Machine learning project "Replication study" Gender classification by voice

Parubchenko Aleksandr, Snorovikhina Viktoriia

Abstract

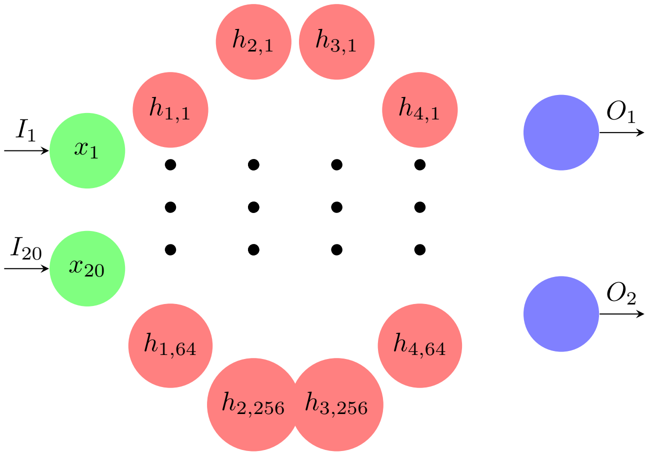
In the following report replication of the article "Voice Gender Recognition Using Deep Learning" is described. Results of the authors were proved with an experiment on their dataset. However, models such as XGboost, Random Forest surpassed the results of author‘s model. To develop the topic, experiments were conducted with other datasets. It contained raw audio with "clean" speech from Audiobooks. Our approach of feature extraction was designed . Experiments were conducted with another dataset from TED-Talks, which contained raw "noisy" audio. Feature extraction was made using R-package warbleR as well as with our approach. Results of all models and datasets were compared.

1 What was stated in the article?

The goal of the article [1] is to predict gender of a person by his/her voice. Authors attached ready csv file with 20 features, which were extracted with package warble-R from audios (did not stated which ones). All of the features represent some statistics of Fast Fourier Transform: median, mean frequency, first, third quantile, interquantile range , skewness, kurtosis, spectral entropy, spectral flatness, mode frequency, frequency centroid, peak frequency, average, min, max of fundamental frequency, average, min, max of dominant frequency, range of dominant frequency, modulation index.
In order to make a classification they used Deep Learning.

1.1 Multilayer Perceptron Neural Network

- 4 hidden layers: 64 perceptrons - 256 perceptrons - 256 perceptrons - 64 perceptrons
- 0.25 dropout between layers
- Activation function in hidden layers: tanh, Activation function in output layer: softmax
- Loss function: Kullback Liebler
- Learning rate 0.01, 150 epochs
- Nadam optimization



1.2 Article Results

Authors states that their MLP gives 96.7% accuracy.
Moreover, they mentioned other Ml methods for solving this task, referring to article [2], but they did not conduct an experiment with these models.
The following methods were mentioned: Random Forest, Decision tree, Stacked Tree, Logistic Regression, SVM classifier, Stacked model (predictions of SVM+RF+XGboost formed new dataset, then XGboost make a prediction on it). We decided to choose **RF, XGboost, SVM, Stacked model** and try out them in order to solve the problem (best one were chosen in advance). We have thought of solving problem by using raw audio itself as input for neural network, but we red that mostly it used for speech recognition. For simple problem such as binary classifications mostly used features extraction, because it gives acceptable results with low computational complexity. So, we decided to focus on this topic.

2 Replicating results

- Was used ready csv file with 20 features, which was attached to the article
- It was checked that classes are balanced
- It was checked that data is normalized
- "Accuracy " metric was used

MLP	**Random Forest**	**XGboost**	**Stacked model**
0.969	0.978	0.976	0.967

It means that results is truly indeed, we got same 96.7% accuracy as authors, BUT more simple model such as RF and XGboost worked better and got higher scores.

3 Moving on - more than a replicating

3.1 Trying another data

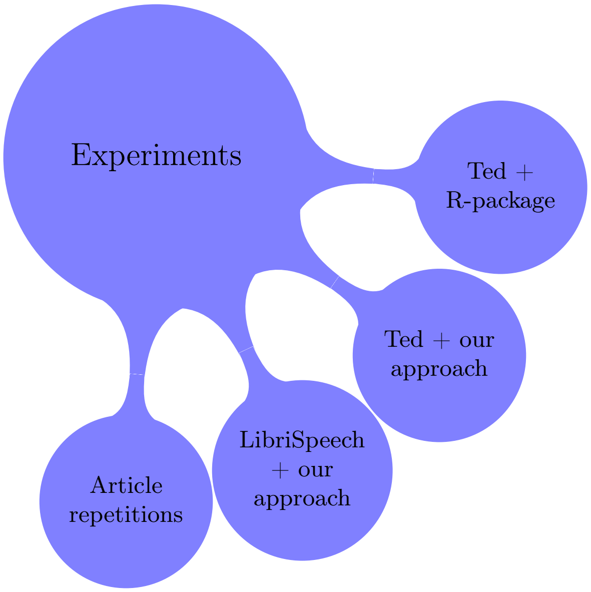
- We have found dataset **LibriSpeech** [3]. It contains audiobooks (kind of, people just red books and it was recorded). It has audio in a flac format. The records of the speech in the corpus are clean, it means that there are no external sounds or noise.
- We have found another dataset from **TED-talks** [4]. It contains records from real public speeches from a well-known persons. It includes clapping, giggles and external noise sometimes.

