Speaker recognition in a cross-lingual dataset

Marine Collery
KTH Royal Institute of Technology
Brinellvägen 8, 114 28 Stockholm
collery@kth.se

Younes Laaboudi
KTH Royal Institute of Technology
Björksätravägen 22, 137 27 Skärholmen
laaboudi@kth.se

François Magny
KTH Royal Institute of Technology
Björksätravägen 36, 137 27 Skärholmen
frhmagny@kth.se

Abstract

Here we compare different speaker recognition methods in a cross-lingual corpus composed of 4 languages and 3 speakers. Two feature sets are exploited. Different supervised and unsupervised approaches are tested through models like SVM, Nearest Neighbors, Decision tree, Gaussian Mixture Model and DBSCAN. The metrics used for evaluation are accuracy, precision, recall, F1score for the supervised approach and Adjusted Rand Index (ARI), Average Cluster Purity (ACP) for the unsupervised approach.

1 Introduction

Speaker recognition can be studied in monolingual mode, cross-lingual mode and multilingual mode as pointed out by [18]. However, language independent speakers identification, recognition or verification problems are rarely considered especially for European languages (cross-lingual and multilingual mode). According to [25], 54% Europeans claim to speak more than one language, which justify the interest of such recognition. [2] and [14] present cross-lingual speaker verification on a bilingual speech database (CSLT-CUDGT2014) where the female speakers speak both Standard Chinese and Uyghur. Cross-lingual or multilingual studies have also been made with English and different Indian languages in [21, 6, 3].

Interesting applications of speaker recognition have been mentioned such as [19] that presents speaker recognition technology has a tool for vocal command processing. This application highlights the fact that speaker recognition models will soon boost actual speech recognition tools. A language independent speaker recognition model would considerably increase its value.

Another use of speaker identification is to do speaker indexing or diarization [17]. The diarization process regroups two parts: speaker segmentation (when a speaker changes) and speaker clustering (who are the speakers). This is useful for applications such as automatic indexing of audio information, which are needed for example to deal with large audio data files such as radio broadcast dataset [12].

In practice, Mel Frequency Cepstrum Coefficients (MFCCs) have proven to be very efficient for speaker recognition [13, 10, 6, 26].

Different methods have been presented for effective multilingual speaker recognition such as Support Vector Machine (SVM) [4, 10], Gaussian Mixture Models (GMM) [24, 16] or DBSCAN [15].

30th Conference on Neural Information Processing Systems (NIPS 2016), Barcelona, Spain.

2 Method

Here we present the method and process we are using.

2.1 Data Collection

The data was collected for the purpose of this project, because we didn't find any database in which the same speakers spoke more than two languages. We recorded in a non-isolated room with a standard smartphone microphone. In order to ensure that the background noises and recording quality were the same for the 3 speakers, the recordings were made successively with the exact same setting.

The recording environment was chosen in order for it to be renewable and for the created models to work with everyday recordings (i.e. phone recordings). Plus, the setup does not require any specific instruments or knowledge in speech processing as shown in Fig.1.

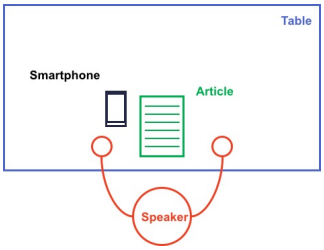


Figure 1: Setup for the recordings from top

Languages	SP1	SP2	SP3	TOTAL
French	07:14	05:40	05:43	18:37
English	13:58	10:51	03:49	26:38
German		3:16		3:16
Spanish			02:24	02:24
TOTAL	21:12	19:47	11:56	52:55

Table 1: Dataset repartition in min:sec

The recordings are in 4 different languages (French, English, German and Spanish), speakers read different newspaper articles.

The database is composed of 8 different combinations of speaker/language as shown in Table1.

