MŰEGYETEM 1782

Budapest University of Technology and Economics
Faculty of Electrical Engineering and Informatics
Department of Telecommunications and Media Informatics

Project Lab Report

Title: Pre-trained Speaker recognition embedding models for disorder speech classifications

Written by: Ismayilzada Ismayil **Neptun#:** DGY785

Field: Computer Science Engineering
Specialization: Internet Architecture and Services
E-mail: ismayil.ismayilzada@edu.bme.hu

Supervisor: Dosti Aziz
E-mail: azizd@edu.bme.hu

Academic year:2022/2023 Autumn semester

Abstract

This project aims to investigate the binary classification of voice disorders using speaker verification embedding models. The embeddings of speech samples from Hungarian and Dutch samples were extracted using two state-of-the-art speaker verification models, particularly in cross-lingual dysphonic speech detection. The severity level of dysphonic speech was then estimated. To test the algorithms' performance in cross-lingual scenarios, speech samples from one language will be used for training and another for testing. Generating new samples or resamples from the existing samples using the bootstrapping method to evaluate the precision of a sample statistic. Boosting is an ensemble method that involves successively training many homogeneous algorithms. A final model with the greatest outcomes is produced by these distinct algorithms. The method generates new hypothetical samples that aid in testing an estimated value using the replacement technique.

Keywords: Machine Learning, x-vector, ECAPA, ensemble learning, SVM, SVR, Boosting, Bagging

1 INTRODUCTION

Speech disorders, also known as speech impairments, are a subset of communication disorders that impair regular speech. This can be interpreted as stuttering, lisping, dysphonia, etc. Researchers estimate that 89% of people with Parkinson’s disease (PD) have speech and voice disorders, including laryngeal, respiratory and articulatory disorders [1]. Dysphonia, or the inability to speak normally, is reportedly one of the illness's most distressing symptoms. Dysphonia is a disorder of the voice. Poor voice quality without any obvious anatomical, neurological, or other organic issues affecting the larynx or voice box is known as functional dysphonia. Additionally known as functional voice difficulty.

In recent years, Artificial Intelligence(AI) has impacted several areas of the healthcare system. AI in healthcare is developed for doctors and researchers in the field to use for helping doctors diagnose patients more accurately, make predictions about patients' future health, and recommend better treatments.

A person's narrative, spontaneous speech can be a valuable source of knowledge about their cognitive state and state of health. We can create models for disease detection through speech. It's easy to use a pre-trained model for this. But firstly, we need to collect recordings of both healthy and disordered voices. But in the healthcare system, it is still harder to find personal records of people diagnosed with different diseases. Privacy is a big part of this problem. But not having actual databases for this is a part of the problem too. For these reasons, we have less than what would be a preferable amount of data, and in less variety and languages too. So some of the problems concerning this kind of pre-trained model lie in being able to train this model using a limited amount of data and languages and make it possible to be used in a professional setting with different languages and accents.z

2 REVIEW OF LITERATURE

“Recent Progress in AI-Based Disease Detection From Audio in a Nutshell” review talks about recent advancements in the AI and machine learning algorithms used for disease diagnosis in healthcare systems. Because of progress in technology, there has been more and more technology developed for the healthcare sector, however being restrained by high requirements for accuracy, robustness, and explainability. This paper focuses on health-oriented AI research as a sub-field of digital health and state of art technologies. It shows that the recent success of deep learning in other fields has been translating more and more to digital health while still using classic Machine Learning methodologies being promised. [2]

A research paper done by Attila Zoltán Jenei, Gábor Kiss and Dávid Sztahó, talks about training models for use in detecting speech-related disorders, first, we need to be able to use the speech samples we have gathered. And features should be extracted from the speech samples. They use x-vector and escape pre-trained models for feature extraction. Using different pre-trained models lets us compare the results and find the best pre-trained model for our purposes. In this paper, it is shown that binary and multiclass classifications can be recognized with similar accuracies. Results show that applied embedding models indeed hold information about voice disorders. Parkinson's disease and organic and structural voice disorders scored highest (91% and 93% UAR(unweighted average recall)), respectively [3].

