Using Deep Learning for Detecting Spoofing Attacks on Speech Signals

Alan Godoy¹,², Flávio Simões³, José Augusto Stuchi³, Marcus de Assis Angeloni¹, Mário Uliani¹, Ricardo Violato¹

¹CPqD Foundation, Campinas, Brazil  ²University of Campinas, Campinas, Brazil
{amello,simoes,jastuchi,massis,uliani,rviolato}@cpqd.com.br

Abstract

It is well known that speaker verification systems are subject to spoofing attacks. The Automatic Speaker Verification Spoofing and Countermeasures Challenge – ASVSpoof2015 – provides a standard spoofing database, containing attacks based on synthetic speech, along with a protocol for experiments. This paper describes CPqD’s systems submitted to the ASVSpoof2015 Challenge, based on deep neural networks, working both as a classifier and as a feature extraction module for a GMM and a SVM classifier. Results show the validity of this approach, achieving less than 0.5% EER for known attacks.

Index Terms: Speaker Verification, Spoofing Countermeasures, Deep Neural Networks

1. Method

1.1. Feature Extraction

Aiming at detecting if an audio is authentic or not, a deep neural network based on a multilayer perceptron (MLP) architecture was used as a feature extraction module [1]. In a bottleneck approach [2], the network output layer is removed and the activations of the last hidden layer neurons are treated as new features for future classification. Figure 1 shows how audio was processed, from feature extraction to network supervised training.

Instead of feeding raw signal directly as input to the network, a pre-processing step was performed in order to transform input signals into sequences of feature vectors. This decision was based on preliminary tests, which indicated such a step was able to improve the learning rate and allowed the use of more compact networks. Therefore, each signal file is divided into a sequence of 20 ms consecutive non-overlapping frames. No window function is applied. In parallel, a voice activity detection method based on ITU G.729B [3] is applied, so each frame is classified as speech/non-speech and only speech frames are preserved.

Different representations were tested as input for the MLP, including the raw speech frame itself, Mel Frequency Cepstral Coefficients (MFCC) [4], Modified Group Delay Cepstral Coefficients (MGDCC) [5] and Discrete Fourier Transform (DFT) coefficients. Nevertheless, better results were achieved with the Discrete Cosine Transform (DCT) coefficients. The DCT has the energy compaction property, which concentrates most of the signal information in a few low-frequency components [6]. For this reason, the first 128 DCT coefficients are used as feature for each active speech frame.

In order to avoid loss of long term information that can possibly be used to distinguish spoofing attacks, when an input is presented to the MLP, each central speech frame is surrounded by its ten previous frames and the ten following ones, including silence frames [7]. Thus, a vector with 2688 features is used as network input.

The backpropagation algorithm, in conjunction with the Stochastic Gradient Descent optimization technique [8], was applied to train the network to classify whether the input represents an authentic (human) or spoofed audio frame. Ground
truth consists of a label indicating if the input audio is authentic or belongs to one of five spoofing categories, named S1, S2, S3, S4 or S5 [9]. Preliminary experiments indicated that using only two classes – spoofing and human – as output led to poor performance in class S1. One hypothesis is that this could happen because S1 distinguishes from other attacks since it is based on a unit selection algorithm, which concatenates pieces of authentic signal to create a new audio. To deal with this, it was decided to drive the network training towards distinguishing S1 from the other spoofing attacks, increasing the relevance (on network performance) of detecting borders between pieces of authentic speech. Thus, three classes were created, as depicted in Table 1: authentic human speech (100), S1 spoofing attack (010) and other spoofing attacks (001).

Table 1: MLP classes output meanings.

| y0 | y1 | y2 | Meaning   |
|----|----|----|-----------|
| 1  | 0  | 0  | human     |
| 0  | 1  | 0  | S1 attack |
| 0  | 0  | 1  | S2, S3, S4, S5 attacks |

Figure 2 shows the MLP deep architecture used in this paper. 1024 neurons were used in the first hidden layer, 512 in the second hidden layer and 32 in the last one. The last hidden layer is artificially small in order to create a bottleneck, which compress signal information useful for spoofing classification in a low-dimensional representation [2]. Each hidden layer uses the logistic function as activation. The output consists of 3 neurons, each one with softmax activation function, returning a real number between 0 and 1. After finishing the network training, the output layer was removed and the activations of the last hidden layer neurons were used as new output, extracting the bottleneck features, as indicated in Figure 2.

1.2. Classification

Three different classifiers were tested: Support Vector Machines (SVM), Gaussian Mixture Models (GMM) and Multilayer Perceptron. In the cases of the SVM and the GMM classifiers, feature extraction took an additional step. Since each audio file has a different duration and, thus, a different number of frames, feature vectors over all frames were averaged so that each file was represented by a single fixed-size 32-dimensional feature vector [10].

