Increasing Speaker Recognition Algorithm Agility and Effectiveness for “Unseen” Conditions

Fred Goodman, MITRE Corporation
Talk Outline

- Issues when using Speech as a Biometric
- Evaluating Speaker Recognition Systems
- Speaker Recognition Techniques
- Expanding Speaker Recognition Applications
- Dealing with “Unseen” Conditions
- Conclusions
Speech as a Biometric

- Speech is “performed”, while many other biometrics (fingerprint and iris) are not. Performances are affected by internal factors (“intrinsic”) as well as external ones (“extrinsic”).

- Modern speaker recognition is concerned with text-independent matching.

- Testing assumes the talker is not “cooperative”; i.e. the talker is unaware of the system.

- Most testing uses a verification paradigm (i.e. an identity is claimed; the system says yea or nay). This generalizes to predict closed-set or even open-set testing results.

- Note: Human SID performance is generally worse than machine performance! (exception: close friends, loved ones).
Generic SID Biometric Block Diagram

Enrollment

Bob's Model

Text-Independent, “Unaware”

Feature Extraction → Model Training

Bob

Sally

Sally's Model

Recognition

???????

???

Feature Extraction → Scoring & Decision

Sally!

N.B. – Must permit “none-of-the-above”
What comes out of a SID verifier?

- A number representing the likelihood that the current speaker is the same as the “model” speaker
- The figure shows actual score histograms (NIST 2008 eval.)

Target PDF: \( \mu=4.5, \sigma=2.01 \)

Impostor PDF: \( \mu=0, \sigma=1.0 \)

More FA, Fewer misses

Fewer FA, More misses

MD: Missed Detection

FA: False Accept

Decision Threshold
Characterizing Performance: The DET Curve

- The Detection Error Tradeoff curve shows performance at all threshold settings simultaneously.

Notice: If \( P(tgt) = 0.001 \) and \( EER=1\% \), for 1000 trials, we get \( \sim1 \) true hits & \( \sim11 \) FAs.
Issues when using Speech as a Biometric
Evaluating Speaker Recognition Systems

Speaker Recognition Techniques
Expanding Speaker Recognition Applications
Dealing with “Unseen” Conditions
Conclusions
Sources of Speaker Identity (Features)

- **Low-level (10 – 30 msec)**
  - Anatomical structure of vocal tract (e.g. nasal passages)
  - Acoustical characteristics of glottal source

- **Medium-level (100s of msec)**
  - Prosodics: rhythm, speed, intonation, volume
  - Idiosyncrasies (e.g. lip smacks, ‘uh-huh’)

- **High-level (100 – 1000 msec)**
  - Word choices
  - Grammatical usages
  - Accent/Dialect/Language
Speech Spectrograms

Analysis Window
\(~=100\) samples (WB)  "Greasy wash water all year"

Analysis Window
\(~=400\) samples (NB)
Spectro-Temporal Receptive Fields (STRFs)

STRF features are extremely robust to wideband noise
Prosodic Features in SID

- Pitch, energy & duration short-time values are converted into “features” as shown below:

- Those features are turned into even more sophisticated features using N-grams, rank normalization, etc; ultimately a classifier is applied (e.g. Support Vector Machine).
- Good performance requires several minutes of speech
- Fuses very well with other methods
MLLR: Deviation from the Average Speaker

- The MLLR (Maximum Likelihood Linear Regression) technique originally used in speech recognition, has proven valuable for SID

\[
\mu_{\text{new}} = A \mu + b
\]
Where \(A\) is a matrix & \(b\) is a vector

- \(A\) is 39x39 and \(b\) is 39x1

- Up to 8 phone classes used

- MLLR relies on speech recognition to find phone boundaries
Gaussian Mixture Modeling (GMM)

■ With a small number of parameters, complex shapes can be modeled (3 1-Dim. Gaussians shown below):

![Graph showing 3 Gaussian distributions with mean values and standard deviations.]

3 µs, 3 σ’s, 3 wts

■ 2-D Example*: Training uses EM iterative algorithm) to build 3-element model

![Diagram illustrating the EM algorithm in 2-D space with red arrows indicating iteration progress.]

Random Starting points

Final- (8 iterations later)

[* Actually 40-dim features, 1-2k mixtures]
"Supervectors" & Dimension Reduction

- Concatenate GMM mixture means to make a "Supervector" (up to 2k*40)=80k length vector
- Reduce "noise" dimensions by applying Joint Factor Analysis or i-vector/PLDA
Expanding Speaker Recognition Applications

- Landline Telephone: 1970
- Consistent “Calibration”: 1996
- Cellular Telephone: 2001
- Language (Multiple/Cross): 2004
- Interview (Cross) Microphone: 2008
- Cross-Channel (tel. vs. interview): 2008
- Aging: 2010
- Vocal Effort/Lombard: 2010
- Additive Noise: 2011
- Room Reverberation: 2011
- Cross-Room (‘bright’ vs. ‘dead’): 2011
- Minimal/No Training Data: 2011
- “Confidence”: 2011
Issues when using Speech as a Biometric
Evaluating Speaker Recognition Systems
Speaker Recognition Techniques
Expanding Speaker Recognition Applications
Dealing with “Unseen” Conditions
Conclusions
Defining the “Unseen” Data Problem