Characteristics

- Bit rate: 127kbps
- Encoding: M4A (converted to WAV)
- Sampling rate: 44,1 KHz

2.2 Feature Extraction

In order to extract features from the collected data (See 2.1) we decided to use the open-source feature extractor openSMILE [8]. This tool is used in a lot of speech related papers such as [23] or [22].

Two feature sets are created:

- The first one was to extract the Mel-frequency cepstral coefficients (MFCC) as well as the associated dynamic features (deltas and deltadelta).
  Frame size: 25ms
  Frame step: 10ms
- The second approach was to create a feature set inspired by the features used by [23] to distinguish multiple speaker traits (those include MFCC).

The work presented in the INTERSPEECH 2012 Speaker Trait Challenge paper by [23] from which we inspired ourselves, presents its extracted features as efficient tools to describe Speaker_traits. We were interested in comparing the results with the MFCC features. The openSMILE [8] distribution provides the configuration file used by the authors. Few changes were made to extract values for every frame rather than computing a mean. The exact list of features and functionals are listed in [23] it gathers MFCCs, spectral and energy low level descriptors (LLDs) and multiple functionals.

We will call this feature set the 'Speaker Trait set'.

2.3 Standardization and Dimensionality reduction

For the standardization, the StandardScaler module from scikit-learn [20] is used. Standardization both on the whole dataset and at the speaker level are generated in order to compare their impact on the predictions.

Dimensionality reduction is not considered as the MFCC feature set is composed of only **39 features** and the Speaker_trait feature set of **241 features**. Computations are therefore possible without it and we keep all information provided by the data.

2.4 Model

Here we present the different models we chose to compare.

2.4.1 Supervised Learning

The supervised learning techniques considered are Support Vector Machines (SVM), Nearest Neighbors (kNN) and Tree Decision based classifiers on both feature sets and with both standardizing methods.

The models are evaluated with common metrics (accuracy, precision, recall, F1score).

2.4.2 Unsupervised methods

For the unsupervised methods, the focus is on KMeans and Gaussian Mixture methods. The latter method was especially interesting as they are considered to be more robust against the interference of non-speaker factors [11]. These are methods for which the number of clusters has to be specified. Another method we considered was Agglomerative Clustering, knowing that it requires more computational time.

These algorithms have been proven to work on speaker clustering problems [11].

Those algorithms are tested with both feature sets for multiple numbers of clusters.

The clusters are evaluated with 3 metrics. The **Adjusted Rand Index (ARI)** which computes a similarity measure between two clusterings by considering all pairs of samples and counting pairs that are assigned in the same or different clusters in the predicted and true clusterings. The **Average Cluster Purity (ACP)** which is another commonly used measure to evaluate speaker clustering methods. It is used to measure how well a cluster represents one speaker. As the data is composed of multiple languages, the **ACP for languages (ACLP)** is also generated to see how well the clusters represent languages.

The DBSCAN algorithm which views clusters as areas of high density separated by areas of low density is also tested and its efficiency evaluated via speaker and language repartitions in the created clusters.

3 Experiments

Here we present the settings and experimental parameters used.

3.1 Supervised models

The first experiment was about recognizing speakers using a language that was not used in the training set. In our case, the models were trained with a dataset using only frames from French speech samples, and the test set contained samples in English, German and Spanish.

With both features set (MFCC or Speaker traits set) training set contained a balanced number of frames in French from our three speakers. The test set contained the other languages spoken by the three speakers.

Balanced training set The purpose of the test dataset is in our case to evaluate the ability of the model to predict the speaker speaking regardless of the language used. We decided to train the model

with a balanced dataset not because the test dataset is balanced, there is no reason for it to be but because there is no reason for the test dataset to be unbalanced in a certain direction. It could be unbalanced in favor of Sp1 Sp2 or Sp3 but the model should not prioritize one of them. The training set size is set to 10000 frames.

Models We decided to compare different types of supervised models and evaluated them with the accuracy, precision, recall and F1score metrics. Precision, recall and F1score are averaged when the test dataset is balanced enough.

