Accepted manuscript of: Affek, M., Tatara, M.S. (2023). Open-Set Speaker Identification Using Closed-Set Pretrained Embeddings. In: Kowalczuk, Z. (eds) Intelligent and Safe Computer Systems in Control and Diagnostics. DPS 2022. Lecture Notes in Networks and Systems, vol 545. Springer, Cham. https://doi.org/10.1007/978-3-031-16159-9_14

Open-Set Speaker Identification using Closed-Set Pretrained Embeddings

Michał Affek[1[0000−0001−6953−1609]] and Marek S. Tatara[1[0000−0001−9753−8429]]

Department of Robotics and Decision Systems, Faculty of Electronics, Telecommunications, and Informatics, Gdańsk Tech, Gdańsk, Poland
s165541@student.pg.edu.pl, martatar@pg.edu.pl

Abstract. The paper proposes an approach for extending deep neural networks-based solutions to closed-set speaker identification toward the open-set problem. The idea is built on the characteristics of deep neural networks trained for the classification tasks, where there is a layer consisting of a set of deep features extracted from the analyzed inputs. By extracting this vector and performing anomaly detection against the set of known speakers, new speakers can be detected and modeled for further re-identification. The approach is tested on the basis of NeMo toolkit with SpeakerNet architecture. The algorithm is shown to be working with multiple new speakers introduced.

Keywords: speaker identification · open-set identification · speaker recognition · feature extraction · anomaly detection.

1 Introduction

Throughout the years the solutions to the task of identifying a person became more accurate and robust. One of the aspects that plays a major role here is the identification of a person by his/her voice, on a basis of short utterances. The problem can be divided into a few subcategories: (i) closed-set identification, where the recording must be assigned to one of the already known speakers, (ii) open-set problem, where apart from the already known speakers there are lectors previously unheard, and (iii) speaker diarization, where subsequent parts of a recording are attributed to particular speakers [2]. Although the closed-set problem is the object of research for quite some time, the higher accuracy for this problem with numerous speakers was achieved just recently [1]. The open-set problem, on the other hand, is still not sufficiently addressed and more scientific efforts are still needed.

Early approaches to address the open-set problem were using Gaussian Mixture Models (GMM) [13], which were further extended by introducing Universal Background Model (UBM) [14] to include speaker-independent background in the model, commonly used across different application of speaker recognition [10, 11, 15]. Later, on top of GMM-UBM, the i-vectors (identity vectors) were introduced [4, 6]. These vectors used to extract fixed-length feature vectors from utterances using UBM combined with Baum-Welch statistics, which were used in

numerous applications [5]. Instead of relying on a predefined, usually statistics-based set of features, deep models map utterances to a set of fixed-length deep features, which are further used for the classification task, usually, with an associated neural model [1, 3]. The approach proposed in this paper, which can be seen as the original contribution of the Authors, takes advantage of the deep neural models trained with an aim of speaker identification in a closed-set, and extend them toward the generalized open-set problems.

2 Proposed Approach

The authors propose to build the proposed solution on the deep neural networks approach used for a closed-set identification with numerous speakers and extend it toward an open-set problem without retraining. A vector of features is extracted by probing the last layer (of a neural network) that is prior to the classifier. If the network is adequately trained, this layer should represent low-level deep features differentiating the speakers. Further, customized algorithms for classifying speakers can be implemented. In the simplest form, it can be implemented as thresholding of the cosine distance between two embeddings. If the distance is smaller than the assumed threshold, then two embeddings belong to the same speaker, otherwise - they do not. If a test feature vector does not belong to any known speaker, a new class (representing a speaker) is created.

2.1 Rationale

The rationale behind the proposed approach is the fact, that each new training requires new data and resources, and might even not return a viable solution. Instead, it is proposed to use an already trained neural network and cut off the part responsible for classification. In that way, the output of the network will be a vector containing features that represent the characteristics of the speaker. Such vectors will be called embedding.

The success of the authors' reasoning depends on whether the embeddings are rich enough and discriminative in a way allowing for the successful classification of new speakers. For a sufficiently large and heterogeneous dataset, the deep neural networks should learn the representation of such discriminative features to make further classification possible. By assuming, that the human voice can be characterized by a finite number of features, it should be sufficient to use these features and perform the identification even for previously unheard speakers.

