SVM-BASED SPEAKER VERIFICATION FOR CODED AND UNCODED SPEECH

Artur Janicki

Institute of Telecommunications, Warsaw University of Technology
ul. Nowowiejska 15/19, 00-665 Warsaw, Poland

ABSTRACT

This paper describes experiments with speaker verification using support vector machines (SVMs). Verification from coded and uncoded speech is analyzed, both in matched and mismatched conditions. A hybrid SVM-GMM approach is used, in which SVM classifiers with Kullback–Leibler kernel make verification decisions based on the mean values of Gaussian mixtures. The most common narrowband codecs are used, such as G.711, G.729, G.723.1, GSM 06.10, GSM 06.60, and Speex. The Equal Error Rate (EER) is presented for various numbers of Gaussian components, and for various testing conditions. Possible reasons for the non-uniform performance degradation in the case of codec mismatch are discussed. Selected ROC curves are presented. The results are compared with a similar investigation of a close-set speaker classification.

Index Terms— speaker verification, speech coding, support vector machine, speaker recognition, ROC curve

1. INTRODUCTION

A speaker verification system, for example the one used by a bank for customer authorization, mostly works based on the transmitted speech signal, i.e., the signal which has been transcoded by a voice codec. It should therefore work robustly, regardless of whether the customer is calling from a land line, a mobile, or an Internet phone. This is why there is a need to make speaker verification robust not only against a change of microphone or against the speaker's inter-session variability, but also against the various speech codecs used in voice transmission. We deal with the same problem in speaker detection, where we aim at finding a speaker in a large corpus of telephonic (i.e., transcoded) speech.

Support Vector Machines (SVMs) [1] have already been successfully employed for speaker classification from coded speech, see [2], where a multi-class SVM classifier was used. Since the SVM algorithm in its basic form is a binary classifier, the authors wanted to test SVMs in a speaker verification task, which is a binary problem (acceptance/rejection) and needs no additional voting mechanism, as is the case for multi-class problems such as classification.

1.1. SVMs in speaker verification

In speaker recognition, including speaker verification (and detection), SVMs have already been successfully used in numerous studies. In [3], the authors proposed using an SVM to process supervectors containing the mean values of GMM Gaussian components. They used the linear Kullback–Leibler kernel, which for M Gaussian components can be expressed as

$$K(utt_a, utt_b) = \sum_{i=1}^{M} \left(\sqrt{\lambda_i}\Sigma_i^{-\frac{1}{2}}\mu_i^a\right)^T \left(\sqrt{\lambda_i}\Sigma_i^{-\frac{1}{2}}\mu_i^b\right) \quad (1)$$

where λ, μ, and Σ are the ith Gaussian parameters (weight, mean values and covariance matrix) of the utterances a and b. The authors showed ROC curves for a speaker detection task from the NIST SRE 2005 challenge, where SVMs outperformed the classical GMM ATNorm approach, requiring considerably less computational power.

In speaker verification, an SVM speaker model is trained to find a discriminant hyperplane in the score space between the correct speaker and a potential impostor. Therefore, unlike the close-set speaker recognition task, there is a need to model the impostor. In the literature two such methods have been described: (i) using a universal background model (UBM) and using it for each speaker, and (ii) using a set of speakers (a cohort), which should be selected separately for each speaker to efficiently cover the impostors' space. In [4], these two approaches were even combined, resulting in a decrease in the verification error.

In [5], the authors used SVMs with the Fisher kernel and the LR (likelihood ratio) kernel with spherical normalization. On the PolyVar speech corpus they achieved up to 33% relative improvement of speaker verification accuracy compared to GMM-UBM systems.

