# Knowledge Acquistion for Speech and Language from a Large Number of Sources

JHU Workshop04

James Baker, CMU

July 23, 2004

# Research Issues

- HMMs and n-grams are reaching limits
  - Not much improvement just from more data
  - Not good at learning structure
- Much existing knowledge is not correctly represented
  - Articulation
  - Speaker variability and dialects
  - Common sense language knowledge

# Salient New Features

- Eliminate dependency on HMMs
- Eliminate dependency on EM algorithm
- Detailed articulatory modeling
- Knowledge acquistion from large number of informants
- Formulation of training as massive constrained optimization problem
- Training on millions of hours of speech and trillions of words of text

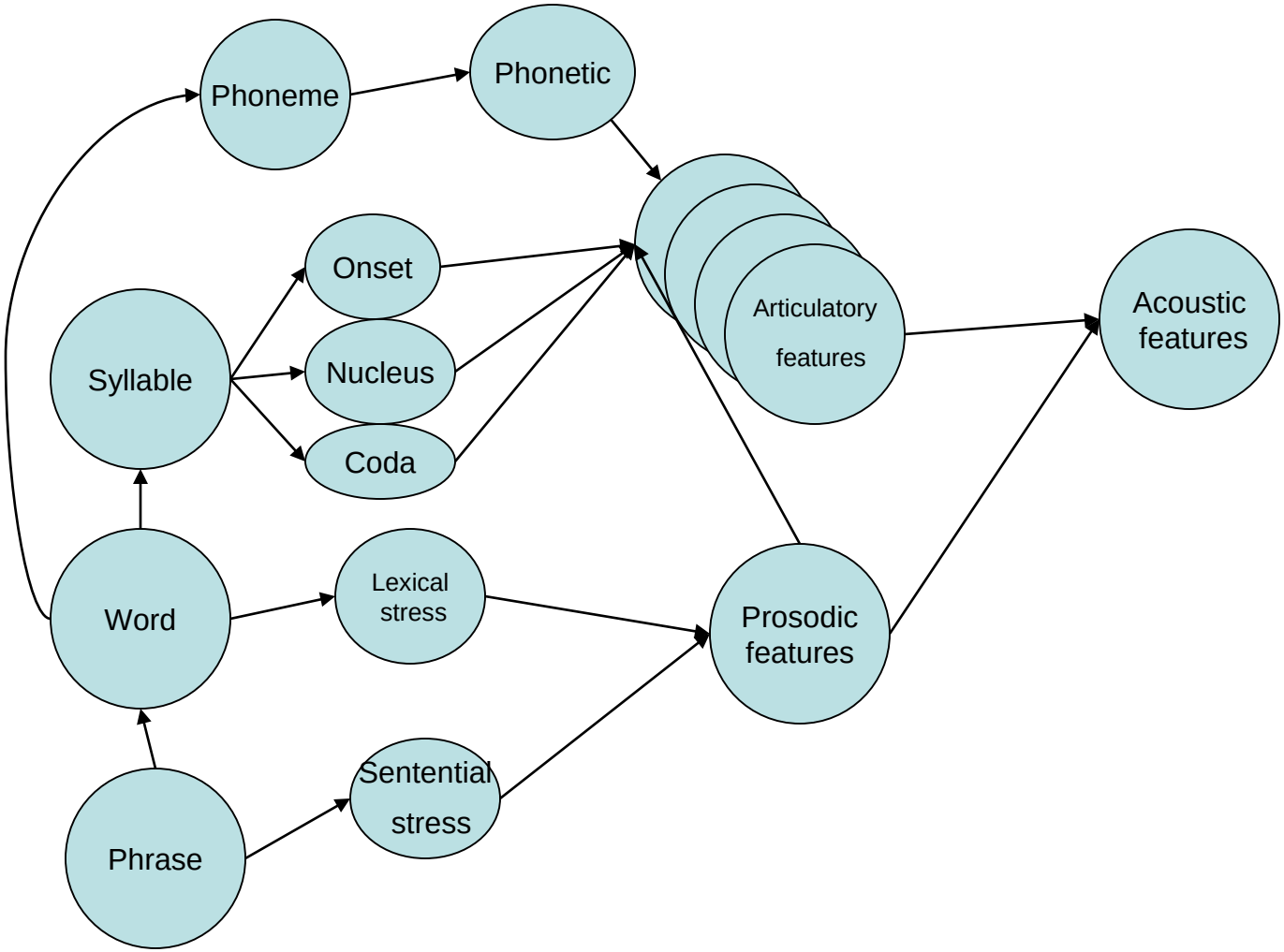
# Elements of New Methodology

- Overall architecture: System of systems
- Knowledge of speech production
- Speaker variation and dialect modeling
- Knowledge acquisition from large number of informants
- Large quantity of data and knowledge
- Virtual reality, role playing game
- Pairwise hypothesis rescoring
- Scaling and distributed computing

# System of systems

- Multi-tiered approach
- Multiple instantiations for each tier for robustness
- Intelligent (“glass box”) cooperation
  - For computational efficiency and robustness
- Subsystems optimized for specialized tasks
  - For computational efficiency and minimum error rate
- Final rescoring does not need to be probability model based

# Multi-Tiered (Simplified)



# Detailed Acoustic Knowledge

- Knowledge of speech production
  - Hypothesis scoring by “mimic” synthesis
    - One score component for parameter tracks conditioned on hypothesis
    - 2nd Score component for match to observed acoustics
- Asynchronous articulatory streams
- Abductive inference
  - Require explanations for all observed events (both in training and in recognition)
  - Score plausibility of explanation

# “Mimic” Synthesis

- Hypothesis specific “verification by synthesis”
- Conditions:
  - Known text
  - Known sample of speech
  - No limit on number of bits
  - Must resynthesize from parametric model conditioned on hypothesized states in “hidden” tiers
- We know more than normal compression (script)
- We know more than normal TTS (speech sample)
- Get knowledge of hidden states
- Each Voice Model is as detailed as TTS voice
  - Use (say) 100 hours of recordings per voice