- **SVM.** The Support Vectors Machine Classifier (SVC in scikit-learn [20]) is computed with the following parameters:
  - coefficient C = 1
  - the class_weight is balanced
  - the probability estimate are turned on
  - all other parameters are set to default
- **Nearest Neighbors.** With Nearest Neighbors Classification (KNeighborsClassifier in scikit-learn [20]) as explained by scikit-learn's [20] documentation the classification is computed from a simple majority vote of the nearest neighbors of each point. After few tests we decided to compute this model with the following parameters:
  - k-nearest neighbors is 10
  - the weight function used in prediction is uniform
  - all other parameters are set to default
- **Decision Tree.** Classification with Decision trees (DecisionTreeClassifier in scikit-learn [20]) consists in learning simple decision rules to classify the data. The results are computed with the following parameters:
  - the class_weight is balanced
  - we also fixed the random_state (to 1) to maintain consistency with reiterations
  - all other parameters are set to default

Then in order to evaluate the influence of using different languages for the test and training set, we computed the same experiments on monolingual datasets for French and English. The training dataset is kept balanced and of size 10000. The same models and configurations are tested. This experiment will let us understand:

- How accurately can we predict the identity of a speaker speaking in language L with a model trained on this language L?
- Are those results linked to the ones on a cross-lingual mode?

3.2 Unsupervised models

For the unsupervised approach, we used clustering algorithms to distinguish speech frames at our disposal. Using the whole MFCC and Speaker Trait feature set, we used different models to see how the frames were clustered. For these models we tried different numbers of clusters ranging from 3 to 10.

Dataset 200000 frames are randomly selected from the whole dataset and only those are clustered for computational efficiency.

Models To compare the performance of different kinds of unsupervised methods the metrics presented in 2.4.2, ARI, ACP and ACLP are used.

- **K-means.** The K-Means algorithm (KMeans in scikit-learn [20]) clusters data into groups of equal variance.
- **Gaussian Mixture.** The Gaussian Mixture model (GaussianMixture in scikit-learn [20]) is a probabilistic model using the EM algorithm. The method iterates until the computed likelihood converges.
- **Agglomerative Clustering.** The Agglomerative Clustering algorithm (AgglomerativeClustering in scikit-learn [20]) is a bottom up hierarchical clustering method.

In the last three cases we set the number of clusters to different values ranging from 3 to 50, and used the default parameters. The choice of high values was made knowing that the results would have to be interpreted carefully: for example with too many clusters the purity could be improved only if fewer frames are in unidentified clusters. However, given the size of the dataset we thought that 50 was still a reasonable number even though the number of speakers is lower. Extra clusters could represent silences or other non-speech sounds that are present in the recordings.

- **DBSCAN.** The Density Based Spatial Clustering of Applications with Noise algorithm [7] consist of creating clusters depending on the density of points in an area. For this model, we can influence two parameters
  - eps: corresponds to the maximum distance between two samples to be consider in the same neighborhood.
  - min_samples: is the minimal number of samples to have a core point.

  In order to find the best parameters, we used a method of trial and error. We try the eps from 1 to 6 with a step of 0.5 and changing the min_samples for 5 to 18.

4 Results

4.1 Supervised models

The first experiment that consisted in training on French samples and testing on English, German and Spanish with a supervised approach provided promising results. It can be seen in Table.1, the best results were obtained with the Speaker trait feature set (standardize per speaker) with the Decision Tree Classifier.