Dysponic voices can be detected using pre-trained models. Most of the previous research has been done using monolingual datasets. But the paper [4] trains and tests their results using two different languages. Hungarian and Dutch speech datasets have been used here. Feature selection and model training has been done on the Hungarian dataset, and later tested on the Dutch dataset. Creating a pre-trained model for cross-language dysphonic voice prediction was found to be acceptable, as the given results are at an acceptable accuracy level. Cross-language classification predictions have shown 0.86 and 0.81 F1 scores with the vowel /E/ included and excluded. [4]

[14] the paper focuses on dysphonia, a voice quality change that affects one in three people at least once in their lifetime. The primary contribution of this paper is an examination and performance evaluation of various machine learning methods for voice pathology detection. A large dataset of 1370 voices chosen from the Saarbruecken Voice Database serves as the basis for all analyses. The outcomes have demonstrated that the Support Vector Machine algorithm provides the best accuracy in voice pathology detection. A voice can be classified as pathological or healthy using all the parameters with an accuracy of about 86% [14].

3. METHODOLOGY

3.1 Embeddings

For speaker recognition, we need to first translate voice and speech to some kind of numerical value for using them in model training. What this means is that we need to extract features from speech, and there are multiple different embedding methods for this type of work.[11] There are numerous ways to recognize speech, including acoustic, deep neural, and wavelet neural networks. Deep neural network (DNN) embeddings are mostly used for speaker recognition. I-vector, x-vector and ECAPA are different kinds of DNN embeddings, with each being a better version of the previous ones.[5]

I am using two different kinds of embedding for my pre-trained models in this paper. They are x-vector and ECAPA embeddings for audio feature extractions.

The i-vectors and x-vectors share the ability to represent speech utterances compactly as a vector of fixed size, regardless of the length of the utterance. The extraction algorithms of i-vectors and x-vector are quite different. The x-vector concept is newer and the name of the method is similar to "i-vector" to suggest that this representation can be used instead of i-vectors in state-of-the-art speaker (or language) recognition systems.

X-vector is a Time Delay Neural Network (TDNN), which is trained to differentiate between speakers, and to map variable-length utterances to fixed-dimensional embeddings. This is what is essentially called an x-vector. Previous studies have found that x-vector embeddings leverage large-scale training datasets better than i-vectors. The performance of the x-vectors is superior on the evaluation datasets and x-vector TDNN successfully uses data augmentation, due to its supervised training. [6]

The neural network is used by the current speaker verification methods to extract speaker representations. Following the application of numerous improvements to this architecture based on current developments in the related fields of face verification and computer vision, ECAPA-TDNN embedding was created. One of these enchantments was the change in the hierarchical changes done to Neural Networks(NN) learning algorithm. It is well known that neural networks can learn hierarchical features, each layer of which operates at a different level of complexity. Firstly, different hierarchical levels of DNN have been aggregated and propagated. Later improving the statistics pooling module enables the network to concentrate on various subsets of frames during each estimation of the channel statistics. This is what is called ECAPA-TDNN. [7]

3.2 Datasets

We are using four different datasets for this research. Two ECAPA and two x-vector for two different languages, Hungarian and Dutch each. There are multiple labels and grades for different kinds of diseases in these datasets, but because we are going to train-test our models for dysphonic and healthy speech. We will not be using all of the entries in our datasets. Health speech is labelled ("HC") and dysphonic speech ("OD") is in the "label" column. So when creating our datasets for training and testing, we will be dropping all labels other than "HC" and "OD".

X-vector and ECAPA embedding for the Hungarian language dataset has 707 rows for each of them inside. There are 374 rows for "HC" and "OD" combined in x-vector embedding datasets. And 60 rows for ecapa embedding datasets.

When creating machine learning models, the first thing we need to do is to preprocess the datasets that we are going to use in our models.

Python predefined libraries must be imported to preprocess data using Python. Some specific tasks are carried out using these libraries. Numpy, Pandas and Sklearn are some of the libraries we will be using.

It's critical to distinguish between independent variables from the dataset and dependent variables in machine learning. Because we are going to train two different kinds of machine learning algorithms, classification and regression, our dependent and independent variables will be different for each algorithm.

3.3 Ensemble learning

By combining the predictions from various models, ensemble learning is a general meta-approach to machine learning that aims to improve predictive performance. The term "ensemble learning" describes algorithms that combine the results of at least two different models. Although there are practically countless ways to accomplish this, the most frequently discussed and used classes of ensemble learning techniques are probably three. Bagging, stacking, and boosting are the three main categories of ensemble learning techniques.