1.2. Impact of speech coding on speaker recognition

Several studies have already been conducted on speaker recognition from coded speech. In the majority of cases, the

The calculations were made in the Interdisciplinary Centre for Mathematical and Computational Modeling (ICM) of the University of Warsaw (computational grant No. G46-2)

researches used speaker recognition based on Gaussian mixture models, where speaker models were adapted, using, e.g., the MAP (maximum a posteriori) algorithm, from a universal background model (GMM-UBM systems) [6]. Usually two cases are considered: (i) *matched conditions*: when the speaker recognition system trained using speech transcoded with codec X is tested on speech also transcoded with codec X; (ii) *mismatched conditions*: when the system trained using speech transcoded with codec X (or not coded at all) is tested on speech transcoded with codec Y.

So it was for example in [7], where the authors showed for the NIST 1998 speaker recognition evaluation corpus how much the recognition accuracy is affected by transcoding using the GSM 06.10, G.723.1, and G.729 codecs. The authors reported that the GSM 06.10 codec had the best results both in matched and mismatched conditions, but G.723.1 proved to be the worst: Equal Error Rate (EER) rose from 4% to 12% for female speakers, so the performance degradation was consistent with decreasing perceptual quality.

GSM speech codecs were examined in [8], but only in matched conditions. The authors showed that both speaker identification and verification performance is degraded by these codecs, blaming the low LPC order in these codecs. Speaker recognition from speech coded with the GSM 06.60, G.729, G.723.1, or MELP codecs was studied in [9] both in matched and mismatched conditions. The authors used the GMM-UBM technique, with gender-dependent UBM models. They found that the recognition accuracy decreases when the mismatch between the quality of the training and testing codecs increases. It was shown that using handset dependent score normalization (HNORM) improved the results. In various experiments with the Speex codec in [10], it was shown that Speex can serve well also for creating speaker models for testing GSM-encoded speech. In [11], wideband codecs were examined: WMA, AAC and MP3; some loss in recognition accuracy was observed without change of the sample rate, and a significant loss was experienced when the sample rate was changed.

SVMs were employed for speaker recognition from coded speech in [2], which presented that the SVM-GMM approach required a higher number of Gaussian components (M) than the GMM-UBM approach. It was shown, however, that for $M = 256$, an SVM-based classifier yielded much higher accuracy for GSM-transcoded speech than was found in [8] for GMM-UBM.

1.3. The aims of this study

Following the promising results of SVM-based speaker recognition in several studies, including the ones for coded speech, we decided to examine it in the speaker verification task. The following questions were posed in this study: (i) how will the SVM algorithm perform for speaker verification of coded and uncoded speech, compared to the GMM-UBM approach, in matched and mismatched conditions? (ii) What is the difference in performance compared to the classification task? (iii) Which codec would be the best for creating speaker models, which would allow efficient verification independently of the codec used in testing?

The results will be compared with the experiments on speaker verification from speech transcoded with the GSM codecs [8] and the study concerning SVM-based speaker classification from coded speech described in [2].

2. TESTING METHODOLOGY

2.1. Speech data

The TIMIT speech corpus [12] was used as the database of recordings. Although it was originally designed for studies of speech recognition, this corpus has been used as well for a number of studies on speaker recognition (e.g., [8] and [11]), as it contains recordings of 630 speakers, which is a relatively large number. The drawback of the TIMIT corpus is that it contains only single-session recordings, so the problem of the speaker's inter-session variability was not investigated in this study.

Each of the speakers utters ten sentences, each one lasting 3.2 s on average. The audio material per single speaker is relatively short (ca. 32 s, in total for training and for testing, compared, e.g., to 120 s in [10]), which makes the verification problem an even bigger challenge.

The experiments were run both for uncoded and coded speech. The uncoded speech was sampled at 16 kHz or 8 kHz, but the coded speech was sampled at 8 kHz only, as narrowband codecs were tested. The codecs researched were those which are the most-used lossy codecs in fixed, mobile, and VoIP telephony: (i) G.711 in A-law option; (ii) G.723.1 in 6.4 kbps option; (iii) GSM 06.10 (known also as GSM Full-Rate), working with the bitrate 13 kbps; (iv) GSM 06.60 (known also as GSM-Enhanced Full Rate), with 12.2 kbps bitrate; (v) G.729, operating at a bitrate of 8 kbps; (vi) Speex, here working in mode 8, as it showed the best performance in mismatched conditions in [10].

