Finding Difficult Speakers in Automatic Speaker Recognition

Lara Lynn Stoll

Electrical Engineering and Computer Sciences
University of California at Berkeley

Technical Report No. UCB/EECS-2011-152
http://www.eecs.berkeley.edu/Pubs/TechRpts/2011/EECS-2011-152.html

December 16, 2011

Finding Difficult Speakers in Automatic Speaker Recognition

by

Lara Lynn Stoll

A dissertation submitted in partial satisfaction of the
requirements for the degree of
Doctor of Philosophy

in

Engineering - Electrical Engineering and Computer Sciences

in the

Graduate Division

of the

University of California, Berkeley

Committee in charge:

Professor Nelson Morgan, Co-chair
Dr. N. Nikki Mirghafori, Co-chair
Professor Michael Jordan
Professor John J. Ohala

Fall 2011

Finding Difficult Speakers in Automatic Speaker Recognition

Abstract

Finding Difficult Speakers in Automatic Speaker Recognition

by

Lara Lynn Stoll

Doctor of Philosophy in Engineering - Electrical Engineering and Computer Sciences

University of California, Berkeley

Professor Nelson Morgan, Co-chair

Dr. N. Nikki Mirghafori, Co-chair

The task of automatic speaker recognition, wherein a system verifies or determines a speaker's identity using a sample of speech, has been studied for a few decades. In that time, a great deal of progress has been made in improving the accuracy of the system's decisions, through the use of more successful machine learning algorithms, and the application of channel compensation techniques and other methodologies aimed at addressing sources of errors such as noise or data mismatch. In general, errors can be expected to have one or more causes, involving both intrinsic and extrinsic factors. Extrinsic factors correspond to external influences, including reverberation, noise, and channel or microphone effects. Intrinsic factors relate inherently to the speaker himself, and include sex, age, dialect, accent, emotion, speaking style, and other voice characteristics. This dissertation focuses on the relatively unexplored issue of dependence of system errors on intrinsic speaker characteristics. In particular, I investigate the phenomenon that some speakers within a given population have a tendency to cause a large proportion of errors, and explore ways of finding such speakers.

There are two main components to this thesis. First, I establish the dependence of system performance on speaker characteristics, building upon and expanding previous work demonstrating the existence of speakers with tendencies to cause false alarm or false rejection errors. To this end, I explore two different data sets: one that is an older collection of telephone channel conversational speech, and one that is a more recent collection of conversational speech recorded on a variety of channels, including the telephone, as well as various types of microphones. Furthermore, in addition to considering a traditional speaker recognition system approach, for the second data set I utilize the outputs of a more contemporary approach that is better able to handle variations in channel. The results of such analysis repeatedly show variations in behavior across speakers, both for true speaker and impostor speaker cases. Variation occurs both at the level of speech utterances, wherein a given speaker's performance can depend on which of his speech utterances is used, as well as on the speaker level, wherein some speakers have overall tendencies to cause false rejection

or false alarm errors. Additionally, lamb-ish speaker behavior (where the speaker tends to produce false alarms as the target) is correlated with wolf-ish behavior (where the speaker tends to produce false alarms as the impostor). On the more recent data set, 50% of the false rejection and false alarm errors are caused by only 15-25% of the speakers.

The second component of this thesis investigates a straightforward approach to predict speakers that will be difficult for a system to correctly recognize. I use a variety of features to calculate feature statistics that are then used to compute a measure of similarity between speaker pairs. By ranking these similarity measures for a set of impostor speaker pairs, I determine those speaker pairs that are easy for a system to distinguish and those that are difficult-to-distinguish. A variety of these simple distance measures could successfully select both easy- and difficult-to-distinguish speaker pairs, as evaluated by differences in detection cost and false alarm probability across a large number of systems. Of the performance measures tested, the best feature-measure at finding the most and least difficult-to-distinguish speaker pairs was the Euclidean distance between vectors of the mean first, second, and third formant frequencies. Even greater success was attained by the Kullback-Liebler (KL) divergence between pairs of speaker-specific GMMs. Furthermore, an examination of the smallest and biggest distances (as computed by the KL divergence) revealed individual speaker tendencies to consistently fall among the most (or least) difficult-to-distinguish speaker pairs.

I then develop an approach for finding those individual speakers who will be difficult for the system, using a set of feature statistics calculated over regions of speech. In particular, a support vector machine (SVM) classifier is trained to distinguish between difficult and easy speaker examples, in order to produce an overall measure of speaker difficulty as a target or impostor. The resulting precision and recall measures were over 0.8 for difficult impostor speaker detection, and over 0.7 for difficult target speaker detection. Depending on the application, the detection threshold can be tuned to improve precision, recall, or specificity in order to best suit the needs of a particular task. The same approach can be taken with single conversation sides, as with a set of conversation sides corresponding to the same speaker, since the input feature statistics can be calculated over any number of speech samples.

To my mother, whom I miss every day

Contents

List of Figures

List of Tables

Acknowledgments

Given the many years of my graduate career, there is a long list of people to thank. I begin with my adviser, Professor Nelson Morgan, who welcomed me into the speech group and gave me a research home at the International Computer Science Institute (ICSI). In addition to providing support throughout my academic career, Morgan was also instrumental in helping me find the last puzzle piece to fit in my dissertation work.

Next, I have to express my deep gratitude for my mentor, Nikki Mirghafori. It is hard to describe all the ways in which Nikki has positively influenced me. When I first started in the speaker recognition group in 2005, she provided excellent technical and professional guidance, helping me to gain understanding and confidence, improve my communication skills, and grow as a contributing member of the group. After an interlude without her at ICSI, Nikki returned in 2010 to once again lead the speaker recognition group, introducing a wonderful balance between research and personal concerns to our meetings, and helping me to learn how to better deal with stress, fatigue, and other distractions that arise in daily life. I am truly appreciative of Nikki's encouragement and support, and it is reassuring to know that it will continue as I move on to the next challenge.

There are many other researchers to be thanked for helping me along the way. One particularly influential person in my thesis work was George Doddington, who was a wonderful source of ideas to try, and a most interesting person to work with. I must also thank Joe Frankel, with whom I collaborated on my Master's work. Additional members of the speaker recognition community who have provided feedback and help throughout my career include Andreas Stolcke, Liz Shriberg, Sachin Kajarekar, Howard Lei, Andy Hatch, Christian Müller, David van Leeuwen, Eduardo Lopez-Gonzalo, and Joaquin Gonzalez.

Of course, I must also mention some of the many students, post-docs, visitors, and staff at ICSI, who have helped make it the wonderful place that it is. Among these are Kofi Boakye, Marios Athineos, Dan Gillick, Arlo Faria, Oriol Vinyals, Jaeyoung Choi, David Imseng, Benoit Favre, Korbinian Riedhammer, Adam Janin, and Jacob Wolkenhauer.

I would be remiss if I did not acknowledge my friendly officemates throughout the years: Madelaine Plauché, Matthew Aylett, and Vijay Ullal. Special recognition goes to my officemates of the past several years, Mary Knox and Suman Ravuri, who are not only lovely work companions (and excellent contributors on Sporcle quizzes), but also dear friends.

Finally, I want to thank my family. My mom made me who I am, and I would not have been successful without her influence in my life. My dad has been truly supportive of me in every way imaginable, despite the fact that it often appeared that I might never finish. My sister and brother (and their spouses as well) have always been there for me, and are due to receive many a dinner in thanks once I finally have a job. Lastly, I thank my nieces, Lynn and Magnolia, for always reminding me of the simple joys in life.

Chapter 1

Introduction

1.1 Automatic Speaker Recognition

The task of automatic speaker recognition, wherein a system verifies or determines a speaker's identity using a sample of speech, has been studied for a few decades. In that time, a great deal of progress has been made in improving the accuracy of the system's decisions, through the use of more successful machine learning algorithms, and the application of channel compensation techniques and other methodologies aimed at addressing sources of errors such as noise or data mismatch. This dissertation focuses on the relatively unexplored issue of dependence of system errors on speaker characteristics. In particular, I investigate the phenomenon that some speakers within a given population have a tendency to cause a large proportion of errors, and explore ways of finding such speakers.

There are a number of tasks that fall into the category of speaker recognition. My work uses the speaker verification paradigm, in which there is a hypothesized target speaker identity, with an associated training speech utterance, and the system must decide whether a given test utterance was spoken by the target speaker. In this case, there are two types of errors: false rejections, in which the true speaker is rejected as such, and false acceptances, in which an impostor speaker is accepted as the target speaker. In general, these errors can be expected to have one or more causes, involving both intrinsic and extrinsic factors. Extrinsic factors correspond to external influences, including reverberation, noise, and channel or microphone effects. Intrinsic factors relate inherently to the speaker himself, and include sex, age, dialect, accent, emotion, speaking style, and other voice characteristics. This dissertation analyzes errors in terms of intrinsic speaker attributes.

1.2 Inherent Speaker Characteristics

As human listeners, we may observe that some speakers sound more alike than others, or we may find it difficult to identify certain speakers because their voices do not always sound

the same from time to time. Similarly, there may be speakers for which an automatic speaker recognition system makes more decision errors. There are many sources of variation within and across speakers that may contribute to causing such errors, including basic physical attributes, language, accent, characteristics of speaking style, and changes in emotional state or health.

This thesis is inspired by the analysis of Doddington et al. [22], in which the authors characterized speakers in terms of their error tendencies. The default, well-behaved speakers are "sheep." Speakers who cause a proportionately high number of false rejection errors as the target speaker are called "goats." Those speakers who tend to cause false acceptance errors as the target speaker are "lambs," and those who tend to cause false acceptance errors as the impostor speaker are labeled "wolves." The existence of such speaker types was demonstrated through statistical tests using the outputs of an automatic speaker recognition system. Further analysis of additional data sets and different types of speaker recognition systems can provide more insight into the dependence that system performance has on the speakers.

Given that automatic speaker recognition system performance does depend on speaker characteristics, knowing which speakers are likely to cause errors is information that could prove useful for improving decision accuracy. Yet, limited work has been done to find these difficult speakers without the benefit of having a system's output.

Furthermore, there are a number of real-world applications that rely on automatic speaker recognition technology, that could benefit from being able to find the most similar speakers or the most difficult trials to make a decision about. Inherent to certain tasks are populations of in-set and out-of-set speakers. That is, there may be a set of known speakers (i.e., in-set speakers), with associated speech samples, that needs to be distinguished from other, unknown speakers (i.e., out-of-set speakers). One example of this type of real-world application is that of fraud detection, where a company is trying to prevent fraud in the use of a call center or other phone-base system. Given a database of speaker models trained using speech samples from people known to have committed fraud, an automatic system may compare new speech data from incoming calls to the database of fraudster speaker models in order to detect possible fraudulent attempts, which must then be verified by a human listener. However, a human expert would be unable to listen to all calls if there are a large number of potential matches between new speech data and the fraudster models. A method for selecting the most error-prone speakers could thus prove very useful for focusing the efforts of a human listener in a smart way.

1.3 Thesis Goals and Overview

There are two main components to this thesis. First, I establish the dependence of system performance on speakers, building upon the previous work of Doddington et al. To this end, I explore two different data sets: one that is an older collection of telephone channel

conversational speech, and one that is a more recent collection of both conversational speech and interview-style speech, recorded on a variety of channels, including landline and cellular telephone, as well as various types of microphones. Furthermore, in addition to considering a traditional speaker recognition system approach, for the second data set I utilize the outputs of a more contemporary approach that is better able to handle variations in channel.

The second component of this thesis investigates a straightforward approach to predict speakers that will be difficult for a system to correctly recognize. I use a variety of features to calculate feature statistics that are then used to compute a measure of similarity between speaker pairs. By ranking these similarity measures for a set of impostor speaker pairs, I determine those speaker pairs that are easy for a system to distinguish and those that are difficult-to-distinguish. I then develop an approach for combining a set of feature statistics in order to produce a comprehensive measure of how likely it is that a speaker will cause errors. In particular, I use support vector machine (SVM) classifiers trained to distinguish between difficult and easy examples, in order to detect difficult impostor and target speakers.

I begin by covering relevant background material in Chapter 2, including typical features and systems for automatic speaker recognition, intrinsic speaker characteristics, and related error analyses of speaker recognition systems. Next, I explore the speaker-dependent performance of systems in Chapter 3. In Chapter 4, I introduce a simple approach to finding difficult-to-distinguish speaker pairs. I then describe a technique for detecting difficult target or impostor speakers in Chapter 5. Finally, I summarize and conclude my work in Chapter 6.

Chapter 2

Background

There are several broad areas of prior work relevant to this dissertation. I begin in Section 2.1 by setting up the speaker recognition problem, while in Sections 2.2, 2.3, 2.4, and 2.5 I provide details about features, system approaches, relevant speech corpora, and measures of system performance, respectively. There are a number of intrinsic speaker qualities, which account for intra-speaker variability, as well as differences between speakers, that I describe in Section 2.6. The most directly related work involves error analysis pertaining to speaker recognition systems, which I discuss in Section 2.7.

2.1 The Speaker Recognition Problem

As its name implies, automatic speaker recognition attempts to recognize, or identify, a given speaker by processing his/her speech automatically, that is to say, in a fully objective and reproducible manner, without the aid of human listening or analysis. In order to be able to recognize the speaker of a given test utterance, it is necessary to have training data first, so that the system can "learn" the speaker of interest. The term speaker recognition can be used to refer to a variety of tasks. One type of task is speaker identification, where the system must produce the identity of the speaker, given a test utterance, from a set of speakers. With closed-set speaker identification, the number of speakers in the set is fixed, and the system must choose which among the given speakers is a match to the speaker of the test utterance. Open-set speaker identification adds a layer of complexity by allowing the test utterance to belong to a speaker not in the set of speakers for whom there is training data available. A second type of task is speaker verification, which involves a hypothetical target speaker match to the test speaker, and the system must determine whether or not the test speaker identity is as claimed.

Regardless of which type of task, the problem may be further characterized as being text-dependent or text-independent. In the text-dependent case, the train and test utterances are required to be a specific word or set of words; the system can then exploit the knowledge

of what is spoken in order to better make a decision. For the text-independent case, there is no constraint on what is said in the speech utterances, allowing for generalization to a wider variety of situations.

The dissertation work focuses on the text-independent speaker verification task. For each target (or hypothesis) speaker and test utterance pair, the system must decide whether or not the speaker identities are the same. In this case, two types of errors arise: false acceptance (or false alarm) and false rejection (or missed detection). A false accept occurs when the system incorrectly verifies an impostor test speaker as the target speaker. A false reject occurs when the system fails to verify a true test speaker as the target speaker. A trial refers to a target speaker and test utterance pair. In general, the training data of a target speaker may include one or more samples of speech, of varying lengths, and the test data may also include varying lengths of speech samples. For my purposes, the train and test utterances will both be a single conversation side, which is typically 2.5-3 minutes of speech. Therefore, a trial will correspond to a pair of train and test conversation sides. For each trial, the corresponding score simply refers to the output of a speaker recognition system given that train and test data. The score may or may not correspond to a likelihood. Furthermore, in order to make a decision for a trial given its score, there must be a decision threshold; then, the system will decide that it's a true speaker trial if the score is above the decision threshold, or decide that it's an impostor trial if the score is below the decision threshold.

In general, speaker recognition errors may be caused by both extrinsic factors, such as channel effects or noise, and intrinsic factors, such as age, sex, speaking style, or other inherent speaker attributes. My focus is on the effects of intrinsic speaker characteristics.

In order to perform a speaker recognition task, a system must first parameterize the speech in a meaningful way that will allow the system to distinguish and characterize speakers and their speech; this step is addressed next in Section 2.2, which discusses some relevant features commonly used in speech processing applications. A number of typical system approaches and methods are then discussed in Section 2.3, while I describe commonly utilized speech corpora and performance measures in Sections 2.4 and 2.5. In Section 2.6, I will describe a variety of intrinsic factors that contribute to variations both within an individual speaker and across different speakers, and consider the potential impacts of such speaker characteristics, before concluding with an overview of relevant error analyses of speaker recognition systems in Section 2.7.

2.2 Speech Features

The process of parameterizing a raw input, for example, speech, is referred to as feature extraction. For speech processing, low-level features are those based directly on frames of the speech signal, where frames correspond to a moving window, typically 25 ms long, with a given step size of typically 10ms. A length of 25ms and step size of 10ms corresponds to an overlap of 15ms between speech frames. High-level features, on the other hand, usually

incorporate information from more than just one frame of speech, and include, for example, speaker idiosyncrasies, prosodic patterns, pronunciation patterns, and word usage. The type of low-level acoustic features most often used in speaker recognition tasks are Mel-frequency cepstral coefficients, or MFCCs, which are described in Section 2.2.1. Section 2.2.2 provides a brief introduction to other acoustic and prosodic features, such as formant frequencies. Finally, Section 2.2.3 introduces various types of speech segments, which may be used to calculate different types of features.

2.2.1 Cepstral Features

MFCCs are generated by the process shown in Figure 2.1. First, an optional pre-emphasis filter is applied, to enhance the higher spectral frequencies and compensate for the unequal perception of loudness at different frequencies. Next, the speech signal is windowed as described above and the squared magnitude of the fast Fourier transform (FFT) is calculated for each frame. A Mel-frequency triangular filter bank is then applied, where Mel refers to an auditory scale based on pitch perception. There are different versions of the transformation from linear frequency scale to Mel frequency. One example, taken from [57], is given by

$$f_{Mel} = 1127 \cdot \ln\left(1 + \frac{f_{linear}}{700}\right) \tag{2.1}$$

A typical number of filters is 24 to 26. After the spectrum has been smoothed, the log is taken. Finally, a discrete cosine transform (DCT) is applied to obtain the cepstral coefficients, c_n:

$$c_n = \sum_{k=1}^{K} S_k \cos\left[n(k - \frac{1}{2})\frac{\pi}{K}\right], \quad n = 1, 2, \ldots, L \tag{2.2}$$

where S_k are the log-spectral vectors from the previous step, K is the total number of log-spectral coefficients, and L is the number of coefficients to be kept (this is called the order of the MFCCs), with $L \leq K$.

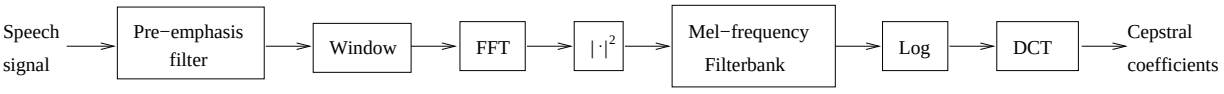

Figure 2.1: *Generation of MFCC features*

In addition to using the MFCCs, it is common to include estimates of their first, second, and possibly third derivatives as additional features. These are referred to as deltas, double-deltas, and triple-deltas. The polynomial approximations of the first and second derivatives

are as follows:

$$\begin{aligned} \Delta c_m &= \frac{\sum_{k=-l}^{l} k c_{m+k}}{\sum_{k=-l}^{l} |k|} \\ \Delta\Delta c_m &= \frac{\sum_{k=-l}^{l} k^2 c_{m+k}}{\sum_{k=-l}^{l} k^2} \end{aligned} \tag{2.3}$$

Furthermore, an energy term and/or its derivative can also be included in the feature parameterization.

Other commonly used cepstral features include linear-frequency cepstral coefficients (or LFCCs), which use a linear rather than Mel-based frequency bank, as well as features based on linear prediction, such as linear predictive coding coefficents (LPCCs) and perceptual linear prediction features (PLPs).

2.2.2 Other Acoustic and Prosodic Features

Formant frequencies correspond to resonances of the vocal tract and can often be measured in spectrograms by amplitude peaks in the frequency spectrum. Vowels in particular can be largely characterized by the first and second formants, though any voiced speech segment will produce formants.

The fundamental frequency, or f0, is an acoustic property corresponding to the lowest harmonic in the frequency spectrum. Pitch and fundamental frequency are often used interchangeably as terms, though pitch is an auditory property that is perceived by human listeners, who place sounds on a pitch scale ranging from low to high. The intonation of speech is the pitch pattern. Jitter is a term to describe varying pitch in the voice. A related feature is shimmer, which describes varying loudness in the voice.

Other commonly used prosodic features include energy distributions and dynamics, and duration and timing information, such as speech rate or average duration of various speech segments. Prosody will be revisited in more detail in Section 2.6.1.

2.2.3 Speech Segments

One concept that arises when considering higher-level features is that of speech segments. The basic linguistic unit of speech is that of a phone, which corresponds to a vowel or consonant speech sound that may be described in terms of articulatory movements and acoustic properties. Phonemes are sounds that are used to differentiate words [42]. For instance, in the words *got* and *not*, /g/ and /n/ are two different phonemes that lead to different meanings. Phonemes may be pronounced in different ways, leading to different phones that are all instances of the same phoneme; although there are differences in pronunciation of these phones, their meaning does not change. In the remainder of this thesis, the term phone is used to refer to phoneme.

Going beyond the phone, segments may be defined as groups of phones or syllables, as well as words, and sentences. All of these types of segments may be used as the basis for calculating various types of features.

2.3 System Approaches and Methodologies

There are a number of statistical and discriminative-training based methods that have been explored for the speaker recognition task. Two of the most successful modeling approaches that have been used are the Gaussian mixture model (GMM) and the support vector machine (SVM), which are discussed here. Other techniques have utilized hidden Markov models (HMMs), artificial neural networks such as multi-layer perceptrons (MLPs), or vector quantization (VQ).

2.3.1 Gaussian Mixture Model (GMM)

The Gaussian mixture model is a powerful tool for modeling certain types of unknown distributions effectively. The GMM uses a mixture of multivariate Gaussians to model the probability density function of observed variables. That is, for a GMM with N Gaussians, with variable x (n-dimensional), the probability density is given by

$$p(x|\lambda) = \sum_{i=0}^{N-1} \pi_i \, \mathcal{N}(x; \mu_i, \Sigma_i) \tag{2.4}$$

where π_i are the mixture weights, which sum to 1, and $\mathcal{N}(x; \mu_i, \Sigma_i)$ are Gaussian distributions with mean vectors μ_i and covariance matrices Σ_i, specifically,

$$\mathcal{N}(x; \mu_i, \Sigma_i) = \frac{1}{(2\pi)^{n/2}|\Sigma_i|^{1/2}} \exp\left(-\frac{1}{2}(x-\mu_i)^{\mathrm{T}}\Sigma_i^{-1}(x-\mu_i)\right) \tag{2.5}$$

The model parameters are denoted by $\lambda = (\pi_i, \mu_i, \Sigma_i)$, for $i = 0, \ldots, N-1$. The expectation-maximization (EM) algorithm iteratively learns the model parameters from the data, which are the observations. The covariance matrix is typically chosen to be diagonal, for improved computational efficiency as well as better performance.

In the context of using features extracted from speech, each feature vector would correspond to x in equation (2.4). Based on the assumption that speech frames are independent, the individual frame probabilities can be multiplied to obtain the probability of a speech utterance. That is, the probability of a speech segment X, composed of feature vectors $\{x_0, x_1, ..., x_{M-1}\}$, is given by

$$p(X|\lambda) = \prod_{j=0}^{M-1} \sum_{i=0}^{N-1} \pi_i \, \mathcal{N}(x_j; \mu_i, \Sigma_i) \tag{2.6}$$

for a mixture of N Gaussians.

In a speaker recognition setting, there are several GMM approaches that can be taken. Here, only the currently prevalent approach, referred to as UBM-GMM, is described. Two GMM models are needed: one for the target speaker and one for the background model [64]. Using training data from a large number of speakers, a speaker-independent universal background model, or UBM, is generated. The UBM training data is a type of system-level training data, which is chosen to be completely disjoint from the training data used to train target models for a given set of trials. So that every target speaker model is in the same space and can be compared to one another, the speaker-dependent models (using the corresponding target speaker training data) are adapted from the UBM using maximum *a posteriori* (MAP) adaptation. For a given test utterance X, and a given target speaker, a log likelihood ratio (LLR) can then be calculated:

$$LLR(X) = \log p\left(X|\lambda_{target}\right) - \log p\left(X|\lambda_{UBM}\right) \tag{2.7}$$

Comparing the LLR to a threshold, Θ, will determine the decision made about the test speaker's identity: if $LLR(X) > \Theta$, the test speaker is identified as a true speaker match, otherwise, the test speaker is determined to be an impostor. The LLR is the score for the UBM-GMM system.

2.3.2 Support Vector Machine (SVM)

Support vector machines, or SVMs, are a supervised learning method that can be used for pattern classification problems [11]. For binary classification, which is the task of interest here, the SVM is a linear classifier that finds a separating hyperplane between data points in each class. The SVM learns the maximum-margin hyperplane that will separate the data, making it a maximum-margin classifier. The input can be transformed, possibly in a nonlinear way, through the use of different kernel functions, allowing for more flexibility and modeling power. With an SVM, the "model" for each target speaker is the defining hyperplane, and instead of probabilities for data given a distribution, distances from the hyperplane are used.

In mathematical terms, the SVM problem can be formulated as

$$\begin{aligned} &\min \|w\|^2 + C \sum_i \xi_i \\ &\text{subject to} \quad y_i(w \cdot x_i - b) \geq 1 - \xi_i,\ 1 \leq i \leq n \end{aligned} \tag{2.8}$$

where ξ_i are slack variables, x_i are the training data points, y_i are the corresponding class labels ($+1$ or -1), C is a constant, and w and b are the hyperplane parameters. Essentially, the goal is to find the hyperplane such that $sign(w \cdot x_i - b) = y_i$, up to some soft margin involving ξ_i.

The SVM is used in speaker recognition by taking one or more positive examples of the target speaker, as well as a set of negative examples of impostor speakers, and producing a hyperplane decision boundary. Since there are far more impostor speaker examples than target speaker examples, a weighting factor is typically used to make the target example(s) count as much as all of the impostor examples. Once the hyperplane for a given target speaker is known, the test speaker can be classified as belonging to either the target speaker or impostor speaker class. Instead of a log likelihood ratio, a score can be produced by using the distance of the test data from the hyperplane boundary.

2.3.3 A Brief Historical Overview of Types of Systems

Automatic speaker recognition systems can be categorized by the type of features they use and by the type of statistical modeling tool that they use. Features may range from low-level and short-term (based directly on the acoustics of the speech) to higher levels incorporating longer lengths of time, including prosodic, lexical, or semantic. MFCCs are an example of low-level, short-term features, while phone n-gram counts are higher-level, longer-term features. The overview of systems provided here, while not exhaustive, covers a variety of feature types and statistical learning methods, and is intended to give an idea of a range of approaches that have proven successful. In some cases, although a system alone may not have very good performance (compared to other systems), it may still be successful by contributing in a system fusion.

One conventional approach that has already been described in Section 2.3 is the cepstral GMM system [64, 61]. The cepstral SVM system utilizes a generalized linear discriminant sequence kernel to train an SVM classifier on a sequence of input cepstral features [12].

Some methods attempt to combine the advantages of the generative modeling of GMMs with the discriminative power of SVMs. One such approach is an SVM classifier that uses GMM supervectors as features [14]. The supervectors are the concatenated mean vectors from a GMM that has been MAP-adapted from a UBM to a speaker's data, with the idea that this mapping from an utterance into a high-dimensional supervector space is similar to an SVM sequence kernel.

Another successful approach is the MLLR-SVM system, which uses maximum-likelihood linear regression (MLLR) transforms from a speech recognition system as features for speaker recognition [69, 68]. In the context of a speech recognition system, MLLR applies an affine transform to the Gaussian mean vectors in order to map speaker-independent means to speaker-dependent means. The coefficients from one of more of these MLLR adaptation transforms are used in an SVM speaker recognition system with very good results.

One type of non-acoustic feature is the word n-gram, where n-gram can encompass unigrams, bigrams, and so forth. The motivation for using such a feature for speaker recognition is that there are idiolectal differences among speakers, i.e., speakers vary in their word usage. Speaker-dependent unigram and bigram language models were first used in a target to background likelihood ratio framework, with promising results [21].

There are also phone-based approaches. Similar to the word n-gram modeling, the phone n-gram system first used frequency counts of phone n-grams, where phones are found using a phone recognizer, or possibly phone recognizers for multiple languages, in a likelihood ratio framework [2]. The use of phonetic information was extended in a number of techniques, including the use of binary trees [55], cross-stream modeling [30], and SVMs [13, 29]. Another example is a pronunciation modeling approach, where word-level automatic speech recognition (ASR) phone streams are compared with open-loop phone streams [39].