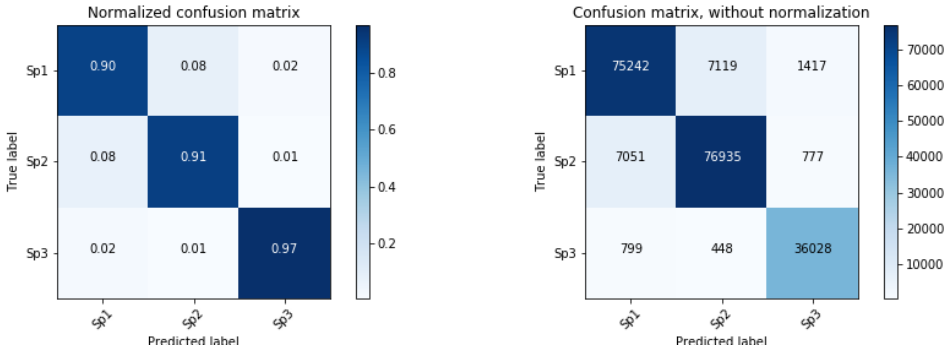
Feature set	**Metrics**	**SVM**			**kNN**			**Tree**		
		Sp1	Sp2	Sp3	Sp1	Sp2	Sp3	Sp1	Sp2	Sp3
	Accuracy		**81.12**%			67.03%			57.25%	
MFCC	Precision	83%	81%	78%	64%	86%	55%	62%	60%	45%
	Recall	80%	83%	81%	85%	49%	68%	55%	59%	59%
	F1	81%	82%	79%	73%	63%	61%	58%	60%	51%
	Accuracy		76.38%			67.24%			52.72%	
MFCC	Precision	79%	80%	65%	71%	69%	54%	59%	61%	33%
Speaker Std*	Recall	79%	76%	72%	69%	70%	56%	53%	54%	50%
	F1	79%	78%	68%	70%	69%	55%	56%	57%	40%
	Accuracy		83.65%			72.28%			66.02%	
Speaker Trait	Precision	88%	83%	76%	71%	83%	60%	69%	70%	54%
	Recall	80%	86%	86%	82%	62%	75%	64%	67%	69%
	F1	84%	85%	81%	76%	71%	67%	66%	69%	61%
	Accuracy		83.56%			72.06%			**91.44**%	
Speaker Trait	Precision	85%	88%	73%	72%	75%	64%	91%	91%	94%
Speaker Std*	Recall	86%	83%	80%	79%	73%	53%	90%	91%	97%
	F1	85%	85%	76%	75%	74%	58%	90%	91%	95%

Table 2: Model comparison with different metrics on both feature set - 10000 samples used for training - No dimensionality reduction

As you can see on the confusion matrix generated for those results Fig.2 and according to the precision and recall values associated, speakers have comparable results especially as Sp1 and Sp2 were more represented in the test dataset (see 2.1 and Table.1)

The second experiment that consisted in training monolingual sets resulted in very high accuracies on the test datasets (See Table.3. If those accuracies were lower we would not have had the same results in the cross-lingual experiment.

We can notice that the Standardization per Speaker worked better with the Speaker Trait feature set as well as the association of Standardization per Speaker on the Speaker trait feature set with the Decision Tree Classifier.



(a) Normalized (b) Not Normalized

Figure 2: Confusion Matrix - Speaker trait feature set standardize per Speaker - Decision Tree classifiers

	Feature set	Metrics	SVM	kNN	Tree
French	MFCC	Accuracy	**86.64**%	74.06%	61.02%
	MFCC - Speaker Std*	Accuracy	82.80%	72.14%	57.07%
	Speaker Trait	Accuracy	89.21%	78.91%	72.0 2%
	Speaker Trait - Speaker Std*	Accuracy	90.91%	77.23%	**95.13**%
English	MFCC	Accuracy	**85.30**%	72.53%	60.19%
	MFCC - Speaker Std*	Accuracy	82.15%	73.39%	56.03%
	Speaker Trait	Accuracy	86.90%	76.24%	70.79%
	Speaker Trait - Speaker Std*	Accuracy	89.53%	79.74%	**93.86**%

Table 3: Model comparison with different metrics on both feature set - 10000 samples used for training - No dimensionality reduction - testing on the left samples in the same language as testing

4.2 Unsupervised models

The different algorithms using the two features set and changing the number of clusters as an input are compared (see Fig 3).