2.2. Verification procedure

A hybrid SVM-GMM approach was used, in which supervectors (SVs) were created by stacking the mean values of the GMM speaker models (generative part), and the verification decision was made by the SVM algorithm with a linear Kullback–Leibler kernel (discriminative part). The speaker models were created by adapting the mean values μ of the Gaussian components in a UBM model using the MAP algorithm with a relevance factor $RF = 1$.

The speech data was parameterized using 19 MFCC parameters (plus the 0th one), with a frame length of 30 ms and a 10 ms analysis step. The UBM models were trained using the GMM EM-ML algorithm, separately for each of the codecs

and for uncoded speech, using 200 speakers. The remaining 430 speakers were used for verification tests, analogously to [2] and [8].

Speaker SVs were created based upon ca. 16 s of training speech material. Recordings of five SX TIMIT sentences of each speaker were concatenated and split into eight equal parts to create eight SVs per speaker, as this proved to be successful in [2]. Speaker SVM verification models were trained by taking eight SVs of the correct speaker and 200 SVs created from the 200 UBM speakers. Hence, the UBM speakers were treated as a generalized impostor model.

Verification tests were run using single SA or SI TIMIT sentences, so testing SVs were created for each of these recordings for all 430 speakers. The SA sentences are the same for every speaker, this is why they were used in the testing part only, to preserve text-independence. Each speaker model was challenged with five attempts of the correct speaker and five impostor attacks, thus making 2 x 5 x 430 = 4300 verification trials in total. When testing verification in matched conditions, the UBM model, training, and tested SVs were all created from speech transcoded with the same codec. In experiments with mismatched conditions, the UBM and training sequences were trained using speech transcoded with codec X, and tested on SVs created from speech transcoded with codec Y.

The experiments were run in the Matlab environment using the LIBSVM [13] and h2m toolboxes [14]. The results were assessed by counting the number of False Acceptance (FA) and False Rejection (FR) errors. Based on them, the False Alarm and Miss probabilities were calculated. The EER value and ROC data were obtained by changing the decision level in the SVMs.

3. RESULTS

Verification tests were first run for clean, uncoded speech at the original sampling frequency (16 kHz), for matched conditions. It turned out that for 16 Gaussian components, the results are inferior to the ones described in [8], which were achieved with the GMM-UBM method: the EER yielded 4.05% compared to 1.1% in [8]. When increasing the number of Gaussians (M) the results got better, reaching 1.12% only for $M = 512$, see Fig. 1. Similar tests were performed for speech transcoded with the GSM 06.10 codec, in matched conditions as well; here the SVM-based verification proved better than GMM-UBM already for $M = 64$, reaching 5.86% compared to 7.30% in [8]. Generally, for coded speech and higher values of M, the SVM-based speaker verification significantly outperformed the GMM-UBM-based verification (see Fig.1); for coded speech it was therefore decided to conduct further experiments with $M = 256$.

Next, experiments were run for 8 kHz-sampled speech, both coded and uncoded, in matched and unmatched conditions. The EER results for 49 different verification configurations (systems trained with seven various speech types, each of them tested with seven speech types) are presented in Table 1. The diagonal (highlighted) presents the verification results for matched conditions. The best results were achieved for uncoded speech, speech transcoded with Speex8 and G.711 (with EER equal 2.93%, 3.40% and 3.53%, respectively); G.723.1 and G.729 yielded the worst scores (EER 5.40% and 5.16%, respectively). These results are consistent with the voice quality offered by these codecs: a similar relationship was observed in [2] and [10]. A ROC curve for the matching condition is presented in Fig. 2(a). It shows, among other things, that the ROC characteristics for speech transcoded with Speex8 and with G.711 almost overlap.