3.2 Feature extraction

- **R-package.** It is stated in the article that article‘s csv file, was made with package warbleR. This package has function Specan, which measures acoustic parameters on acoustic signals. This functions suits for wav-format audio files only.
- **Our approach.** As soon as we downloaded first dataset LibriSpeech with flac-format, we came up with the idea to design our implementation of Specan. Some of the statistics of Fast Fourier Transform such as median, mean frequency, first, third quantile, IQR, skewness, kurtosis were implemented in python. For more complicated features was used package Rpy2, which enables usage of R function in Python notebook. So, other features were calculated using functions from seewave package - R package for sound analysis and synthesis. In order to understand the nature of features and the right way of creating it, we used article [5].

3.3 Final csv-datasets

- For **LibriSpeech** we had audios with different length (somewhere very small like 3 seconds), so we decided not to cut them. Average length is around 10 seconds per each persons. The classes "female" and "male" were balanced (602 and 564). The data was normalized. Dataset contains 1166 rows.
- For **TED-talks** inverse to LibriSpeech we had audios with long speeches like 1 hour, so we cut all of them and the length became 30 seconds per each person. Records were in sph-format, in order to use it with Specan we converted it in wav-format. The classes "female" and "male" were unbalanced (523 male and 251 female). We made a bootstrap of female samples and add copies to dataset in order to make classes balanced. The final dataset contains 523 males and 521 females. The data was normalized. Dataset contains 774 rows.
- We created file **process_input.py**, which includes all the preprocessing details and needs from raw audio for any format to csv files.



4 Obtained Results

Model/ Data	**MLP**	**Random Forest**	**XGboost**	**Stacked model**	**SVM**
Article	0.969	0.978	0.976	0.967	0.718
LibriSp.+ f.reduction	0.820	0.849	0.860	0.829	0.834
LibriSp.	0.826	0.840	0.871	0.860	0.800
Ted+our approach	0.733	0.687	0.713	0.809	0.814
Ted+R-package	0.813	0.745	0.832	0.846	0.814

4.1 Conclusions

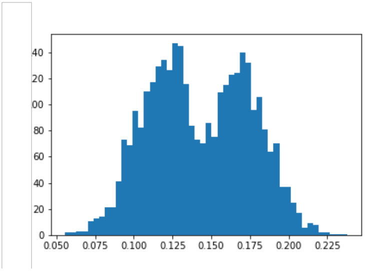
- **LibriSpeech + our approach**
  - Best model is Xgboost with 86% accuracy
  - MLP from the article did not give such a high score as 97.6%, so the architecture of the network can not be generalize as universal model for voice-gender classification.
  - Generally, we see decreasing of accuracy on this dataset in compare with data from the article, we have an assumption why it is happened, which would be described later.

- Feature reduction was made , only 5 most important features were left. It has decreased and increased accuracy for different models, but result is around the same. However, we prefer 5-dimensional model: simple models are more preferable (according to AIC criterion for example) and interpret .

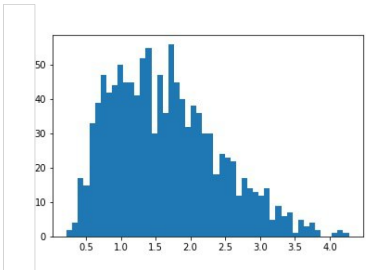
- **Ted datasets**
  - Generally scores decreased in compare with previous datasets. We expected such results, because of different kind of noises in the records.
  - Generally scores of TED with our approach are lower in compare with TED from R-package.
  - Best model for TED+R-package is Stacked model with 84,6% accuracy.
  - Best model for TED+our approach is SVM with 81,4% accuracy. Stacked model is failed just by 0.005 and it can be seen that probably that is because RF worked badly.
  - We see that Random Forest started to work worse with adding noise in both TED datasets.
  - Still MLP from the article did not achieved high scores, this again indicates impossibility of generalizing its architecture for this task.

4.2 Some guesses of decreasing scores in compare with the article and with our approach of feature extraction

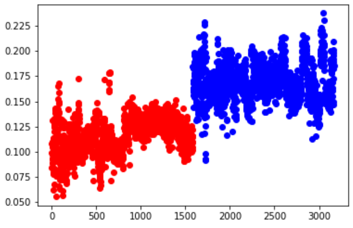
- It is a well-known fact that males and females with at least some accuracy could be separable with a fundamental frequency of a voice. We made a feature importance in LibriSpeech and article datasets and saw mean fundamental frequency on the top. We decided to look at the distribution.