2.2 Data Flow

The proposed solution attempts to take advantage of the pretrained models (especially those trained with a high volume of data) without the need to retrain them or collect new data. The proposed high-level data flow for the speaker identification task is presented in Fig. 1. The utterances to be analyzed are provided to the input of a neural network. Then, the weights on one of the

last layers are read, forming a feature vector (embedding). The feature vector is compared with all the models of known speakers and if a match according to specified criteria is found, then the utterance is classified as the known speaker that was matched. Otherwise, the utterance is assigned to a new speaker, and based on the extracted embedding a new model is created and added to the speakers' models database.

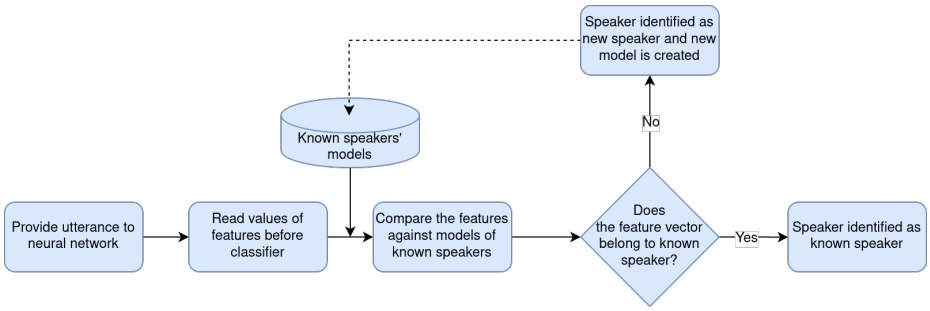

Fig. 1. Chart presenting the data flow in the proposed method.

Note that it is advisable to analyze utterances belonging to a single person in a batch so that the model representing a speaker can be built on a set of features containing representative statistical characteristics.

2.3 Speaker Recognition Backbone

NeMo (Neural Modules) is a toolkit developed by the international company NVIDIA and published under Apache 2.0 license in 2019 [8]. Since its very first publication, it is still under development and major bug fixes are constantly being introduced. The representative idea behind the system is to show the capacity of NVIDIA's products in Machine Learning (ML) applications to the public.

Both training and inference functions from NeMo were used to establish and test the selected model. The exact structure of the neural model is entirely based on SpeakerNet - available in pretrained collection in NeMo [7]. The model, with the help of structural scripts from NeMo, was fine-tuned using a selected part of LibriSpeech [12]. This establishes the the neural backbone for the proposed solution. Theoretically, any other backbone can be used but the latter part (classifier and anomaly detection) must be adapted to a specifically chosen approach. The architecture of SpeakerNet with the indicated part where embeddings are extracted is shown in Fig. 2. Note that the SpeakerNet is used in the context of the proposed approach only as a feature extractor, and the open-set classification scheme is implemented by the Authors.

In summary, the NeMo backbone provides utilities for training the model and produces embeddings for each utterance, serving as inputs to the anomaly detection algorithm, followed by the speaker identification part. The embeddings are