3.4 Bagging

When creating a machine learning model, we can either use a single algorithm or combine several algorithms. Ensemble learning refers to the use of multiple algorithms. Compared to single algorithms, ensemble learning produces better prediction results. Bagging and boosting are the two most popular types of ensemble learning techniques. [8]

In Bagging, the average of the model is calculated after the independent training of several homogenous algorithms.

The Bootstrap Aggregation method also referred to as the Bagging method, can be used to solve classification and regression issues. Bagging algorithms also raise a model's accuracy rating. These algorithms lessen variance and prevent model overfitting. [9]

4 RESULTS

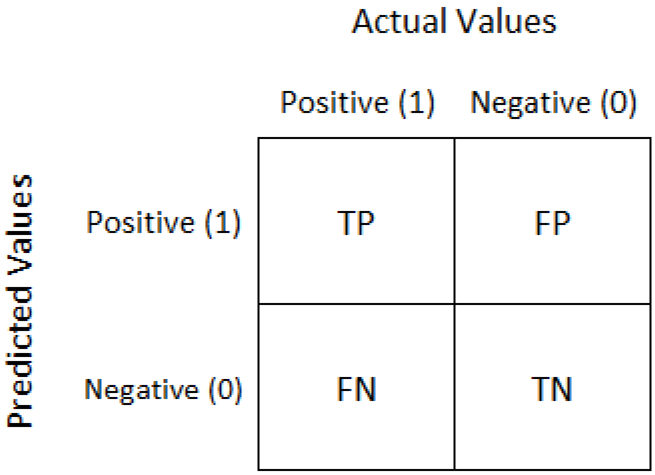
For this paper, we will be using different metrics for both the x-vector and ECAPA embedding datasets. And we will be doing cross-lingual detection of dysphonic speech for Hungarian and Dutch datasets. We will have Hungarian to Dutch language (Table 1 and Table 4) measurements, Dutch to Hungarian (Table 2 and Table 5) measurements and lastly Mixed measurements (Table 3 and Table 6). What this means is that we will be training our models on Hungarian, and later testing them on Dutch datasets, and vice versa. For mixed measurements, we will be combining Hungarian and Dutch datasets and training our model on this newly created dataset.

Each table will show results for four categories: x-vector SVM, x-vector bagging, ECAPA SVM, and ECAPA bagging. We make our tables like these because we want to be able to compare both embeddings and algorithms used in our model. [10]

4.1 Evaluation metrics

For our measurements, we are going to use different metrics for different models. For classification, we will use accuracy score, sensitivity, specificity and F1-scores.

We can find accuracy scores just using our predicted values and test datasets. But to calculate sensitivity, specificity and F1-score, we need to first create a confusion matrix (Picture 1). A table called a confusion matrix is used to describe how well a classification algorithm performs. The output of a classification algorithm is visualized and summarized in a confusion matrix. And after creating a confusion matrix for our classification algorithms, we will need to calculate the other 3 measurements using the values inside this confusion matrix.[13]



		Actual Values	
		Positive (1)	Negative (0)
Predicted Values	Positive (1)	TP	FP
	Negative (0)	FN	TN

Picure 1. Confusion Matrix

True positive (TP) = the number of cases correctly identified as patients.
False positive (FP) = the number of cases incorrectly identified as patients.
True negative (TN) = the number of cases correctly identified as healthy.
False negative (FN) = the number of cases incorrectly identified as healthy.

The accuracy score returns the fraction of correctly classified samples. Having a higher accuracy score means that our model will make fewer errors during prediction, and it will be a successful model.

$$\text{Accuracy} = (TP + TN) / (TP+TN+FP+FN)$$

The metric known as sensitivity assesses a model's capacity to forecast true positives for each available category. The metric known as specificity assesses a model's capacity to forecast the actual negatives for each available category.

$$\text{Sensitivity} = TP / (TP+FN)$$

A measure similar to sensitivity is called specificity. Specificity assesses a model's capacity to forecast the actual negatives for each available category.

$$\text{Specfity} = TN / (TN+FP)$$

The accuracy of a model on a dataset is gauged by the F-score, also known as the F1-score. It's employed to assess binary classification schemes that label examples as "positive" or "negative."

$$F1 = TP / (TP + 0.5(FP+FN))$$

For regression, we are going to use three metrics, Root-mean-square error (RMSE), Pearson correlation (ρ) and spearman correlation (r). They will be shown similarly to the Classification tables, in Table 4, Table 5 and Table 6.