- Top down analysis in context of rescoring

# Top Down Analysis

- Computational efficiency
  - Top down search is computationally expensive
  - Bottom-up detection can be computationally very efficient (FastMatch, look-ahead, etc.)
  - Top down analysis in rescoring takes much less computation than in search
- Combines with abductive inference
  - Analysis must explain all observed events
  - Give special attention to “unusual” events

# Speaker Variation and Dialect Modeling

- Each voice is modeled as a parametrizied manifold
  - Requires full voice models for a set of prototype voices
- Speaker variation and dialect are modeled as smooth manifold transformations
  - Interpolate to model other voices
- Speaker variability is **not** a random variable independently sampled every 10 milliseconds

# Knowledge Acquired from Large Number of Informants

- All speakers of a language have common everyday language knowledge that far exceeds our best systems
- Acquire this knowledge as a large number of small factoids
- Acquire factoids through high volume computer applications with interactive use of speech and language
  - The application may be based on a speech or language task important to the user
  - The application may simply be fun (a game)
- Can also acquire large quantity of recorded speech and data of speech perception

# Kinds of Knowledge from Informants

- Language knowledge (e.g. can the informant correct an error from given context)
  - Correction from text only
  - Correction from limited context
- Speech perception
  - Knowledge complementary to production knowledge
  - Can find boundaries of decision surface
- Read speech
- Correction of recognition
- Can present informants with artificially generated errors
  - Warning: statistically biased sampling
  - Advantage: Can directly measure decision boundary

# How to Get Millions of Informants

- Have the knowledge acquisition process be an integral part of a large volume application
  - Example: A MMORPG within a fantasy/sci-fi setting requiring communication among the characters across multiple natural and artificial languages
    - Players must communicate with other characters using errorful speech-to-speech translation devices
    - Players actively work to teach their devices to get better
    - The game uses simulated errors as well as real errors
  - Other examples: real translation, real speech recognition, language learning