We can see that the MFCC features produce generally higher metrics, and that the Gaussian Mixture Model (GMM) is better to separate speakers and languages. With a number of clusters lower than 10 the speakers are not really separated and the clusters are close to the value 1/3. The languages are separated better as the ACLP values are higher. These high values could also be interpreted as a bias in the dataset given that Spanish and German were spoken by one speaker each. For higher numbers of clusters we can get better results in terms of purity, but it the ARI tends to drop, meaning that the speakers become more "diluted" in the clusters.

4.2.1 DBSCAN

We tried the DBSCAN model with the purpose of clustering the different speakers or languages. This is was not successful as shown in Table.4 where 0.02 portion of the MFCC dataset was used. The table is a sample of the result we get with the DBSCAN and illustrate the difficulty we have encountered this this method. In the best case (the first presented), only 2.7% of the data are affected to a cluster. The repartition values in the table indicate after creation of the clusters, in which clusters the different speaker/language are affected. The last cluster entitled 'left' corresponds to the numbers of data of the current identity that have not been affected to any clusters.

As shown in Table.4, we can see that the number of clusters detected is not always the right one. And even if it is (i.e. for the first case, they could have correspond to the number of speaker), the repartition of the data does not match an expected clustering. The represented clusters do not correspond to any defined group (neither the speakers, nor the languages). This also explains the very low percentage of

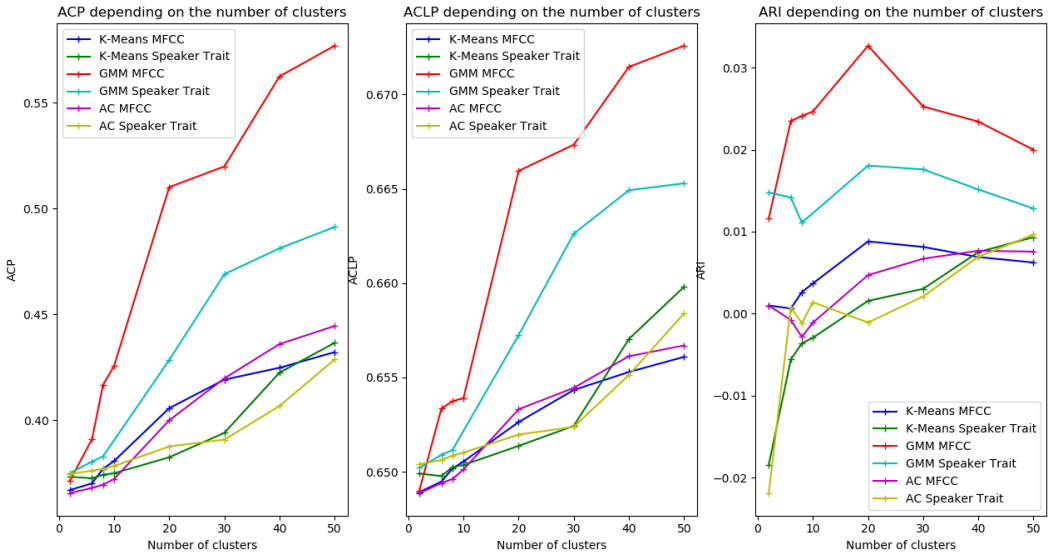


Figure 3: Model comparison and datasets for different numbers of clusters

data that have been clustered. We also ran tests on the Speaker_trait features but the results were very similar.

