2nd Hellenic Conf. on AI, SETN-2002, 11-12 April 2002, Thessaloniki, Greece,
Proceedings, Companion Volume, pp. 525-530

Development of a Text-Dependent Speaker Identification System with the OGI Toolkit

Asterios Toutios, K. G. Margaritis

PDP Lab, Department of Applied Informatics,
University of Macedonia, 54006 Thessaloniki, Greece
{toutios, kmarg}@uom.gr

Abstract. This paper discusses the development a text-dependent speaker identification system. In particular, we develop a neural network, collect the input corpus, train, and measure its efficiency. The development is done with the CSLU-NN environment, a set of programs in Tcl and C, part of the OGI toolkit, originally built for speech recognition tasks, and adapted to our speaker recognition needs. Thus we propose a well-described and quite simple way for building pattern recognizers that deal with .wav files, in general. Our results demonstrate the efficiency of our implementation.

1 Introduction

Speaker recognition is the process of automatically recognizing who is speaking on the basis of individual information included in speech waves. This technique can make it possible to use the speaker's voice to verify their identity and control access to various services. These services include voice dialing, banking by telephone, telephone shopping, database access services, information services, voice mail, security control for confidential information areas, and remote access to computers [1]. From the viewpoint of technology, speaker recognition is a general term, which refers to any task to discriminate people based upon their voice characteristics [2].

Speaker recognition can be divided into speaker verification and speaker identification. Speaker identification is the process of determining which registered speaker provides a given utterance. Speaker verification, on the other hand, is the process of accepting or rejecting the identity claim of a speaker. Speaker identification is the most difficult problem of the two.

Speaker recognition systems can also be divided into text-dependent and text-independent. By text-dependent, we mean that the text in both training and testing is the same or is known. On the contrary, text-independent systems do not rely on a specific text being spoken.

Moreover, speaker identification can be subdivided into two further categories, closed-set and open-set problems. The closed set problem is to identify a speaker from a group of N known speakers. In the open set problem, a reference model for an unknown speaker may not exist. In this situation, an additional decision alternative, that the unknown does not match any of the models, is required.

In this paper we consider only text-dependent speaker identification, for both the open-set and closed-set problems.

The remainder of the paper is organized as follows. Section 2 presents our main working tool, the OGI, or CSLU, Toolkit. In section 3 an overview of our speaker identification system is addressed. In section 4 our experimental results are presented. Conclusions are drawn in the final section. In the appendix, we present the changes done to the original CSLU method for training a digit recognizer.

2 The OGI Toolkit

Since the early nineties, the Center for Spoken Language Understanding (CSLU) at the Oregon Graduate Institute of Science and Technology (OGI), has been working on the development of new tools for creating spoken language systems. The result of this effort is the CSLU Toolkit, an integrated set of software and documentation that represents the state of the art in tools for research, development and learning about spoken language systems. The CSLU Toolkit is freely available for non-commercial use and may be downloaded from http://cslu.cse.ogi.edu/toolkit.

The CSLU Toolkit has been developed to support speech-related research and development activities for a wide range of users and uses. Among various other topics, the Toolkit is designed to enable domain experts to rapidly design spoken language systems for real applications, even in languages other than English, with easy-to-use authoring tools, generate state-of-the-art spoken language systems automatically from high level design specifications, learn about spoken-dialogue systems through coursework incorporated into the tools, easily perform research on human-computer interaction in many tasks using spoken-dialogue systems, perform research on the underlying technologies, and incorporate research advances into working systems for evaluation in real applications.

The architecture of the Toolkit has three main components: a set of libraries containing core technologies modules specific to speech recognition, speech synthesis and facial animation, an interactive programming shell (CSLUsh) and a graphically-based Rapid Application Development environment (RAD). A part of the first one of these components is the CSLU-NN development environment.

The CSLU-NN development environment is a tool that contains various Tcl/Tk and C functions that implement the general steps needed to create a neural-network based recognizer, such as specifying the categories that the network will recognize, training a network to recognize these categories, and evaluating the network's performance [3,4].

Our speaker identification system is developed using the CSLU-NN development environment.

3 Overview of the System

The text-dependent speaker identification system that we have developed can be divided into four, quite discrete, "subsystems" or, in other words, has to accomplish four tasks: Digitize the spoken utterance; divide it into frames and compute features for each frame; classify each frame as belonging to a specific speaker with a neural network; and, finally, given the neural network's outputs for each frame, determine who the speaker is [5].

3.1 Digitization

The analog spoken utterance is sampled with a frequency of 8000 Hz by a 16-bit A/D converter. This is a CSLU standard and is equivalent to passing the speech signal through a low pass filter with a cut off frequency of 4000 Hz, thus losing the information contained at higher frequencies. It makes up for telephone quality of speech.