The square root of the residuals' variance yields the RMSE. It shows how closely the observed data points match the values predicted by the model, or how well the model fits the data in its entirety. RMSE is an absolute measure of fit whereas R-squared is a relative measure.

The most popular method for determining a linear correlation is the Pearson correlation coefficient (r). It represents the strength and direction of the relationship between two variables and ranges from -1 to 1.
The strength of association between two rank-ordered variables is measured by the Spearman correlation coefficient, a nonparametric correlation statistic.

4.2 Classification

In Table 1, we can see that all of the values give better results for ECAPA embedding compared to the x-vector. When we did bagging, we were expecting better results compared to SVM values, but in about every category, the Hungarian to Dutch Classification table's measurements shows simple SVM does a better job than bagging.
For x-vector embedded datasets, the difference between SVM and bagging is minimal, but for ecapa embedding, it starts to show bigger differences.

Table1. Hungarian to Dutch Classification (round to 2 decimals)

		Accuracy	Sensitivity	Specificity	F1-score
X-VECTOR	SVM	0.78	0.73	0.83	0.79
	Bagging	0.77	0.73	0.8	0.76
ECAPA	SVM	0.85	0.77	0.93	0.86
	Bagging	0.82	0.73	0.9	0.83

In Table 2, results for x-vector embeddings show the same results for both SVM and bagging algorithms. But for ecapa embedded datasets, bagging shows clear improvement compared to SVM, such as there are 0.02, 0.03, 0.01 and 0.01 increases for accuracy, sensitivity, specificity and f1-score respectively.

Table 2.Dutch to Hungarian Classification

		Accuracy	Sensitivity	Specificity	F1-score
X-VECTOR	SVM	0.75	0.99	0.49	0.65
	Bagging	0.75	0.99	0.49	0.65
ECAPA	SVM	0.85	0.85	0.85	0.85
	Bagging	0.87	0.88	0.86	0.86

In Table 3, Mixed Classification, different from the previous 2 Tables, x-vector embedding shows better results than its ECAPA counterpart. And for each embedded dataset, SVM shows better results than bagging algorithms.

Table 3. Mixed Classification

		Accuracy	Sensitivity	Specificity	F1-score
X-VECTOR	SVM	0.91	0.98	0.83	0.9
	Bagging	0.9	0.96	0.84	0.89
ECAPA	SVM	0.89	0.97	0.81	0.88
	Bagging	0.89	0.97	0.79	0.87

4.3 Regression

In table 4, ECAPA Support Vector Regression (SVR) shows better results than x-vector SVR, such as root means the squared error is 0.66 for ECAPA, but 0.83 for x-vector. But, bagging algorithms shows different results, as the x-vector bagging algorithm shows better results than its ECAPA counterpart. At the same time, bagging algorithms still show clear improvement compared to SVR for its intended embedding. Root means squared error, spearman and Pearson correlation show better results for bagging in both x-vector and ECAPA bagging.

Table 4. Hungarian to Dutch Regression

		RMSE	Spearman correlation	Pearson correlation
X-VECTOR	SVR	0.83	0.64	0.48
	Bagging	0.67	0.83	0.83
ECAPA	SVR	0.66	0.76	0.76
	Bagging	0.72	0.82	0.81

The results we get from table 5 show similar results to the ones we got from table 4, as ECAPA is still the better-embedded dataset. At the same time, bagging shows clear improvement in its measurements compared to SVR.

Table 5. Dutch to Hungarian Regression

		RMSE	Spearman correlation	Pearson correlation
X-VECTOR	SVR	0.97	0.68	0.63
	Bagging	0.66	0.82	0.81
ECAPA	SVR	0.83	0.76	0.75
	Bagging	0.69	0.81	0.81

For table 6, compared to the previous two tables, we do not get better results from ECAPA. And even bagging algorithms does not do better than SVR, as the best measurements of bagging are the same as SVR in Pearson and spearman correlation. But for root means squared error, bagging gets worse, but because it is small, it can be ignored.

Table 6. Mixed Regression

		RMSE	Spearman correlation	Pearson correlation
X-VECTOR	SVR	0.64	0.83	0.83
	Bagging	0.65	0.83	0.83
ECAPA	SVR	0.67	0.81	0.8
	Bagging	0.7	0.81	0.81

5 CONCLUSION

In this research, we were able to see that we get better results most of the time while using ECAPA embeddings, as it is considered better than x-vector embedding. But there can be some exceptions to this.