Additional methods seek to take advantage of the speaker information present in words, by using word-conditioning. A keyword HMM system trains background HMMs for a number of keywords, and adapts them to speaker; a likelihood ratio between the background and speaker models for each word are then calculated for a given test utterance, and the likelihood ratios are combined to produce a final system score [6]. The word-conditioned phone n-gram system considers phone n-grams only for a specific set of keywords [43].

A number of approaches have used prosodic features, including pitch and energy distributions or dynamics [1], and prosodic statistics including duration and pitch related features [59]. Nonuniform Extraction Region Features (NERFs) consider a number of features, including maximum or mean pitch, duration patterns, and energy contours, for various regions of speech, which are delimited by some sort of event, such as short pauses, long pauses, or schwas [35].

2.3.4 Channel Compensation Techniques

One obvious component to a speech signal that is unrelated to the speech (or speaker) itself is the channel on which the speech is recorded. Although most speech corpora have been collected using the telephone, there are different types of handsets, including cellular, and there has also been a recent collection of data using different types of microphones. The biggest effect of having different types of channels present in the data occurs when there is a channel mismatch between the training and test data. That is, if a system's target speaker model is trained using data from an electret telephone handset, for instance, but the test speech was collected from a carbon-button telephone handset, it will "sound" different to the system, even if the speaker is the same for both. In speaker recognition systems, the effects of channel variation are typically addressed using normalizations, on the feature-level, the model-level, or the score-level. Since various approaches are taken in different domains and in varying ways, they often improve performance when applied on top of each other.

Historically, channel effects have been the dominating cause of errors in automatic speaker recognition tasks. In early speaker recognition work, mismatch in the type of telephone handset of train and test data caused error rates over four times as great as in the case of matched handsets [62]. In the most recent 2010 NIST Speaker Recognition Evaluation, the effects of channel mismatch still exist, but to a far lesser extent, with very low overall error rates for the best systems, despite increased amounts of channel variability.

Feature-level Normalizations

Cepstral mean subtraction (CMS) is a fairly simple technique that is applied at the feature-level [3]. CMS subtracts the time average from the output cepstrum in order to produce a zero mean log cepstrum. That is, for a temporal sequence of each cepstral coefficient c_m,

$$\hat{c}_m(t) = c_m(t) - \frac{1}{T}\sum_{\tau=1}^{T} c_m(\tau) \tag{2.9}$$

The purpose of CMS is to remove the effects of the transmission channel, yielding improved robustness. However, any non-linear channel effects will remain, as will any time-varying linear channel effects. Furthermore, CMS can remove some of the speaker characteristics, as the average cepstrum does contain speaker-specific information.

Another feature-level channel compensation method is feature mapping [63]. Feature mapping aims to map features from different channels into the same channel-independent feature space. A channel-independent root GMM is trained, and channel-dependent background GMMs are adapted from the root. Feature-mapping functions are obtained from the model parameter changes between the channel-independent and channel-dependent models. The most likely channel is detected for the speaker data, which is then mapped to the channel-independent space. Adaptation to target speaker models is done using mapped features, and during verification, the mapped features of the test utterance are used for scoring. The root GMM is used as the UBM for calculating the log likelihood ratios.

Within-class covariance normalization (WCCN) is a feature normalization technique for SVM systems [28]. In this method, a generalized linear kernel is trained, using class label information (i.e., a target or impostor speaker), in order to find orthonormal directions in the feature space that maximize information relevant to the task. The weights of those directions are optimized to minimize an upper bound on the error rate.

Model-level Normalizations

Speaker model synthesis (SMS) is a GMM model-based technique that utilizes channel-dependent models [70]. Rather than having one speaker-independent UBM, the SMS approach begins with a channel- and gender-independent root model, and then uses Bayesian adaptation to obtain channel- and gender-dependent background models. Channel-specific target speaker models are also adapted from the appropriate background model, after the gender and channel of the target speaker's training data have been detected. Furthermore, a transformation for each pair of channels is calculated using the channel-dependent background models; this transformation maps the weights, means, and variances of a channel a model to the corresponding parameters of a channel b model. During testing, if the detected channel of the test utterance matches the type of channel of the target speaker model, then that speaker model and the appropriate channel-dependent background model are used to calculate the LLR for that test utterance. On the other hand, if the detected channel of the

test utterance is not a match to the target speaker model, then a new speaker model is synthesized using the previously calculated transformation between the target and test channels. Then, the synthesized model and the corresponding channel-dependent background model are used to calculate the LLR for the test utterance.

Nuisance attribute projection (NAP) is another model-based technique, designed for use in SVM systems [67]. This method aims to remove "nuisance" dimensions, that is, those irrelevant to the task of speaker recognition, by projecting points in the expansion space of the SVM onto a subspace designed to be more resistant to channel effects. A projection matrix is created (using a training data set) in order to minimize the average cross-channel distance, with a weight matrix which can be formulated to not only reduce cross-channel distances, but also increase cross-speaker distances. This minimization problem reduces to an eigenvalue problem, where the eigenvectors with the largest eigenvalues must be found.

Score-level Normalizations

Although it does not specifically address the channel variation problem, one type of score-level normalization is zero normalization, or Z-norm [44]. In Z-norm, an impostor score distribution is obtained by testing a speaker model against impostor speech utterances. Then, the statistics of this speaker-dependent impostor distribution, namely the mean and variance, are used to normalize the scores produced for that speaker. That is, for a test utterance X, and a target speaker model T,

$$S_{ZN}(X) = \frac{S(X) - \mu_{impostor}(T)}{\sigma_{impostor}(T)} \tag{2.10}$$

where $S_{ZN}(X)$ is the normalized score, $S(X)$ is the original score, and $\mu_{impostor}(T)$ and $\sigma_{impostor}(T)$ are the mean and standard deviation of the distribution of impostor scores for target model T.

A variant of Z-norm is handset normalization, or H-norm, which aims to address the issue of having different handsets for the training and testing data [62]. H-norm tries to remove the handset dependent biases present in the scores produced, and it requires having a handset detector to label the handset of the speech segments. For each speaker, handset-dependent means and variations are determined for each type of handset (typically electret and carbon-button) by generating scores for a set of impostor test utterances from each handset type. Then, the score is normalized by the mean and standard deviation of the distribution corresponding to the handset of the test utterance, as determined by the handset detector. For test utterance X,

$$S_{HN}(X) = \frac{S(X) - \mu(HS(X))}{\sigma(HS(X))} \tag{2.11}$$

where $S_{HN}(X)$ is the new score, $S(X)$ is the original score, and $HS(X)$ is the handset label of X.

The final normalization of interest is test normalization, or T-norm, which generates scores for a test utterance against impostor models (in addition to the target model), in order to estimate the impostor score distribution's statistics [4]. T-norm is a test-dependent normalization, since the same test utterance is used for testing and for generating normalization parameter estimates. In mathematical terms,

$$S_{TN}(X) = \frac{S(X) - \mu_{impostor}(X)}{\sigma_{impostor}(X)} \tag{2.12}$$

where $S_{TN}(X)$ is the normalized score, $S(X)$ is the original score, and $\mu_{impostor}(X)$ and $\sigma_{impostor}(X)$ are the mean and standard deviation of the distribution of scores for test utterance X against the set of impostor speaker models.

2.3.5 Current State-of-the-Art Systems

One current state-of-the-art approach utilizes joint factor analysis (JFA), which models speaker and session variability in GMMs [38]. A target speaker GMM is adapted from a UBM, and the speaker is represented by the means, covariance, and weights of the GMM. JFA assumes that a speaker- and channel-dependent supervector can be decomposed into the sum of a speaker supervector, s, and a channel supervector, c. Furthermore, the speaker supervector is modelled as

$$s = m + Dz + Vy,$$

where m is the speaker- and channel-independent supervector from the UBM, D is a diagonal matrix, V is a low-rank rectangular matrix, and y and z are independent normally distributed random vectors, with components corresponding to the speaker and residual factors, respectively. The channel-dependent supervector is modelled as

$$c = Ux,$$

where U is a low-rank rectangular matrix and x is a normally distributed vector whose components corresponding to the channel factors. By estimating the speaker space matrix V, the channel space matrix U, and the residual matrix D, the speaker, channel, and residual factors can be calculated, and a score for a trial can be computed using a simple linear product. A simplified version of factor analysis can also be applied to a UBM-GMM system, using only the channel space matrix U, to do eigenchannel MAP adaptation [71, 48].

Another current approach that developed from JFA is the i-vector system [19]. In this method, the total variability is modeled in a single matrix, rather than as separate speaker and channels, i.e.,

$$s = m + Tw$$

where T is the total variability matrix, and w is the i-vector (which stands for an intermediate size vector). The matrix T is trained in a similar way as V is in the previous approach, and i-vectors are extracted. Linear discriminant analysis (LDA) and WCCN are applied to the i-vectors as channel compensation, and a score is produced using cosine distance scoring.

2.4 Speech Corpora

There are a number of conversational speech corpora utilized for speaker verification tasks. Older corpora include Switchboard-1, Switchboard-2, and Fisher [45, 46, 17]. They contain speech data collected from telephone conversations between pairs of speakers; these conversations are typically around 5 minutes in length, so that each conversation side (i.e., the side of the conversation corresponding to one speaker) is roughly 2.5 minutes in length. In addition to landline telephone data, there is a cellular telephone data set of Switchboard-2.

The National Institute of Standards and Technology (NIST) has coordinated Speaker Recognition Evaluations since 1997, and there are multiple corpora available from these evaluations; the most commonly used data sets correspond to the NIST 2004, 2005, 2006, 2008, and 2010 Speaker Recognition Evaluations (SREs) [50, 51, 52, 53, 54]. The evaluation data is taken from various stages of the larger Mixer collection [15, 16]. Each of the aforementioned SRE data sets include conversational telephone speech. Conversational speech recorded on a variety of microphones was included starting in SRE05. SRE08 introduced a different style of speech, specifically that of an interview; in these cases, most speech belongs to the interviewee, though some interviewer speech may be present. I will refer to each speech sample or utterance, whether obtained from a conversation or an interview, as a conversation side.

2.5 Performance Measures for the Speaker Verification Task

The NIST Speaker Recognition Evaluations use two performance measures for speaker recognition systems, namely the detection cost function (DCF) and the equal error rate (EER). As mentioned previously, there are two types of errors that occur in speaker verification tasks: false acceptances, or false alarms, in which an impostor speaker is incorrectly verified as the target, and false rejections, or misses, in which a true speaker is rejected as the target. For every decision threshold, there will be false alarm and miss rates that indicate the probability of each type of error occurring.

The DCF is defined as a weighted sum of the miss and false alarm error probabilities:

$$\mathrm{DCF} = \mathrm{C}_{\mathrm{Miss}} \times \mathrm{P}_{\mathrm{Miss|Target}} \times \mathrm{P}_{\mathrm{Target}} + \mathrm{C}_{\mathrm{FalseAlarm}} \times \mathrm{P}_{\mathrm{FalseAlarm|NonTarget}} \times (1 - \mathrm{P}_{\mathrm{Target}}) \quad (2.13)$$

In Equation (2.13), $\mathrm{C}_{\mathrm{Miss}}$ and $\mathrm{C}_{\mathrm{FalseAlarm}}$ are the relative costs of detection errors, and $\mathrm{P}_{\mathrm{Target}}$ is the *a priori* probability of the specified target speaker. I will use the values from SRE08, namely, $\mathrm{C}_{\mathrm{Miss}} = 10$, $\mathrm{C}_{\mathrm{FalseAlarm}} = 1$, and $\mathrm{P}_{\mathrm{Target}} = 0.01$. When DCF is given here, it refers to the minimum possible DCF, i.e., to a cost that has been minimized over possible values of the decision threshold. The equal error rate (EER) is simply the rate at which false alarm and miss probabilities are equal.

The minimum DCF and EER capture only two possible operating points for a system. In order get a better sense for how good a system is overall, there are detection error tradeoff (DET) plots, which plot the false alarm rate against the miss rate over the entire range of decision thresholds [47]. By using a logarithmic scale, a receiver operating characteristic (ROC) curve becomes a line. The better the system, the closer the DET curve will be to the lower left of the plot (i.e., smaller error rates).

2.6 Intrinsic Speaker Qualities

In general, a speech sample is affected by both intrinsic and extrinsic factors, where extrinsic factors include noise, room acoustics, and channel effects. Since the focus of my dissertation work is on intrinsic speaker characteristics, I now discuss a variety of issues and concepts relevant to a discussion of inherent speaker qualities. A brief overview of some of the major sources of variation within and among speakers is given in Section 2.6.1, including physical attributes, accent or dialect, prosody, and emotion. Additionally, in order to further explore the inherent difficulties of a speaker recognition task, the concept of the distinctiveness or recognizability of a speaker is covered in Section 2.6.2, along with various studies in which human listening has been applied to a speaker-related task. Finally, Section 2.6.3 presents work that deals with voice modifications attempted in order to fool an automatic speaker recognition system, as these studies are indicative of the effects that varying speaker characteristics can have.

2.6.1 Sources of Speaker Variation

Physical Attributes

At the most basic level, a person's voice is characterized by his vocal apparatus. The length of the vocal tract, the size of the vocal folds in the larynx, the size and shape of the nasal cavity, and other anatomical features all contribute to the acoustic properties of a person's speech, affecting formant frequencies of vowels, average pitch, pitch range, and qualities such as breathiness and nasality [20]. While an individual has a certain amount of control over the frequency characteristics of his speech and can speak outside of his typical range of everyday speech frequencies, the effects of other physical attributes, such as the size and shape of the nasal cavity, cannot be manipulated.

A person's voice will also be affected by his age and health. Physical changes that occur as a child grows into an adult are the most obvious example of aging effects, especially for male voices. However, the voice quality also changes as an adult grows older. Examination of voice spectrograms for a set of subjects over a period of years showed that the frequency of the point of concentration of formants and the mean pitch frequency decreased with increasing age, and the individual distribution curves of mean pitch frequency became more narrow, i.e., the ability to vary fundamental frequency was lost in the aging process [23].

Furthermore, a person's health will impact the way his voice sounds; for instance, a cold may make the voice hoarse or more nasal.

Language, Dialect, and Accent

The language choice of a speaker is another source of speaker individuality. In the case of multi-lingual speakers, their native language will typically influence the way they produce the speech sounds of other languages, giving rise to a foreign accent.

Furthermore, word and phone pronunciation can vary widely, even within the same language, leading to accents among native speakers. There are many accents in English, for instance: not only are there British, American, and Australian accents, but there are local regional accents within each of those groups.

In addition to different accents, languages often include different dialects, which may vary in the usage of certain words or grammatical forms, as well as word pronunciations. Variations in dialect may reflect geographical, age, socio-economic, or educational differences between speakers.

Variability in Speech Production and Prosody

Humans are able to listen to speech and identify the words and phones that are spoken. However, the same word or phone may be produced in varying ways. Speakers will differ in the precise ways of articulating a sound, as well as the degree of coarticulation between consecutive sounds. Speech rate, often measured by the number of words or phones per second, is another characteristic that will vary from speaker to speaker.

In linguistics, prosody refers to various acoustic properties of speech that can convey additional information about the utterance or speaker. Types of prosodic information include loudness, pitch, tone, intonation, rhythm, and lexical stress. Variations in prosody may indicate things such as sarcasm, speaker emotion, emphasis, or whether an utterance is a statement or a question. Furthermore, prosody is suprasegmental, meaning that prosodic features are not limited to any one segment, but occur at a higher level, across multiple segments.

The concept of speech rhythm involves a number of timing parameters, including the tempo, pauses, and various durational patterns, which may for example, be measured as the mean and standard deviation of word or phone lengths. The prosodic tendencies of a given speaker help to define his speaking style. Additional lexical information such as word usage, and the relative frequency of disfluency classes (including pause-fillers, discourse markers, or backchannel expressions) can also contribute to a speaker's individual speaking style. As described in Section 2.3.3, several of the higher-level systems for speaker recognition attempt to capture such individual variations in order to differentiate between speakers.

Emotion

The emotional state of a speaker can also impact the characteristics of his speech. A number of acoustic parameters can be involved in conveying an emotion: the level, range and contour of the fundamental frequency (perceived as pitch); the vocal energy or amplitude (perceived as voice intensity); the energy distribution across the frequency spectrum (perceived in voice quality or timbre); formant location (related to articulation perception); and a number of timing parameters, such as tempo and pauses [5].

As an example, joy typically manifests in speech as increases in the mean, range, and variability of fundamental frequency, along with an increase in mean energy. Joy may also cause a higher rate of articulation.

2.6.2 Speaker Recognizability or Inherent Challenges

A concept that is related to inherent speaker characteristics is the recognizability of a person's voice. One human listening experiment asked subjects to rate the distinctiveness of different speakers, in terms of a seven point scale describing how easy or hard the voice would be to remember [40]. An error analysis of a speaker recognition system that will be discussed in Section 2.7 also attempted to find speakers who were hard for the system to recognize. Though the results of human listening tasks may not always correspond to results obtained by automatic systems, they provide insight into the nature of challenges inherent to speaker recognition tasks.

Speaker verification by human listeners was compared to machine performance using NIST 1998 Speaker Recognition Evaluation data [65]. The human task was designed to emulate the paradigm of the NIST evaluation as closely as possible, though human constraints due to memory and fatigue imposed a limit on both the number of the trials as well as the length of speech samples. Listeners were asked to make a same or different speaker discrimination with confidence ratings (10 levels). Results showed that human listening, when individual decisions were combined, was comparable to or even better than typical computer algorithms, especially in the case of mismatched train and test handsets.

Recently, the 2010 NIST Speaker Recognition Evaluation included a human assisted speaker recognition task [27]. Participating sites evaluated a subset of trials, selected to be difficult, using any human assisted technique, including listening and examination of spectrograms or other features. The decision could be based on a group of humans, with no restriction on the use of experts or naive listeners. Analysis of results showed that this was largely a challenging task for humans, with fairly high error rates on many of the selected trials. For these difficult trials, automatic systems performed better than humans.

A study of voice identification by human listeners, relating to the reliability of the testimony of an earwitness (in a legal setting), examined a variety of issues, including familiar versus unfamiliar voices, the reliability or accuracy of voice identification, reliability as a function of time, and reliability as a function of whether or not the listener is trying to remember

the voice [18]. Examination of various studies yielded a number of conclusions. First, the length of the heard speech does not seem to have too great of an effect. Voice disguise and even unintentional changes in tone were found to greatly reduce identification accuracy, even under ideal conditions. When comparing incidentally and intentionally memorized voices, there was little evidence that voice identifications by witnesses who were unprepared or had little time to initiate efficent encoding strategies would be reliable. In terms of delay between the time of hearing the initial speech and making a voice identification, the greater the delay, the greater the likelihood of error and unreliability. Examination of the relationship between witness accuracy and confidence level showed promising, but inconclusive results.

2.6.3 Voice Modifications

As mentioned in Section 2.6.1, speakers can manipulate their voices in certain ways, even if they cannot change certain physical attributes, like vocal tract lengths or the size and shape of their nasal cavities. Changes in a speaker's voice, intentional or not, can impact speaker recognition performance.

One early study examined the effects of voice disguise and voice imitation on spectrograms [23]. For voice disguise, subjects kept the speech content the same across samples, but were allowed to differ from their normal voice in terms of pitch frequency, rate of articulation, pronunciation, and dialect. Comparison of the formant positions indicated that the formants could be shifted higher or lower than the normal voice, though the first formant was comparatively stable. In terms of voice imitation, the imitator was able to vary his mean fundamental frequency considerably in order to be more similar to a target, though he was generally unable to precisely match the formants or instantaneous fundamental frequencies of the speaker being imitated. It makes sense that the imitator could successfully change his overall average fundamental frequency , even if precise instantaneous fundamental frequencies could not be matched, given that the imitator is changing his voice according to his memory of perceived pitch of the target speaker (which may not match the actual instantaneous values). Similarly, although formant frequencies can potentially be changed, a speaker has certain habits of articulating speech sounds (leading to certain formant frequencies) that are often difficult to manipulate consciously over a continuous speech utterance. The imitator was largely successful in imitating the speech melody of a given target.

A later study examining mimicry also aimed to determine how closely an impersonator could match certain acoustic parameters of his speech to those of speech from the target figure [24]. The professional impersonation artist was given three excerpts of speech from well-known figures and asked to imitate these speakers as closely as possible, in terms of voice quality, speech style, and speech rate. A comparative recording of the same speech material was made with the artist using his natural voice and speaking style in order to find the extent to which the artist had to change his voice. The impersonator was able to successfully change his global speech rate, though he had less control over more local articulatory timing. Global fundamental frequency was also successfully matched by the impersonator, who was able to

both increase and decrease his mean fundamental frequency (by 15-30 Hz) in order to do so. The impersonator had varying degrees in success at matching the first three formant frequencies of his speech to the targets.

There have also been a number of studies exploring the effects of voice modification on an automatic speaker recognition system. The effects of intentional voice alterations (such as changing pitch or adopting an accent) were tested both for human listening experiments as well as for automatic speaker recognition system performance [36]. The speech was collected from normal subjects (that is, people who are not professional or expert mimics), in a setting that simulated a telephone conversation. Speakers were asked to disguise their voice in a variety of ways, including changing pitch, changing duration, and mimicking an accent. Automatic speaker recognition performance using a cepstral UBM-GMM system was evaluated for two conditions: training and test data from normal voice; and training from normal voice and testing from disguised voice. The normal-normal condition produced an EER of almost 0%, while the normal-disguised condition had an EER of 7.5%. However, using the decision threshold from the normal-normal system on the normal-disguised trials yielded an increase in false rejection rate from 7% to 40%, suggesting that systems are vulnerable to intentional voice disguises. A human listening experiment asked subjects to listen to two samples of about 5 seconds of speech and decide whether the utterances were spoken by the same speaker; if unsure, listeners could hear additional 5 second speech utterances, up to a limit of 20 seconds, when they had to make a final decision. The results indicated that in the normal-normal condition, automatic performance was similar to the lower quartile of human performance, though the automatic performance was better than humans in the normal-disguised case.

Another study investigated the effects of a transfer function-based voice transformation on automatic speaker recognition performance [8]. In the source-filter model of speech production, speech is modeled as a convolution of a sound source (i.e., the vocal cords) and a linear acoustic filter (i.e., the vocal tract). In the spectral domain, a speech signal X is then given by $X(f) = H(f)S(f)$, where $S(f)$ is the Fourier transform of the source signal and $H(f)$ is the transfer function corresponding to the filter characteristics of a speaker, where transfer function refers to the mapping of input to output in the frequency domain for a linear time-invariant system (such as a filter). Given knowledge of the speaker recognition method, the voices of impostors were modified to target a specific speaker. By transforming the impostor speech in such a way as to match the transfer function of a targeted speaker, they were able to increase the false alarm rate of the system from less than 1% to 97%, when using the targeted speaker's training utterance, and to 50% when using a different utterance of the targeted speaker. A previous study also tested computer voice-altered impostors, using a speech synthesis algorithm to model the spectral characteristics of a target voice [58]. In this case, the false acceptance rate increased from 1.5% to 86%.

2.7 Speaker Recognition Error Analysis

2.7.1 A Speaker Menagerie

One of the inspirations for this thesis is the work of Doddington et al., who classified speakers in groups according to the types of speaker recognition errors they cause [22]. There are 4 types of speakers defined: "goats," speakers who cause a large number of false rejections as a target speaker; "lambs," speakers who cause a large number of false accepts as a target; "wolves," speakers who cause a large number of false accepts as an impostor test speaker; and "sheep," the default type of speaker. Through the use of statistical tests, the presence of goats, lambs, and wolves was shown for a UBM-GMM system using data from NIST's 1998 Speaker Recognition Evaluation, for female speakers only.

The score for each trial of target-test pairs was considered a function of the test speaker index j and the model speaker index k. Thus, a score probability density function for a given test speaker (j) and model speaker (k) would be $f_s(\bullet|j,k)$. By asserting the null hypothesis that there are no speaker differences, the existence of goats, lambs, and wolves could be shown by considering different score distributions and disproving the null hypothesis. For the case of goats, the density function need only include the case where $j = k$, in which the density should not depend on k if goats do not exist; that is, without goats, the distribution of true speaker scores should be the same for each true speaker. For lambs and wolves analysis, the case of interest is $j \neq k$, in which the density should not depend on k if lambs do not exist, and should not depend on j if wolves do not exist. That is, if there are no lambs, the distribution of impostor scores should be the same regardless of the model speaker, while if there are no wolves, then the distribution of impostor scores should be the same regardless of test speaker.

For goats, analysis comprised computing means and variances for the sets of scores belonging to the same true speaker, and then determining if the means and variances depend on the speaker. Under the assumption that the means and variances do not depend on the speaker, only 5% of the true speaker score means should lie outside the 2.5 and 97.5 percentiles of the hypothetical speaker-independent underlying score distribution with appropriate mean and variance; if this does not hold true, then the speakers below the hypothetical 2.5 percentile can be categorized as goats. The results showed that there were, in fact, more outliers than could be accounted for by a single speaker-independent distribution.

For lambs, graphical analysis involved plotting the maximum impostor score for a model speaker against each true speaker score for that model speaker. Although this plot did not indicate any lamb sub-population of models in this analysis, the models with high maximum impostor score may be considered lamb-like.

For wolves, after computing the maximum impostor score for each test utterance, then the means and variances of sets of maximum impostor scores for the same test speaker can be calculated. As with the distribution considered in the goat speaker analysis, the means are compared with the 2.5 and 97.5 percentiles of a hypothetical speaker-independent underlying

score distribution; if more than 5% of the means lie outside these hypothetical percentiles, then there is a speaker dependence, and the test speakers with means above the hypothetical 97.5 percentile may be considered wolves. Once again, there were more outliers than could be accounted for by a single distribution, indicating the existence of wolf-ish speakers.

Furthermore, the F-test, Kruskal-Wallis test, and Durbin test were used to reject the null hypotheses at the 0.01 significance levels for goats, lambs, and wolves. The F-test is a one-way analysis of variance test used to determine statistically whether there is a speaker effect. The F-test was applied to test for potential goats by using all true speaker scores for each speaker, while it tested for potential lambs and wolves by first averaging the scores corresponding to the same model-test speaker pair (over all test utterances), and then using all impostor trials for the model speakers (in the lamb case) or test speakers (in the wolf case). The Kruskal-Wallis test is also a one-way analysis of variance, but it is non-parametric and uses ranks. For speakers with at least 5 true speaker trials, all the true speaker scores were used (goats). As with the F-test, the impostor scores were averaged for each model-test speaker pair before the test is applied (for lambs and wolves). Ranks are assigned to all of the mean scores, and ranks are summed for each speaker. Finally, the Durbin test is a two-way analysis of variance by ranks test, and was applied only to impostor scores (for lambs and wolves testing), for which the data could be viewed as conditioned on the two different speakers (i.e., the model and test speakers for each impostor score). As with the previous tests, impostor scores were first averaged across test utterances, and then the Durbin test assigned ranks to the averaged scores. The ranks were then summed for each test or model speaker, corresponding to the lamb or wolf test, respectively.

Using the rank sums from the Durbin test, a mild correlation of about 0.26 was found to exist between lambs and wolves. There were no correlations found between goats and either lambs or wolves. Furthermore, the speakers were ranked according to how goat-like they were (using the Kruskal-Wallis test) and to how wolf-like and lamb-like they were (using the Durbin test). Then, a cumulative distribution of errors for the rank ordered speakers showed that the 25% most goat-like speakers contributed 75% of the false rejection errors, though false alarm errors were more evenly distributed across speakers.

2.7.2 Related Work

Poh et al. extended the work of Doddington et al. by developing a user-specific score normalization (referred to as F-norm's variant) in order to address "badly behaved" users of the system, i.e., those users who degrade system performance [60]. Furthermore, for a multimodal biometrics context, Poh et al. developed a fusion technique that decides whether or not to fuse the output of several systems on a per user basis.

For a closed set speaker identification task, Jin and Waibel implemented a "naive delambing method" in order to reduce the effects of speakers who were likely to be identified as another speaker [31]. In the context of a vector quantization (VQ) based technique, in which codebooks are trained for each speaker, Jin and Waibel found that the closest match

in cross-validation testing for some speakers was not the correct speaker himself, and thus developed a method for modifying the codebooks in such cases. Additionally, to further reduce the effects of lamb-like speakers, these lamb speakers were located in the set (using cross-validation testing), and a threshold was set for each lamb speaker's belief heuristic value, so that identification as that lamb speaker could occur only if the score was above the belief heuristic.