3.2 Feature Computation

The utterance is divided into non-overlapping 10 msec frames. For each frame 13 Mel-Frequency Cepstrum Cefficients (MFCC) [6], and 13 Perceptual Linear Prediction (PLP) [7] features are computed. The PLP analysis technique was originally designed to suppress speaker dependent components in features used for automatic speech recognition, but later experiments [8] demonstrated the efficiency of their use for speaker recognition tasks.

For each frame a 130-dimensional vector is constructed. It consists of the MFCC and PLP features of the frame of interest plus the MFCC and PLP features of the frames at –60, -30, 30 and 60 msec relative to it. These vectors will be the input of the neural network.

3.3 Neural Network

A neural network is used to classify each frame as belonging to a specific speaker. The network has a three-layered architecture and is trained using the back-propagation algorithm [9]. The number of the input nodes is equal to the size of the input vectors e.g. 130. The number of the output nodes is equal to the number of the registered to the system speakers. Finally, the number of the hidden nodes is chosen by the user.

The learning rate of the network decreases exponentially with each iteration, with the amount of decrease dependent on the number of training vectors. The initial learning rate and the momentum of the network are set by the user. Negative penalty is used to compensate for the fact that there are different numbers of input vectors for each speaker [10].

The outputs of the neural network are used as estimates of the probability, for each speaker, that the current frame belongs to the specific speaker.

3.4 Final Determination of the Speaker

For each utterance, the outputs of the neural network make up a matrix of how the probabilities that the utterance belongs to a specific speaker change with time. To determine who the most likely speaker is we need to search through this matrix and calculate a "score" for each speaker. This is done by with a Viterbi search [11]. The name of the speaker with the highest score is the final output of the system.

4 Experimental Results

We have collected two corpuses for our experiments. The first one (we will call it Set-1) consists of 20 speakers uttering 15 times each the phrase "open sesame" in Greek, summing up to a total of 300 utterances. The speakers were asked to color their voices in various ways during the recording session, a fact that adds difficulty to the speaker recognition task. The second corpus (Set-2) consists of the same 20 speakers uttering a various number of times (15 to 25) the same phrase, with the addition of a relatively large number of "garbage" utterances, summing up to a total of 479 utterances. Our "garbage" utterances make up for the case that the speaker that tests the system is unknown, or a different than the selected phrase is spoken to the system. They comprise of original utterances shifted in the frequency domain, noise and silence samples, and completely irrelevant speech samples.

Set-1 makes up for the case of closed-set identification, while Set-2 makes up for the case of open-set identification.

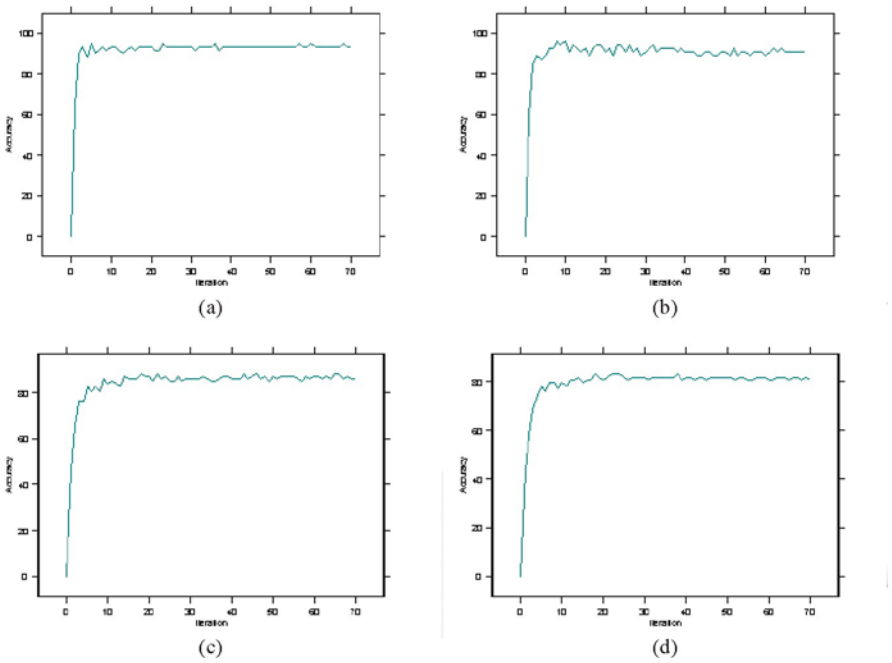
From Set-1, 15 utterances are culled out for third-party evaluation, 173 constitute the training set, 58 the "development set" and 54 the test set. The same numbers for Set-2 are 24, 279, 93 and 83, respectively. Our speaker recognition system is trained on the training sets, and tested on the development and test sets.