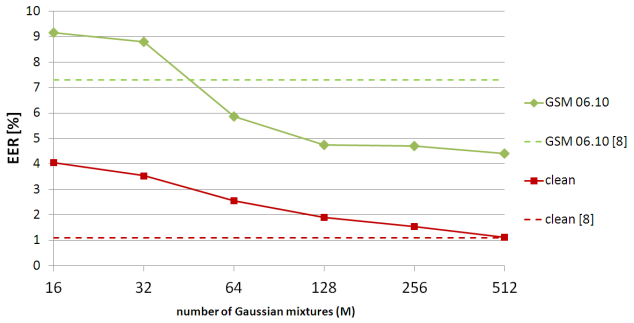


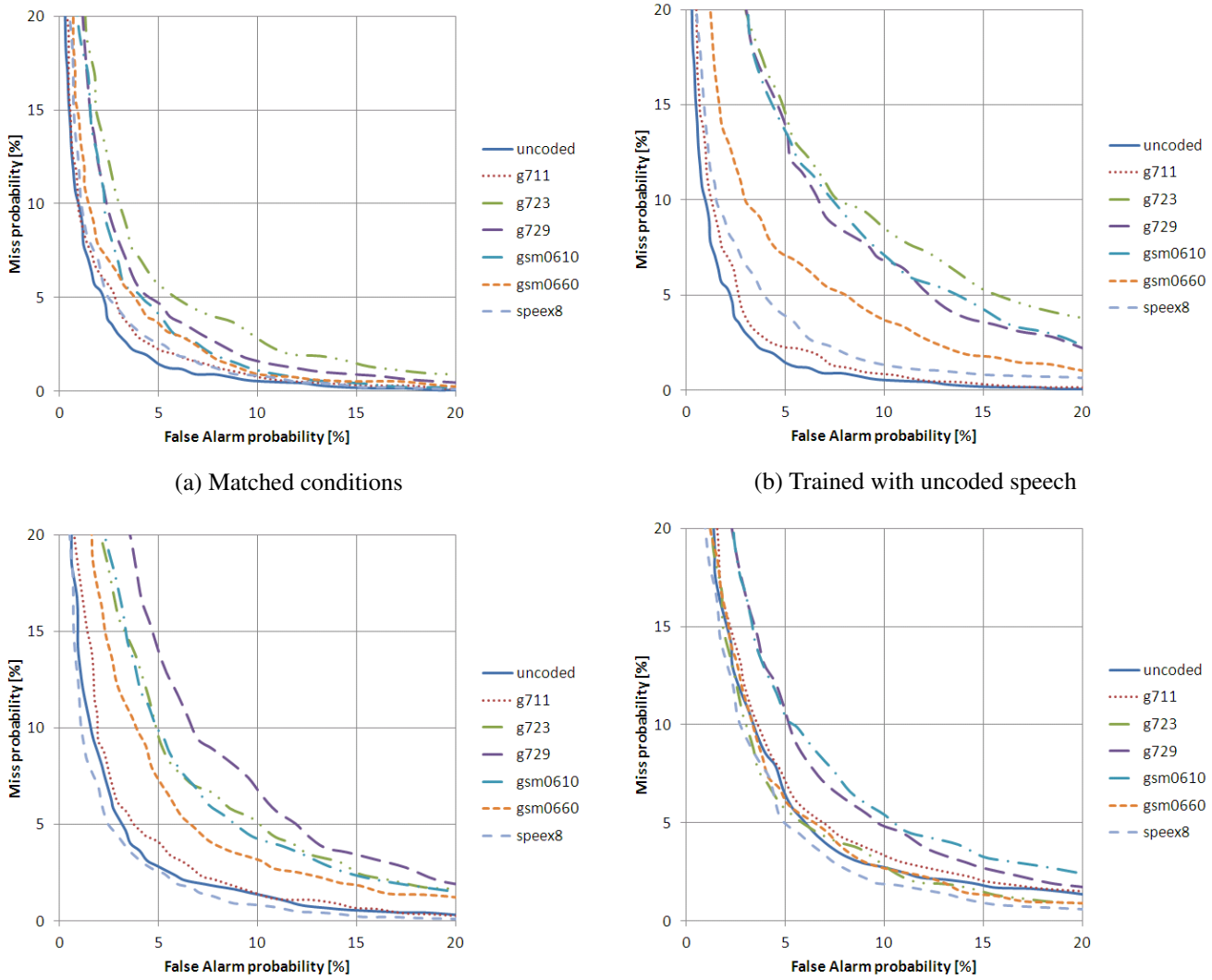
Fig. 1. EER results for uncoded speech ($fs = 16$ kHz) and GSM 06.10-transcoded speech for various numbers of Gaussian components, compared with the results in [8].

