIT/VBX: Variational Bayes HMM over X-vectors diarization*. GitHub. https://github.com/BUTSpeechFIT/VBx

[10] Landini, F., Profant, J., Diez, M., & Burget, L. (2022). Bayesian HMM clustering of X-vector sequences (VBX) in speaker diarization: Theory, implementation and analysis on standard tasks. *Computer Speech &amp; Language*, *71*, 101254. https://doi.org/10.1016/j.csl.2021.101254

[11] Sarfjoo, S. S., Madikeri, S., & Motlicek, P. (2021). Speech activity detection based on multilingual speech recognition system. *Interspeech 2021*. https://doi.org/10.21437/interspeech.2021-1058

[12] Cretois, B., Rosten, C., & Sethi, S. S. (2022). *Automated Speech Detection in Eco-Acoustic Data Enables Privacy Protection and Human Disturbance Quantification*. https://doi.org/10.1101/2022.02.08.479660

[13] Yue, Y., Du, J., He, M.-K., Yeung, Y., & Wang, R. (2022). Online speaker diarization with core samples selection. *Interspeech 2022*. https://doi.org/10.21437/interspeech.2022-10363

[14] Serafini, L., Cornell, S., Morrone, G., Zovato, E., Brutti, A., & Squartini, S. (2023). An experimental review of speaker diarization methods with application to two-speaker conversational telephone speech recordings. *Computer Speech &amp; Language*, *82*, 101534. https://doi.org/10.1016/j.csl.2023.101534

[15] Karra, K., & McCree, A. (2021). Speaker diarization using two-pass leave-one-out gaussian PLDA clustering of DNN embeddings. *Interspeech 2021*. https://doi.org/10.21437/interspeech.2021-1807

[16] Garcia-Romero, D., Snyder, D., Sell, G., Povey, D., & McCree, A. (2017). Speaker diarization using Deep Neural Network embeddings. *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. https://doi.org/10.1109/icassp.2017.7953094

[17] Fujita, Y., Kanda, N., Horiguchi, S., Nagamatsu, K., & Watanabe, S. (2019). End-to-end neural speaker diarization with permutation-free objectives. *Interspeech 2019*. https://doi.org/10.21437/interspeech.2019-2899

[18] Snyder, D., Garcia-Romero, D., Povey, D., & Khudanpur, S. (2017). Deep neural network embeddings for text-independent speaker verification. *Interspeech 2017*. https://doi.org/10.21437/interspeech.2017-620

[19] Jeancolas, L., Petrovska-Delacrétaz, D., Mangone, G., Benkelfat, B.-E., Corvol, J.-C., Vidailhet, M., Lehéricy, S., & Benali, H. (2021). X-vectors: New quantitative biomarkers for early parkinson’s disease detection from speech. *Frontiers in Neuroinformatics*, *15*. https://doi.org/10.3389/fninf.2021.578369

[20] Kizitskyi, M., Turuta, O., & Turuta, O. (2023). Improving Speaker Verification Model for Low-Resources Languages. *COLINS-2023: 7th International Conference on Computational Linguistics and Intelligent Systems*. https://ceur-ws.org/Vol-3403/paper8.pdf

[21] Levow, G.-A. (2023). *Investigating Speaker Diarization of Endangered Language Data*. https://aclanthology.org/2023.computel-1.6.pdf

[22] Povey, D., Burget, L., & Khudanpur, S. (2013, October). *Kaldi-ASR*. GitHub. https://github.com/kaldi-asr/kaldi

[23] Sun, L., Du, J., Gao, T., Lu, Y.-D., Tsao, Y., Lee, C.-H., & Ryant, N. (2018). A novel LSTM-based speech preprocessor for speaker diarization in realistic mismatch conditions. *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. https://doi.org/10.1109/icassp.2018.8462311