2.7.3 Session Variability

Beyond considering the effects of different types of speakers, there has also been work investigating the impact that the particular training and test utterances used have on system performance [34]. A UBM-GMM system with factor analysis on male telephone data from the 2008 NIST Speaker Recognition Evaluation was first analysed with respect to performance dependence on the target speaker, focusing on the lambs and wolves of the aforementioned Doddington menagerie. Results showed an uneven distribution of false alarm errors, with 26% of the speakers causing 50% of the errors, and the 6% worst speakers accounting for 17% of the errors. The distribution of false rejection errors was also uneven, with 8% of the target speakers causing 50% of the false rejection errors, and 25% of these errors were due to 6% of the speakers.

The study also investigated the effect of the training sample used for each target speaker. Baseline performance corresponded to the training segment selected in the NIST evaluation. The best and worst training utterances were also defined for each speaker by finding the utterance that minimized or maximized the sum of false acceptance and false rejection rates, respectively. The baseline NIST performance had an EER of 12.1%, while using the best training data yielded an EER of 4.1% and using the worst training data generated an EER of 21.9%. The variability in performance demonstrated that the choice of training segment can have a significant impact.

Additional work investigated possible causes for the variable performance [33]. In particular, using data from NIST SRE08 as well as a French database of controlled read speech, BREF 120, the dependence of performance on training session was further analyzed. When switching the train and test segments of the sets used in the aforementioned work on SRE08, they found that the ranking of performance remained the same. That is, the inverted case corresponding to the original worst training segments (which become test segments in the inversion) still had the highest EER (17%) and the inverted case corresponding to the original best training segments (which are test segments in the inversion) had the lowest EER (7.4%), with the inverted NIST set performing in between the two (at 13.5%). However, the differences in performance were smaller than in the original case, suggesting that the choice of training excerpts have a greater effect than the choice of testing excerpts.

Analysis of system performance on the BREF 120 database for both male and female speakers also showed a range of performance between choosing the best training utterances and the worst, with random selection of training segments yielding performance in between

the best and the worst. The distribution of phonetic content between different training excerpts was examined as a possible contributing cause for the difference in performance. However, the results of MANOVA indicated that the phonetic distribution across the sets did not differ significantly for either female or male speakers, nor did the number of selected frames. A MANOVA testing differences across the acoustic features, in particular linear frequency cepstral coefficients (LFCCs), delta LFCCs and, delta-delta LFCCs did show some significant differences in the case of LFCCs and delta LFCCs.

Chapter 3

Speaker-Dependent System Performance

The first component of this thesis work is to establish the effects that inherent speaker qualities have on automatic speaker recognition system performance. To this end, I analyze scores from a GMM-UBM system, as well as UBM-GMM system with simplified factor analysis, in several ways. I begin with a small subset of data with limited channel variability, and gradually extend this to further exploration.

3.1 Preliminary UBM-GMM System Analysis

3.1.1 System and Data

The corpus under investigation in the following analysis is Switchboard-1 [45]. This corpus of conversational telephone speech, which has roughly 2.5 minutes of speech per conversation side, was chosen for several reasons. First, there is less channel variability than in more recently collected corpora. This is desirable for my analysis because my focus is on intrinsic speaker effects, rather than extrinsic factors like channel. Second, there is a variety of information available for the speakers, including age, education level, and dialect area.

In order to further control for channel effects, I consider only those conversation sides with electret handset labels (as determined by SRI's automatic handset labeler). This results in 3429 conversation sides from 407 speakers, of whom 199 are female and 208 are male. For my analysis, I obtain the full set of one conversation side training and testing scores, i.e., training on each conversation side, and testing every model against every conversation side, for a total of 11,754,612 trials (not including the trials where the train and test conversation sides are the same). Of these, 38,676 are target trials.

The automatic speaker recognition system used for this data is a basic cepstral gender-independent UBM-GMM. Specifically, the input features are 12th order MFCCs plus energy,

with deltas and double-deltas, with CMS applied. There are 1024 Gaussian mixtures, and the UBM is trained using a small set of 286 conversation sides from the Fisher corpus [17], a conversational speech corpus collected on the telephone. This set was chosen to be balanced in terms of sex and handset type. The conversations are about 5 minutes in length, so each conversation side contains roughly 2.5 minutes. I use SRI's UBM-GMM system implementation [37].

For additional channel compensation, I apply T-norm to this UBM-GMM system, using conversation sides from Fisher and Switchboard-1 (separate from conversation sides used for the aforementioned Switchboard-1 experimental set) as the impostor cohort. There are 327 impostor models in total, 163 female and 164 male.

3.1.2 Speaker Subset

Due to the large number of trials in this experiment, it is not feasible to visualize all of the scores at once. However, it is informative to consider a confusion matrix in order to see how the system scores vary depending on the speaker(s). Thus, limiting the speakers to those with 10 electret conversation sides, I obtain a set of scores for 15 male speakers and 19 female speakers, with a total of 340 conversation sides. A plot of the scores for these speakers is shown in Figure 3.1 for the UBM-GMM system without T-norm applied. The blocks of 10 conversation sides are labeled according to speaker number. The first 15 speakers are male, and the last 19 are female (labels 16-34). Thus, the target trials correspond to 10x10 blocks along the diagonal, with impostor trials elsewhere. The lower left and upper right quadrants are same-sex trials (male and female, respectively), while the upper left and lower right quadrants correspond to mixed-sex trials. The male only and female only quadrants are shown in Figures 3.2 and 3.3 for closer examination.

One thing to notice is the variation among target trial scores. Different speakers vary in the degree to which their target trials produce high scores. For instance, male speaker 14 and female speaker 29 tend to have higher target trial scores, while female speaker 33 tends to have lower target scores. Furthermore, speakers vary in the degree of consistency across their target scores. While some speakers appear to have fairly similar scores across all target trials, e.g. male speaker 3 and female speaker 16, others have much more variation in the range of target scores, e.g. male speaker 13 and female speaker 20.

In terms of impostor trials, it is also clear that scores are more confusable for certain speaker pairs, such as male speakers 3 and 5 or female speakers 19 and 29, and less confusable for other speaker pairs, such as male speakers 5 and 13 or female speakers 16 and 20. Additionally, we can observe tendencies across the same speaker to produce higher or lower scores as the impostor model or test segment. Those speakers with higher scores as the target model (column blocks) are potential lambs, while those speakers with higher scores as the test segment speaker (row blocks) are potential wolves. Another observation of note is that some higher scores are even produced for mixed-sex trials, such as those for male speaker 8. Finally, it is apparent that scores are not symmetric, indicating that for the UBM-GMM

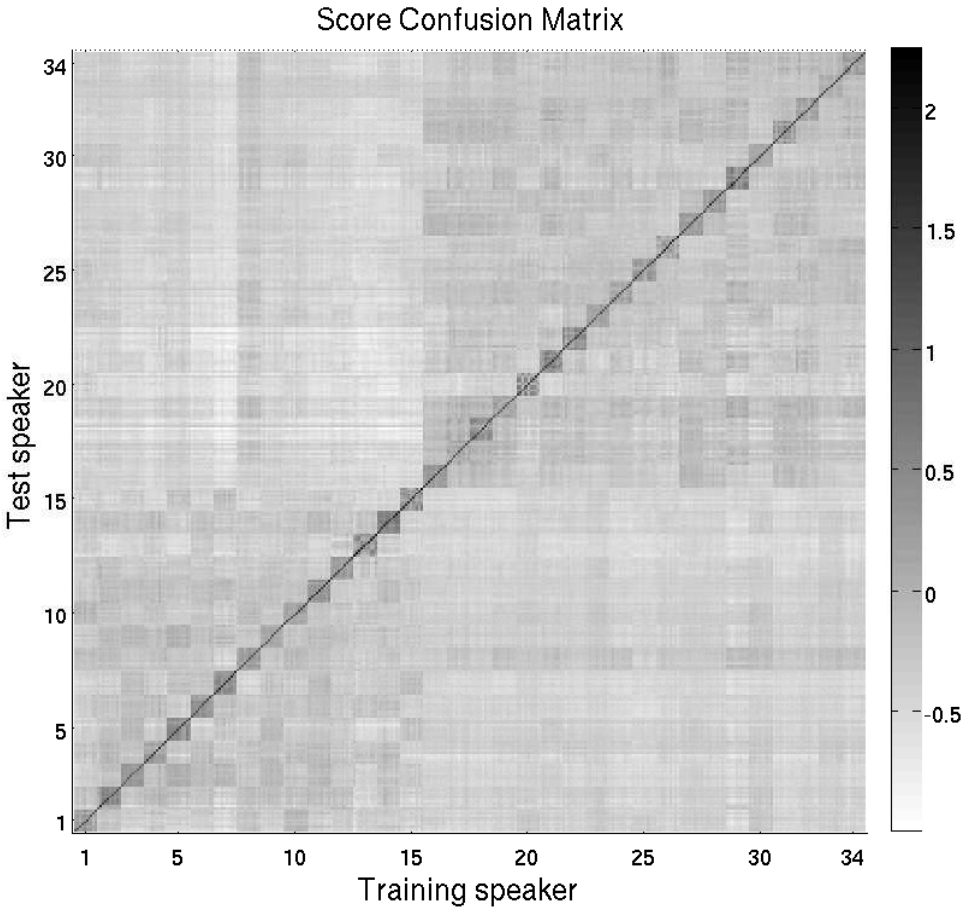

Figure 3.1: *Score confusion matrix for 34 Switchboard-1 speakers with 10 electret conversation sides each.*

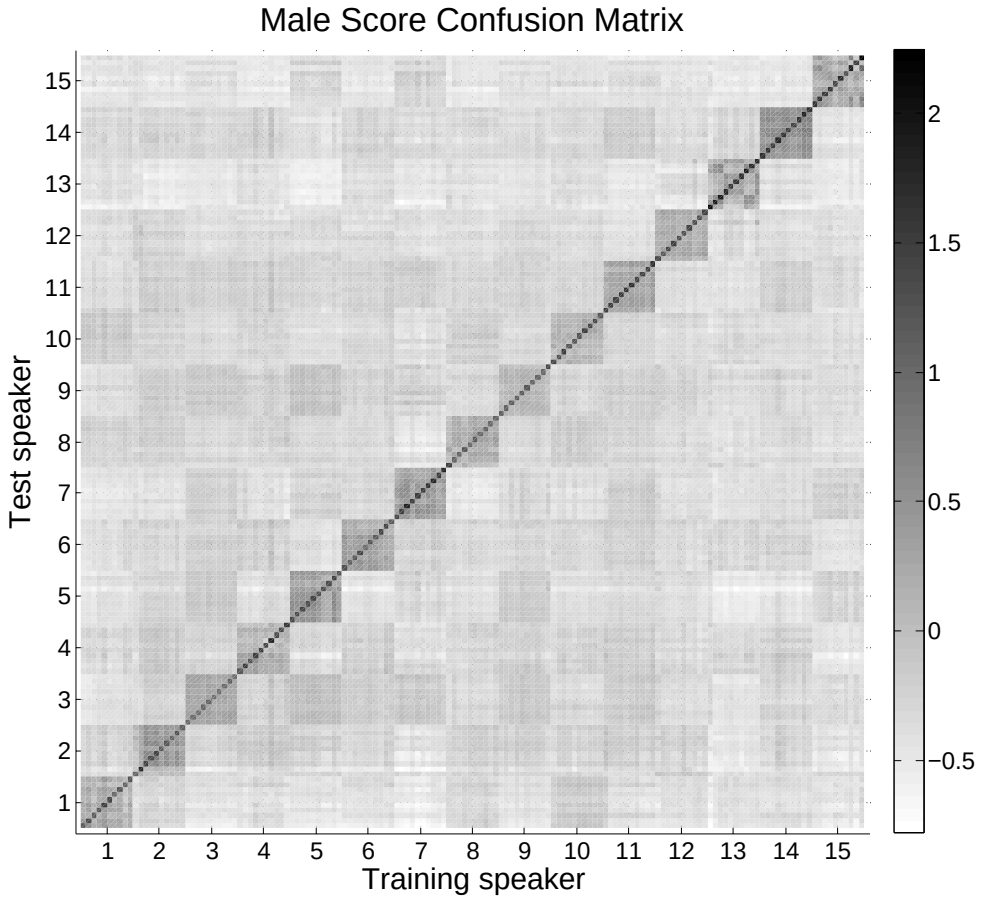

Figure 3.2: *Score confusion matrix for 15 male speakers with 10 electret conversation sides each.*

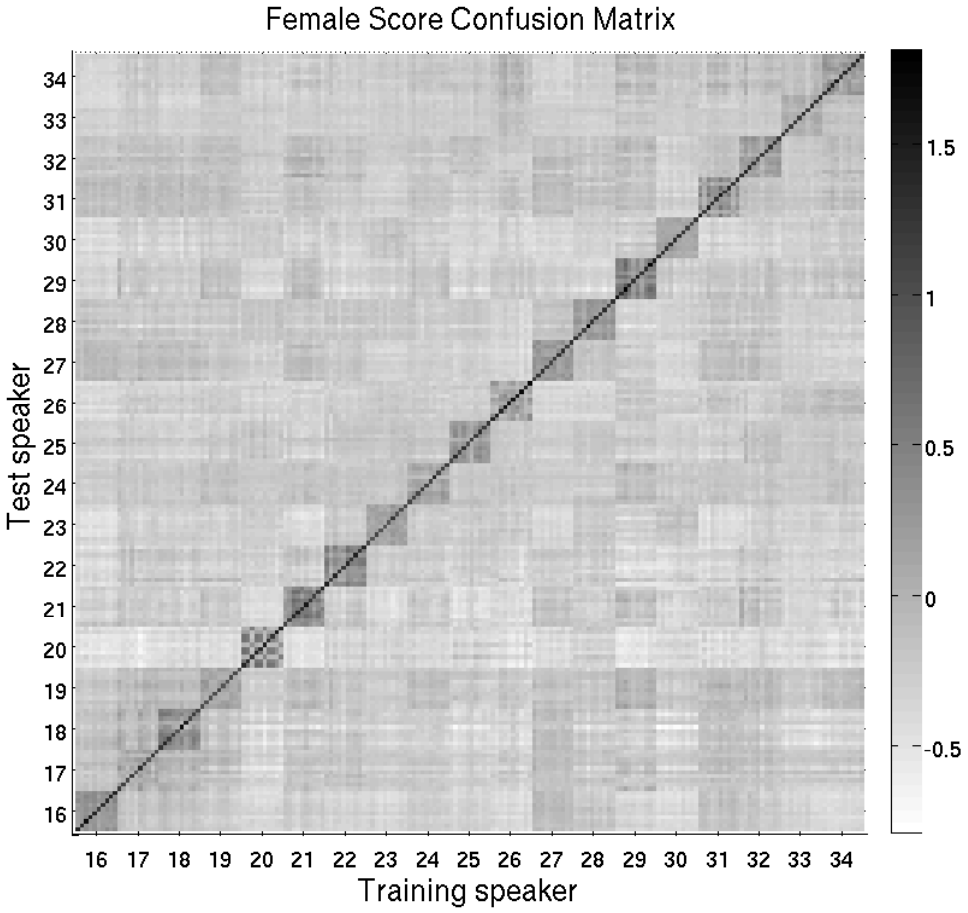

Figure 3.3: *Score confusion matrix for 19 female speakers with 10 electret conversation sides each.*

system, there is a dependence which conversation is used to train the target model.

To further get a sense of speaker behavior, I average the scores in various ways. First, consider on the true speaker trials. For each target model (of each speaker), I average the true speaker scores over the nine such trials for that target model. A plot of these averages is shown across all male and all female speakers, in Figures 3.4 and 3.5, respectively. Male speaker 14 has high average true speaker scores (an observation consistent with the notes from the confusion matrix plot), while male speakers 9 and 10 tend to have average true speaker scores on the lower end of the range. Similarly, female speakers 21 and 29 have higher average target scores and female speaker 33 has lower averages; again, such observations are consistent with those made from examination of the plot of the confusion matrix, though Figures 3.4 and 3.5 are better able to give a sense of the relative performance of speakers as true targets. It is interesting to consider the differences between averages for different target models corresponding to the same speaker. In certain cases, there are outlier target models, whose average true speaker scores are much lower than the rest, as with male speakers 2 and 13 and female speakers 20 and 24. Female speaker 22 appears to have two sets of target models, which cluster among relatively higher or lower average true speaker scores. Male speaker 3 and female speakers 19 and 23 appear to be the most consistent across target models. Clearly, the degree of consistency across true speaker trial scores is a factor in how difficult it is for a system to make a correct decision about whether the train and test speakers are the same.

Next, for a given target speaker, I average the impostor speaker scores for every test segment over all the target models of the given speaker. To begin, Figure 3.6 shows these average impostor scores for four of the male speakers. The plots in the figure contain clusters of points denoted by a symbol+color combination. Each symbol+color combination designates one particular (impostor) test speaker, and each point with that symbol+color combination corresponds to one conversation side of that test speaker. The value at each point is the average score for the given impostor conversation side, averaged over all models of the target speaker (who is the constant across all points). If the average impostor scores are typically on the high end of the range over all impostor speakers, this suggests the target speaker in question has lamb-ish tendencies, i.e., a tendency to produce high impostor scores as the target model.

Among the male target speakers, male speakers 1 and 3 have a lot of variation across the average scores for different test segments of the same impostor speaker. Speaker 3 appears to have more lamb-ish qualities, with higher average impostor scores across several impostor speakers. On the other hand, speakers 8 and 15 appear to have greater consistency in average scores across test segments of the same impostor speaker. Speaker 15 is the least lamb-ish, with fairly low average impostor scores over all impostor speakers. In many instances, the target speakers produce average impostor scores that vary across impostor speakers.

Figure 3.7 shows similar plots of average impostor scores for four of the female speakers. Examination of the female speaker plots indicates the most lamb-ish tendencies for speaker 18, and the least lamb-ish for speaker 20. Female speaker 18 shows a great deal of variation

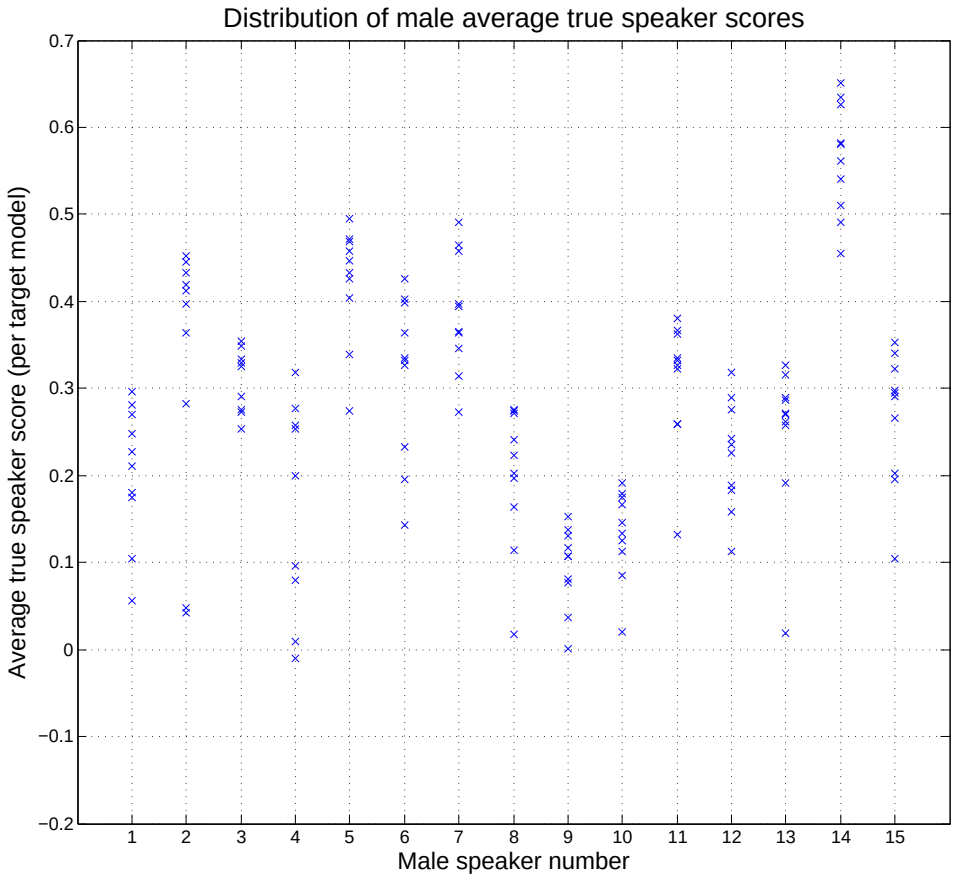

Figure 3.4: *Average true speaker score for each male target model.*

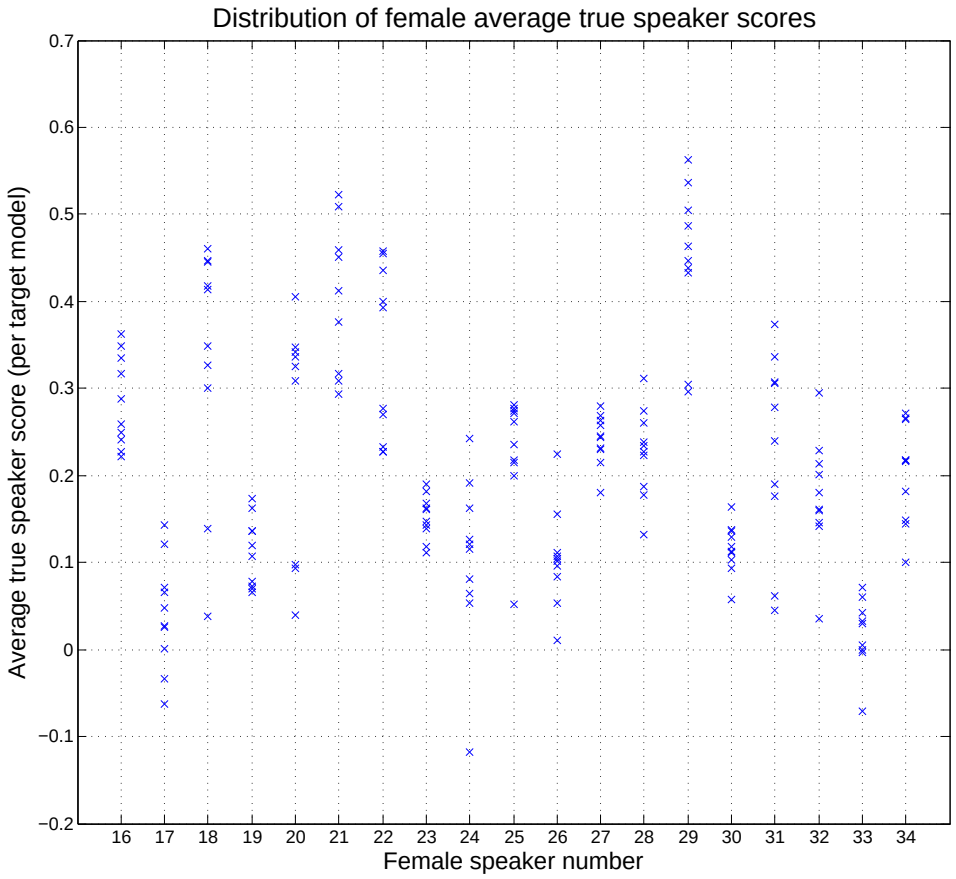

Figure 3.5: *Average true speaker score for each female target model.*

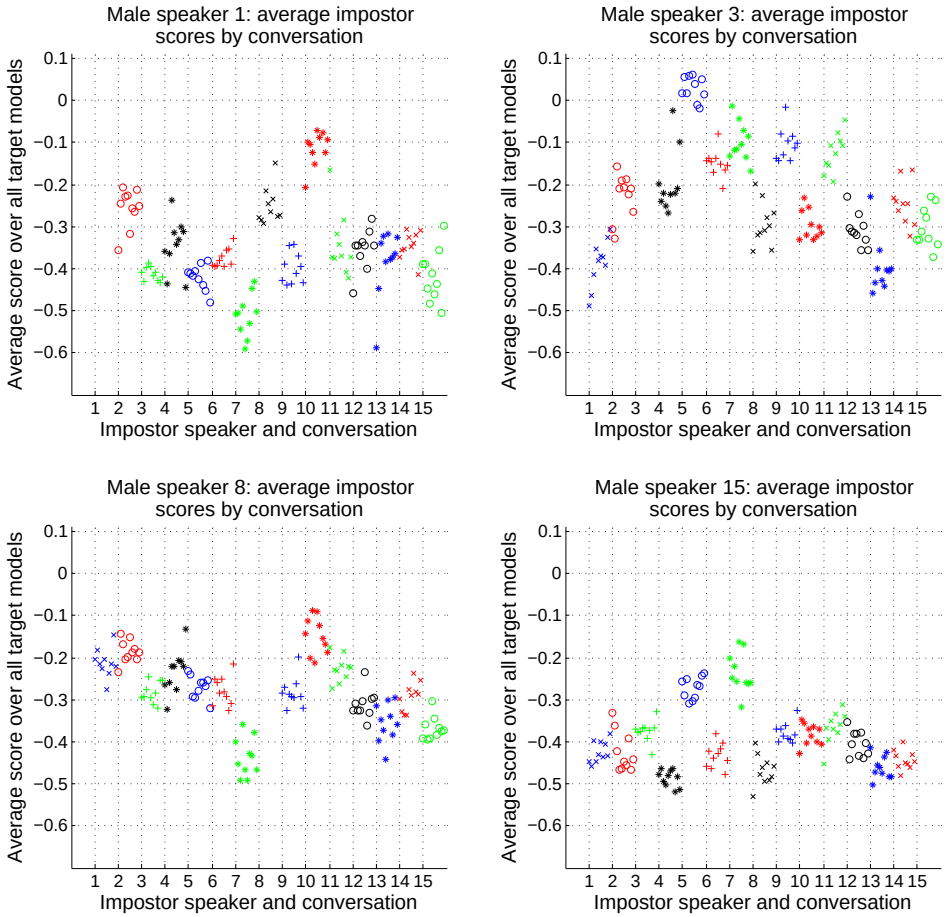

Figure 3.6: *Average scores for each impostor test segment, averaged over all target models of male speakers 1, 3, 8, and 15. Each color+symbol combination designates a particular (impostor) test speaker, whose corresponding speaker number is labeled on the abscissa. Each individual point within a color+symbol combination corresponds to a particular test utterance of that test speaker.*

in average scores across impostor speakers, and impostor test segments of the same speaker. In contrast, female speaker 33 (and to a lesser extent speaker 20) shows very similar average scores across most of the impostor speakers.

These plots clearly show different types of score distributions depending on the speaker. While some speakers produce low impostor scores and high target scores, making them less likely to cause errors given a threshold, other speakers have tendencies towards high impostor scores, or a wide range of target scores, making them more likely to produce false alarms or false rejections. Furthermore, differences have been observed not only at the speaker level, but also at the level of train and test conversation sides.

3.1.3 All Electret Trials

In order to see and analyze more speaker data, I extend the data to include all speakers, and all trials with conversation sides labeled as electret, using scores from the UBM-GMM system with T-norm. Again, the aim of such analysis is to gain better understanding of the different types of speaker behavior.

Some interesting scatter plots are shown below for female speakers. Figure 3.8 shows the average impostor score for each impostor speaker (averaged over all targets) versus the average impostor score for each target speaker (averaged over all impostors); these values have a correlation coefficient $\rho = 0.598$, implying that lambs (target speakers with high impostor scores) also have a tendency to be wolves (test speakers with high impostor scores). This correlation is reasonable since the same speaker pairs are used in the trials for calculating both impostor score averages; the only difference between the averages is whether the constant speaker is the target or the impostor. Furthermore, if false acceptance errors are caused by "average" speakers being confusable, then it makes sense that an "average" speaker would be confusable with other speakers, both as the target and as the test.

Figure 3.9 shows the average impostor score versus the average target score for speakers as the target speaker. With a correlation coefficient of $\rho = -0.485$, the implication is that goats (speakers with low target scores) also have a possibility of being lambs (target speakers with high impostor scores).

The same plots are shown for male speakers in Figures 3.10 and 3.11.

In the first plot, showing average impostor score as the test versus average impostor score as the target, there is an even higher correlation of $\rho = 0.682$, again suggesting that lamb-ish and wolf-ish behavior are related.