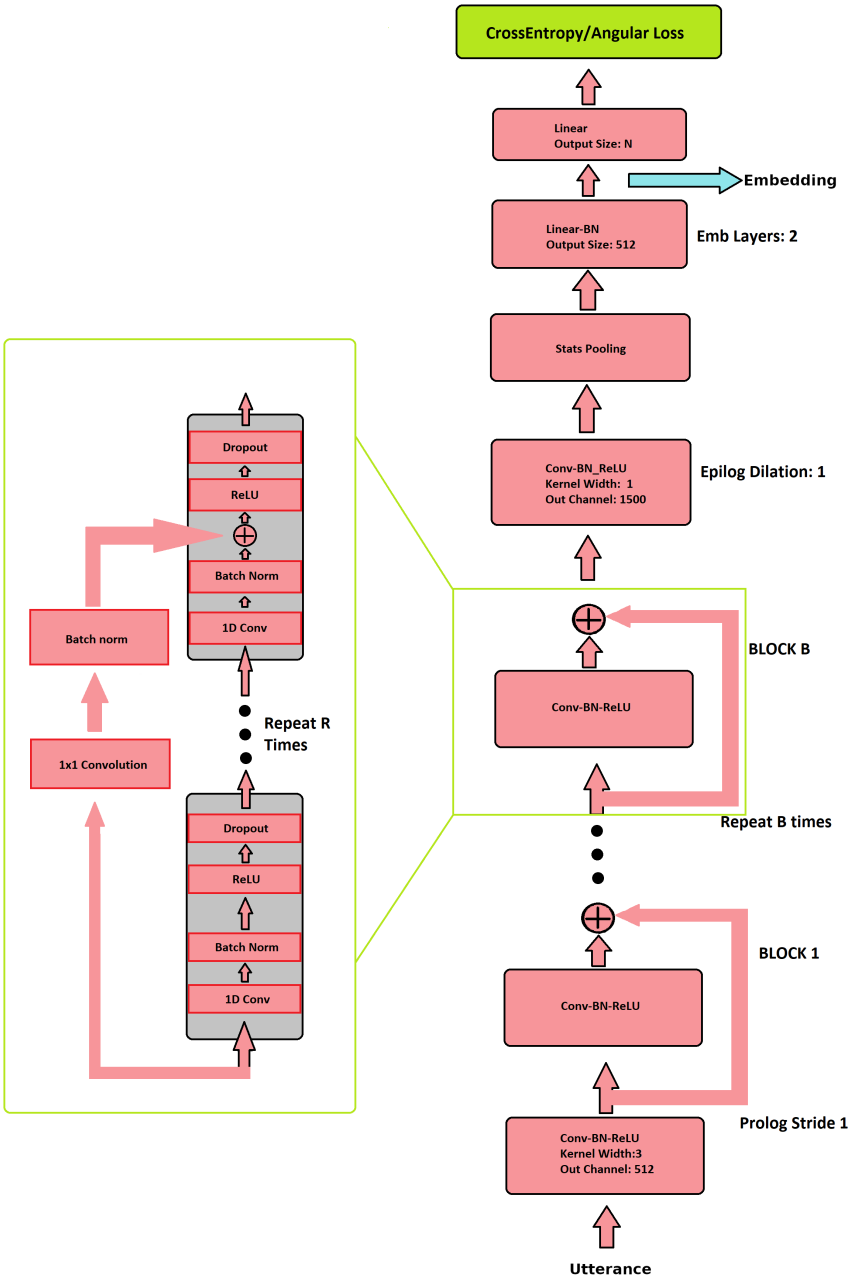

Fig. 2. SpeakerNet architecture with the embedding extraction point indicated (teal arrow). The architecture is taken from [7].

represented as vectors with 512 elements each (*i.e.*, containing one-dimensional numeric data of size 512). The values inside mentioned embedding vectors are ranging between -0.2 and 0.2 with the mean value close to 0. For example, the embedding of the speaker with ID number 1455 has the mean value equal to 0.001, maximal value in the vector equal to 0.143 and minimal equal to -0.079. This is part of characteristic of LibriSpeech dataset - it shows that the data is quite consistent and correctly cleaned from noise or significant outliers. For clarification, these characteristics are extracted by loading the trained model and using NeMo's inference functions. Then the results are saved inside *Pickle* (*.pkl*) file (Python package which provides the user with the possibility to transform Python object into a stream of bytes). Thanks to the byte representation of the data a full portability of embeddings is assured, which is important in case of evaluation in various environments (*i.e.* different operating systems).

3 Embedding-based Classification of Speakers

Embeddings (*i.e.* feature vectors extracted from audio by SpeakerNet consisting of 512 values each) described previously are the main output of the model's inference. General usage of such embeddings which provides the information on the model's sufficiency is based on a trial approach. Users with the help of scripts generate trial files that include two utterances with the ground truth information (same class - 1, mismatch - 0). After preparing such files, the evaluation of cases starts. Program loads data from embeddings *.pkl* file pointed by the provided trial text file. With both vectors loaded the script evaluates the similarity between them. The similarity function used is the basic cosine similarity function. Since it produces values in the range (-1, 1) the decision to scale them to the range (0, 1) was made. By obtaining the new file with scores, it is possible to see how the model behaves in each case. Greater value indicates that vectors are more similar, while lower values can impede that the utterances are not coming from the same speaker.