- Traditional pattern recognition techniques require substantial training data from the same source.
- Without such training data, getting a valid log-likelihood ratio is problematic.
- But real-world applications may not cooperate with our needs:
  - Infinite number of room sizes, microphone positions, wall materials, noise sources, etc.
  - Unlike telephone where standards limit variation.
- Algorithms historically never self-modified, based on conditions. Even now, they do very little....
- What can be done to limit the damage when a new source of data appears?
- “Solving” this problem means getting close to clean performance.
Solving the “Unseen” Data Problem

- Use simulation to create extrinsic conditions (noise, reverb)
  - Feed simulated data to make backend (JFA, i-vectors) better
- Collect intrinsic conditions
  - Whisper to shout (effort), fast to slow (rate)
  - Read vs. oration vs. telephony vs. interview (style)
  - Illness, drunk, sleepy, aging
- Understand the effects on Speaker models
  - Automatically detect conditions (e.g. SNR, speech rate)
  - Modify algorithms according to the differences between training and test conditions
- For a brand-new condition:
  - Use unsupervised adaptation to improve performance over time
  - Learn to detect data too bad to process effectively (no-decision)
  - Use supervised adaptation with a few known “true” cuts
Example Condition-Driven Algorithm Mods

- Modify front-end feature extraction based on conditions, because a feature set is robust against reverb.
- Decide to weight certain speech sounds (phonemes) differently because noise is distorting them (fricatives, mixed-excitation sounds – “zh”)
- Change fusion weights based on SNR or Reverb (RT) because (e.g.) prosodic energy features degrade quickly in that condition.
- Modify decision threshold to reflect large differences in either extrinsic or intrinsic conditions (e.g. vocal effort) between training and recognition samples.
Conclusions

- Speaker recognition is still a serious research issue 40 years after its birth
- The expansion of application conditions since 2006 has been dramatic
- But we are coming to a crossroads:
  - Collecting hundreds of speakers is expensive
  - Exposing them to many extrinsic/intrinsic conditions is time-consuming & difficult
- Encouraging algorithm developers to use simulated extrinsic data to become more robust
- Must continue to collect intrinsic variations until better models of speech behavior can be built
- Encourage algorithm developers to estimate extrinsics/intrinsics & modify algorithms accordingly
Thanks for inviting me and listening!
Extra Slides
Mel-Warped Cepstrum Features

The mel-scale, based on human perception, is approximately linear for frequencies less than 1000 Hz and logarithmic for frequencies greater than 1000 Hz.

\[ \text{mel} = 2595 \log_{10}((f/700)+1) \]

Triangular, Mel-Weighted Filter Bank

12 < \( N \) > 20, plus Velocity and (perhaps) Acceleration terms

Window \( \rightarrow \) \(|\text{DFT}|\) \( \rightarrow \) Mel-Warp \( \rightarrow \) log \( \rightarrow \) DCT \( \rightarrow \) Take 1\(^{st}\) N \( \rightarrow \) Time Diff.
**Frequency Domain Linear Prediction**

Alternative Feature set, shows robustness to reverb

- **DCT**
- **Sub-band Windowing (96 bands)**
- **FDLP**
- **Gain Norm.**
- **Mel-scale Short-term Integration (32 ms)**
- **Cepstral Xform**

![Graph representing frequency domain linear prediction](image)

© 2012 The MITRE Corporation. All rights reserved. Approved for Public Release 12-0099. Distribution Unlimited.
I - Vector Generation/ PLDA

- M = m + Tw (m is the UBM Supervector, M is the incoming Supervector)
- Estimate the Total variability matrix T, given training GMM Supervectors (using the EM algorithm).
- The i-vectors (w) are the speaker/session factors of the T matrix (analogous to the factors in JFA)
- Results in a ~400 element vector w
- PLDA breaks it down further, with the i-vectors as an input:
  - w = m + Vy + Ux + ε, where
  - V = speaker subspace (y are the factors)
  - U = channel subspace (x are the factors)
  - m = mean vector over all training data
  - ε = residual noise (covariance matrix Σ)
“Shoebox” Room Reverberation Simulation

- Allows the user to specify:
  - Materials for the 4 walls, ceiling & floor
  - Dimensions (x,y,z)
  - Positions of the sound source & receiver
  - HRTF for receiver
- Results in a Room Impulse Response
  - Characterized by “RT60” metric
  - Which can then be convolved with clean speech
- Key Limitation: can’t put humans in the room – bodies soak up sound. As a result RIR is overly “bright”.
- Much more sophisticated room simulations exist ($$$)
Collecting Interview Room Data (NI ST/ LDC)

Each room has ~16 microphones. In addition, telephone calls are made by the same speakers.
Vocal Effort Collections?

Lombard Effect

White, Pink, Babble

dB Level

Fixed or Variable

MIXER

Noisy

Clear Voice

Output

VE Effect (Oration)

10 meters

5 meters

2.5 meters
Score-Level Fusion

- Fusion weights and offset developed using a small development data set

Fusion DET Curve

- Subsystem #1
- Subsystem #2
- Subsystem #3
- Subsystem #7
- Subsystem #8

Fusion Weights (A)

Fusion offset (b)