These results show us that, SVM does a better job than bagging for the classification models. But we were not able to accomplish the expected results for bagging classification algorithms. We wanted to use bagging classification, as in bagging, to find the average of the model, several homogeneous algorithms are independently trained. If there was overfitting in our algorithms, then bagging would prevent model overfitting and reduce variance. But the result we got showed that, in most cases, bagging will be giving worse results than simple SVM algorithms. We got better results using bagging only in the Dutch to the Hungarian Classification model.

For regression, as expected, we got better results from ECAPA compared to the x-vector. Only Mixed Regression (Table 6) shows some better results for the x-vector, but they are minimal. At the same time, we were able to get better measurements for bagging algorithms, meaning the bagging algorithms showed better results than SVR algorithms.

We can use other ensemble methods for further research, such as boosting and so on, to try and get better results than simple SVM or linear regression algorithms.

6 REFERENCES:

[1] Lorraine O Ramig, Cynthia Fox & Shimon Sapir (2008) Speech treatment for Parkinson’s disease, Expert Review of Neurotherapeutics, 8:2, 297-309, DOI: 10.1586/14737175.8.2.297.

[2] Milling, M., Pokorny, F. B., & Schuller, B. W. (2022). Is Speech the New Blood? Recent Progress in AI-Based Disease Detection From Audio in a Nutshell. *Frontiers in Digital Health*, *4*. https://doi.org/10.3389/fdgth.2022.886615.

[3] Attila Zoltán Jenei, Gábor Kiss and Dávid Sztahó, “Detection of Speech Related Disorders by Pre-Trained Embedding Models Extracted Biomarkers” pp. 1-11.

[4] Dávid Sztahó, Miklós Gábriel Tulics, Jinzi Qi, Hugo Van hamme, Klára Vicsi, “Cross-lingual Detection of Dysphonic Speech for Dutch and Hungarian Datasets”, pp.1-6.

[5] Dawalatabad, N., Ravanelli, M., Grondin, F., Thienpondt, J., Desplanques, B., & Na, H. (2021). ECAPA-TDNN Embeddings for Speaker Diarization. *arXiv*. https://doi.org/10.21437/Interspeech.2021-941.

[6] David Snyder, Daniel Garcia-Romero, Gregory Sell, Daniel Povey, Sanjeev Khudanpur. X-VECTORS: ROBUST DNN EMBEDDINGS FOR SPEAKER RECOGNITION. Center for Language and Speech Processing & Human Language Technology Center of Excellence The Johns Hopkins University, Baltimore, MD 21218, USA.

[7] Desplanques, B., Thienpondt, J., & Demuynck, K. (2020). ECAPA-TDNN: Emphasized Channel Attention, Propagation and Aggregation in TDNN Based Speaker Verification. *arXiv*. https://doi.org/10.21437/Interspeech.2020-2650.

[8] González, S., García, S., Del Ser, J., Rokach, L., & Herrera, F. (2020). A practical tutorial on bagging and boosting based ensembles for machine learning: Algorithms, software tools, performance study, practical perspectives and opportunities. *Information Fusion*, *64*, 205-237. https://doi.org/10.1016/j.inffus.2020.07.007.

[9] Kadiyala, A. and Kumar, A. (2018), Applications of python to evaluate the performance of bagging methods. Environ. Prog. Sustainable Energy, 37: 1555-1559. https://doi.org/10.1002/ep.13018.

[10] Carvalho, D.V.; Pereira, E.M.; Cardoso, J.S. Machine Learning Interpretability: A Survey on Methods and Metrics. Electronics 2019, 8, 832. https://doi.org/10.3390/electronics8080832.

[11] Sztahó, D., & Fejes, A. (2022). Effects of language mismatch in automatic forensic voice comparison using deep learning embeddings. arXiv. https://doi.org/10.48550/arXiv.2209.12602.

[13] Baratloo, A., Hosseini, M., Negida, A., & Ashal, G. E. (2015). Part 1: Simple Definition and Calculation of Accuracy, Sensitivity and Specificity. Emergency, 3(2),48-49. https://doi.org/https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4614595/.

[14] L. Verde, G. De Pietro and G. Sannino, "Voice Disorder Identification by Using Machine Learning Techniques," in IEEE Access, vol. 6, pp. 16246-16255, 2018, DOI: 10.1109/ACCESS.2018.2816338.