After collecting the scores on the test utterances it was essential to calculate Equal Error Rate (EER) for this trial. EER is an error measure associated with a threshold, for which the probability of false rejection (false negative) is equal to the probability of false acceptance (false positive). The measurement is equal to 0.83% of EER, which was estimated with a threshold equal to 0.71. With increasing the number of trials the EER went down to 0.58% and the threshold changed insignificantly to 0.72. Further tests showed that the convergence on EER is going to around 0.35% and the threshold stabilizes at 0.7. Note that these values, in general, can change with new data, as it is specific to a particular set of utterances. Nevertheless, due to low error and approximately stable threshold value, the threshold equal to 0.7 was used for further analysis. Table 1 presents the exemplary output of the test which evaluates classification capability by performing a verification task (same class examples are not using the same utterances for comparison).

Note that in case of applying the solution fitted to the closed-set problem, the detection of unseen speakers can be referred to as anomaly detection, as this speaker is previously unseen (anomaly) to the model.

Table 1. Exemplary scores produced by cosine similarity function on two feature vectors (IDs, rescaled cosine similarity value, ground truth.

Indexes	**Cosine similarity**	**Ground truth**
211 and 211	0.830	1
7402 and 7402	0.890	1
233 and 233	0.812	1
3242 and 3242	0.718	1
6848 and 6848	0.917	1
211 and 211	0.868	1
4340 and 4340	0.893	1
1069 and 226	0.493	0
2136 and 2911	0.524	0
8324 and 2436	0.518	0
4014 and 730	0.468	0
6415 and 5561	0.599	0
4640 and 1235	0.603	0
2836 and 27	0.368	0

To visualize the data, the t-distributed stochastic neighbor embedding (t-SNE) algorithm is introduced [9]. It performs the projection of multidimensional data from embeddings to two- or three-dimensional space. It is worth noting that in this method there is a random initialization of points in the target projection. This is causing the results to diverge between runs. The projected space contains datapoints representing embeddings for particular utterances in a human-comprehensible way.

The main observation from analyzing the distribution of embedding in these spaces is that additional unseen points most of the time are isolated in the space. The model which did not "heard" this speaker's utterances in the training stage is still able to produce embeddings, yet they are distinct enough to not fall close to a known class. In addition, in the case of more than one unseen speaker the datapoints are organized in clusters which proves that the model is recognizing similar characteristics in data from an unknown source, and can be applied to the open-set problem.

The anomaly detection algorithm is implemented in its simplest form and brings down to the calculation of cosine distance between available datapoints. The distance threshold is used to categorize the results. The exact threshold level is indicated empirically (here 0.7 was assumed), so that the ratio of correctly classified new speakers is maximized, and at the same time the number of misclassification of known speakers is minimized. Tested behavior covers two-dimensional projections of embeddings.

4 Results

To check the correctness of the trained model it is advisable to visualize the embeddings in an identifiable manner. It is possible to represent feature vectors in reduced-dimensionality form. Again, to make it human-readable and possible to visually evaluate, reduction to two latent dimensions (with no identified physical meaning) is performed. In Fig. 3 groups of samely colored points hint that embeddings of speakers accurately portray the speaker's voice characteristics. In addition, standard inference (created by NeMo authors) tests are performed. These with usage of cosine similarity on embeddings yielded more than 99% of accuracy on the test set (known speaker, unseen utterances). It should be noted that every dot in the mentioned figure is from the test subset of the LibriSpeech dataset. They were not previously seen by the trained model. Also it is worth mentioning general characteristics of used LibriSpeech dataset subset. In training 251 speakers were used (each speaker has portion of utterances saved for test subset), 2 speakers from completely new LibriSpeech subset (for evaluating anomaly detection methods). Duration of utterances is not exceeding 20 seconds; they are preprocessed by dataset authors - cleaned from noise. In following figures all 251 seen speakers are showed (plus unseen ones) except Fig. 3 where the speaker's count was reduced to emphasize NeMo identification accuracy.