Our neural networks have 130 input layers, 200 hidden layers, and 20 or 21 output layers, for Set-1 or Set-2, respectively. Several experiments showed that the initial learning rate should be 0.1, while the momentum should be set to 0.

Training is done for 70 iterations (or epochs). In Figure 1 we present the learning curves for our networks. Table 1 summarizes our experimental results.

Table 1. Accuracy percentage rates for the various Sets.

	Development Set-1	**Test Set-1**	**Development Set-2**	**Test Set-2**
Best Iteration	94,83%	94,44%	88,17%	83,13%
Converged Network	93,10%	90,74%	86,02%	81,93%



(a)

(b)

(c)

(d)

Fig 1. Percentage of accuracy vs. number of iterations plots for: a) Development Set-1, b) Test Set-1, c) Development Set-2, d) Test Set-2

5 Conclusion

We have developed a text-dependent speaker identification system, using the CSLU Toolkit, and specifically the CSLU-NN environment. We have achieved good results on a quite difficult recognition task, thus suggesting the efficiency of the use of the CSLU Toolkit for building such systems.

References

1. Cole R.A., Mariani J., Uszkoreit H., Zaenen A., Zue V. (eds.): Survey of the State of the Art in Human Language Technology. Cambridge University Press, 1997.
2. Doddington G.R.: Speaker Recognition-Identifying People by Their Voices. Proceedings of IEEE, Vol. 73, No. 11, 1986, pp. 1651-1644.
3. Cosi P., Hosom J.P., Schalkwyk J., Sutton S., Cole R.A..: Connected Digit Recognition Experiments with the OGI Toolkit's Neural Network and HMM-Based Recognizers. 4th IEEE Workshop on Interactive Voice Technology for Telecommunications Applications (IVTTA-ETWR98), Turin, Sep. 1998, pp. 135-140.

4. Schalkwyk J., de Villiers J., van Vuuren S., Vermeulen P.: CSLUsh an Extendible Research Environment. In Proceedings of EUROSPEECH ’97, Rhodes, Greece 1997.
5. Hosom J.P., Cole R., Fanty M.: Speech Recognition Using Neural Networks at the Center for Spoken Language Understanding. http://cslu.cse.ogi.edu/tutordemos/nnet_recog/recog.html.
6. Davis S.B., Mermelstein P.: Comparison of Parametric Representations for Monosyllabic Word Recognition in Continuously Spoken Sentences. IEEE Trans Acoust. Speech and Signal Processing, Vol, ASSP-28, No. 4, August 1990, pp. 357-366.
7. Hermansky H., Pavel M.: Psychophysics of Speech Engineering Systems. In Proceedings of the 13th International Congress on Phonetic Science, pp. 42-45, Vol. 3, Stockholm, Sweden, 1995.
8. van Vuuren S.: Comparison of Text-Independent Speaker Recognition Methods on Telephone Speech with Acoustic Mismatch. In Proceedings of ICSLP ’96, pp. 1784-1787,October 1996.
9. Haykin S.: Neural Networks, a Comprehensive Foundation. Macmillan College Publishing Company, 1994.
10. Hosom J.P., Cole R., Fanty M., Schalkwyk J., Yan Y., Wei W.: Training Neural Networks for Speech Recognition. http://cslu.cse.ogi.edu/tutordemos/nnet_training/tutorial.html.
11. http://www.cedar.buffalo.edu/Linguistics/Forms/VitMethod.html.

Appendix

The CSLU method for building a digit recognizer is presented in detail in [10]. Here, we present the changes to this method in order to develop a speaker recognizer.

A typical .phn file is like this:

```
MillisecondsPerFrame: 1.0
END OF HEADER
0 1549 speaker1
```

where the numbers 0 1549 indicate the whole duration of the utterance, while “speaker1” simply indicates the name of the speaker..

A .txt file simply contains the name of the speaker:

```
speaker1
```

The .vocab file is like this:

```
speaker1       {speaker1}                  ;
speaker2       {speaker2}                  ;
speaker3       {speaker3}                  ;
separator      {.pau [.garbage] .pau}      ;
$person   = speaker1 | speaker2 | speaker3 ;
$grammar = ([separator%%] $person [separator%%] );
```

And, finally, the .parts file:

```
.pau            1 ;
speaker1        1 ;
speaker2        1 ;
speaker3        1 ;
$sil   = .pau tc .garbage ;
```