The second plot, showing average impostor score versus average target score as the target speaker, yields a smaller correlation in the male case, with $\rho = -0.277$, though it is still negative. There is less evidence to suggest that goats may have a tendency to also be lambs. It is possible that the correlations in these plots may be due to other factors, such as differing numbers of target and impostor trials per speaker, or poor audio quality for some conversation sides.

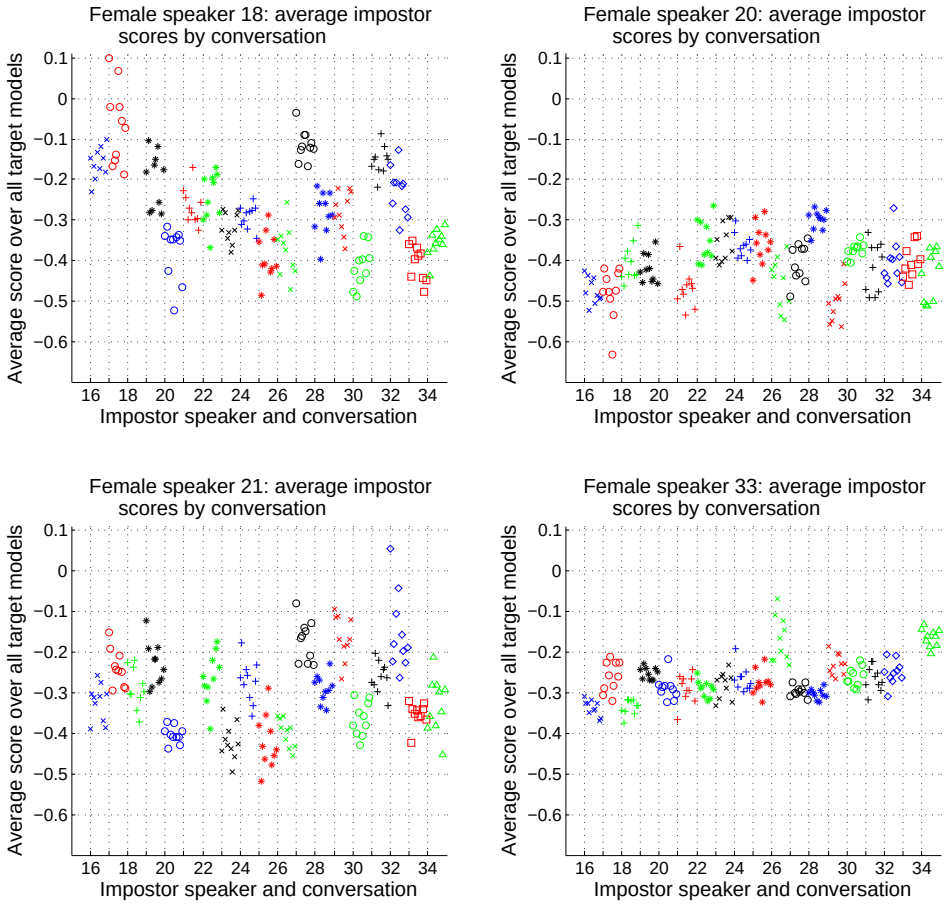

Figure 3.7: *Average scores for each impostor test segment, averaged over all target models of female speakers 3, 5, 6, and 18. Each color+symbol combination designates a particular (impostor) test speaker, whose corresponding speaker number is labeled on the abscissa. Each individual point within a color+symbol combination corresponds to a particular test utterance of that test speaker.*

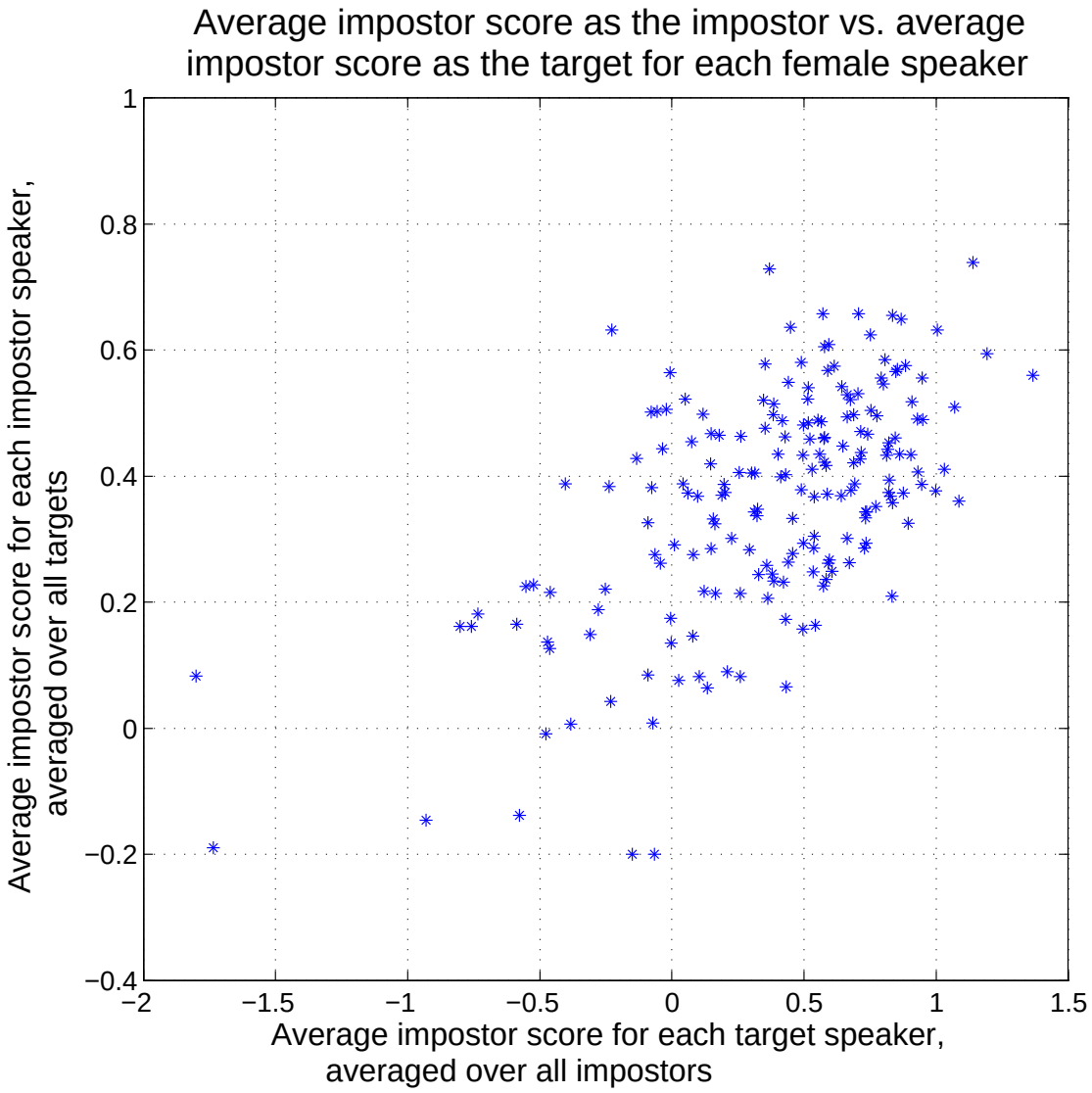

Figure 3.8: *Average impostor score for speaker as the impostor versus average impostor score for speaker as the target, for female speakers.*

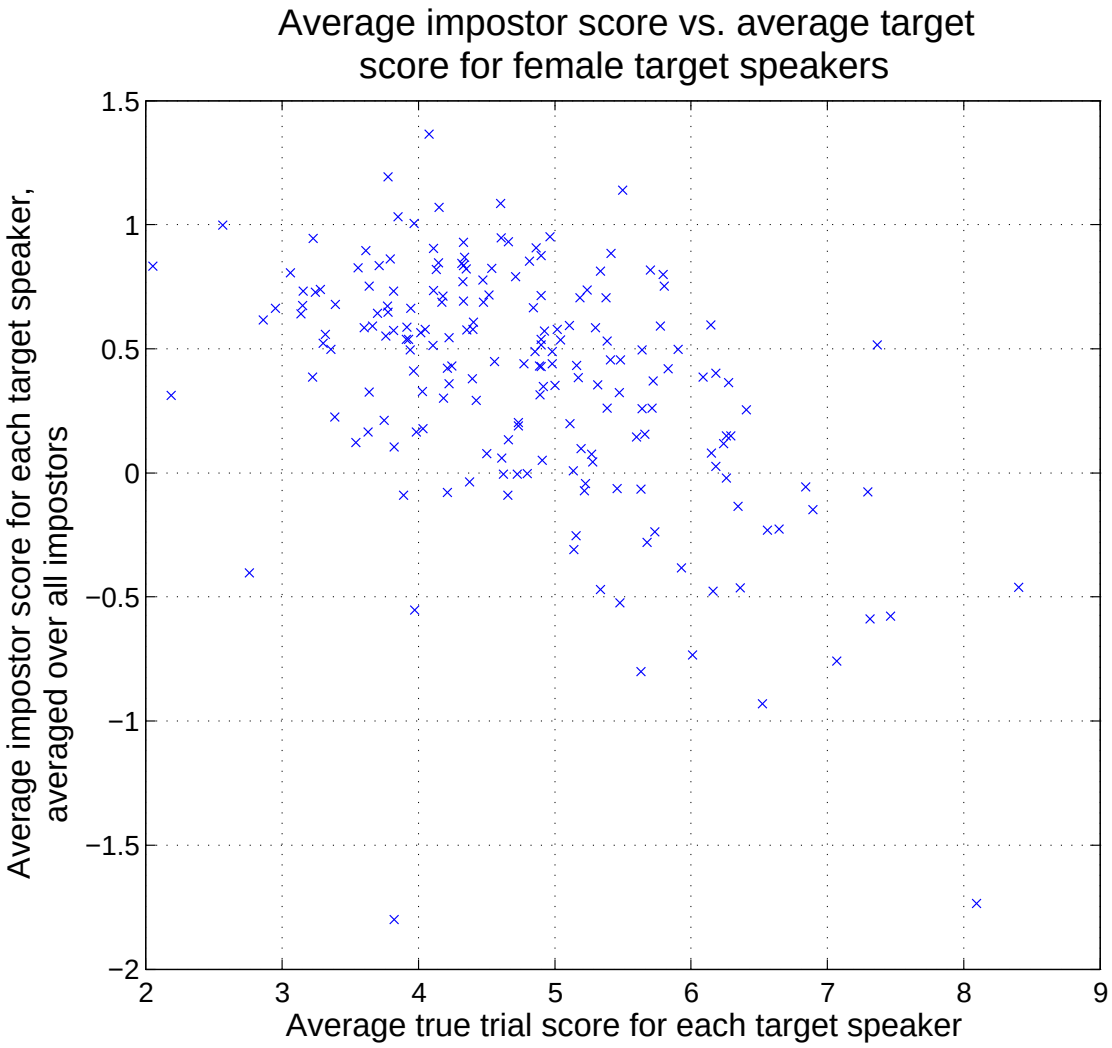

Figure 3.9: *Average impostor score versus average target score for female target speakers.*

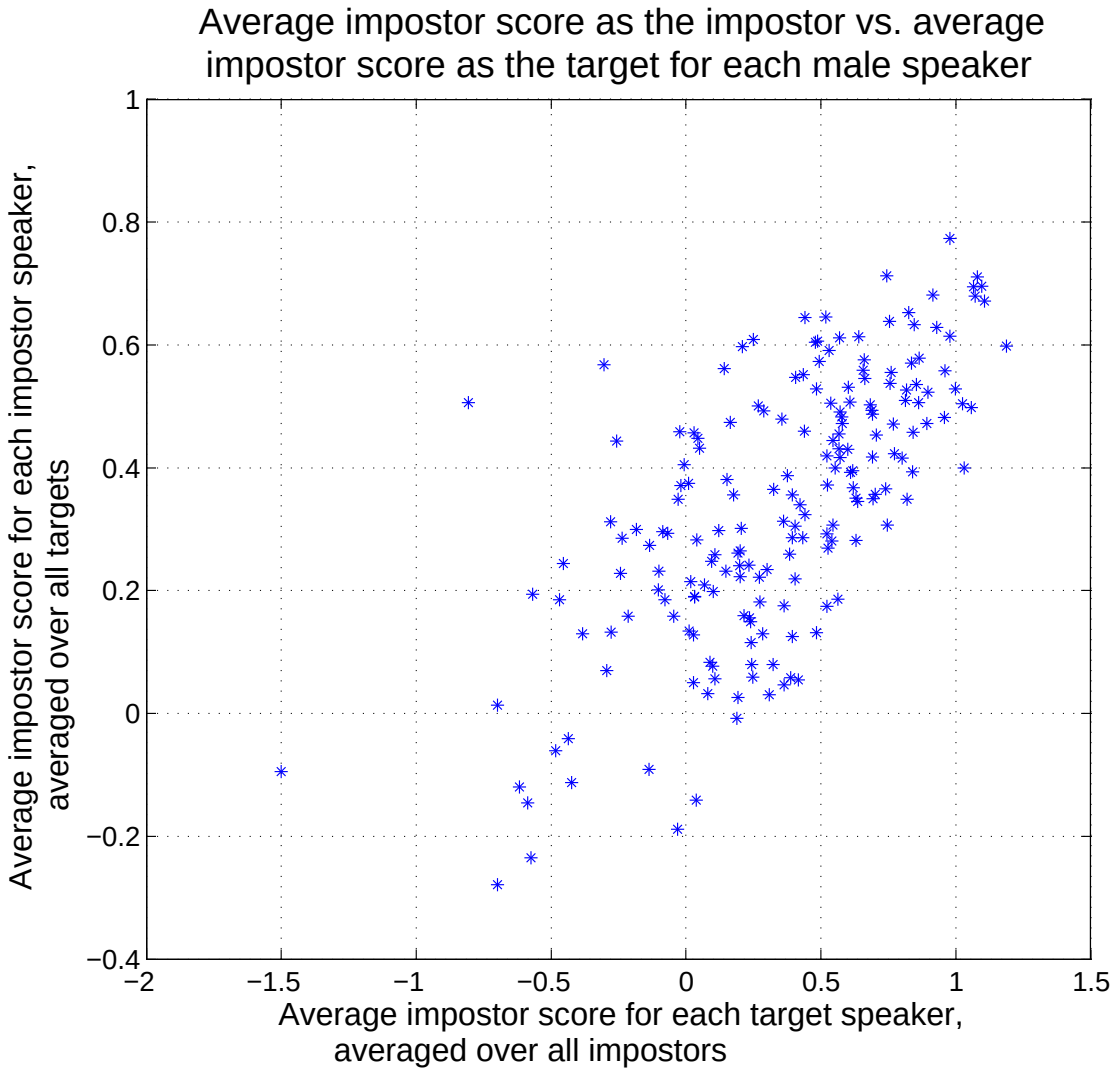

Figure 3.10: *Average impostor score for speaker as the impostor versus average impostor score for speaker as the target, for male speakers.*

Figure 3.11: *Average impostor score versus average target score for male target speakers.*

Next, I consider a number of plots focusing on showing evidence of goat-, lamb-, and wolf-type speaker populations, similar to those shown in the prior work of Doddington, et al. [22]. The first plot, addressing goat-like tendencies, that is, causing missed detection errors in true speaker trials, is shown in Figures 3.12 and 3.13 for males and females, respectively. Here, the average true speaker score is plotted against the number of true speaker trials for each speaker. In these plots a large amount of outlying average target scores would indicate greater variability across speakers. However, it is not clear in either plot that there are more outliers than would be expected if the target score distribution did not depend on the speaker, though there appear to be a handful of goat-ish speakers among females with fewer than five target trials (indicated by those points showing the lowest average target scores).

To look for a population of lambs, namely those speakers who cause false alarm errors as target speakers, I plot the true speaker scores for a target model against the highest impostor score for that target model. This plot is shown in Figure 3.14 for males, and in Figure 3.15 for female speakers. In both male and female plots there is a large cluster of points indicating speakers without lamb-ish tendencies, i.e., those with maximum impostor scores less than, or on par with, true speaker scores. However, there are also many instances showing maximum impostor scores greater than the target scores for target models, and also greater than most other maximum impostor scores, suggesting lamb-like tendencies for some speakers.

Finally, in Figures 3.16 and 3.17, I plot the average maximum impostor score against the number of test conversation sides for each impostor speaker, for male and female speakers, respectively. As was the case with the earlier plots of average target scores, there is no clear evidence that the average maximum impostor score distributions are speaker-dependent. Interestingly, there seem to be more outliers on the low end, i.e., with low average maximum impostor scores, than on the high end (which would indicate wolf-ish tendencies).

3.1.4 Effects of Speaker Demographics on System Scores

Continuing with the UBM-GMM system with T-norm, using Switchboard-1 electret conversation sides, I now switch focus to consider whether speaker demographics are evident in system scores. For the Switchboard-1 corpus, the following information is available for each speaker: sex, birth year, education level, and dialect area. The possible education levels are less than high school, less than college, college, and more than college. The dialect area corresponds to the region where the speaker lived for his first 10 years; the possible areas include New England, North Midland, South Midland, Western, New York City, Northern, Southern, and Mixed. In order to assess what characteristics have an impact on the scores produced by the system, I performed an analysis of variance (ANOVA) test for a number of different score distributions, described below. In each case, the probability (p) given to show significance level is the probability of being incorrect in concluding that the distributions are not the same.

Since trial independence is an incorrect assumption, target scores were averaged for target

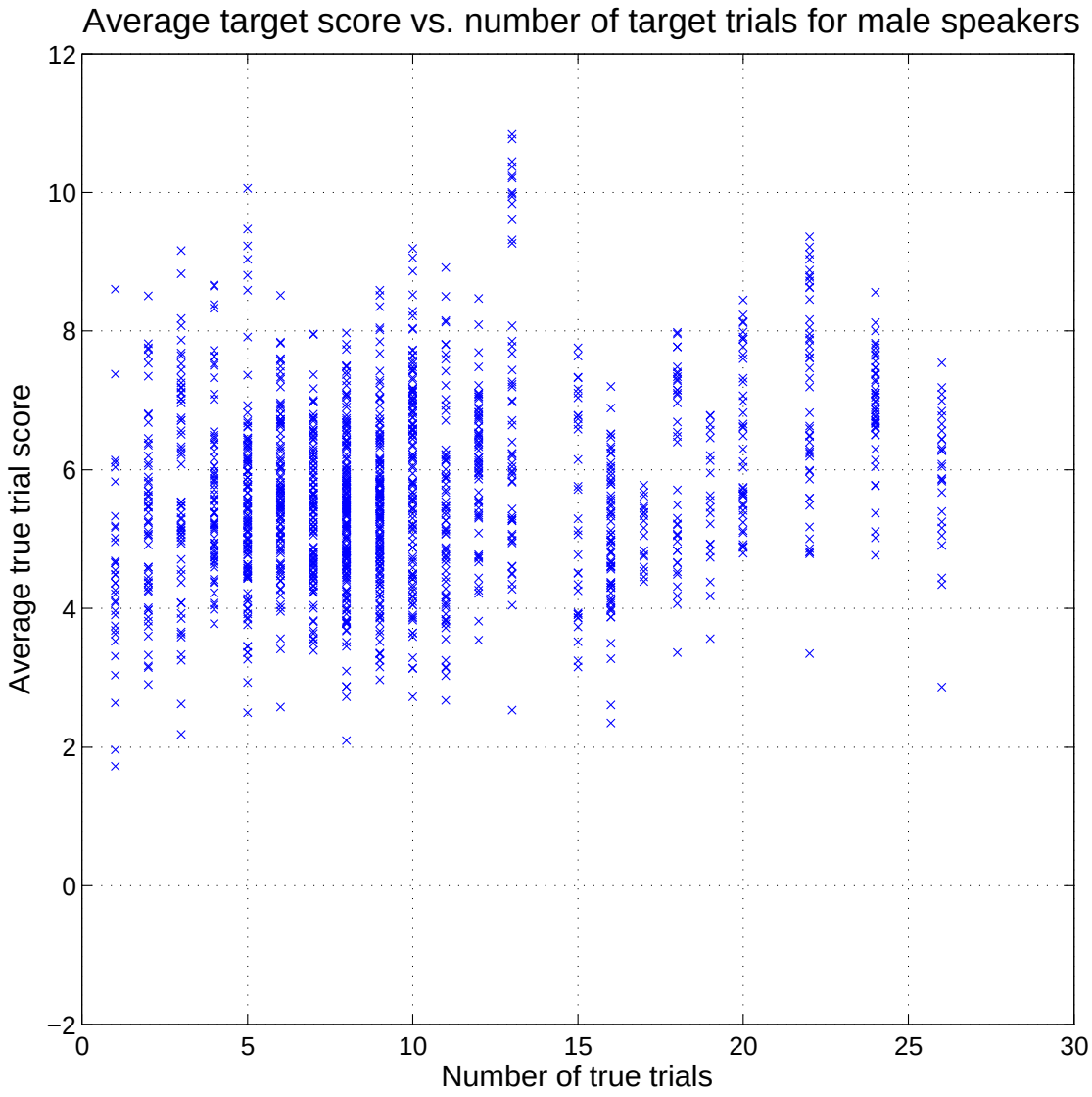

Figure 3.12: *Average true speaker score versus number of true speaker trials, for male speakers.*

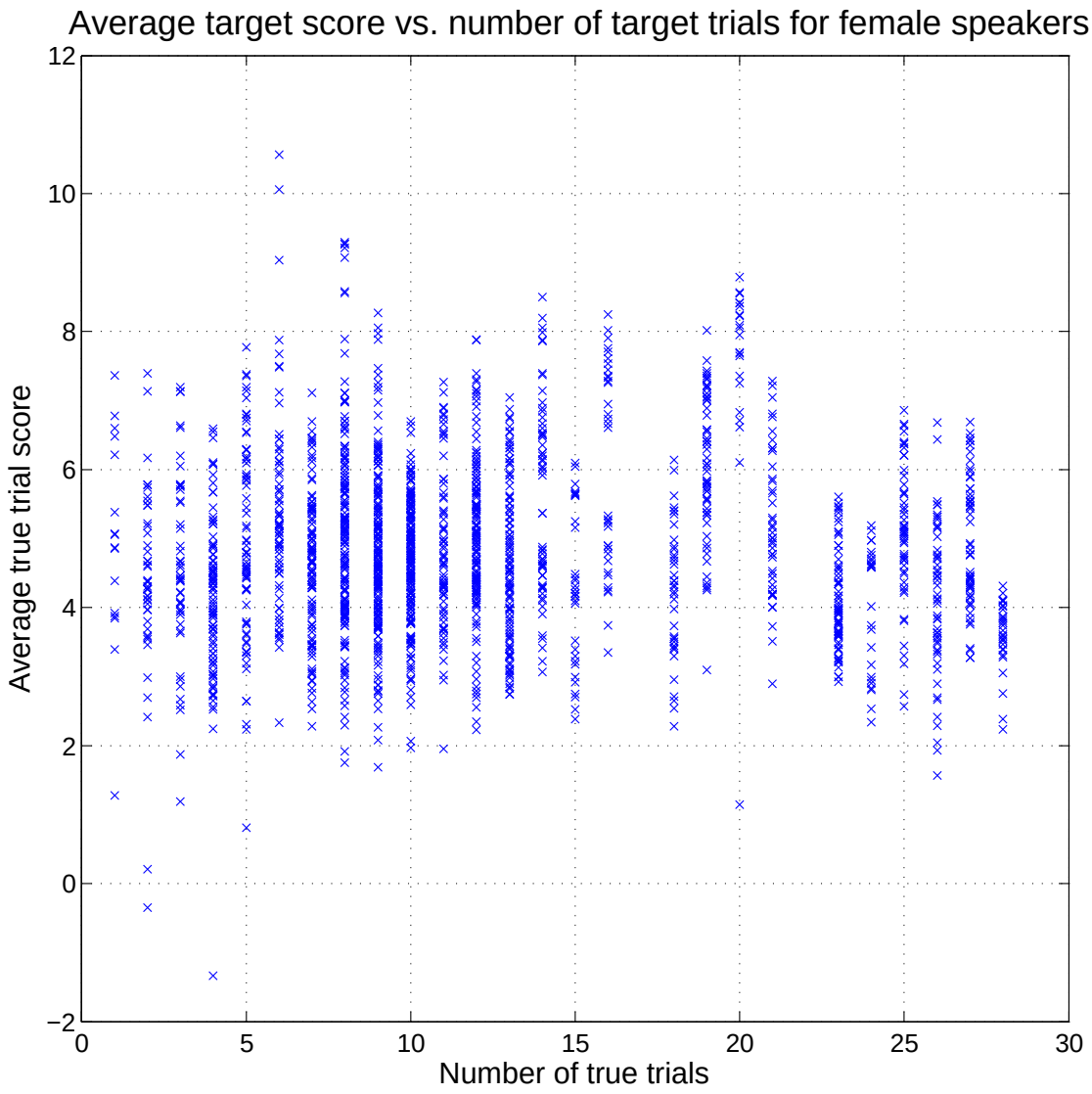

Figure 3.13: *Average true speaker score versus number of true speaker trials, for female speakers.*

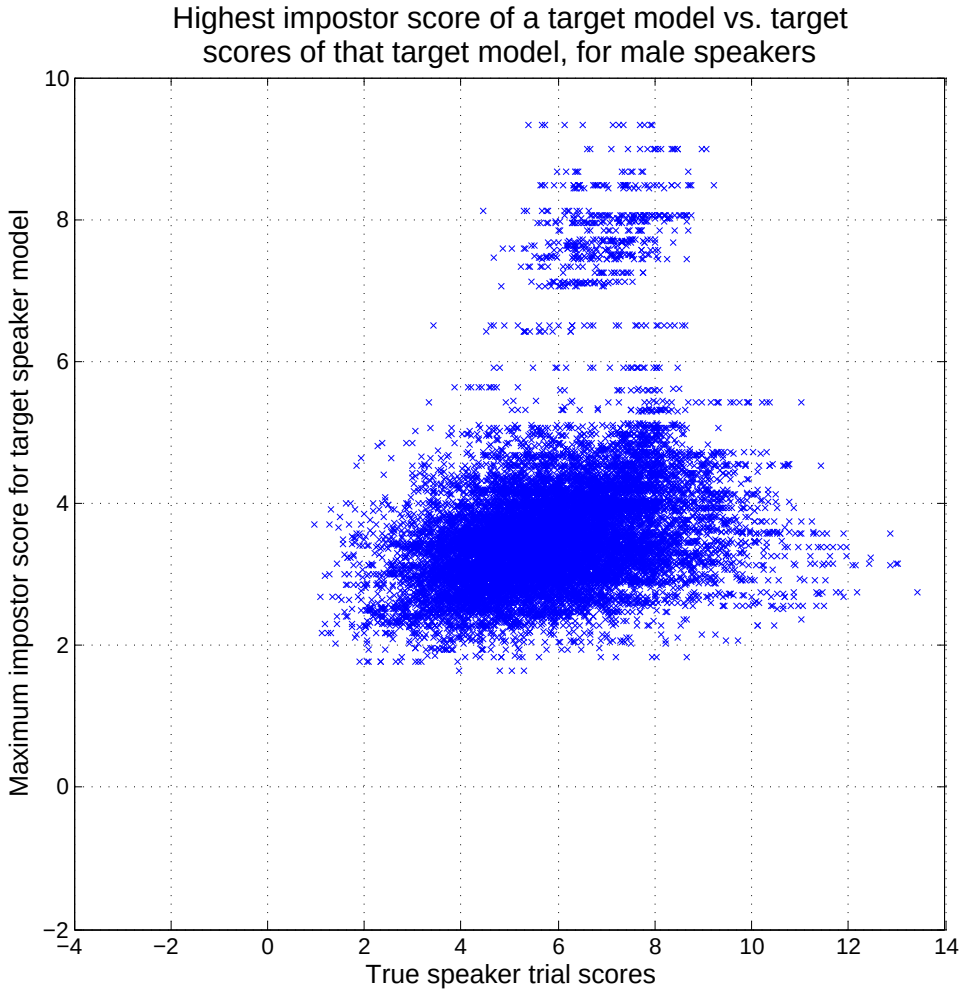

Figure 3.14: *Highest impostor score for a target model versus the true speaker scores for that target model, for male speakers.*