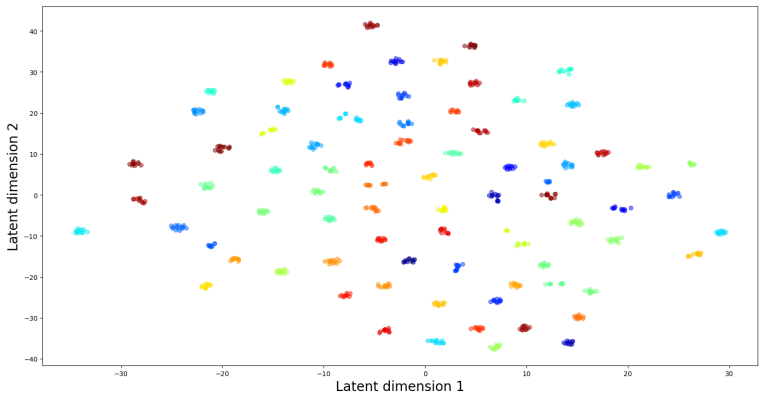

Fig. 3. Classification of generated embeddings by membership to speaker class.

Another test's results can be observed in Fig. 4. They present visualization fn mentioned dimensionality principles with the division between male/female speakers. Such differentiation and self-organization of males and females is a premise that the trained model is suitable for further open-set evaluation.

Unseen speaker is a lector which was not present during the training process of the model. However, the model is used in inference for evaluating utterances

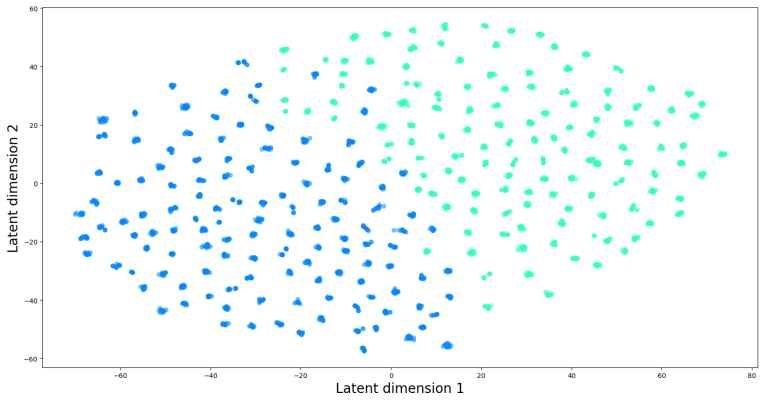

Fig. 4. Visualization of male (blue) and female (cyan) speakers' embeddings.

and producing embeddings for this unseen speaker. Further anomaly detection algorithm decides which utterances provided in testing phase were from outside of training set. This method can be tuned for specific datasets (or parts of it) with tunable threshold. The parameter is chosen empirically. The exact value in example presented in Fig. 5 converges to 0.7. Multiple trials with different setups did not disclosed noticeable deviation from this value (oscillations were ± 0.1).

Setting too high threshold increases the number of False Negatives - the embeddings which are really close to particular classes (set of similar characteristics in voice is substantial), but are not aggregated into a single cluster. The effect of too high threshold can be observed in Fig. 6. On the other hand, too small value leads to too high number of False Positives and embeddings are assigned wrongfully to other classes.

Additionally, another unseen speaker was added to evaluate how anomaly detection algorithm behaves with more than one unseen speaker. The visualizing results Fig. 7 contains two new unseen speakers. The t-SNE algorithm detects that these embeddings are highly different from the rest of the data points. The anomaly detection algorithm does not have much problems with solving such cases (the threshold can be even enlarged).

5 Summary

The paper has shown a methodology how to build a functional speaker open-set recognition system. The system's operation has been tested with functional tests. These preliminary study shows the correctness of the proposed approach.

The speaker recognition backbone can classify the provided utterances into a set of speakers seen during the training with high recognition accuracy exceed-