# Quantity of Data

- Millions of hours of speech
- Trillions of words of text

- Such a quantity of data is available
- New training algorithms are proposed for distributed computing
- The methodology provides for semi-supervised training
- New multi-tiered micro-detailed models can utilize the knowledge from such a quantity of data

# Different Objectives for Different Components

- The purpose of FastMatch is to get the correct answer on the short list, not to get it to rank 1
- The purpose of the Search Match is to produce a lattice with the correct hypothesis, not to get the correct hypothesis to have the best score
- Only the final rescoring has to give the correct answer the best score

# Paradox: Language Model for Search

- The correct language model (even if known exactly) is not the best language model for search
  - Example: A large grammar with many canned phrases in a noisy environment
    - When the best scoring (but wrong) hypothesis is in the middle of a long canned phrase it's language model may cause the correct hypothesis to be pruned
    - On the other hand, correctly giving the correct hypothesis a very good LM score only reduces computation for an answer we would get right anyway
    - A less "sharp" LM will give more accurate search
- All models must be optimized for their specific task (which is not the same as the ML estimator)

# Constrained Optimization

- In rescoring, all that really matters is the correction of errors!!
  - This can always be formulated as a constrained optimization problem, regardless of the underlying models used in the base system
  - For training data, focus on just two hypotheses:
    - The current best scoring hypothesis
    - The correct hypothesis
    - Build a custom discriminator for “difference events” R(H1,H2)
      - Note: This is a completely different process than conventional training
      - Completely different (new) models
      - Also different even from “corrective” training
- Each error or close call produces a constraint
- Each factoid from an informant produces a constraint
- Training for either acoustic models or language models

# Compare Just Two Hypotheses

- Only “difference events” matter
  - Do not need to match against a sequential stochastic process
  - Build a context-dependent rescoring model for each difference event (does not need to be a probability model)
- The error is corrected if the revised score, taking account of all the difference events, is better for the correct hypothesis
- Arbitrary models may be used
  - Not restricted to corrective training of parameters in existing models
  - Not restricted to models interpreted as probability distributions
  - Exponential models do not necessarily require evaluation of partition function (because it is the same in the numerator and denominator of log likelihood ratio)

# Probability Modeling vs Hypothesis Comparison

- Corrective training of GMM parameters is an improvement over Maximum Likelihood
- GMM parameters are a special case of general non-linear regression
- Regularization improves generalization
- ML < CT < NR < RNR

# Form of Optimization Problem

$$Minimize: E = \sum_j f(w_j) + C\sum_i g(\xi_i)$$

$$SubjectTo: \forall_i \sum_j w_j a_{i,j} + \xi_i \geq d_i; \xi_i \geq 0;$$

$$a_{i,j} = \varphi_j(\vec{x}_i) y_i$$

Φ may be based on a kernel function:

$$\varphi_j(\vec{x}_i) = K(\vec{x}_i, \vec{x}_j)$$

Problem: There may be millions of constraints (i) and millions of terms (j).

Potential Solution: Distribute problem among millions of computers.

New Problem: How to do massively distributed computation. Avoid computing a(i,j) if i data and j are on different computers.

