OPPORTUNITIES AND CHALLENGES OF PARALLELIZING SPEECH RECOGNITION

Jike Chong, Gerald Friedland, Adam Janin, Nelson Morgan, Chris Oei
OUTLINE

• Motivation
• Improving Accuracy
• Improving Throughput
• Improving Latency
Meeting Diarist Application “Parlab All”
MEETING DIARIST

Audio Signal

Speaker Diarization

Speech Recognition

Relevant Web Scraping

"who spoke when"

"who said what"

"what was said"

"what's relevant to this"

"what are the main points"

Higher-level analysis

Question Answering

Indexing, Search, Retrieval

Summarization
MOTIVATION

• Speech technology has a long history of using up all available compute resources.

• Many previous attempts with specialized hardware with mixed results.
1: IMPROVING ACCURACY

• Speech Technology works well when:
  • Large amounts of training data match application data
  • Small vocabulary; simple grammar
  • Quiet environment
  • Head-worn microphones
  • “Prepared” speech
  • Each change adds 10% error!
FEATURES

• Most state-of-the-art features are loosely based on perceptual models of the cochlea with a few dozen features.

• Combining multiple representations almost always improves accuracy, especially in noise.

• Typical systems combine 2-4 representations.

What if we used a LOT more?
MANYSTREAM

• Based on cortical models
• Large number of filters
• Each filter feeds an MLP.

• Current combination method uses entropy-weighted MLP, but many other possibilities.
MANYSTREAM

It helps!

• 47% relative improvement over baseline for noisy “numbers” using 28-stream system.

• 13.3% relative improvement over baseline for Mandarin Broadcast News using preliminary 4-stream system.
MANYSTREAM

- Next steps:
  - Fully parallel implementation
  - Many more streams
  - Other combination methods
2: IMPROVING THROUGHPUT

• Serial state-of-the-art systems can take 100 hours to process one hour of a meeting.

• Analysis over all available audio is generally more accurate than on-line systems.

• Batch processing per utterance is “embarrassingly” parallel.
INFEERENCE ENGINE

Features from one frame

Gaussian Mixture Model for One Phone State
- Computing distance to each mixture components
- Computing weighted sum of all components

HMM Acoustic Phone Model

Pronunciation Model
- HOP hh aa p
- ON aa n
- POP p aa p

Bigram Language Model

WFST Recognition Network

Features from one frame

Gaussian Mixture Model for One Phone State

HMM Acoustic Phone Model

Pronunciation Model
- HOP hh aa p
- ON aa n
- POP p aa p

Bigram Language Model

WFST Recognition Network
INFEREN CE ENGINE

• At each time step, compute likelihood for each outgoing arc using the acoustic model.

• For each incoming arc, track all hypotheses.

• Regularize data structures to allow efficient implementation.

• The entire inference step runs on the GPU.
INFERENC ENGINE

• 11x speed-up over serial implementation.
• 18x speed-up for compute intensive phase.
• 4x speed-up for communication intensive phase.
• Flexible architecture
• Audio/visual plugin added by domain expert.
INFERENC ENGINE

• Next steps:
  • Generate lattices and/or N-best lists.
  • Explore other parallel architectures.
  • Distribute to clusters.
  • Explore accuracy/speed trade-offs.
3: IMPROVING LATENCY

• For batch, latency = length of audio + time to process.

• On-line applications require control of latency.

• Parallelization allows lower latency and potentially better accuracy.
SPEAKER DIARIZATION

Speaker Diarization – Definition

Estimate “who spoke when” with no prior knowledge of speakers, number of speakers, words, or language spoken.

Audiotrack:

Segmentation:

Clustering:

Speaker A  Speaker B  Speaker C  Sp. A  Speaker B
OFFLINE SPEAKER DIARIZATION

Initialization

Cluster2  Cluster1  Cluster2  Cluster2

(Re-)Training

Yes

Cluster1  Cluster2  Cluster1  Cluster2

(Re-)Alignment

Merge two Clusters?

No

End
ONLINE SPEAKER DIARIZATION

• Precompute models for each speaker.
  • Run offline diarization on the start of a meeting.
  • Train models on first 60 seconds from each resulting speaker.
  • Another approach: stored models per speaker.

• Every 2.5 seconds, compute scores for each speaker model and output the highest.
HYBRID ONLINE/OFFLINE DIARIZATION

Audio Signal

Online Subsystem
- 2.5 sec Buffer
- Online Decision
- Online Decision
- MAP Training

Offline Subsystem
- History Buffer
- Segmentation
  - Diarization Engine
  - Clustering
- Speaker Mapping

"who is speaking now"
HYBRID ONLINE/OFFLINE DIARIZATION

Online Diarization: DER/Core

Error %

Cores Dedicated to Offline Subsystem
DIARIZATION

• Next steps:
  • CPU/GPU hybrid system
  • Implement serial optimizations in parallel version
  • Integrate with manystream approach
CONCLUSION

• Speech technology can use all resources that are available.

• Parallelism enables improvements in several areas:
  • Accuracy
  • Throughput
  • Latency

• Programming parallel systems continues to be challenging.