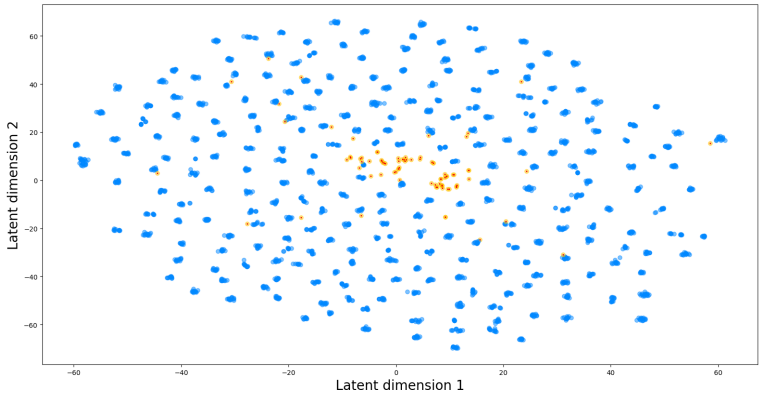

Fig. 5. Visualization of anomaly detection algorithm with threshold equal to 0.7. Blue stands for seen speakers, yellow for unseen utterances in the training stage. Small red dots constitute the results of anomaly detection method (red dot - anomalous data).

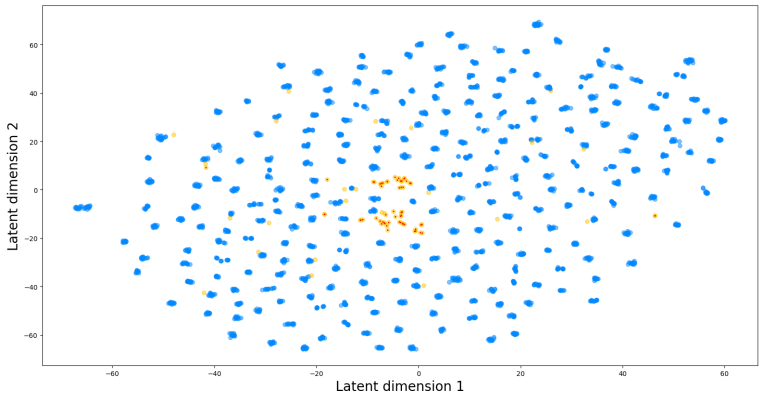

Fig. 6. Visualization of anomaly detection algorithm with too high threshold. Blue stands for seen speakers, yellow for unseen utterances in the training stage. Small red dots constitute the results of anomaly detection method (red dot - anomalous data).

ing 99% for known speakers. It is almost infallible in the new speaker detection task, even for multiple new speaker, yet the accuracy for associating new speakers with the models has to be determined on a more diverse dataset. Created tests have confirmed the usability and accuracy of the constructed system. Performed checks were taking data (recordings) from different LibriSpeech collections. In-

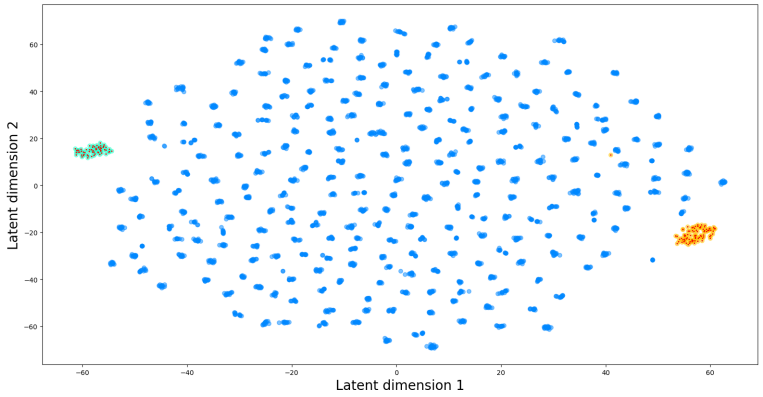

Fig. 7. Visualization of anomaly detection algorithm with threshold equal to 0.7. Blue stands for seen speakers, yellow and teal (accordingly label 464 and 7314 from train_clean_360 subset of LibriSpeech) for not seen in training stage utterances. Small red dots constitute the results of anomaly detection method (red dot - anomalous data).

teresting task would be evaluating the approach on noisy data (for example on the SITW datasets).

The proposed anomaly detection algorithm is not entirely universal and has some limitations. Firstly, the t-SNE projection is changing every run, which makes the visual analysis more difficult. Also the threshold is now predefined, but ideally, it should be adaptive and calculated on the basis of the already collected database of speakers. This would prevent the incorrect anomaly detection in case of providing embeddings from retrained model. Another challenge is the time of computation. The distance is calculated between all possible points. In case of huge number of inputs, it is time consuming.