	Values	% of data clustered	#cluster	Speaker Repartition			Language Repartition			
				Sp1	Sp2	Sp3	EN	FR	DE	ES
eps	4.5	2.70	1	16451	11545	7689	25114	5751	2589	2231
min_s	17		left	7178	12401	4735	16767	4248	2123	1176
eps	3.5	1.05	1	5044	2781	2431	7049	1749	733	725
			2	14	23	19	35	7	8	6
min_s	15		3	0	26	0	17	1	8	0
			left	18571	21116	9974	34780	8242	3965	2676
eps	3.5	1.06	7							
min_s	11									
eps	5	1.16	1	20322	16468	10122	32808	7750	3505	2849
			2	11	0	0	11	0	0	0
min_s	17		3	10	0	1	7	4	0	0
			left	3286	7478	2301	9055	2245	1207	558

Table 4: Results obtained with DBSCAN for the MFCC dataset

5 Discussion

The results obtained show us that some algorithms worked better than others to model our data. The idea of classifying speakers in a cross-lingual dataset is certainly achievable especially with supervised algorithms.

However, it would be interesting to try to generalize the results on a bigger dataset with more speakers especially, and eventually get better results for the unsupervised approach.

Indeed, the results obtained can be biased by the data collection step. The dataset has only 3 speakers, 4 languages and the dataset has never been validated by an other entity. Also the noise in the samples may have influenced the obtained results for the unsupervised approach, as the clusters may represent

different kind of noises[1] instead of different speakers or languages. This problem would require more precise attention.

In addition, when people speak in a foreign (non native) language, they tend to speak with different attitudes and social languages cues [9]. This modification in the speech behavior can jeopardized the results of our project and explain the difficulty to find a pattern for a speaker across language.

The first path that one should follow to improve our results would be to focus on improving the quality of the features by removing true silences and other non-speech sounds that do not characterize the speakers (keeping breathing for example). A secondary solution could be to try out deep architectures models on our dataset following the idea of [5]. However, the current simple models have proven to be very efficient and require low computational time and memory.

6 Conclusion

All in all, we saw how different supervised and unsupervised methods could be used to recognize speakers talking in different languages. In the case of supervised learning we confirmed that training a model with a cross-lingual approach provided worst results as a mono-lingual approach, but very decent performances were still obtained for both.

For unsupervised learning methods, more difficulties were found to isolate the speakers. Created clusters tend to represent more the languages used that the speakers themselves. However, neither languages nor speakers were purely represented. One way to corroborate these results would be to use a larger set of recordings with more speakers and more languages. Other extracting features techniques and audio pre-processing could improve the clustering process and should be tried as well.

References

[1] X. Anguera, C. Wooters, and J. Hernando. Frame purification for cluster comparison in speaker diarization. 01 2006.

[2] R. Askar, D. Wang, F. Bie, J. Wang, and T. F. Zheng. Cross-lingual speaker verification based on linear transform. In *2015 IEEE China Summit and International Conference on Signal and Information Processing (ChinaSIP)*, pages 519–523, July 2015.

[3] U. Bhattacharjee and K. Sarmah. A multilingual speech database for speaker recognition. In *2012 IEEE International Conference on Signal Processing, Computing and Control*, pages 1–5, March 2012.

[4] W. M. Campbell, J. P. Campbell, D. A. Reynolds, E. Singer, and P. A. Torres-Carrasquillo. Support vector machines for speaker and language recognition. *Computer Speech & Language*, 20(2):210 – 229, 2006.

[5] K. Chen and A. Salman. Learning speaker-specific characteristics with a deep neural architecture. *IEEE Transactions on Neural Networks*, 22(11):1744–1756, Nov 2011.

[6] S. Chougule and P. P. Rege. Language independent speaker identification. In *2006 IEEE International Conference on Industrial Technology*, pages 364–368, Dec 2006.

[7] M. Ester, H.-P. Kriegel, J. Sander, X. Xu, et al. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Kdd*, volume 96, pages 226–231, 1996.

[8] F. Eyben, F. Weninger, F. Gross, and B. Schuller. Recent developments in opensmile, the munich open-source multimedia feature extractor. In *Proceedings of the 21st ACM International Conference on Multimedia*, MM '13, pages 835–838, New York, NY, USA, 2013. ACM.

[9] H. Giles and A. C. Billings. Assessing language attitudes: Speaker evaluation studies. *The handbook of applied linguistics*, pages 187–209, 2004.