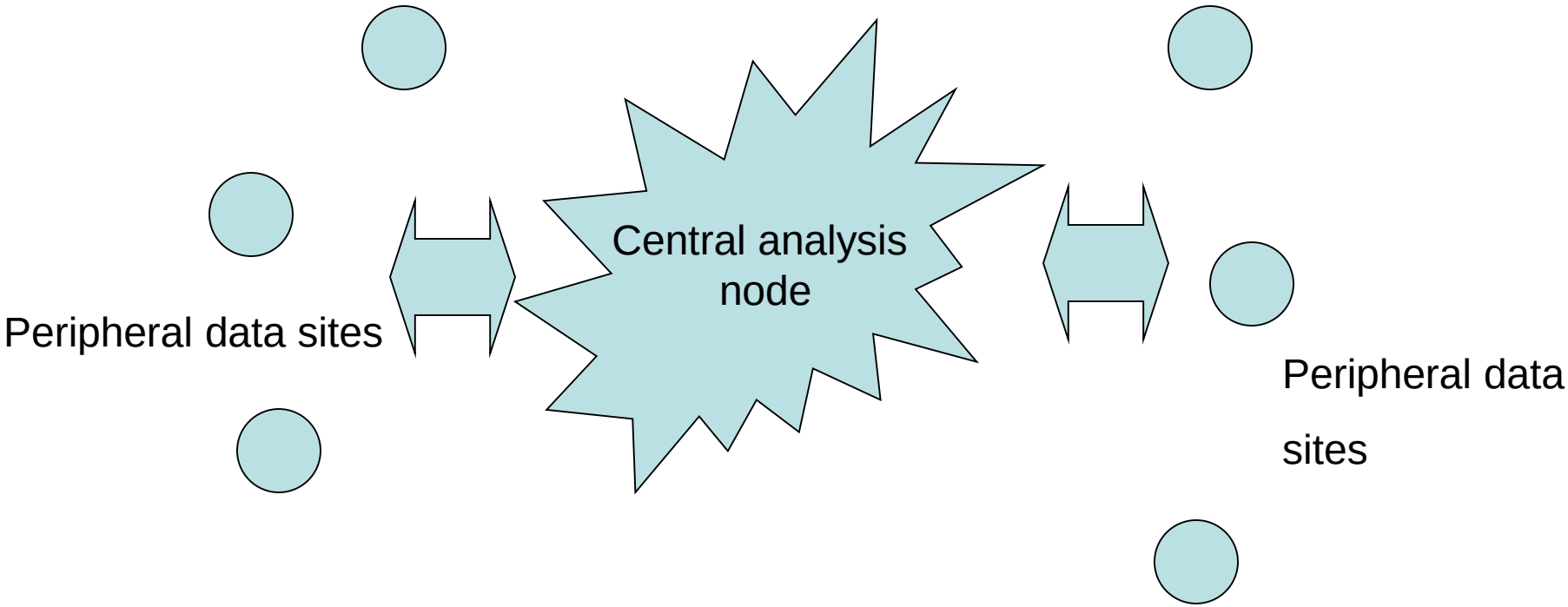
# Scaling Problem Size

- Some algorithms do not scale
  - Standard SVM training (quadratic programming)
    - Generally regarded as scaling only to thousands of constraints
  - Simplex method (linear programming)
    - Selecting new variable to enter basis appears to require products of data available only at different distributed sites

# Distributed Computing

- Special property of our problem: “soft” constraints
  - No point is truly “infeasible”
  - Idea: Combine phase 1 “feasibility” computation with phase 2 optimization computation
- Possible solution methods for distributed computing:
  - Interior point/barrier function methods
  - Primal/dual active set methods
- Additional benefit – new data can be incorporated incrementally without redoing “phase 1” feasibility computation

# Star Topology



# Distributed Knowledge Acquisition

- One shot learning (can be learned in isolation)
  - New words
  - New pronunciations
- Naturally distributed
  - Estimation of hidden variables
  - Collection of factoids
- Must be coordinated
  - Context-dependent rules or conditional probabilities
  - Systematic relationships (e.g. dialects)

# Summary of New Features (1)

- Multi-tiered
- Use of Model of reliability of components
- Intelligent combining for computational efficiency and robustness
- Detailed speech production modeling
- Abductive inference
- Speaker variability as smooth manifold transform
- Large number of language “factoids”
- Valency based structured language model
- Knowledge acquisition from large number of informants
- Knowledge acquisition from artificially created errors

# Summary of New Features (2)

- Not dependent on HMM modeling
- Pairwise comparison of hypotheses
- Minimum error rate as a constrained optimization problem
- Large quantity of data and knowledge
  - Millions of hours of speech
  - Trillions of words of text
  - Semi-supervised training (millions of factoids)
- Massively distributed computing
  - Millions of constraints
  - Millions of variables
  - Thousands (perhaps millions) of computers