When testing speaker verification in mismatched conditions, the results obviously were worse. However, the degradation of the verification performance was not uniform. If the verification system was challenged with high-quality speech (transcoded with G.711 or Speex8, or uncoded), then the increase in the EER was not high. Remarkably, a system trained with G.723.1 and tested with Speex8-transcoded speech yielded even better results than in matched conditions. If the verification was based on lower-quality speech (transcoded with the G.723.1, G.729, or GSM codecs) the performance varied. Sometimes the increase of EER was not significant (e.g., for the pair G.723.1/GSM 06.60: it increased by only 0.27% over the matched condition). On the other hand, some codecs did not complement each other well: especially the GSM 06.10 codec was "disliked" by the others – the EER often doubled or almost tripled if they were challenged with GSM 06.10-transcoded speech. Similar behavior was observed for the classification task in [2].

Analyzing Table 1 row-wise can help find the most suitable codec for creating universal models for speaker verification. The average EER values show that Speex8 and G.723.1 seem to be the best candidates. In addition, G.723.1 yielded the lowest EER variance, which is also visible when analyzing the ROC curves in Fig. 2(d) – they are very close to each other, unlike those for Speex8 in Fig. 2(c).

Table 1. EER [%] for systems trained (in rows) and tested (in columns) with different codecs. The diagonal (in bold) shows results for matched conditions.

training/testing	un-coded	G.711	G.723	G.729	GSM 06.10	GSM 06.60	Speex 8	average	stddev
uncoded	**2.93**	3.30	8.93	7.77	8.74	6.33	4.42	6.06	2.15
G.711	3.58	**3.53**	9.58	8.47	8.51	6.56	4.74	6.43	2.12
G.723.1	5.30	5.77	**5.40**	7.02	7.44	5.67	5.12	5.96	0.73
G.729	5.86	6.98	8.05	**5.16**	10.23	5.07	6.23	6.80	1.39
GSM 06.10	6.60	6.47	8.93	10.65	**4.70**	8.42	6.37	7.45	1.62
GSM 06.60	5.16	6.00	8.09	7.35	9.72	**4.42**	5.86	6.66	1.48
Speex 8	4.05	4.37	6.84	8.70	6.88	6.14	**3.40**	5.77	1.57



(a) Matched conditions

(b) Trained with uncoded speech

(c) Trained with Speex-transcoded speech

(d) Trained with G.723.1-transcoded speech

Fig. 2. ROC curves for various testing scenarios: (a) matched conditions, and the models trained with (b) uncoded speech, (c) Speex, (d) G.723.1.

The ROC curves for the speaker models created with uncoded speech, Fig. 2(b), show that such a verification system would perform well if challenged with G.711 and Speex-transcoded speech, but it would have a high probability of errors when tested with the G.723.1, G.729, or GSM 06.10 codecs.

4. CONCLUSIONS

SVM-based speaker verification performs comparably to the GMM-UBM technique, provided that the number of Gaussians (M) is increased. For the uncoded 16 kHz-sampled speech, it was necessary to increase M from 16 to 512 to achieve nearly the same EER of 1.1%. For GSM-coded speech it was enough to increase M to 64 to get much better results than the GMM-UBM approach. It is believed that this is caused by the fact that in the SVM-GMM technique, we need to create a *model* of the tested speech in order to submit it to the SVM classifier, while in the GMM-UBM technique, the speaker model is challenged directly by the parameters of the tested speech. Therefore the SVM-based speaker verification requires more precise speaker modeling.