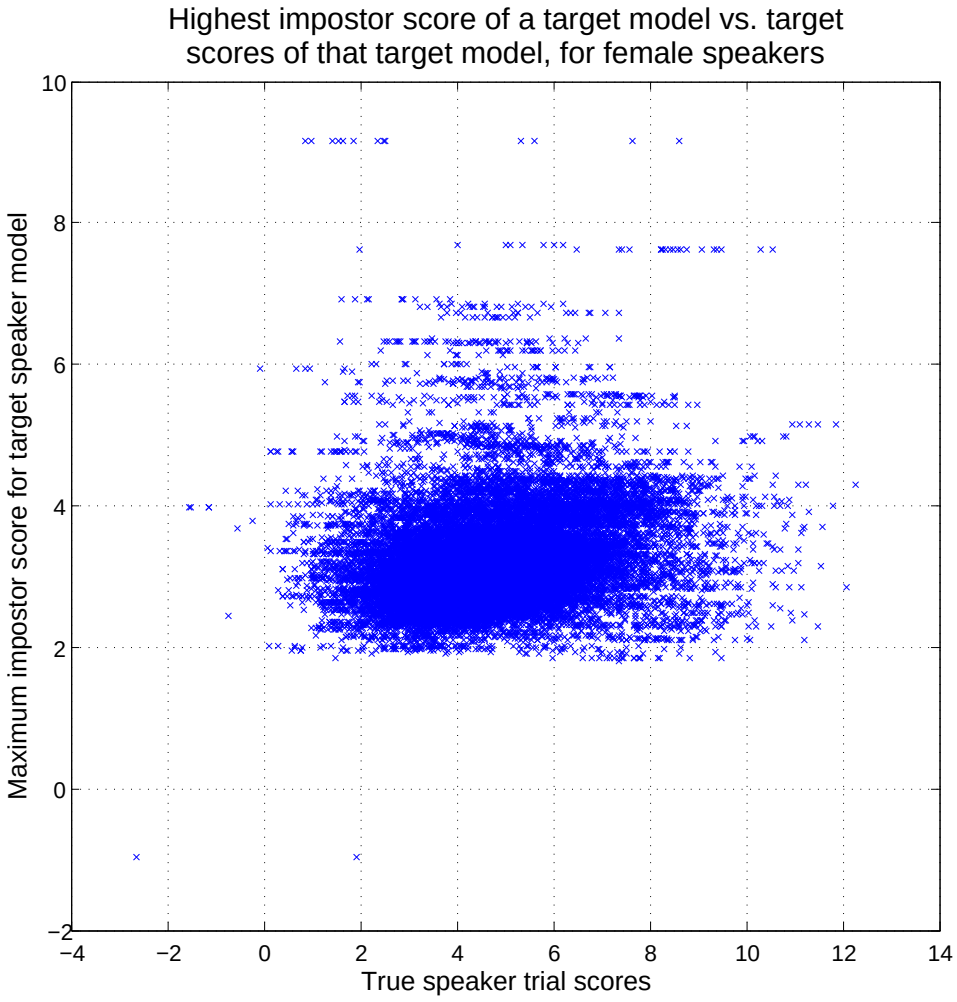

Figure 3.15: *Highest impostor score for a target model versus the true speaker scores for that target model, for female speakers.*

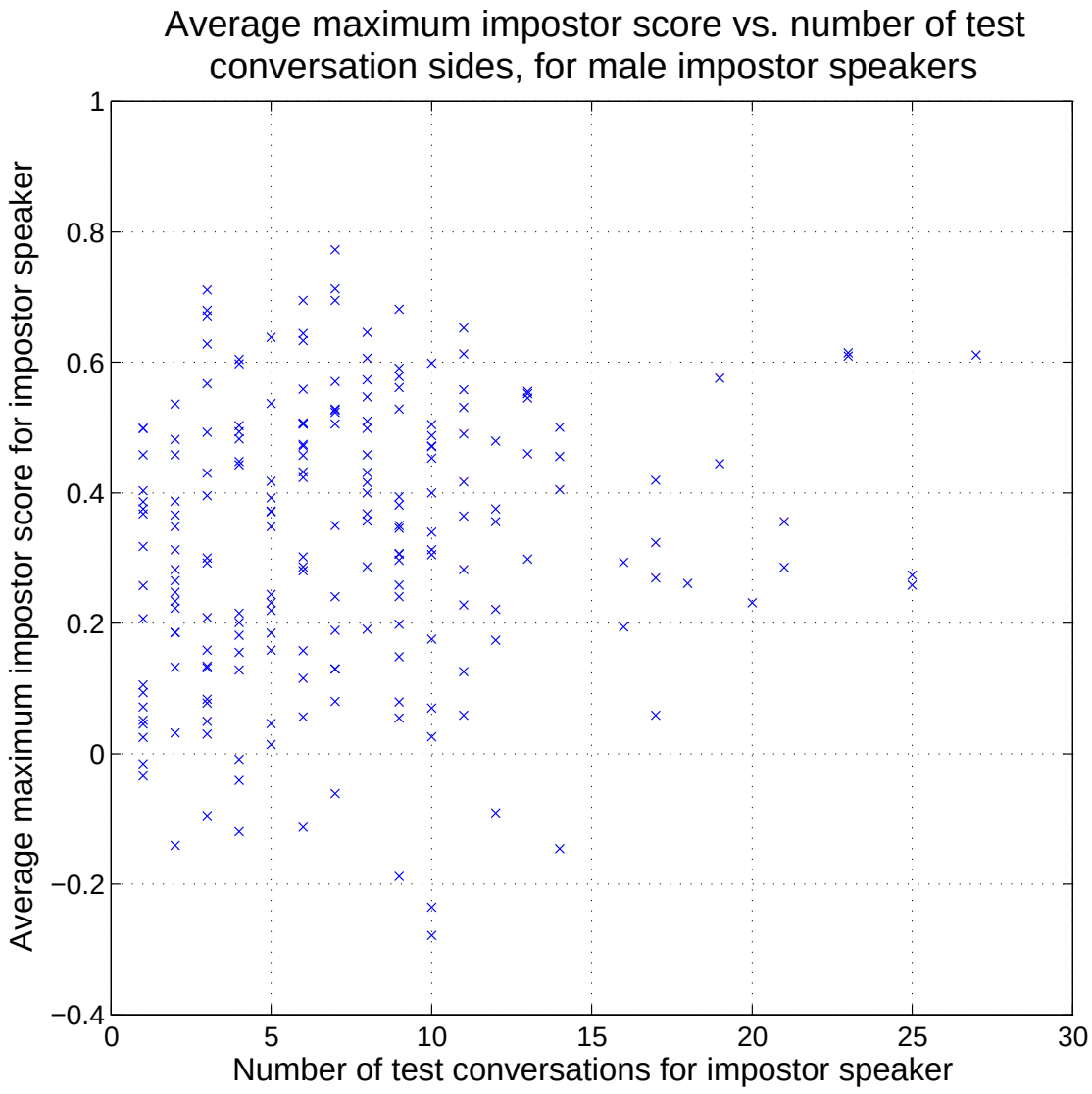

Figure 3.16: *Average maximum impostor score versus number of test conversation sides, for male impostor speakers.*

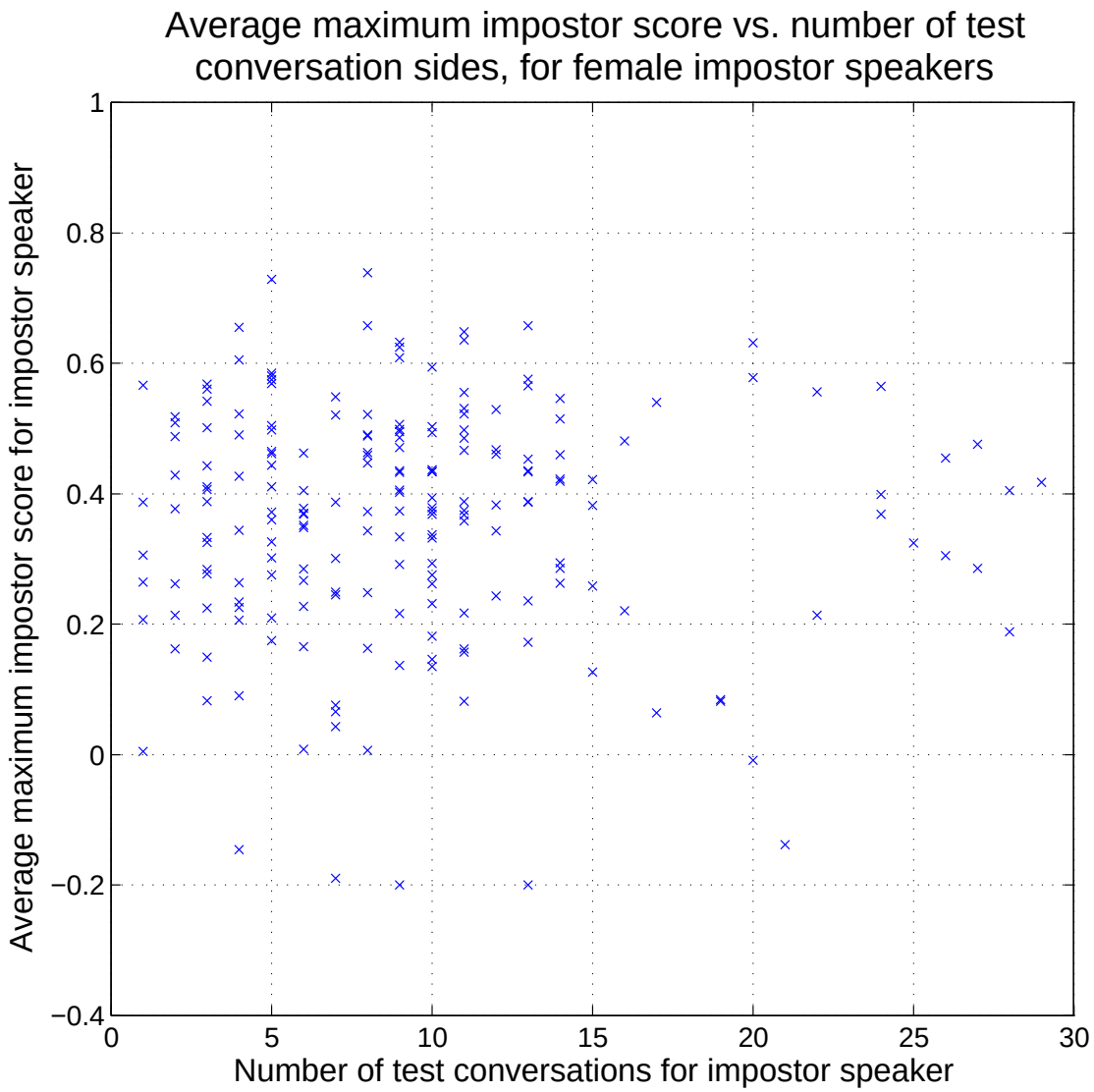

Figure 3.17: *Average maximum impostor score versus number of test conversation sides, for female impostor speakers.*

speakers over all target trials for each speaker before ANOVA analysis was done. I first looked at the target scores for female speakers compared to the target scores for male speakers. In this case, I found that the distribution of male target scores differed significantly from the distribution of female target scores ($p \ll 0.01$), with male target trials having a higher average score. Next, for female and male speakers separately, I considered the effect of age, education level, and dialect. The target score distributions for different age groups (20-29, 30-39, 40-49, and 50-69) did show a significant difference ($p = 0.054$ for females and $p = 0.013$ for males), meaning that the score distribution for at least one age group differed from the rest. However, a pair-wise comparison test (designed to keep the total probability of error to less than 10%) showed that significant differences only occurred between two pairs of distributions: 20-29 versus 30-39 (for males only) and 20-29 versus 50-69 (for both males and females). Education level did not result in differing target score distributions for either sex. Finally, although there appeared to be some differences in distributions for different dialects, only males showed a significant difference (i.e., at least one dialects distribution was different, with $p = 0.013$), and pair-wise comparisons found only two pairs of dialects to have significantly different score distributions (New England versus New York City and Northern versus New York City).

For impostor scores, the assumption of trial independence is again incorrect. In this case, I considered three different approaches: an assumption of target-test speaker pair independence, wherein impostor scores are averaged for each target-test speaker pair; an assumption of target speaker independence, wherein impostor scores are averaged for each target speaker; and an assumption of impostor speaker independence, wherein impostor scores are averaged for each impostor speaker. As would be expected, there is a significant difference in score distributions for same-sex speaker trials and different-sex speaker trials ($p \ll 0.001$ for all averaging approaches). When comparing scores for which the target and impostor speaker have an age difference of 5 years or less to scores for which the age difference between speakers is greater than 5 years, there is also a significant difference for both females and males ($p \ll 0.001$ when averaging for each speaker pair or for impostor speakers, $p < 0.028$ when averaging for target speakers). A comparison of the scores where the target and impostor speakers have the same education level to scores where the speakers have different education levels did not show any significant differences for either sex. Finally, looking at trials with speakers of the same dialect area versus trials with speakers of different dialect areas, there were significant differences when treating the speaker pairs independently ($p = 0.079$ for females and $p = 0.013$ for males), and for females when treating the impostor speakers independently ($p = 0.063$). Perhaps more significant differences are not found in this case because the dialect region information collected does not accurately reflect dialectal differences for all the speakers.

3.2 Analysis of Recent System and Data Set

I now move on from Switchboard-1 analysis to the more recent SRE08 corpus, which contains greater degrees of channel variability. The SRE08 short2-short3 condition uses roughly 2.5-3 minutes of speech for both training and testing [53]. This speech may be taken from one side of a conversation between two people, or from part of an interview. Furthermore, the data includes both telephone and microphone channels (there are 14 types of microphones).

Using the short2 and short3 conversation sides, I generate a set of trials different from those used in the NIST evaluation. For my purposes, I use conversation sides from all speakers with at least 5 available speech utterances. In some cases, the same conversation side was recorded on multiple channels (telephone and microphones, or just microphones). In these cases, I selected only one instance of that conversation side, in order to prevent the introduction of confounding factors due to having the same lexical content across different speech samples. There are 416 speakers (256 female, 160 male), with 3049 conversation sides, and a total of 22,210 target trials. For each impostor speaker pair, five impostor trials are chosen (along with the corresponding trials that have the train and test data switched), for a total of 453,600 impostor trials.

In order to better address the effects of channel variability, I use a UBM-GMM system with simplified factor analysis applied, implemented with the ALIZE toolkit [9]. The UBM is trained using 1553 conversation sides from Fisher and Switchboard-2. The rank 70 eigenchannel U matrix for simplified factor analysis is trained using 1900 conversation sides from SRE04 telephone data (99 speakers with 10 conversation sides each) and SRE05 microphone data (91 speakers with 10 conversation sides each). For the given set of trials, the system has a minimum DCF of 0.382 and an EER of 8.93%.

3.2.1 Target Trials and Goat-ish Behavior

I begin by performing an analysis of variance (ANOVA) test using all target trial scores for each speaker in order to determine if there is a speaker effect on the means. With a resulting $p \ll 0.001$, the null hypothesis that the target scores come from the same (speaker-independent) distribution can be rejected. Figure 3.18 shows a box plot for the male target scores, by speaker. It is clear that the distributions vary across speakers in this case.

Similarly, application of the Bartlett multiple-sample test for equal variances to the target scores also rejects the hypothesis that the scores come from normal distributions with the same variance.

Next, I perform a Kruskal-Wallis test, a non-parametric analysis of variance test that uses ranks and avoids the need for an assumption that the scores are normally distributed. Once again, the results of such a test for the target scores are conclusive in rejecting the null hypothesis that the score distributions do not depend on speaker, with $p \ll 0.001$.

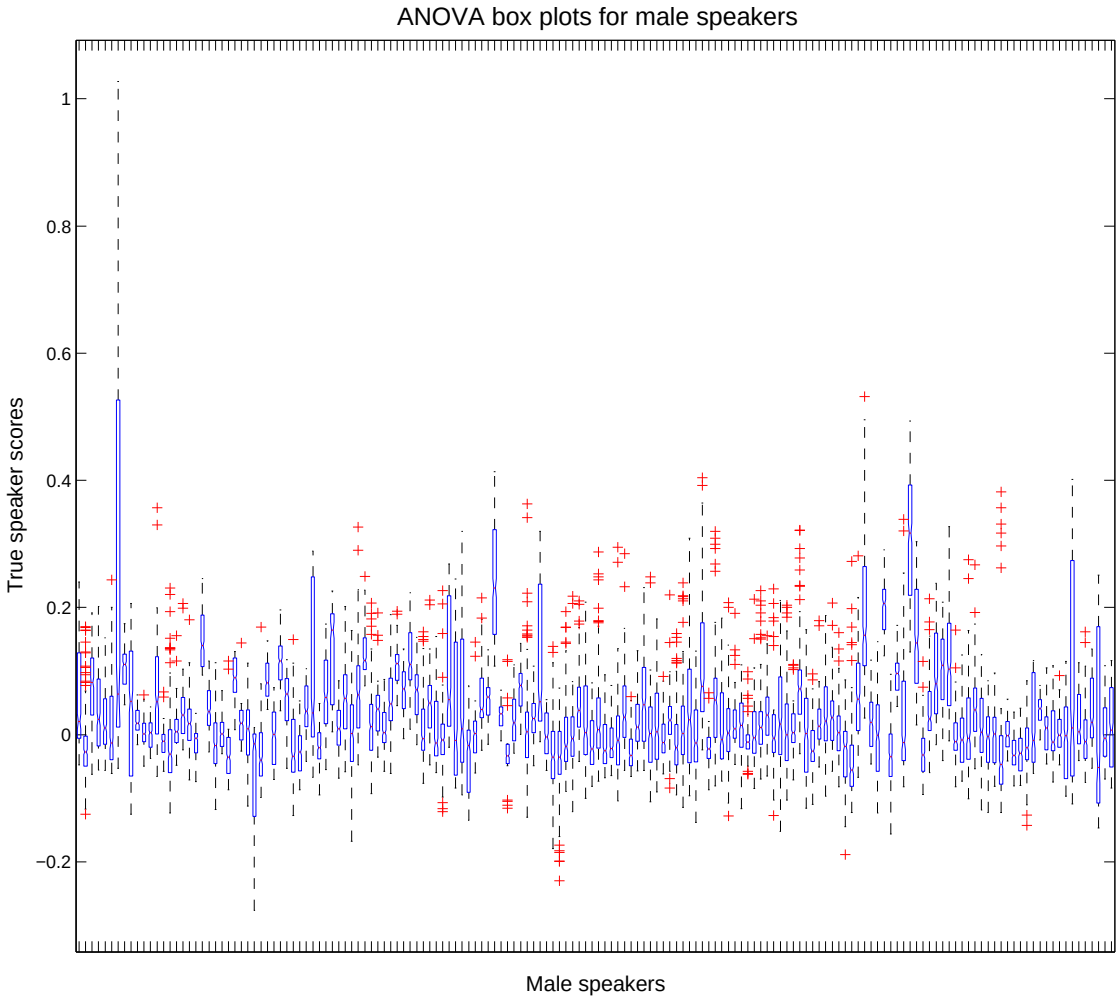

Figure 3.18: *Box plots of target score distributions per speaker, for male speakers, using SRE08 data.*

3.2.2 Impostor Trials and Lamb-ish or Wolf-ish Behavior

After averaging impostor scores for each impostor speaker pair, I considered both the set of these average impostor scores for each target speaker (looking for lambs) and the set of average impostor scores for each test speaker (looking for wolves). In both cases, application of ANOVA did not reject the null hypothesis ($p > 0.44$ for female, male, and all speakers). Similarly, the Kruskal-Wallis Test did not reject the null hypothesis that these scores do not depend on the speaker, though the female speakers came closest to significant differences, with $p = 0.11$.

3.2.3 Distribution of Errors Across Speakers

Using the threshold corresponding to the minimum DCF, errors for each speaker are counted. In particular, I count the number of false rejections (to find goats), the number of false acceptance errors as the target speaker (to find lambs), and the number of false acceptance errors as the test speaker (to find wolves). Cumulative distributions of these errors are plotted for female and male speakers in Figures 3.19 and 3.20, respectively.

There is a very speaker-dependent distribution of errors for female speakers, for all three types of errors. In the case of false rejections, 50% of the errors are due to 38, or roughly 15% of the speakers. This is even more drastic for false acceptances as the target speaker, for which 18, or roughly 7% of the speakers cause 50% of the errors. For false acceptances as the test speaker, 61, or about 24% of speakers account for 50% of the errors.

The story is similar for male speakers. Once again, a speaker-dependent distribution of missed detection errors is observed, with 23, or about 14%, of the speakers producing 50% of the errors. Only 25, or 16%, of the speakers account for 50% of the false alarms as targets, while 33, or 21%, of the speakers produce 50% of false alarms as impostor speakers.

The uneven distribution of errors across speakers suggest goat-like, lamb-like, and wolf-like tendencies for both male and female speakers.

3.3 Discussion

The examination and analysis of system scores presented here has demonstrated that automatic speaker recognition system performance is dependent on the speakers. Speakers may be difficult to correctly verify as the true speaker, and speakers may generate high impostor scores, as either the target speaker, the test speaker, or both.

However, I have also observed a dependence on which segments are selected for training and testing; certain conversation side train-test pairings may produce errors, while others corresponding to the same speaker or speaker pair may not, and scores are not symmetric for a given pair of conversation sides (i.e., switching which utterance is used to train the target model will change the score). Such results suggest that any attempts to predict or use information about how a system will respond to speakers may need to take an approach involving

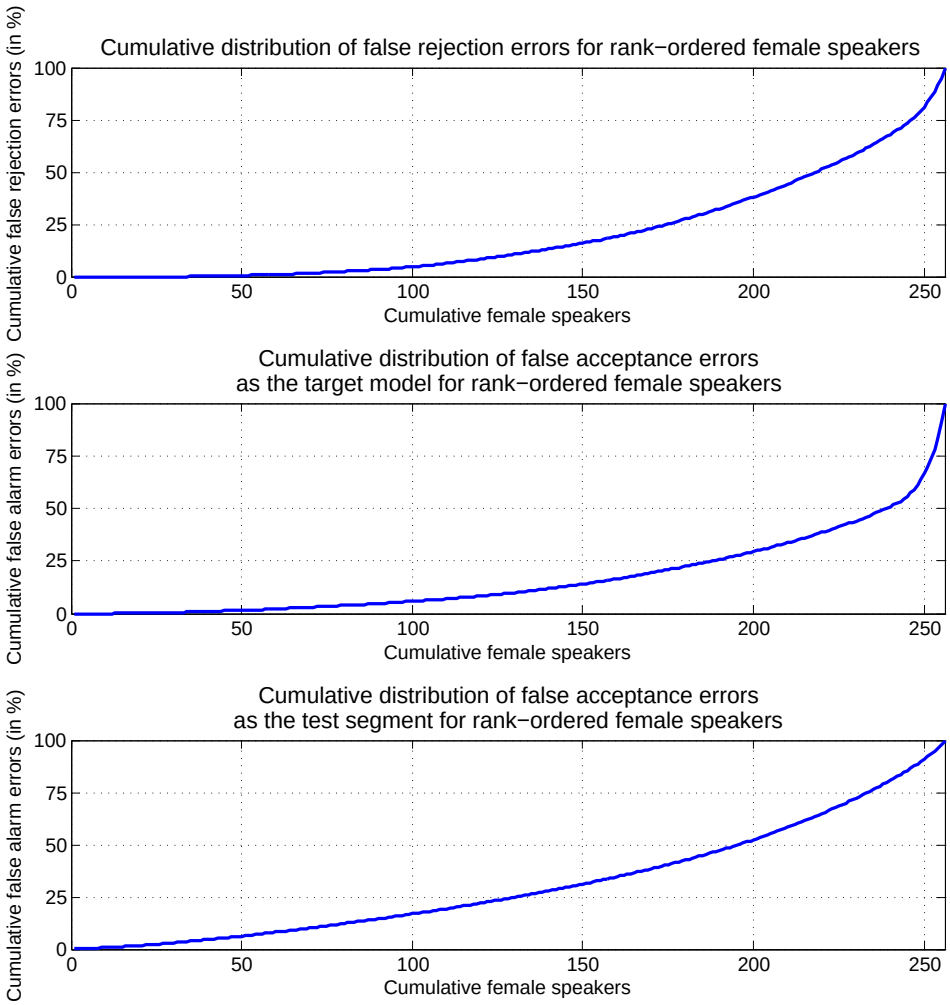

Figure 3.19: *Cumulative distribution of errors across female speakers, for false rejections, false acceptances as the target, and false acceptances as the impostor.*

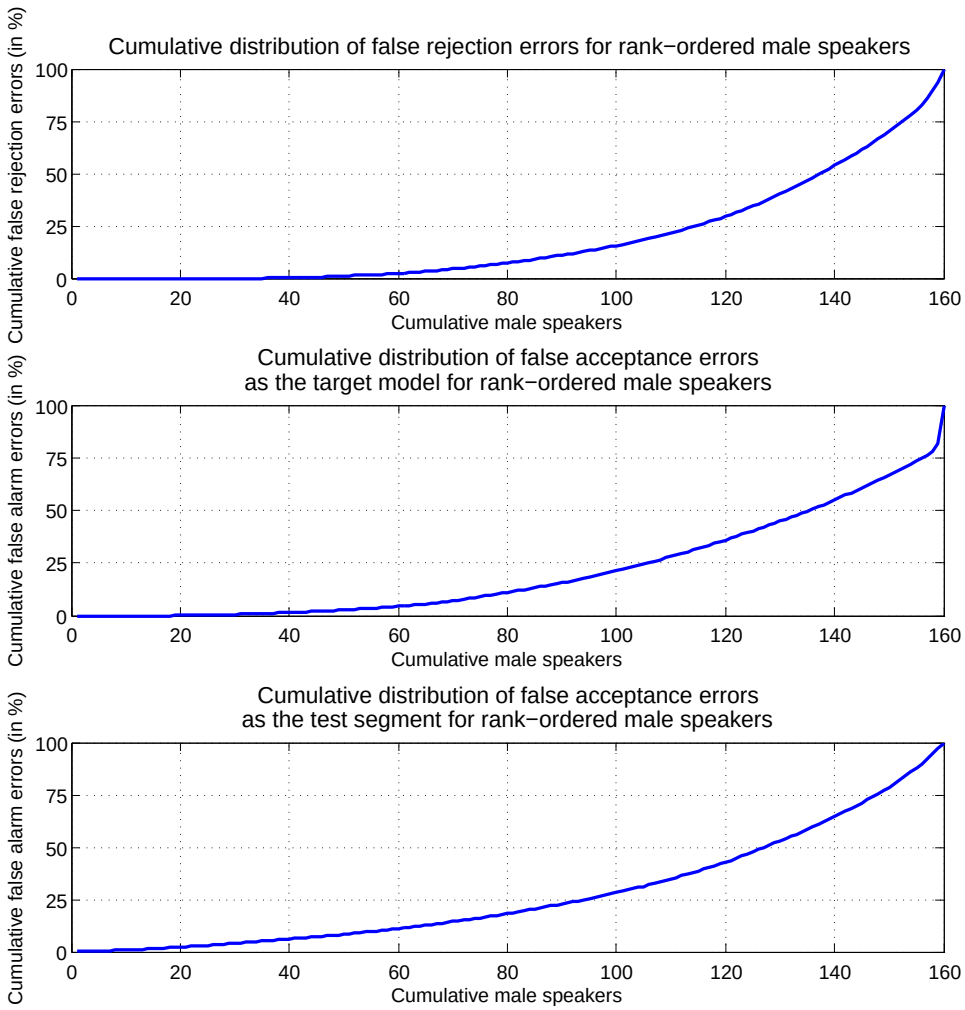

Figure 3.20: *Cumulative distribution of errors across male speakers, for false rejections, false acceptances as the target, and false acceptances as the impostor.*

conversation pairs. At the same time, averaging scores over sets of trials corresponding to a speaker can give a better sense of overall tendencies.

Furthermore, I have observed that there can often be a large degree of variation across speaker pairs; for the same target speaker, impostor scores may change significantly from impostor speaker to impostor speaker. As such, I move away from the separate concepts of lamb and wolf, into a discussion of difficult-to-distinguish impostor speaker pairs, i.e., those pairs for whom the system is likely to produce false alarm errors. At the same time, it is useful to keep in mind that within a given speaker population, there may well be an overall tendency for a particular speaker to cause false alarms, for a number of speaker pairings.