The focus should also be directed on the way of creating the model of the detected anomaly, which may use more sophisticated method like, for instance, Gaussian Mixture Model. That would make the system less prone to misclassification and can increase the overall accuracy in the open-set problem.

References

1. Bai, Z., Zhang, X.L.: Speaker recognition based on deep learning: An overview. Neural Networks **140**, 65–99 (2021). https://doi.org/https://doi.org/10.1016/j.neunet.2021.03.004, https://www.sciencedirect.com/science/article/pii/S0893608021000848
2. Brew, A., Cunningham, P.: Combining cohort and ubm models in open set speaker identification. In: 2009 Seventh International Workshop on Content-Based Multimedia Indexing. pp. 62–67 (2009). https://doi.org/10.1109/CBMI.2009.30

3. Chung, J.S., Nagrani, A., Zisserman, A.: VoxCeleb2: Deep Speaker Recognition. In: Proc. Interspeech 2018. pp. 1086–1090 (2018). https://doi.org/10.21437/Interspeech.2018-1929
4. Dehak, N., Kenny, P.J., Dehak, R., Dumouchel, P., Ouellet, P.: Front-end factor analysis for speaker verification. IEEE Transactions on Audio, Speech, and Language Processing **19**(4), 788–798 (2011). https://doi.org/10.1109/TASL.2010.2064307
5. Ibrahim, N.S., Ramli, D.A.: I-vector extraction for speaker recognition based on dimensionality reduction. Procedia Computer Science **126**, 1534–1540 (2018). https://doi.org/https://doi.org/10.1016/j.procs.2018.08.126, https://www.sciencedirect.com/science/article/pii/S1877050918314042, knowledge-Based and Intelligent Information & Engineering Systems: Proceedings of the 22nd International Conference, KES-2018, Belgrade, Serbia
6. Kenny, P., Boulianne, G., Ouellet, P., Dumouchel, P.: Joint factor analysis versus eigenchannels in speaker recognition. IEEE Transactions on Audio, Speech, and Language Processing **15**(4), 1435–1447 (2007). https://doi.org/10.1109/TASL.2006.881693
7. Koluguri, N.R., Li, J., Lavrukhin, V., Ginsburg, B.: Speakernet: 1d depth-wise separable convolutional network for text-independent speaker recognition and verification (2020). https://doi.org/10.48550/ARXIV.2010.12653, https://arxiv.org/abs/2010.12653
8. Kuchaiev, O., Li, J., Nguyen, H., Hrinchuk, O., Leary, R., Ginsburg, B., Kriman, S., Beliaev, S., Lavrukhin, V., Cook, J., et al.: Nemo: a toolkit for building ai applications using neural modules. arXiv preprint arXiv:1909.09577 (2019)
9. Linderman, G.C., Steinerberger, S.: Clustering with t-sne, provably. SIAM Journal on Mathematics of Data Science **1**(2), 313–332 (2019)
10. Liu, M., Dai, B., Xie, Y., Yao, Z.: Improved gmm-ubm/svm for speaker verification. In: 2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings. vol. 1, pp. I–I. IEEE (2006)
11. McLaughlin, J., Reynolds, D.A., Gleason, T.: A study of computation speed-ups of the gmm-ubm speaker recognition system. In: Sixth European conference on speech communication and technology. Citeseer (1999)
12. Panayotov, V., Chen, G., Povey, D., Khudanpur, S.: Librispeech: An asr corpus based on public domain audio books. In: 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). pp. 5206–5210 (2015). https://doi.org/10.1109/ICASSP.2015.7178964
13. Reynolds, D.A.: Speaker identification and verification using gaussian mixture speaker models. Speech Communication **17**(1), 91–108 (1995). https://doi.org/https://doi.org/10.1016/0167-6393(95)00009-D, https://www.sciencedirect.com/science/article/pii/016763939500009D
14. Reynolds, D.A.: Comparison of background normalization methods for text-independent speaker verification. In: EUROSPEECH (1997)
15. Zheng, R., Zhang, S., Xu, B.: Text-independent speaker identification using gmm-ubm and frame level likelihood normalization. In: 2004 International Symposium on Chinese Spoken Language Processing. pp. 289–292. IEEE (2004)