The classification task [2] required less increase in M: 256 and 32 Gaussians, respectively. This suggests that the verification task in the scenario used (an equal number of client and impostor access trials) was more demanding than the closed-set classification task within a group of 430 speakers.

The problem of a major EER increase in some of the tested mismatch conditions is probably caused by the fact that each of the tested lossy codecs is lossy to a different extent. If speaker models are created with codec X and the client tries to verify its identity using codec Y, the degradation of accuracy will be minimal if the codec Y is either lossy to a low degree (e.g., it is a waveform codec G.711) or it loses a piece of information which was already lost by the codec X. Due to this fact, in our opinion, out of the seven tested types of narrowband speech signals, both coded and uncoded, the speech transcoded either with Speex in mode 8 or with the G.723.1 codec proved to be the most suitable for creating the most universal speaker models.

Future work can involve verifying these results in multi-session conditions, so it will require using another corpus, e.g., YOHO.

5. REFERENCES

[1] V. N. Vapnik, *The Nature of Statistical Learning Theory*, Springer, New York, 1995.

[2] A. Janicki and T. Staroszczyk, "Speaker recognition from coded speech using support vector machines," *LNAI*, vol. 6836, pp. 291–298, 2011.

[3] W. M. Campbell, D. E. Sturim, and D. A. Reynolds, "Support vector machines using gmm supervectors for speaker verification," *IEEE Signal Processing Letters*, vol. 13, pp. 308–311, 2006.

[4] A. Brew and P. Cunningham, "Combining cohort and ubm models in open set speaker identification," in *Proc. of the 2009 Seventh International Workshop on Content-Based Multimedia Indexing*, Washington, DC, USA, 2009, pp. 62–67, IEEE Computer Society.

[5] V. Wan and S. Renals, "Speaker verification using sequence discriminant support vector machines," *IEEE Transactions on Speech and Audio Processing*, vol. 13, no. 2, pp. 203–210, 2005.

[6] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn, "Speaker verification using adapted gaussian mixture models," in *Digital Signal Processing*, 2000.

[7] T. F. Quatieri, E. Singer, R. B. Dunn, D. A. Reynolds, and J. P. Campbell, "Speaker and language recognition using speech codec parameters," in *Proc. EUROSPEECH '99*, 1999, vol. 2, p. 790.

[8] L. Besacier, S. Grassi, A. Dufaux, M. Ansorge, and F. Pellandini, "Gsm speech coding and speaker recognition," in *Proc. ICASSP*, 2000, pp. 1085–1088.

[9] R. B. Dunn, T. F. Quatieri, D. A. Reynolds, and J. P. Campbell, "Speaker recognition from coded speech and the effects of score normalization," in *Proc. 35th Asilomar Conference on Signals, Systems and Computers*, 2001, vol. 2, pp. 1562–1567.

[10] A. R. Stauffer and A. D. Lawson, "Speaker recognition on lossy compressed speech using the speex codec," in *Proc. INTERSPEECH'09*, 2009, pp. 2363–2366.

[11] T. Jiang, B. Gao, and J. Han, "Speaker identification and verification from audio coded speech in matched and mismatched conditions," in *Proc. of the IEEE International Conference on Robotics and Biomimetics (ROBIO 2009)*, 2009, pp. 2199–2204.

[12] J. Garofolo, L. Lamel, W. Fisher, J. Fiscus, D. Pallett, N. Dahlgren, and V. Zue, "Timit acoustic-phonetic continuous speech corpus," 1993, Linguistic Data Consortium, Philadelphia.

[13] C. C. Chang and C. J. Lin, "Libsvm: a library for support vector machines," 2001, http://www.csie.ntu.edu.tw/~cjlin/libsvm/.

[14] Olivier Cappe, "h2m toolkit," http://www.tsi.enst.fr/~cappe/.