Finally, the preliminary work regarding the effects of speaker demographics suggests that while sex is a factor in the score distributions, the other differences are not particularly informative with respect to system scores. Besides the ANOVA analysis, I observed other differences in behavior between male and female speakers. In general, male speakers appear to vary more widely from one another, in the sense that a given male target speaker will produce different ranges of scores for different male test speakers. On the other hand, female target speakers may often produce similar scores for different female test speakers. Going forward, my work will continue to consider results over the entire population, as well as for males and females separately.

Chapter 4

Predicting Difficult-to-distinguish Speaker Pairs

As I have shown, automatic speaker recognition system performance depends at least in part on intrinsic speaker characteristics, and speakers may have a tendency to produce false alarms or false rejection errors. More specifically than a general per-speaker tendency to produce false alarm errors, there is an expectation that automatic speaker recognition systems will vary across impostor speaker pairs in how successfully those pairs are correctly classified. By comparing the performance for a given speaker pair to performance over all speaker pairs, one can determine which speaker pairs are most (or least) difficult for a given system. Although these difficult-to-distinguish impostor speaker pairs may vary to some degree from system to system, I am most interested in finding the speaker pairs that will be poorly performing for any speaker recognition system. Thus, rather than relying on a particular speaker recognition system's output to select such speaker pairs, I aim to find the universally difficult-to-distinguish speaker pairs by utilizing a variety of features, such as pitch, formant frequencies, or energy.

There are several motivations for trying to predict the difficult-to-distinguish impostor speaker pairs. First of all, if the speaker pairs most likely to cause errors can be identified, such information may be able to open a line of research into determining some of the issues related to intrinsic factors that remain in speaker recognition. Another possible application of this work would be as a tool for NIST to select more difficult trials for future Speaker Recognition Evaluations, in order to present an even more challenging task. Finally, being able to find the speaker pairs that are difficult for an automatic system to distinguish could prove particularly useful in selecting a focus for a human expert in a speaker recognition task that utilizes both automatic system scores as well as human analysis, or as a method for sub-sampling the most salient speech samples in a speaker recognition task where it is impractical to fully process all the data that exists.

This investigation considers a basic set of features, including fundamental frequency statistics, energy statistics, long-term average spectrum (LTAS) energy statistics, formant

frequency statistics, histograms of frequencies obtained from linear predictive (LP) analysis, and spectral slope statistics. These feature choices are motivated by prior work in speaker recognition and other tasks involving characterization of speaker differences. For instance, speaker recognition approaches have used features like pitch and energy distributions or dynamics [1], prosodic statistics including duration and pitch-related features [59], and jitter and shimmer [25]. Formant frequencies and bandwidths, obtained using linear predictive analysis, were used as descriptors for perceptual speaker characterization by Necioğlu et al. [56], while McDougall and Nolan showed that formant frequency dynamics are speaker discriminative [49]. Kuwabara and Sagisaka considered many acoustic parameters as influences upon voice individuality, including pitch frequency, contour and fluctuation, formant frequencies, trajectories and bandwidths, and LTAS [41].

The aforementioned features, along with appropriate distance measures, are utilized as a way to select speaker pairs that are closer, or more similar (in terms of that feature-measure pair). The goal is to find feature-measures for which similar speaker pairs correspond to speaker pairs that are difficult for automatic speaker recognition systems to distinguish. As a more complex measure that may better predict speaker recognition system behavior, I also test the approximated Kullback-Liebler (KL) divergence between speaker-adapted Gaussian mixture models (trained on MFCC features).

I begin by describing my approach in greater detail in Section 4.1. Results are given in Section 4.2, and Section 4.3 provides a summary and discussion of findings.

4.1 Approach

This approach tests a variety of measures calculated from different features as a criterion for selecting similar (or dissimilar) speaker pairs for speaker recognition. I describe the features considered in Section 4.1.1, and the measures and process of speaker pair selection are discussed in Section 4.1.2. The data used is covered in Section 4.1.3.

4.1.1 Features

The features described below are examined as potentially useful for speaker pair selection. Features are calculated either using MATLAB, and the Voicebox toolkit [10], or using Praat [7]. The terms given in brackets indicate the terms we will use to refer to the features. Note that the feature statistics calculated using Praat are computed over the entire input file, including both speech and non-speech regions. The features calculated with MATLAB compute statistics over only those regions of the input designated as speech by the voice activity detection (VAD) provided by NIST.

1. Pitch statistics (Praat): mean, median, range, and mean average slope of the fundamental frequency [f0_mean, f0_med, f0_range, f0_mas]. The range was set to consider

fundamental frequencies between 75Hz and 600Hz, with all other settings corresponding to default Praat parameters.

2. Jitter and shimmer (Praat): jitter relative average perturbation, and shimmer 5-point amplitude perturbation quotient [jitt_rap, shim_apq5]. Jitter describes the variations in pitch. The relative average perturbation (RAP) computes the absolute difference between a pitch period and the average of that period and its two neighbors, then takes the average of this absolute difference and divides it by the average pitch period. Settings for computing the jitter RAP include a minimum fundamental frequency of 75Hz, a maximum fundamental frequency of 600Hz, a minimum period of 0.0001, a maximum period of 0.02, and a maximum period factor of 1.3 (which denotes the largest difference between consecutive intervals that will be included in the jitter computation). Shimmer describes varying loudness (or amplitude) in the voice. The five-point Amplitude Perturbation Quotient (APQ5) calculates the average absolute difference between the amplitude of a period and the average of the amplitudes of it and its four closest neighbours, and then divides this average absolute difference by the average amplitude. Parameter settings for computing the shimmer APQ5 include a minimum fundamental frequency of 75Hz, a maximum fundamental frequency of 600Hz, a minimum period of 0.0001, a maximum period of 0.02, a maximum period factor of 1.3, and a maximum amplitude factor of 1.6 (denoting the largest possible difference in amplitude between consecutive intervals that will be included in the shimmer computation).

3. Formant frequency statistics (Praat): mean and median of the first three formants [f1_mean, f1_med, f2_mean, f2_med, f3_mean, f3_med]. The relevant parameter settings for formant frequency calculation include a window length of 25ms, a step size of 6.25ms, a +3 dB point for an inverted low-pass filter (with a slope of +6 dB/octave) of 50Hz (this is a pre-emphasis filter used to create a flatter spectrum), a maximum number of 4 formants, and a maximum formant frequency of 4000Hz (due to the bandlimited nature of the data used here).

4. Energy statistics (Praat): mean and median energy [en_mean, en_med]. Default Praat settings were used, including a designation to subtract the overall mean energy.

5. Long term average spectrum energy statistics (Praat): mean, standard deviation, range, slope, and local peak height of LTAS energy [ltas_mean, ltas_stddev, ltas_range, ltas_slope, ltas_lph]. Praat parameter settings include a filter bandwidth of 100Hz and a frequency range from 0 to 4000Hz. Furthermore, for local peak height calculation, there is a minimum peak height of 2400 and a maximum peak height of 3200.

6. Histograms of frequencies from roots of the LPC polynomial (MATLAB/Voicebox): frequencies obtained from linear predictive coding (LPC) order 8 or order 14 polynomial coefficient roots (both with and without a minimum magnitude requirement of 0.78

and 0.88, respectively[1]) contribute to a histogram with a bin size of 5 Hz covering the 5-3995 Hz range [hist8all, hist8minmag, hist14all, hist14minmag]. A frame length of 25ms and step size of 10ms were used for calculating the LPC coefficients.

7. Spectral slope statistics (MATLAB): mode and median of spectral slope, calculated over frequency range 0-4000 Hz [mode_specsl, med_specsl]. A frame length of 30ms and step size of 10ms were used to calculate per-frame spectral slope values, from which the mode and median values were computed.

4.1.2 Measures and speaker pair selection

Features are calculated for each speech sample, and a measure is computed for every unique speaker pair in two different ways. First is to average the feature values over all conversation sides of each speaker, and then calculate the measure for each speaker pair using these average per-speaker feature values [featavg]. The second method calculates a measure for each possible pairing of conversation sides for a given speaker pair (with one conversation side for each speaker), and then averages these measure values to obtain a single value for each unique speaker pair [measavg].

For scalar features, absolute difference [absdiff] and percent difference [pctdiff] are used as measures, where percent difference for values x and y is defined as

$$\text{Percent difference} = \frac{|x-y|}{\frac{(x+y)}{2}}, \tag{4.1}$$

when x and y have the same sign (it is not used for features with both positive and negative values). In addition to the individual formants, sums of formants are used as scalar features (with absolute and percent difference measures), and the Euclidean distance [eucldist] is also calculated for vectors of formant frequencies, e.g. (f1,f2,f3). For the histograms of frequencies from LP analysis, a correlation coefficient [corr] is calculated as a measure of similarity. Table 4.1 summarizes the possible feature-measure combinations, grouped according to feature type.

Based on the measure for each unique speaker pair, those pairs with the highest and lowest 1% (or 5%) of values are selected to determine if the measure of speaker similarity corresponds to the degree of difficulty for a speaker recognition system. For absolute difference, percent difference, and Euclidean distance, smaller values should indicate more similar speakers, while for correlation coefficients, higher values indicate greater speaker similarity.

[1]These values were chosen based on a preliminary inspection of histograms, and were not optimized for selecting speaker pairs.

Feature group	Features	Measures
Pitch statistics	f0_mean f0_med f0_range f0_mas	absdiff pctdiff
Jitter and shimmer	jitt_rap shim_apq5	absdiff pctdiff
Formant statistics	f1_mean f1_med f2_mean f2_med f3_mean f3_med	absdiff pctdiff
Sum of formant frequencies	f1+f2+f3_med f1+f2+f3_mean	absdiff pctdiff
Formant frequency vectors	(f1, f2)_mean (f1, f3)_mean (f2, f3)_mean (f1, f2)_med (f1, f3)_med (f2, f3)_med (f1, f2, f3)_mean (f1, f2, f3)_med	eucldist
Energy statistics	en_mean en_med	absdiff pctdiff
LTAS energy statistics	ltas_mean ltas_stddev ltas_range ltas_slope ltas_lph	absdiff pctdiff
LPC frequency histograms	hist14all hist14minmag	corr
Spectral slope statistics	mode_specsl med_specsl	absdiff pctdiff

Table 4.1: *Feature and measure combinations.*

4.1.3 Speech corpora

The 2008 NIST Speaker Recognition Evaluation (SRE08) includes a condition (short2-short3) which uses roughly 2.5-3 minutes of speech for each training and testing [53]. This speech is taken either from one side of a conversation between two people over the telephone (possibly recorded on a microphone), or from part of an interview recorded on a microphone (some interviewer speech may be present). Additional interview data was released for a followup evaluation experiment designed to further explore the new interview style of data collection.

Corpus for feature-measure calculation

Speech data from the followup evaluation is used to calculate features for the speakers. In particular, speech recorded on microphone 2 (a lavalier microphone placed on the subject) is used since it has good sound quality. These speaker features are then used in conjunction with a similarity measure in order to predict difficult- and easy-to-distinguish speaker pairs. The majority of speakers have four conversation sides used for the measure calculation (a small minority have three or five conversation sides).

Corpus for evaluation of selected speaker pairs

The data used to evaluate speaker-pair selection is different in several respects from the data used to perform the selection. Specifically, the selection data were collected in an interview, while the evaluation data were collected in either an interview or a telephone conversation. Also, the selection data were collected using a lavalier microphone, whereas the evaluation data were collected using a variety of microphones, including a telephone handset. Furthermore, though the speakers contained in each set are the same, the selection data does not overlap with evaluation data.

Speaker recognition system submissions from the SRE08 short2-short3 condition are used to compute performance on trials for the selected 1% (or 5%) of most and least similar speaker pairs. Of the 34 sites who shared their system submissions for the short2-short3 condition, 33 of these are used in the results. The total number of trials for short2-short3 (after removing trials for speakers not found in the selection data) is 55013, with 1815 unique impostor speaker pairs. When keeping 1% (or 19) of the speaker pairs, there are around 4000 trials on average, while 5% (or 91) of the speaker pairs corresponds to an average of roughly 11000 trials. When filtering trials for selected speaker pairs, I removed target trials of speakers not included in any of the selected speaker pairs.

4.2 Results

System performance for the selected speaker pairs is reported using the minimum detection cost function (DCF) and false alarm (FA) rate, since I am concerned with finding difficult-to-distinguish impostor pairs. The DCF is defined as a weighted sum of the miss (i.e., not identifying a target speaker match) and false alarm (i.e., identifying an impostor speaker as the target speaker) error probabilities:

$$\mathrm{DCF} = \mathrm{C}_{\mathrm{Miss}} \times \mathrm{P}_{\mathrm{Miss|Target}} \times \mathrm{P}_{\mathrm{Target}} + \mathrm{C}_{\mathrm{FalseAlarm}} \times \mathrm{P}_{\mathrm{FalseAlarm|NonTarget}} \times (1 - \mathrm{P}_{\mathrm{Target}}) \quad (4.2)$$

In Equation (4.2), $\mathrm{C}_{\mathrm{Miss}}$ and $\mathrm{C}_{\mathrm{FalseAlarm}}$ are the relative costs of detection errors, and $\mathrm{P}_{\mathrm{Target}}$ is the *a priori* probability of the specified target speaker. SRE08 used $\mathrm{C}_{\mathrm{Miss}} = 10$, $\mathrm{C}_{\mathrm{FalseAlarm}} = 1$, and $\mathrm{P}_{\mathrm{Target}} = 0.01$. For a given decision threshold, the FA rate is defined as:

$$\mathrm{P}_{\mathrm{FalseAlarm}} = \frac{\text{number of false alarm errors}}{\text{total number of nontarget trials}} \quad (4.3)$$

For each speaker recognition system, I compute the percent difference in minimum DCF for the most (and least) similar speaker pairs relative to the minimum DCF over all speaker pairs. Relative to a FA rate of 1% on all speaker pairs, I calculate the percent difference in FA rate (at the decision threshold yielding 1% FA on all trials) for the most (and least) similar pairs. These relative differences are then averaged over all systems. With each feature-measure, if more similar (i.e., closer) speaker pairs correspond to difficult-to-distinguish speaker pairs, then differences in the DCF and FA rate should be positive and significant. The converse holds for less similar speaker pairs, which will have significant negative differences if they are easier for systems to distinguish.

Figures 4.1 and 4.2 show performance differences for the top 1% most and least similar speaker pairs, respectively. For each feature group, the feature-measure pair yielding the largest DCF and FA changes is presented. Similarly, Figures 4.3 and 4.4 show results when considering the top 5% most and least similar speaker pairs, respectively.

Features of each type can select speaker pairs for which the most (or least) similar have worse (or better) performance than all speaker pairs. Furthermore, this difference in performance typically increases when a smaller fraction of speaker pairs is used, i.e., there is a bigger difference for the most similar 1% of speaker pairs than for the most similar 5%. It should be noted that differences in performance are not uniform across different speaker verification systems.

The feature-measure that yields the largest average difference in performance for the 1% most similar speaker pairs is the Euclidean distance between vectors of the mean first, second, and third formant frequencies. The next best feature-measures include other formant-based measures, the percent difference of median energy, and the correlation of histograms of LPC freqencies with minimum magnitude requirement. For the 1% least similar speaker pairs, results are fairly similar across feature-measures, with the correlation of LPC frequency

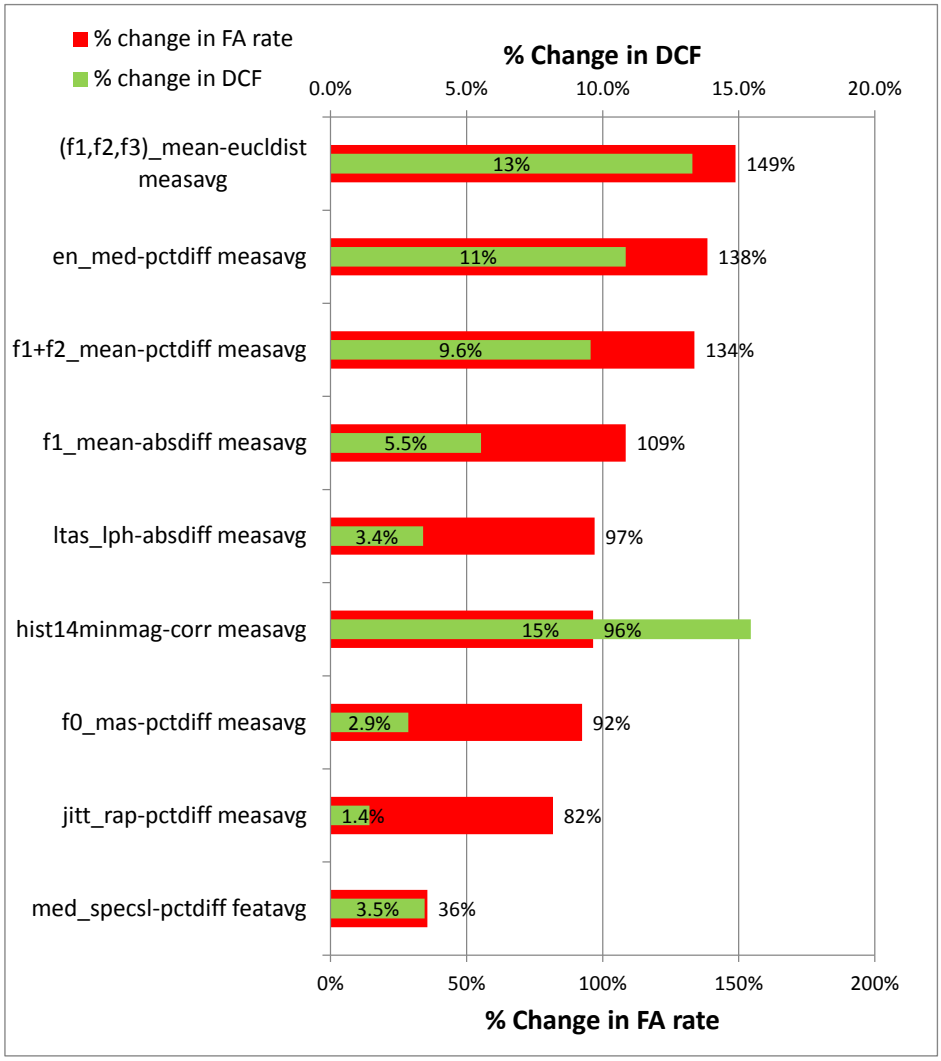

Figure 4.1: *Relative differences in DCF and FA rate for the most similar 1% of speaker pairs, compared to all speaker pairs.*

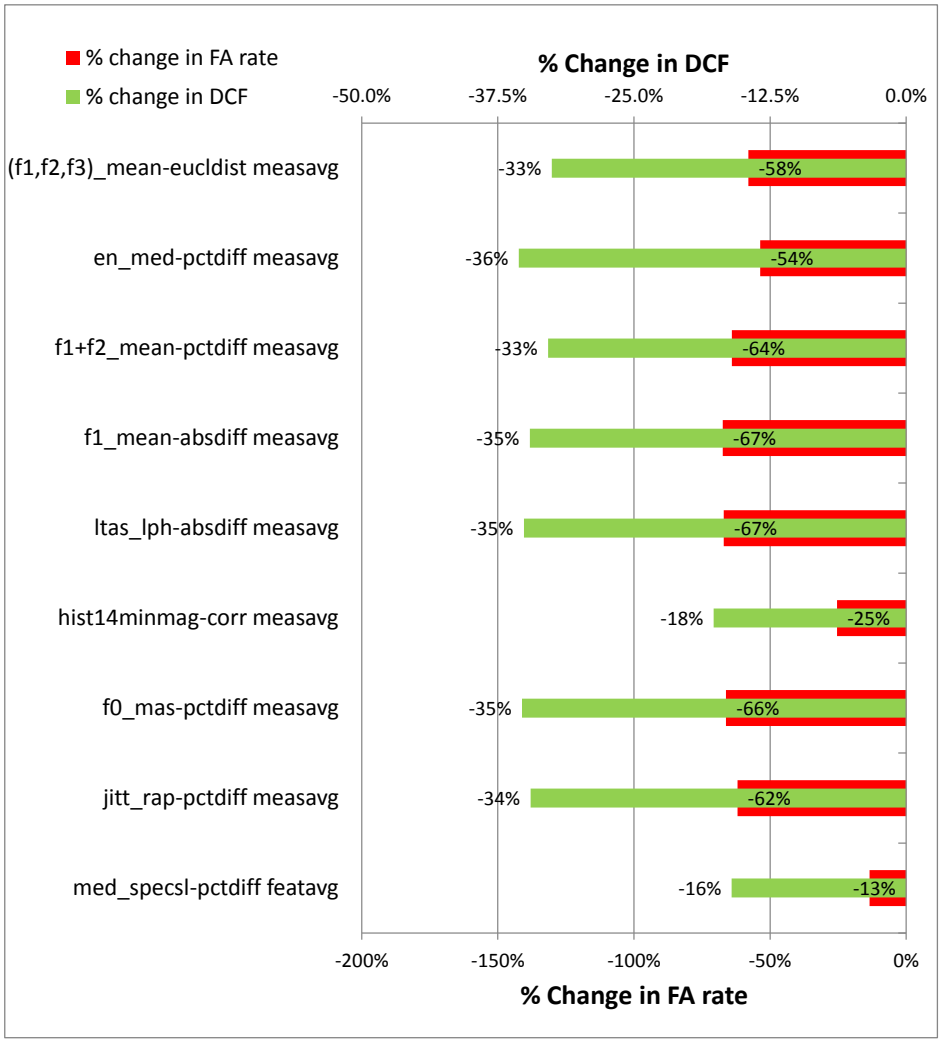

Figure 4.2: *Relative differences in DCF and FA rate for the least similar 1% of speaker pairs, compared to all speaker pairs.*

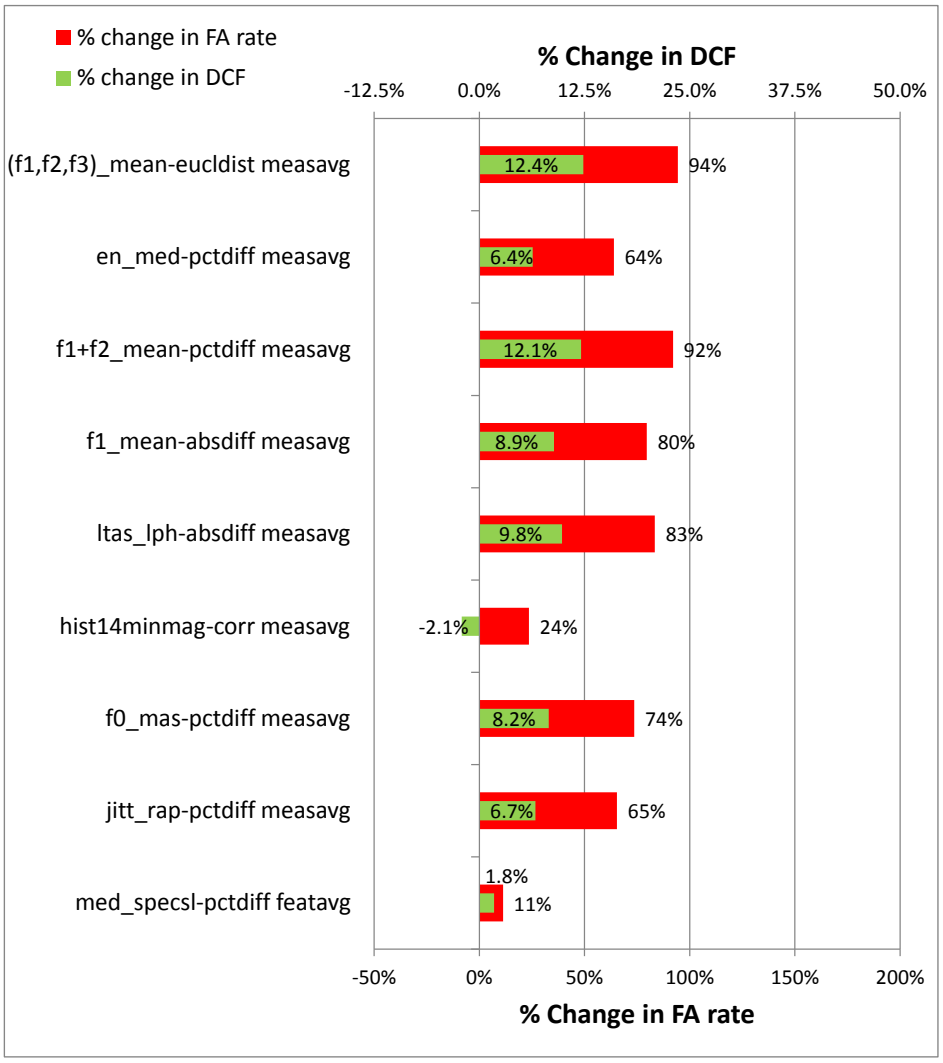

Figure 4.3: *Relative differences in DCF and FA rate for the most similar 5% of speaker pairs, compared to all speaker pairs.*

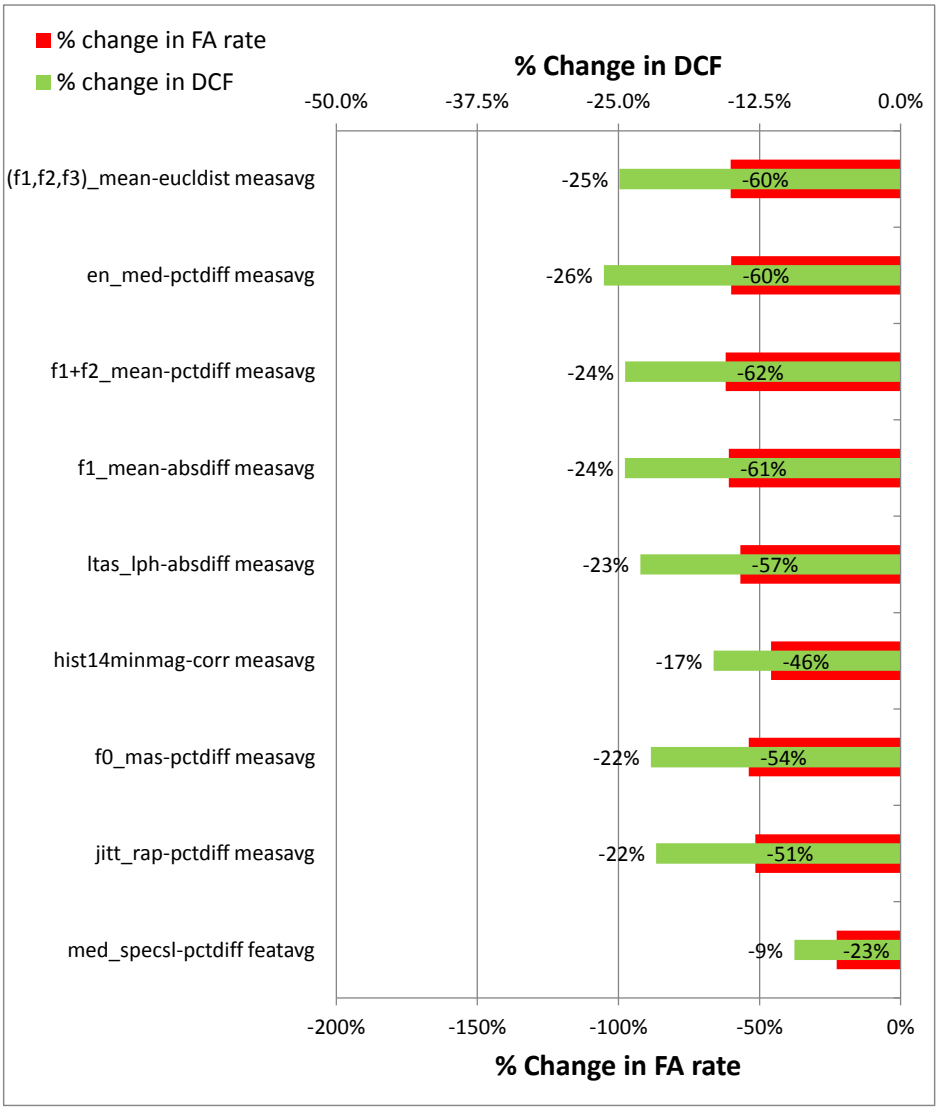

Figure 4.4: *Relative differences in DCF and FA rate for the least similar 5% of speaker pairs, compared to all speaker pairs.*

histograms and spectral slope yielding the smallest differences. The Euclidean distance between vectors of the mean, first, second, and third formant frequencies also appears to be the best feature-measures for finding the 5% most difficult-to-distinguish speaker pairs, with the percent difference of the sum of formants and the absolute difference in LTAS local peak height being the next best. As with the 1% least similar speaker pairs, the 5% least similar show very consistent results across feature-measures, with reduced effectiveness for the correlation of LPC frequency histograms and spectral slope.