[10] S. M. Kamruzzaman, A. N. M. R. Karim, M. S. Islam, and M. E. Haque. Speaker Identification using MFCC-Domain Support Vector Machine. *arXiv:1009.4972 [cs]*, Sept. 2010. arXiv: 1009.4972.

[11] M. Kotti, V. Moschou, and C. Kotropoulos. Review: Speaker segmentation and clustering. *Signal Process.*, 88(5):1091–1124, May 2008.

[12] G. L. Lan, S. Meignier, D. Charlet, and P. Deléglise. Speaker diarization with unsupervised training framework. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5560–5564, March 2016.

[13] F. Y. Leu and G. L. Lin. An mfcc-based speaker identification system. In *2017 IEEE 31st International Conference on Advanced Information Networking and Applications (AINA)*, pages 1055–1062, March 2017.

[14] L. Li, D. Wang, A. Rozi, and T. F. Zheng. Cross-lingual Speaker Verification with Deep Feature Learning. *arXiv:1706.07861 [cs]*, June 2017. arXiv: 1706.07861.

[15] L. Li, W. Wang, and S. He. Grid-density based feature classification for speaker recognition. In *Anti-Counterfeiting, Security and Identification (ASID), 2012 International Conference on*, pages 1–4. IEEE, 2012.

[16] Z. Liu, Z. Wu, T. Li, J. Li, and C. Shen. Gmm and cnn hybrid method for short utterance speaker recognition. *IEEE Transactions on Industrial Informatics*, pages 1–1, 2018.

[17] M. H. Moattar and M. M. Homayounpour. A review on speaker diarization systems and approaches. *Speech Communication*, 54:1065–1103, 2012.

[18] B. G. Nagaraja and H. S. Jayanna. Multilingual Speaker Identification with the Constraint of Limited Data Using Multitaper MFCC. In S. M. Thampi, A. Y. Zomaya, T. Strufe, J. M. Alcaraz Calero, and T. Thomas, editors, *Recent Trends in Computer Networks and Distributed Systems Security*, pages 127–134, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg.

[19] J. W. Nicholson, R. A. Bowser, and A. Kumaki. Verbal command processing based on speaker recognition, Mar. 28 2017. US Patent 9,607,137.

[20] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[21] R. Ranjan, S. K. Singh, A. Shukla, and R. Tiwari. Text-dependent multilingual speaker identification for indian languages using artificial neural network. In *2010 3rd International Conference on Emerging Trends in Engineering and Technology*, pages 632–635, Nov 2010.

[22] B. Schuller, S. Steidl, A. Batliner, F. Burkhardt, L. Devillers, C. Müller, and S. Narayanan. The interspeech 2010 paralinguistic challenge. In *In Proc. Interspeech*, 2010.

[23] B. Schuller, S. Steidl, A. Batliner, E. Nöth, R. Vinciarelli, F. Burkhardt, R. V. Son, F. Weninger, F. Eyben, T. Bocklet, G. Mohammadi, and B. Weiss. The interspeech 2012 speaker trait challenge. In *in Proc. Interspeech 2012*, 2012.

[24] E. Simancas-Acevedo, A. Kurematsu, M. Nakano Miyatake, and H. Perez-Meana2. Speaker Recognition Using Gaussian Mixtures Models. In J. Mira and A. Prieto, editors, *Bio-Inspired Applications of Connectionism*, pages 287–294, Berlin, Heidelberg, 2001. Springer Berlin Heidelberg.

[25] TNS Opinion. Special Eurobarometer 386: Europeans and their Languages - ecodp.common.ckan.site_title, Mar. 2012.

[26] R. Vergin, D. O'Shaughnessy, and A. Farhat. Generalized mel frequency cepstral coefficients for large-vocabulary speaker-independent continuous-speech recognition. *IEEE Transactions on Speech and Audio Processing*, 7(5):525–532, Sep 1999.