A SVM classifier [11] based on the Radial Basis Function (RBF) kernel was generated. Samples from the training set were computed and used to train the SVM-RBF. All spoofing attacks were considered as a single negative class for training.

The SVM-RBF classifier parameters C (controls the cost of misclassification on the training data) and \( \gamma \) (parameter of a Gaussian kernel to handle nonlinear classification) were tuned by performing grid search with K-fold cross-validation over the train set, using 5 folds. Values of 0.001, 0.01, 0.1, 1.0, 10.0, 100.0, 1000.0 and 10000.0 were searched both for C and \( \gamma \). Optimum parameters were chosen aiming at minimizing the average equal error rate (EER) over all 5 folds. After this search, optimum values of \( C = 0.1 \) and \( \gamma = 10 \) were found and the SVM-RBF classifier was retrained with the whole training set. SVM-RBF outputs vary in the interval \([0, 1]\) and represent the likelihood of the test sample belonging to positive class, i.e., authentic speech audio.

Figure 2: MLP used for feature extraction and classification

For the GMM based classifier, two GMMs were trained, one with authentic audios and another with spoofed audios. The following number of Gaussian mixtures were tested: 4, 8, 32, 64, 128, 256 and 512, wherein 8 mixtures gave the lowest EER on the development set. The classifier output is given by the log-likelihood ratio of authentic GMM with respect to spoofing GMM.

The third and last tested approach consisted of using the MLP trained for feature extraction directly as a classifier, without the removal of the output layer. In this case, the feature extraction was merged with the classification step.

As the network last layer returns three values using the softmax function, according to presented in Figure 2 only \( y_0 \) is considered, since it represents the likelihood of being an authentic speech. Thus, values for this third approach vary in the interval \([0, 1]\). A score \( y_0 \) was then calculated for each frame in the audio file, generating a score array for the entire audio. This array was used to compute a unique score for the audio sample. To do so, aiming at removing outliers within the audio file, the first 15% lower array values are removed as well as the 25% higher values. The remaining 60% of the scores were then averaged, resulting in the final score.

2. Acknowledgements

We would like to gratefully thank MSc Christopher Dane Shulby for this paper review and helpful comments.
3. References

[1] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A.-r. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath et al., “Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups,” Signal Processing Magazine, IEEE, vol. 29, no. 6, pp. 82–97, 2012.

[2] D. Yu and M. L. Seltzer, “Improved bottleneck features using pre-trained deep neural networks,” in INTERSPEECH, vol. 237, 2011, p. 240.

[3] A. Benyassine, E. Shlomot, H.-Y. Su, D. Massaloux, C. Lamblin, and J.-P. Petit, “Itu-t recommendation g. 729 annex b: a silence compression scheme for use with g. 729 optimized for v. 70 digital simultaneous voice and data applications,” Communications Magazine, IEEE, vol. 35, no. 9, pp. 64–73, 1997.

[4] S. Davis and P. Mermelstein, “Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences,” Acoustics, Speech and Signal Processing, IEEE Transactions on, vol. 28, no. 4, pp. 357–366, 1980.

[5] Z. Wu, X. Xiao, E. S. Chang, and H. Li, “Synthetic speech detection using temporal modulation feature,” in Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on. IEEE, 2013, pp. 7234–7238.

[6] N. Ahmed, T. Natarajan, and K. R. Rao, “Discrete cosine transform,” Computers, IEEE Transactions on, vol. 100, no. 1, pp. 90–93, 1974.

[7] E. Variani, X. Lei, E. McDermott, I. L. Moreno, and J. Gonzalez-Dominguez, “Deep neural networks for small footprint text-dependent speaker verification,” in Acoustics, Speech and Signal Processing (ICASSP). 2014 IEEE International Conference on. IEEE, 2014, pp. 4052–4056.

[8] S. O. Haykin, Neural networks and learning machines. Pearson Education Upper Saddle River, 2009, vol. 3.

[9] Z. Wu, T. Kinnunen, N. Evans, J. Yamagishi, C. Hanilci, M. Sahidullah, and A. Sizov, “Asvspoof 2015: the first automatic speaker verification spoofing and countermeasures challenge,” in INTERSPEECH 2015 – 16th Annual Conference of the International Speech Communication Association, September 6–10, Dresden, Germany, Proceedings, 2015, p. Submitted.

[10] M. J. R. F. Correia, “Anti-spoofing: Speaker verification vs. voice conversion,” Master’s thesis, Instituto Superior Técnico Lisboa, 2014.

[11] C. Cortes and V. Vapnik, “Support-vector networks,” Machine Learning, vol. 20, no. 3, pp. 273–297, 1995. [Online]. Available: http://dx.doi.org/10.1007/BF00994018