Detection error tradeoff (DET) curves are shown for example systems in Figures 4.5 and 4.6, using the Euclidean distance between vectors of the means of the first, second and third formants, and the percent difference of the median energy, respectively. Although the system in Figure 4.6 has good separation among the different DET curves, there is more overlap in the DET curves of Figure 4.5. Furthermore, Figure 4.5 reveals an asymmetry in behavior for dissimilar and similar speaker pairs, showing that the performance on difficult-to-distinguish speaker pairs is closer to performance on all speaker pairs. While this asymmetry does not exist for all systems and all sets of selected speaker pairs (as evidenced by Figure 4.6), the trend does hold in most cases.

Given that I am using at most a few coarsely calculated features, it is impressive to see the differences in performance that can be obtained using these measures to select easy- or difficult-to-distinguish speaker pairs. It is worth noting that a large reason for such success is due to the information gained by the relative ranking of speaker pairs. As a single, standalone number, a feature-measure may not have much use. However, when taken in the context of a group of feature-measures corresponding to a set of speaker pairs, the absolute values of the feature-measures no longer matter; instead, the gain lies in being able to order a set of speaker pairs from least to most similar.

While the results presented thus far are indeed promising, the differences in performance for similar speaker pairs (relative to all speaker pairs) still have potential to increase further. Accordingly, I test a measure that utilizes Gaussian mixture models, with the motivation that GMMs may better predict speaker recognition system performance, given that many systems utilize cepstral feature-trained GMMs. Using SRI's tools for training GMMs for speaker recognition [37], I trained speaker-specific GMMs via maximum *a posteriori* (MAP) adaptation from a universal background model trained on Fisher data. The input features were 12th order MFCCs plus energy, with deltas and double-deltas, and the models used 1024 Gaussians. For each unique pair of speaker-specific GMMs, an approximation to the Kullback-Leibler (KL) divergence (based on the unscented transform [26]) was used to measure similarity. Results are shown in Figure 4.7.

Compared to previous feature-measures, the KL divergence is indeed more effective at finding difficult- and easy-to-distinguish speaker pairs. DET curves for an example system are shown in Figure 4.8. Again, relative to performance on all speaker pairs, there is a larger performance gap for dissimilar speaker pairs than for similar speaker pairs.

Returning to the groups of speaker pairs selected by the KL divergence approximation for GMMs, I more closely examine the 1%, 3%, 5%, 10%, and 20% most and least similar speaker

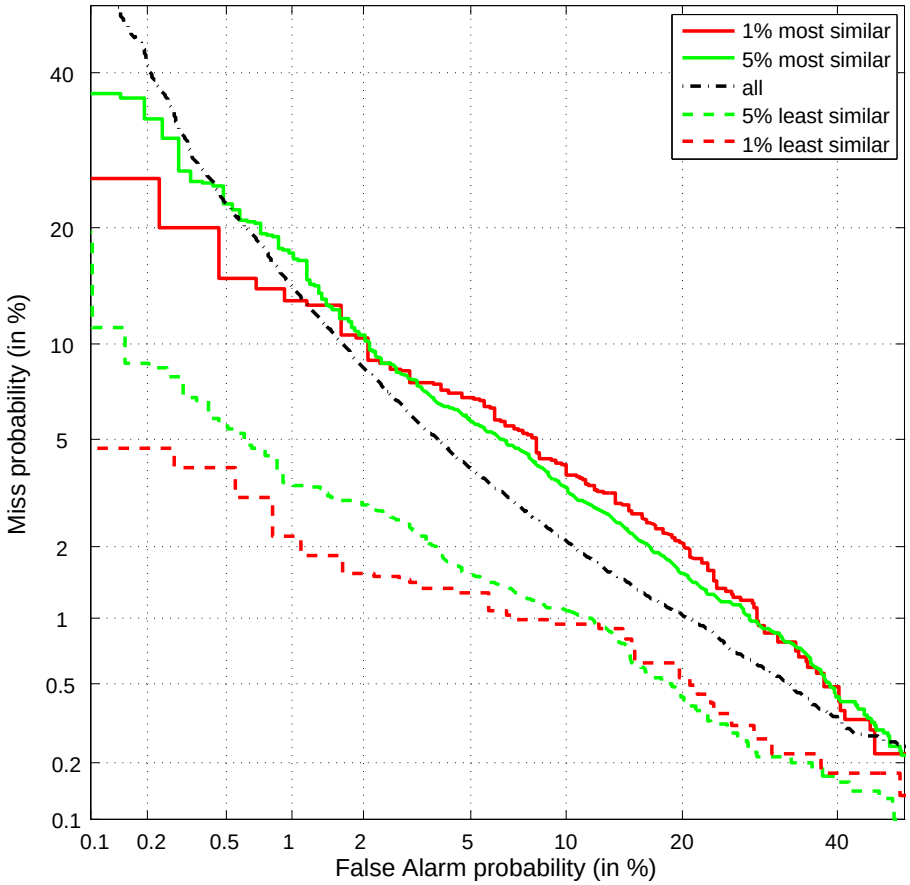

Figure 4.5: *DET curves for an illustrative speaker recognition system, using the Euclidean distance between vectors of the mean first, second, and third formant frequencies for speaker pair selection.*

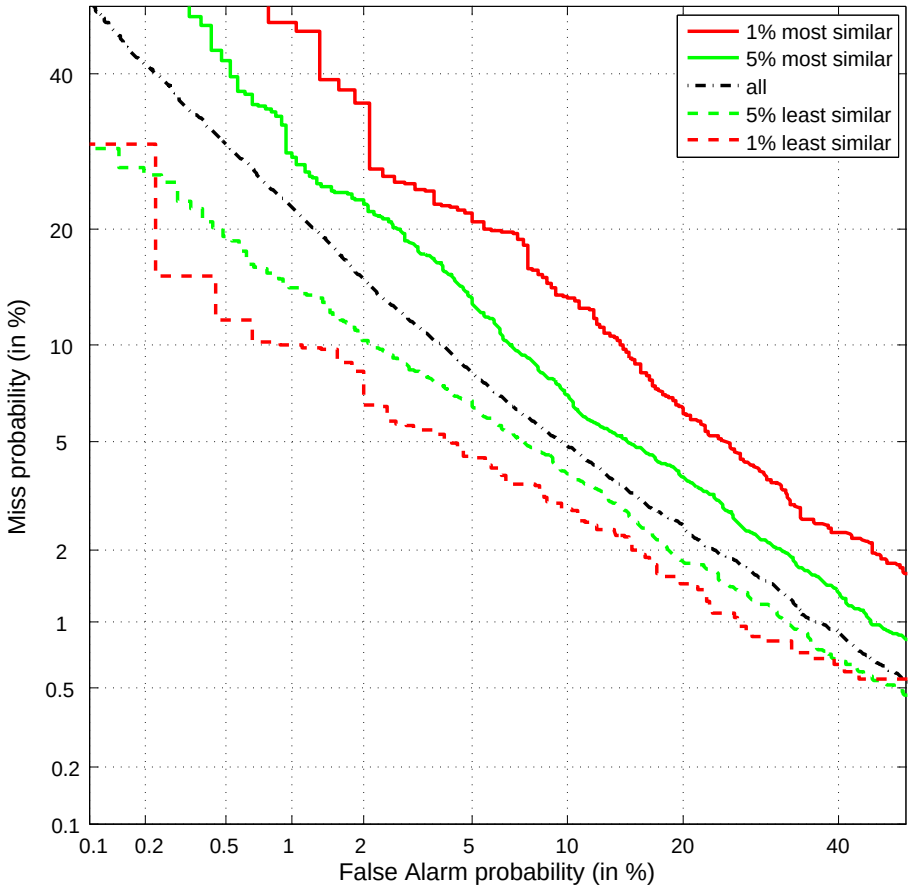

Figure 4.6: *DET curves for an illustrative speaker recognition system, using the percent difference of median energy for speaker pair selection.*

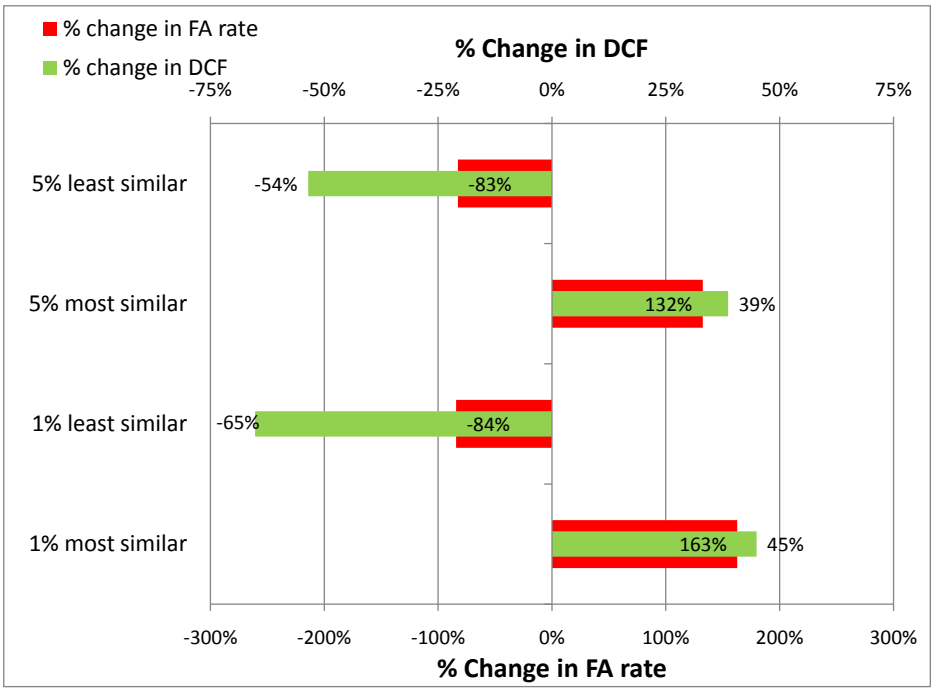

Figure 4.7: *Relative differences in DCF and FA rate for the most and least similar 1% and 5% of speaker pairs selected by the approximated KL divergence between speaker-specific GMMs.*

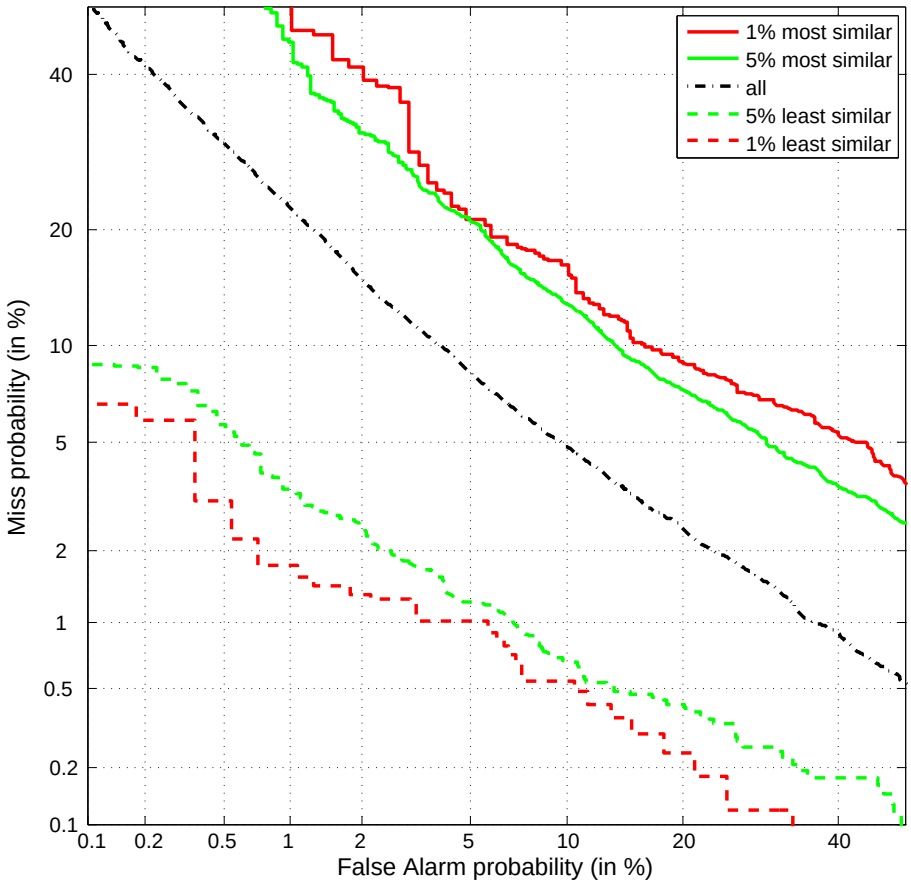

Figure 4.8: *DET curves for an illustrative speaker recognition system, using the approximated KL divergence between speaker-specific GMMs to select speaker pairs.*

pairs. Overall, there are 150 speakers, with 87 female and 63 male, for which there are 1815 same-sex impostor speaker pairs with impostor trials in the SRE08 short2-short3 task. For the groups of speaker pairs with larger values for KL divergence, that is, those speaker pairs that are expected to be easier for systems to distinguish, the majority are male (close to 75% on average). The opposite tendency holds to a lesser extent for more similar pairs tending to be female, although the groups with the lowest 1% and 3% of KL divergence values still have more male speaker pairs. These results suggest that there is a greater range of differences among male speakers, so that there are likely to be more dissimilar male speaker pairs.

Furthermore, examining the number of times a particular speaker appears in a group of similar or dissimilar speaker pairs, we note that there tend to be two types of speakers: those who appear frequently as members of difficult-to-distinguish speaker pairs, and those who occur frequently as members of easy-to-distinguish speaker pairs. In fact, there are 15 speakers (1 male, 14 female) that never appear in the most-dissimilar groups, and 24 speakers (10 male, 14 female) that never appear in the most-similar groups. Such a result is consistent with the existence of wolves and lambs, that is, the tendencies of a speaker to cause false alarm errors.

4.3 Discussion

In summary, the results of this investigation demonstrate that it is possible to predict which speaker pairs will be difficult for a typical speaker recognition system to distinguish. Both difficult- and easy-to-distinguish speaker pairs can be selected using a measure of similarity calculated from features like pitch, energy, or spectral slope. For the features considered here, using the Euclidean distance between vectors of mean first, second, and third formant frequencies produces the largest difference in performance for similar and dissimilar speaker pairs. An even more successful measure is the KL divergence calculated between speaker-specific GMMs. Overall, the degree of success is higher for selecting dissimilar speaker pairs than it is for selecting similar speaker pairs, possibly because similarity in a single characteristic is not necessarily sufficient to identify a difficult-to-distinguish speaker pair. Although the feature-measures cannot match the effectiveness of finding difficult-to-distinguish speaker pairs by actually selecting such pairs using results for a given system, they still provide potentially useful information about speakers. In particular, one may be able to determine an overall tendency of a speaker to be similar or dissimilar to other speakers. Additionally, being able to rank a set of speaker pairs can be quite informative.

In the next chapter, I build upon this approach by using a set of feature statistics in order to detect difficult speakers. I consider the task of finding difficult target speakers, who are prone to causing false rejection errors, separately from the task of finding difficult impostor speakers, who are prone to causing false alarms. Specifically, I train support vector machine (SVM) classifiers using examples of the most and least difficult target and impostor speakers.

Chapter 5

Detecting Difficult Speakers

It has been observed that simple feature statistics can be used to provide measures of similarity between speakers. Up to this point, I have used these feature statistics individually. Now, I investigate one method for using them jointly in order to make a prediction about whether a speaker will be difficult, either as a true speaker or an impostor speaker. In particular, I train a support vector machine (SVM) to distinguish between examples of the speakers who cause the most and fewest errors, corresponding to the most and least difficult speakers, respectively. Since speaker behavior is different for target and impostor speakers, I train separate SVMs for detecting difficult true speakers (who will cause false rejections) and difficult impostor speakers (who will cause false alarms).

I begin by discussing the data set that will be used for these experiments in Section 5.1. Section 5.2 describes the selection of feature statistics used as input to the SVMs. Details of SVM training are covered in Section 5.3, including the method for determining the difficult and easy speakers to use for training. The results of experiments are given in Section 5.4, and Section 5.5 concludes with a discussion of lessons learned.

5.1 Data Set for SVM Experiments

For this approach, I need to find speakers who cause very many or very few errors (of either the false rejection or false alarm type). Accordingly, these speakers need to have enough true speaker and impostor trials available for us to make a reliable decision about these error tendencies. This is especially an issue for the true speaker errors, given the limited number of target trials that are available.

In order to maximize the number of true speaker trials, as well as have a reasonable number of impostor trials, I use the same set of SRE08 data that I used for the analysis of 3.2. In particular, I take selected conversation sides from the SRE08 short2 and short3 train and test conditions, which correspond to roughly 2.5-3 minutes of speech per sample. I choose conversation sides from all speakers with at least 5 available speech utterances. Some

conversation sides were recorded on multiple channels (telephone and microphones, or just microphones). In these cases, I select only one instance of that conversation side, in order to prevent the introduction of confounding factors due to having the same lexical content across different speech samples. There are 416 speakers (256 female, 160 male), with 3049 conversation sides, and a total of 22,210 target trials. For each impostor speaker pair, five impostor trials are chosen (along with the corresponding trials that have the train and test data switched), for a total of 453,600 impostor trials.

Although the training and test sets are disjoint, they are selected from the same database of conversation sides of SRE08. In practice, it is not unreasonable to make an assumption that there will be a set of domain-specific data available for training that is representative of the data used in a given type of speaker recognition application. Due to data sparsity, I take a round robin approach (specifically, 10-fold cross-validation) in order to best utilize the available data.

5.2 Selection of Feature Statistics

The feature statistics under consideration include statistics of energy, spectral slope, fundamental frequency, formant frequency, and MFCC features, where the statistics can be calculated over frames corresponding to various regions, including phones, groups of phones, and all speech. In the previous work on finding difficult-to-distinguish impostor speaker pairs, I had success using feature statistics calculated over the whole utterance or all speech regions. I take the same approach here by choosing to calculate the feature statistics over all frames of speech. One additional motivation for such a choice is that it is generally more convenient to simply calculate statistics using speech frames rather than frames of particular phonetic regions, given that it is less computationally expensive to implement a speech/non-speech detector than it is to obtain phonetic transcripts from an automatic speech or phone recognition system.

The complete set of features is as follows.

1. Energy [en], calculated in MATLAB, using 25ms frames with a 10ms stepsize

2. Spectral slope [spsl], calculated in MATLAB, using 30ms frames with a 10ms stepsize

3. Fundamental frequency [f0], calculated with the Snack sound toolkit [66], using the ESPS method, which relies on the normalized cross correlation function and dynamic programming, with a default window length of 7.5ms and a stepsize of 10ms, default minimum pitch of 60Hz and default maximum pitch of 400Hz

4. First three formant frequencies, [f1,f2,f3], calculated with the Snack sound toolkit, which estimates speech formant trajectories using dynamic programming for continuity constraints and the roots of a 12th order linear predictor polynomial as candidates; a

default window length of 49ms, a stepsize of 10ms, default $\cos^4$ windowing function, default preemphasis of 0.7, and a nominal first formant frequency of 500Hz, specifying the number of formants to be 3

5. First four formant frequencies, [g1,g2,g3,g4], calculated with the same settings as [f1-f3], except for the specification that the number of formants is 4 (note that looking for 3 formants produces different outputs than looking for 4 formants)

6. 19th order MFCCs plus energy [C0-C19], calculated using the Hidden Markov model Toolkit (HTK) [72], using 26 filter banks ranging from 200Hz to 3300Hz, frame length of 25ms, stepsize of 10ms, no normalizations

7. Mean- and variance-normalized 19th order MFCCs plus energy [N0-C19], calculated with HTK with the same settings as [C0-C19]

The set of statistics computed for each feature over speech regions are mean, median, standard deviation, skewness, kurtosis, minimum, and maximum.

I include each type of feature and statistic in order to obtain feature statistics that may be informative in differing ways. However, since the two sets of formant frequencies (calculated by finding the first three [f1-f3] or the first four [g1-g4]) are related, as are the normalized and non-normalized MFCCs ([N0-N19] and [C0-C19]), I consider three groups of features, with differing degrees of similarity among the features:

1. energy [en], spectral slope [spsl], fundamental frequency without zeros [f0no0], fundamental frequency including zeros [f0with0], the set of the first three formant frequencies [f1-f3], and non-normalized MFCCs [C0-C19], for a total of 187 statistics [speech1]

2. same as (1), with addition of normalized MFCCs [N0-N19], for a total of 327 statistics [speech2]

3. same as (2), with addition of the first four formant frequencies [g1-g4], for a total of 355 statistics [speech3]

5.3 SVM Training

In order to train an SVM classifier to detect difficult speakers, there must be training data that corresponds to such difficult speakers, as well as to non-difficult speakers who will provide negative examples. To determine these speakers, I utilize the scores from an automatic speaker recognition system. Given a particular decision threshold, I can then evaluate how many false rejection and false acceptance errors occur among the trials of a given speaker, and rank the speakers according to these error rates. For each speaker, false acceptance errors as the target are counted along with false acceptance errors as the impostor (in other words, I do not distinguish between lamb-ish and wolf-ish speaker tendencies).

Roughly the top and bottom 20% of speakers (ranked according to their error rates) are used for training and testing. In particular, I take 80 speakers from each end of the difficulty spectrum. Those speakers with the lowest frequency of errors provide negative training examples, while the speakers with most frequently occurring errors provide positive examples. For this speaker selection, I utilize scores from a UBM-GMM system with simplified factor analysis applied. Details of the implementation may be found in Section 3.2. In order to count errors, I use the decision threshold corresponding to an overall false alarm rate of 1%.

As mentioned previously, there is limited data available in SRE08; to deal with this data sparsity, I utilize a round robin (or 10-fold cross-validation) approach, with 10 splits of the data. Given 10 disjoint sets of 4 difficult and 4 easy speakers, I use 9 of the sets to train the SVM, and the remaining 1 to test, with each set being the test set exactly once. The results are then calculated across the ten test sets. To further ensure that these results are representative, I run the experiment 10 times, with random selection of the 10 splits each time.

Each speaker has 5 or more conversation sides that are used as separate examples. I consider two separate SVMs: one to detect difficult true speakers, i.e., those that are prone to causing false rejection errors, and one to detect difficult impostor speakers, i.e., those that are prone to causing false alarms.

In addition to considering a linear kernel for the SVM, I also test polynomial kernels of orders 2 and 3, in the event that a nonlinear mapping may prove useful for the detection task at hand. Furthermore, I use the input feature statistics both as they are as well as with a rank normalization applied. Rank normalization, wherein the features are assigned a relative ranking from minimum to maximum, is a technique that often yields nice improvements in the context of speaker recognition systems with SVM classifiers. The rank normalization mapping is learned from the examples used to train the SVM, and then applied to both the train and the test data. The SVMs are implemented using the SVM light toolkit [32].

5.4 SVM Testing

Given that I want to use the SVM to find difficult speakers, I will present results for a detection task. As with a speaker verification task, the decisions in a detection task may be broken up into four groups. In this case, either a difficult speaker will be detected or not. This positive detection result may be either correct or incorrect, or in other words, may be either a true positive or a false positive. In the event that a negative result is produced, this may be a true negative or a false negative.

I will report the precision, recall, specificity, and F-measure. The precision is a measure of how accurately the positive class is detected. It is given by the ratio of true positives to the total number of positive decisions, that is,

$$\text{Precision} = \frac{\text{tp}}{\text{tp} + \text{fp}},$$

where tp denotes the number of true positives and fp denotes the number of false positives. Recall is a measure of how many of the positive class instances were detected, also known as the true positive rate, designated by

$$\text{Recall} = \frac{\text{tp}}{\text{tp} + \text{fn}},$$

where tp again denotes the number of true positives, and fn denotes the number of false negatives (missed detections). Similarly, the true negative rate is given by the specificity, which is

$$\text{Specificity} = \frac{\text{tn}}{\text{tn} + \text{fp}},$$

where tn is the count of true negatives and fp is the count of false positives. Finally, the F-measure combines precision and recall, and is given by

$$\text{F-measure} = \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}.$$

First, I will show results for detecting a difficult impostor speaker in Section 5.4.1, followed by results of detecting a difficult target speaker in Section 5.4.2.

5.4.1 Detecting Difficult Impostor Speakers

The recall, precision, specificity, and F-measure are given in Table 5.1 for three versions of the SVM classifier using the [speech1] set of feature statistics, which may or may not be rank normalized [rank,nonorm]. In these results, the SVM is detecting difficult impostor speakers, who are likely to cause false acceptance errors, either as the target model or the test speaker. The three SVMs differ in the kernel that they use, which may be a linear kernel [linear], a second order polynomial kernel [poly2], or a third order polynomial kernel [poly3]. These results are averages across the 10 runs of a 90% train - 10% test round-robin approach.

These results show that it is, in fact, possible to detect difficult impostor speakers, most of the time, with recall and precision rates of around 0.84 and 0.86, respectively. Furthermore, note that rank normalization yields much more balanced results. Additionally, the linear and polynomial kernels perform about the same in terms of recall, with polynomial kernels giving a slight gain in precision, specificity, and F-measure.

Depending on the application, the recall or the precision may be more important. For those situations where it is important to find all of the difficult speakers, at the cost of including some easy speakers, the threshold for making the difficult distinction can be lowered, thereby increasing the recall. On the other hand, it may be important to be very accurate about any difficult speaker labels, at the cost of missing some difficult speakers. In this scenario, the threshold for making the decision can be raised, and the precision increased.

For the linear kernel SVM using rank normalized feature statistics, results are given in Table 5.2 for three different thresholds: -0.5, 0 (corresponding to the values in Table 5.1), and 0.5.

The choice of threshold for the detection of a difficult speaker allows one to adjust according to the most important criterion. Even when improving results for one particular measure, the results stay fairly good across all performance measures, though the specificity (or true negative rate) does drop to 0.616 when the recall is increased. Since many applications might find it very important to be correct about the speakers that are labeled as difficult, I will examine the low false alarm case (this corresponds to a high specificity and precision). In particular, consider a false alarm rate of 5%, meaning that 5% of the difficult labels will be incorrect. For the corresponding threshold (which is around 0.83), average recall is 0.612, and average precision is 0.959, for the SVM with a linear kernel and rank normalization of the input features. In other words, in order to be 95% correct about difficult speaker decisions, over 60% of the difficult speakers are found. Though this is not a very high recall rate, it may still be sufficient for some applications, and it provides a reasonable starting point on a first try at this task.

Now, let us compare performance for the linear kernel SVM, when using more feature statistics, in particular, the [speech2] and [speech3] sets, which add normalized MFCCs and a different set of four formant frequencies. Results for the three feature sets are given in Table 5.3.

In this case, the additional speech-based feature statistics do not add much information for distinguishing between easy and difficult impostor speakers.

5.4.2 Detecting Difficult Target Speakers

Now, I present results for an SVM classifier trained to detect difficult target speakers, who tend to cause false rejection errors. Table 5.4 shows the recall, precision, specificity, and F-measure for SVMs trained using the set of [speech1] input feature statistics, both with and without rank normalization, for three SVM kernels, namely linear, order two polynomial, and order three polynomial. In each case, the results presented correspond to an average over ten runs of a round robin approach using a 90% - 10% split of the data.

In this case, there are fairly reasonable results, though detection of difficult target speakers is not as successful the detection of difficult impostors. The intuition behind why difficult target speakers are not detected as successfully as difficult impostor speakers is as follows. To cause false alarm errors, impostor speakers must be confusable with other speakers; so, there may be overall characteristics that make a speaker more average or more similar to other speakers within the population. On the other hand, the characteristics that make a target speaker hard to recognize as himself may vary from speaker to speaker, so that it is harder to capture all the ways in which a single conversation side may indicate a tendency to cause false rejections.