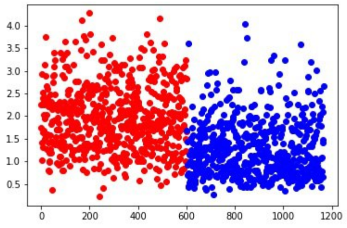
a) Distribution of mean fundamental freq, article dataset



b) Distribution of mean fundamental freq, LibriSpeech dataset

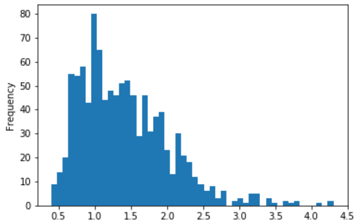


a) Distribution of classes w.r.t. mean fundamental freq, article dataset

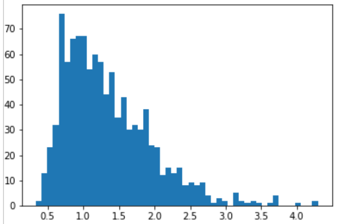


b) Distribution of classes w.r.t. mean fundamental freq, LibriSpeech dataset

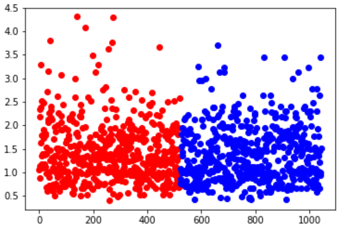
- Here we see obviously 2 Gaussians in data from article and clearly see separability of males and females, what can‘t be said about our dataset made from LibriSpeech. Here we came up with the idea that our approach of feature extraction is wrong and decided to compare distribution of mean fundamental frequency in same dataset TED talks.



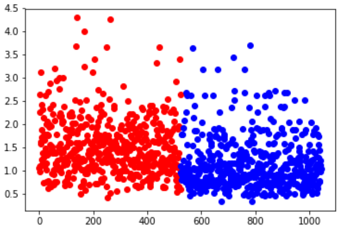
a) Distribution of mean fundamental freq, TED+our approach



b) Distribution of mean fundamental freq, TED + R-package



a) Distribution of mean classes w.r.t. fundamental freq, TED+our approach

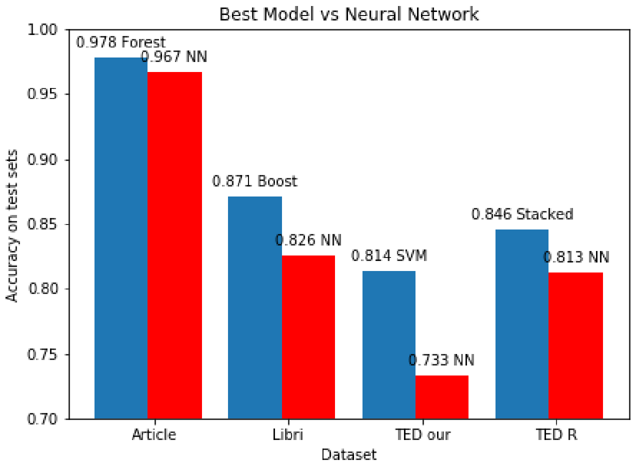


b) Distribution of classes w.r.t. mean fundamental freq, TED + R-package

- There are no 2 peaks even with feature extraction with R-package. It seems to us very strange and suspicious. There is nothing could be done wrong with just putting audio in specan function, but still we can not see 2 Gaussians. It consequences questions about data from the article.

5 Conclusions

- Architecture of neural network can no be generalized for solving gender classification problem by voice.
- We have found more simple methods like Random Forest and Xgboost, which managed to surpassed scores of neural network.
- Our approach of R package is differ from specan from warbleR. It decreased accuracy.
- Noise in TED-talks dataset decreased the accuracy.
- Here is final result with best scores and models for 4 datasets.



6 Work split

- Victoria Snorovikhina made Jupyter notebooks called "Project_Self_Made", "Process_TED", so it means all the research related to LibriSpeeach corpus and TED with our approach of feature extraction, including designing MLP neural network used through all datasets. She also created this approach of feature extraction, which can be found in the file process_input.py : "process_audio".
- Alexander Parubchenko made Jupyter notebooks called "Project_Data_from_article" , "Process_TED_via_specan", so it means all the research related to Dataset from the article and TED with features used R-package. He also created all preprocessing needs, which contains parallelization of audio processing. It means all the functions in file process_input.py, besides "process_audio".

7 References

1 Voice Gender Recognition Using Deep Learning, Mücahit BüyükyılmazAli ,Osman ÇıbıkdikenAli, 2016

2 K. Becker, “Identifying the Gender of a Voice using Machine Learning”, 2016, unpublished

3 http://www.openslr.org/12

4 http://www.openslr.org/7/

5 Speaker Gender Recognition system, Zimeng Hong, 2017