Returning to the results of Table 5.4, observe that rank normalization once again really helps to improve performance overall. In the case of difficult target speakers, there are small

SVM kernel	Normalization	Recall	Precision	Specificity	F-measure
linear	nonorm	0.997	0.585	0.010	0.738
linear	rank	0.838	0.851	0.794	0.844
poly2	rank	0.840	0.861	0.811	0.850
poly3	rank	0.839	0.866	0.818	0.852

Table 5.1: *Recall, precision, specificity, and F-measure values for detecting difficult impostor speakers using SVMs with different kernels (linear, second order polynomial [poly2], and third order polynomial [poly3]), with the [speech1] set of feature statistics as input, with or without rank normalization applied [rank,nonorm].*

Threshold	Recall	Precision	Specificity	F-measure
-0.5	0.915	0.770	0.616	0.836
0	0.838	0.851	0.794	0.844
0.5	0.726	0.914	0.904	0.809

Table 5.2: *Recall, precision, specificity, and F-measure values for detecting difficult impostor speakers using a linear kernel SVM trained with rank normalized feature statistics, comparing three different decision thresholds for difficult impostor speaker detection.*

Feature set	Recall	Precision	Specificity	F-measure
speech1	0.838	0.851	0.794	0.844
speech2	0.831	0.852	0.798	0.841
speech3	0.830	0.859	0.809	0.844

Table 5.3: *Recall, precision, specificity, and F-measure values for detecting difficult impostor speakers using a linear kernel SVM trained with rank normalized feature statistics, comparing three sets of speech feature statistics, [speech1], [speech2], and [speech3].*

SVM kernel	Normalization	Recall	Precision	Specificity	F-measure
linear	nonorm	1	0.551	0	0.710
linear	rank	0.729	0.715	0.643	0.722
poly2	rank	0.733	0.726	0.661	0.729
poly3	rank	0.736	0.737	0.677	0.737

Table 5.4: *Recall, precision, specificity, and F-measure values for detecting difficult target speakers using SVMs with different kernels (linear, second order polynomial, and third order polynomial), with the [speech1] set of feature statistics as input, with or without rank normalization applied.*

gains from using polynomial kernels instead of linear, with a third order polynomial kernel improving results more than a second order polynomial.

Table 5.5 shows the tradeoff in recall, precision, specificity, and F-measure values observed when varying the threshold for making a difficult speaker decision, again considering threshold values of -0.5, 0 (corresponding to the results of Table 5.4), and 0.5. These results are given for the SVM using a third order polynomial kernel, with rank normalized feature statistics.

Threshold	Recall	Precision	Specificity	F-measure
-0.5	0.895	0.653	0.415	0.755
0	0.736	0.737	0.677	0.737
0.5	0.561	0.848	0.877	0.675

Table 5.5: *Recall, precision, specificity, and F-measure values for detecting difficult target speakers using a third order polynomial kernel SVM trained with rank normalized feature statistics, comparing three different decision thresholds for difficult target speaker detection.*

In the difficult target speaker case, the drop in specificity (for a high recall threshold) and the drop in recall (for a high precision threshold) are larger than in the difficult impostor speaker case. In order to obtain close to 90% recall, the false alarm rate becomes almost 60%. Again considering the operating point for low false alarms, with 5% of the difficult speaker labels being incorrect (a threshold around 0.95), the average recall is 0.374, and the average precision is 0.922. Thus, in order to avoid incorrectly labeling difficult target speakers, almost two-thirds of the difficult target speakers will not be found. Such a low recall rate may not be sufficient in many applications. Given the difficult nature of the task, it nevertheless provides an initial starting point that may be improved upon in the future.

Next, Table 5.6 shows results for an SVM using a third order polynomial kernel and rank normalized input features, for the three sets of speech feature statistics.

Feature set	Recall	Precision	Specificity	F-measure
speech1	0.736	0.737	0.677	0.737
speech2	0.747	0.749	0.694	0.748
speech3	0.753	0.746	0.686	0.749

Table 5.6: *Recall, precision, specificity, and F-measure values for detecting difficult target speakers using a third order polynomial kernel SVM trained with rank normalized feature statistics, comparing three sets of speech feature statistics, [speech1], [speech2], and [speech3].*

As with the difficult impostor speaker detection task, adding feature statistics (mean and variance normalized MFCCs or formant frequencies g1-g4) does not change results by much, though there are some small improvements.

To this point, my approach has treated male and female speakers together. However, male and female speakers may behave differently. In order to see if difficult target speaker detection improves when females and males are two different cases, female- and male-specific SVMs are trained. One disadvantage to this approach is that there are fewer easy and difficult speakers to use for training. I consider two sets of MLPs, one trained using the 20% most and least difficult speakers (female or male), and one trained using 25% most and least difficult speakers. Table 5.7 shows recall, precision, specificity, and F-measure values for male and female difficult target speaker detection. In each case, the number of easy and difficult speaker examples is given (note that there are 256 female speakers and 160 male speakers total). These results are all for SVMs using third order polynomial kernels, and rank normalized input feature statistics.

Sex	Number of example speakers	Recall	Precision	Specificity	F-measure
female	50 (20%)	0.752	0.725	0.613	0.739
female	60 (25%)	0.739	0.710	0.608	0.724
male	30 (20%)	0.807	0.658	0.467	0.725
male	40 (25%)	0.748	0.651	0.516	0.696

Table 5.7: *Recall, precision, specificity, and F-measure values for detecting difficult target speakers using a third order polynomial kernel SVM trained with rank normalized feature statistics, using SVMs trained separately for female and male speakers, with either 20% or around 25% of speakers taken as difficult or easy examples.*

In both female and male cases, the results do not improve over treating both sexes together. Recall increases slightly, at the cost of lower precision and specificity. Furthermore, note that increasing the number of speakers used as difficult and easy examples does not improve results. Including more speakers also means that the speakers used for training are not necessarily the best examples of difficult (or easy) ones, which potentially counteracts any gain from having more training examples. Training separate female and male SVMs for finding difficult impostor speakers gave results similar to those observed here: there were no gains over using a sex-independent SVM, and increasing the number of training examples to 25% also failed to improve results compared to using the top and bottom 20%. Given more female and male speakers for training, an approach using separate female and male SVMs may yield improvements. However, for the data available here, it is better to maximize the training examples and use the same SVM to detect difficult female and male speakers.

5.5 Discussion

In order to combine a set of feature statistics for detecting difficult speakers, who tend to cause a large number of errors, I trained SVMs to distinguish between examples of easy

and difficult speakers, for both target (true) speakers and impostor speakers. As input, I used a set of feature statistics calculated over speech regions, where the features include fundamental frequency, formant frequencies, energy, spectral slope, and MFCCs (both with and without mean and variance normalization).

Based on the results for the data set used here, this approach is more successful at finding difficult impostor speakers than difficult target speakers. One reason why finding difficult target speakers is more challenging than finding difficult impostors is that while there may be similar characteristics across difficult impostor speakers (which make them confusable with other speakers), the characteristics that make target speakers difficult may vary more from speaker to speaker. In both cases, however, recall and precision rates over 0.7 (or 0.8 in the case of difficult impostors) can be obtained. Furthermore, the threshold for picking a difficult speaker can be varied according to what errors are most important to minimize. For a false alarm rate of 5%, over 60% of difficult impostor speakers will still be found, and 37% of difficult target speakers. Given the challenging nature of the task, these recall rates are not particularly high (especially in the case of target speakers). However, for certain applications, the loss in recall may still be worth the gain in precision and specificity. Given enough training examples of difficult and easy speakers, there may be gains from treating female and male speakers separately. With limited data, though, better results are obtained by using the combined set of training examples in one sex-independent SVM.

One advantage of using feature statistics as the input to the SVM is that the statistics can be calculated over an individual conversation side or a set of conversation sides for the given speaker. This allows difficult speaker detection to work for varying amounts of available data. In my approach here, each conversation side of the easy and difficult speakers is used separately, with no exploitation of having more than one conversation side per speaker. One avenue for future exploration is to see how results change depending on the number of utterances used for each speaker. It may also be possible to find better feature statistics for detecting difficult speakers; the optimal feature statistics may be different for difficult target and impostor speakers, as well as for female and male speakers.

Another possible direction for future investigation is to see how well difficult conversation sides can be detected. The results of my error analysis, as well as the related work of Kahn et al. [34, 33], have shown that there can be particular conversation sides of a speaker that cause more errors than others. Being able to detect these "bad" utterances may provide very useful information for improving system performance.

Chapter 6

Conclusions and Future Work

This focus of this dissertation was on the intrinsic, speaker-based factors that contribute to errors in automatic speaker recognition systems. Inspired by the well-known work of Doddington et al. [22], which both categorized speakers according to their tendencies to cause errors and demonstrated the existence of such speaker types, I aimed to further explore the phenomenon of speaker-dependent system performance. In particular, there are two main components of this exploration, which are reviewed in the following sections. Section 6.1 describes the analysis of speaker behavior for two data sets and two types of automatic speaker recognition systems, with which I both confirm and build upon previous results demonstrating that system performance depends on speaker characteristics. Having established that certain speakers are more likely to cause errors than others, I then discuss a simple approach for finding these difficult speakers in Section 6.2. Section 6.3 concludes with a discussion of contributions and possible future work.

6.1 Analysis of Speaker Behavior

The aforementioned work of Doddington et al. analyzed errors only for female speakers, using data from the NIST 1998 Speaker Recognition Evaluation. In order to expand such analysis, I examined two data sets and two types of automatic speaker recognition systems, looking for speaker-dependent behaviors for both male and female speakers. The first data set was Switchboard-1, a corpus of conversational speech collected from the telephone. I further restricted this data to one type of telephone handset in order to limit the effects of extrinsic channel variability. Using scores from a GMM-UBM system, I began by considering a score confusion matrix for a set of 34 speakers with 10 conversation sides each. It was observed that the speakers varied both in how high their average true speaker scores were, as well as in how consistent the true speaker scores were across target-test pairs. There was also variability in how different target models of the same speaker behaved; for some speakers, scores were consistent across all models, while for others there was greater score variation.

Some impostor speaker pairs were more confusable than others, and some speakers had overall tendencies to have higher impostor scores.

Extending this analysis to include a large number of trials and speakers in Switchboard-1, I continued to show examples of varying speaker behavior, in terms of tendencies to have high or low target or impostor scores. For both female and male speakers, there was a correlation (around 0.6) between a tendency to cause high impostor scores as the target speaker and a tendency to cause high impostor scores as the test speaker.

For the Switchboard-1 data, I also investigated the possible effects of speaker sex, age, education level, and dialect area on system scores. Using analysis of variance (ANOVA) tests, I found significant differences between male and female score distributions. Significant differences were also found for score distributions with impostor speakers who have less than a five year age difference compared to impostor speakers with more than a five year age difference. The results for education level and dialect area were inconclusive. Based on such findings, I concluded that the most salient of these speaker demographics was sex, a result in line with other observations regarding differences in speaker recognition behavior between males and females.

For the second data set, I used a more recent collection of conversational and interview speech used in the 2008 NIST Speaker Recognition Evaluation (SRE08); this data contains much more channel variability, including not only landline and cellular telephone data, but also data from a variety of microphones. For this corpus, I used a GMM-UBM system with simplified factor analysis, in order to better handle the differences in channel. Once again, a variety of speaker-dependent system performance was observed, including tendencies to cause false alarm or false rejection errors. For both female and male speakers, 50% of the false rejection and false alarm errors were caused by only 15-25% of the speakers.

6.2 Difficult Speaker Detection

My approach for finding difficult speakers began with a method for calculating measures of similarity between impostor speaker pairs. Using statistics of features such as energy, formant frequencies, fundamental frequency, and spectral slope, calculated over all speech, I successfully obtained a variety of simple distance measures that could successfully select both easy- and difficult-to-distinguish speaker pairs, as evaluated by differences in detection cost and false alarm probability across a large number of systems. Of the performance measures tested, the best feature-measure at finding the most and least difficult-to-distinguish speaker pairs was the Euclidean distance between vectors of the mean first, second, and third formant frequencies. Even greater success was attained by the Kullback-Liebler (KL) divergence between pairs of speaker-specific GMMs. Furthermore, an examination of the smallest and biggest distances (as computed by the KL divergence) revealed individual speaker tendencies to consistently fall among the most (or least) difficult-to-distinguish speaker pairs.

I then used a set of feature statistics calculated over speech regions to train a support

vector machine (SVM) classifier to distinguish between difficult and easy speakers. Using scores from an automatic speaker recognition system, I ranked speakers according to the rate at which they caused false rejection and false alarm errors, taking the 20% of speakers with the most and least errors as difficult and easy training examples. Two separate SVMs were trained: one to detect difficult target speakers (who will cause false rejections) and one to detect difficult impostor speakers (who will cause false alarms). The resulting precision and recall measures were over 0.8 for difficult impostor speaker detection, and over 0.7 for difficult target speaker detection. Depending on the application, the detection threshold can be tuned to improve precision, recall, or specificity in order to best suit the needs of a particular task. At a 5% false alarm threshold, over 60% of difficult impostor speakers are found, and over 37% of difficult target speakers. These low recall rates (especially in the case of difficult target speakers) are indicative of the level of difficulty present in the task of finding error-prone speakers using a single conversation side. Nevertheless, the results are promising for a first attempt at such a task. The same approach can be taken with single conversation sides, as with a set of conversation sides corresponding to the same speaker, since the input feature statistics can be calculated over any number of speech samples.

6.3 Contributions and Future Work

The analysis showing the ways in which system scores depend on the speakers built upon and added to prior error analysis work. Considering two data sets, with differing degrees of channel and other extrinsic variability, along with two types of speaker recognition systems, I found that in both cases, speaker-dependent behavior is observed. I also noted differences between female and male speakers: there tend to be more confusable female impostor speaker pairs, perhaps due to the more limited range of certain acoustic characteristics, such as fundamental frequency, for female speech. Additionally, not only are there differences in tendencies for certain speakers to cause errors, there is also variability at lower levels, across different conversation sides of the same speaker. Furthermore, the tendency to produce false alarms as the target speaker is correlated with the tendency to produces false alarms as the impostor speaker.

Given such observations, I was then able to successfully predict difficult-to-distinguish impostor speaker pairs through the use of distance measures calculated with statistics of features such as fundamental frequency, formant frequencies, energy, and spectral slope. In addition to considering feature-measures that can give relative rankings of similarity between a pair of speakers, I also generalized the approach to simply detect a difficult individual speaker. Distinguishing between difficult target speakers and difficult impostor speakers, I trained SVMs using examples of the easiest and most difficult speakers in terms of causing errors. Both of these are novel approaches that can be used to address the effects of inherent speaker characteristics on automatic speaker recognition systems. Further exploration of this problem may yield better feature statistics or other improved approaches for finding

difficult speakers. Additionally, it may be possible to adapt this technique in order to detect particular conversation sides of a given speaker that will produce errors.

Bibliography

[1] Andre G. Adami, Radu Mihaescu, Douglas A. Reynolds, and John J. Godfrey. Modeling prosodic dynamics for speaker recognition. In *Proceedings of ICASSP*, 2003.

[2] Walter D. Andrews, Mary A. Kohler, and Joseph P. Campbell. Phonetic speaker recognition. In *Proceedings of Eurospeech*, 2001.

[3] Bishnu S. Atal. Effectiveness of linear prediction characteristics of the speech wave for automatic speaker identification and verification. *Journal of the Acoustical Society of America*, 55(6):1304–1312, 1974.

[4] Roland Auckenthaler, Michael Carey, and Harvey Lloyd-Thomas. Score normalization for text-independent speaker verification systems. In *Digital Signal Processing*, volume 10, pages 42–54, 2000.

[5] Rainer Banse and Klaus R. Scherer. Acoustic profiles in vocal emotion expression. *Journal of Personality and Social Psychology*, 70(3):614–636, 1996.

[6] Kofi Boakye. Speaker recognition in the text-independent domain using keyword hidden markov models. Master's thesis, University of California at Berkeley, 2005.

[7] Paul Boersma and David Weenink. Praat: doing phonetics by computer (version 5.0.3.0). http://www.praat.org.

[8] Jean-François Bonastre, Driss Matrouf, and Corinne Fredouille. Transfer function-based voice transformation for speaker recognition. In *Proceedings of Odyssey*, 2006.

[9] Jean-François Bonastre, Nicolas Scheffer, Driss Matrouf, Corinne Fredouille, Anthony Larcher, Alexandre Preti, Gilles Pouchoulin, Nicholas Evans, Benoit Fauve, and John Mason. ALIZE/SpkDet: a state-of-the-art open source software for speaker recognition. In *Proceedings of Odyssey*, 2008.

[10] Mike Brookes. VOICEBOX: Speech Processing Toolbox for MATLAB. http://www.ee.ic.ac.uk/hp/staff/dmb/voicebox/voicebox.html.

[11] Christopher J. C. Burges. A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, 2(2):121–167, 1998.

[12] William M. Campbell. Generalized linear discriminant sequence kernels for speaker recognition. In *Proceedings of ICASSP*, May 2002.

[13] William M. Campbell, Joseph P. Campbell, Douglas A. Reynolds, Douglas A. Jones, and Timothy R. Leek. Phonetic speaker recognition with support vector machines. In *Advances in Neural Information Processing Systems 16*, 2004.

[14] William M. Campbell, Douglas E. Sturim, and Douglas A. Reynolds. Support vector machines using GMM supervectors for speaker verification. *IEEE Signal Processing Letters*, 13(5):308–311, May 2006.

[15] Christopher Cieri, Walt Andrews, Joseph Campbell, George Doddington, John Godfrey, Shudong Huang, Mark Liberman, Alvin Martin, Hirotaka Nakasone, Mark Przybocki, and Kevin Walker. The mixer and transcript reading corpora: Resources for multilingual, crosschannel speaker recognition research. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC)*, pages 117–120, 2006.

[16] Christopher Cieri, Linda Corson, David Graff, and Kevin Walker. Resources for new research directions in speaker recognition: The mixer 3, 4 and 5 corpora. In *Proceedings of Interspeech*, 2007.

[17] Christopher Cieri, David Miller, and Kevin Walker. The Fisher corpus: a resource for the next generations of speech to text. In *4th International Conferenced on Language Resources and Evaluation, LREC*, pages 69–71, 2004.

[18] Brian R. Clifford. Voice identification by human listeners: On earwitness reliability. *Law and Human Behavior*, 4(4):373–394, 1980.

[19] Najim Dehak, Patrick J. Kenny, Reda Déhak, Pierre Dumouchel, and Pierre Ouellet. Front-end factor analysis for speaker verification. *Audio, Speech, and Language Processing, IEEE Transactions on*, 19(4):788 –798, May 2011.

[20] Volker Dellwo, Mark Huckvale, and Michael Ashby. How is individuality expressed in voice? an introduction to speech production & description for speaker classification. In Christian Müller, editor, *Speaker Classification*, volume 4343 of *Lecture Notes in Computer Science / Artificial Intelligence*. Springer, Heidelberg - Berlin - New York, 2007.

[21] George Doddington. Speaker recognition based on idiolectal differences between speakers. In *Proceedings of Eurospeech*, 2001.

[22] George Doddington, Walter Liggett, Alvin Martin, Mark Przybocki, and Douglas Reynolds. SHEEP, GOATS, LAMBS and WOLVES: A statistical analysis of speaker performance in the NIST 1998 speaker recognition evaluation. In *Proceedings of ICSLP*, 1998.

[23] W. Endres, W. Bambach, and G. Flosser. Voice spectrograms as a function of age, voice disguise, and voice imitation. *Journal of the Acoustical Society of America*, 49(6):1842–1848, 1971.

[24] Anders Eriksson and Par Wretling. How flexible is the human voice? - a case study of mimicry. presented at European Conference Speech Technology, Rhodes, 1997.

[25] Mireia Farrús, Javier Hernando, and Pascual Ejarque. Jitter and shimmer measurements for speaker recognition. In *Proceedings of Interspeech*, 2007.

[26] Jacob Goldberger and Hagai Aronowitz. A distance measure between gmms based on the unscented transform and its application to speaker recognition. In *Proceedings of Eurospeech*, 2005.

[27] Craig Greenberg, Alvin Martin, Linda Brandschain, Joseph Campbell, Christopher Cieri, George Doddington, and John Godfrey. Human assisted speaker recognition in NIST SRE10. In *Proceedings of Odyssey*, 2010.

[28] Andrew Hatch and Andreas Stolcke. Generalized linear kernels for one-versus-all classification: Application to speaker recognition. In *Proceedings of ICASSP*, 2006.

[29] Andrew O. Hatch, Barbara Peskin, and Andreas Stolcke. Improved phonetic speaker recognition using lattice decoding. In *Proceedings of ICASSP*, 2005.

[30] Qin Jin, Jiri Navratil, Douglas A. Reynolds, Joseph P. Campbell, Walter D. Andrews, and Joy S. Abramson. Combining cross-stream and time dimensions in phonetic speaker recognition. In *Proceedings of ICASSP*, 2003.

[31] Qin Jin and Alex Waibel. A naive de-lambing method for speaker identification. In *Proceedings of ICSLP*, 2000.

[32] Thorsten Joachims. Making large-scale support vector machine learning practical. In Bernhard Schlkopf, Chris Burges, and Alex J. Smola, editors, *Advances in Kernel Methods - Support Vector Learning*. MIT Press, 1999.

[33] Juliette Kahn, Nicolas Audibert, Solange Rossato, and Jean-François Bonastre. Intra-speaker variability effects on speaker verification performance. In *Proceedings of Odyssey*, 2010.

[34] Juliette Kahn, Solange Rossato, and Jean-François Bonastre. Beyond doddington menagerie, a first step towards. In *Proceedings of ICASSP*, 2010.

[35] Sachin Kajarekar, Luciana Ferrer, Kemal Sonmez, Jing Zheng, Elizabeth Shriberg, and Andreas Stolcke. Modeling NERFs for speaker recognition. In *Proceedings of Odyssey*, 2004.

[36] Sachin S. Kajarekar, Harry Bratt, Elizabeth Shriberg, and Rafael de Leon. A study of intentional voice modifications for evading automatic speaker recognition. In *Proceedings of Odyssey*, 2006.

[37] Sachin S. Kajarekar, Luciana Ferrer, Elizabeth Shriberg, Kemal Sonmez, Andreas Stolcke, Anand Venkataraman, and Jing Zheng. SRI's 2004 NIST speaker recognition evaluation system. In *Proceedings of ICASSP*, volume 1, pages 173–176, 2005.

[38] Patrick Kenny, Pierre Ouellet, Najim Dehak, Vishwa Gupta, and Pierre Dumouchel. A study of interspeaker variability in speaker verification. *Audio, Speech, and Language Processing, IEEE Transactions on*, 16(5):980 –988, july 2008.

[39] David Klusacek, Jiri Navratil, D.A. Reynolds, and J.P. Campbell. Conditional pronunciation modeling in speaker detection. In *Proceedings of ICASSP*, 2003.

[40] Jody Kreiman and George Papcun. Comparing discrimination and recognition of unfamiliar voices. *Speech Communication*, 10:265–275, 1991.

[41] Hisao Kuwabara and Yoshinori Sagisaka. Acoustic characteristics of speaker individuality: Control and conversion. *Speech Communication*, 16:165–173, 1995.

[42] Peter Ladefoged. *A Course in Phonetics*. Thomson Wadsworth, University of California, Los Angeles, fifth edition, 2006.

[43] Howard Lei and Nikki Mirghafori. Word-conditioned phone n-grams for speaker recognition. In *Proceedings of ICASSP*, 2007.

[44] Kung-Pu Li and Jack E. Porter. Normalizations and selection of speech segments for speaker recognition scoring. In *Proceedings of ICASSP*, pages 595–598, 1988.

[45] Linguistic Data Consortium. Switchboard-1 corpus. http://www.ldc.upenn.edu.

[46] Linguistic Data Consortium. Switchboard-2 corpus. http://www.ldc.upenn.edu.

[47] Alvin Martin, George Doddington, Terri Kamm, Mark Ordowski, and Mark Przybocki. The DET curve in assessment of detection task performance. In *Proceedings of Eurospeech*, volume 4, pages 1895–1898, 1997.

[48] Driss Matrouf, Nicolas Scheffer, Benoit Fauve, and Jean-François Bonastre. A straightforward and efficient implementation of the factor analysis model for speaker verification. In *Proceedings of Interspeech*, 2007.

[49] Kirsty McDougall and Francis Nolan. Discrimination of speakers using the formant dynamics of /u:/ in british english. In J. Trouvain and W. Barry, editors, *Proceedings of ICPhS*, pages 1825–1828, 2007.

[50] National Institute of Standards and Technology. The NIST year 2004 speaker recognition evaluation plan. http://www.nist.gov/speech/tests/spk/2004/SRE-04_evalplan-v1a.pdf, 2004.

[51] National Institute of Standards and Technology. The NIST year 2005 speaker recognition evaluation plan. http://www.itl.nist.gov/iad/mig/tests/spk/2005/sre-05_evalplan-v6.pdf, 2004.

[52] National Institute of Standards and Technology. The NIST year 2006 speaker recognition evaluation plan. http://www.itl.nist.gov/iad/mig/tests/spk/2006/sre-06_evalplan-v9.pdf, 2004.

[53] National Institute of Standards and Technology. The NIST year 2008 speaker recognition evaluation plan. http://www.nist.gov/speech/tests/sre/2008/sre08_evalplan_release4.pdf, 2008.

[54] National Institute of Standards and Technology. The NIST year 2010 speaker recognition evaluation plan. http://www.itl.nist.gov/iad/mig/tests/sre/2010/NIST_SRE10_evalplan.r6.pdf, 2010.

[55] Jiri Navratil, Qin Jin, Walter Andrews, and Joseph Campbell. Phonetic speaker recognition using maximum likelihood binary decision tree models. In *Proceedings of ICASSP*, 2003.

[56] Burhan F. Necioğlu, Mark A. Clements, and Thomas P. Barnwell III. Objectively measured descriptors applied to speaker characterization. In *Proceedings of ICASSP*, 1996.

[57] Douglas O'Shaughnessy. *Speech communications: human and machine*. Institute of Electrical and Electronics Engineers, 1999.

[58] Bryan L. Pellom and John H.L. Hansen. An experimental study of speaker verification sensitivity to computer voice-altered impostors. In *Proceedings of ICASSP*, 1999.

[59] Barbara Peskin, Jiri Navratil, Joy Abramson, Douglas Jones, David Klusacek, Douglas A. Reynolds, and Bing Xiang. Using prosodic and conversational features for high-performance speaker recognition: Report from JHU WS'02. In *Proceedings of ICASSP*, 2003.

[60] Norman Poh, Samy Bengio, and Arun Ross. Revisiting Doddington's zoo: A systematic method to assess user-dependent variabilities. In *Proceedings of Multimodal User Authentication*, 2006.

[61] Douglas A. Reynolds. Automatic speaker recognition using gaussian mixture speaker models. *The Lincoln Laboratory Journal*, 8(2):173–192, 1995.

[62] Douglas A. Reynolds. The effect of handset variability on speaker recognition performance: Experiments on the switchboard corpus. In *Proceedings of ICASSP*, volume 1, pages 113–116, 1996.

[63] Douglas A. Reynolds. Channel robust speaker verification via feature mapping. In *Proceedings of ICASSP*, 2003.

[64] Douglas A. Reynolds, Thomas Quatieri, and Robert Dunn. Speaker verification using adapted gaussian mixture models. *Digital Signal Processing*, 10:19–41, 2000.

[65] Astrid Schmidt-Nielsen and Thomas H. Crystal. Speaker verification by human listeners: Experiments comparing human and machine performance using NIST 1998 speaker evaluation data. *Digital Signal Processing*, 10:249–266, 2000.

[66] Kare Sjolander. The snack sound toolkit. http://www.speech.kth.se/snack/, 2004.

[67] Alex Solomonoff, William M. Campbell, and Ian Boardman. Advances in channel compensation for SVM speaker recognition. In *Proceedings of ICASSP*, 2005.

[68] Andreas Stolcke, Luciana Ferrer, and Sachin Kajarekar. Improvements in MLLR-Transform-based speaker recognition. In *IEEE Odyssey Speaker and Language Recognition Workshop*, 2006.

[69] Andreas Stolcke, Luciana Ferrer, Sachin Kajarekar, Elizabeth Shriberg, and Anand Venkataraman. MLLR transforms as features in speaker recognition. In *Proceedings of Eurospeech*, pages 2425–2428, 2005.

[70] Remco Teunen, Ben Shahshahani, and Larry Heck. A model-based transformational approach to robust speaker recognition. In *Proceedings of ICSLP*, 2000.

[71] Robbie Vogt, Brendan Baker, and Sridha Sridharan. Modelling session variability in text-independent speaker verification. In *Proceedings of Interspeech*, 2005.

[72] Steve J. Young, Dan Kershaw, Julian Odell, Dave Ollason, Valtcho Valtchev, and Phil Woodland. *The HTK Book Version 3.4*. Cambridge University Press, 2006.