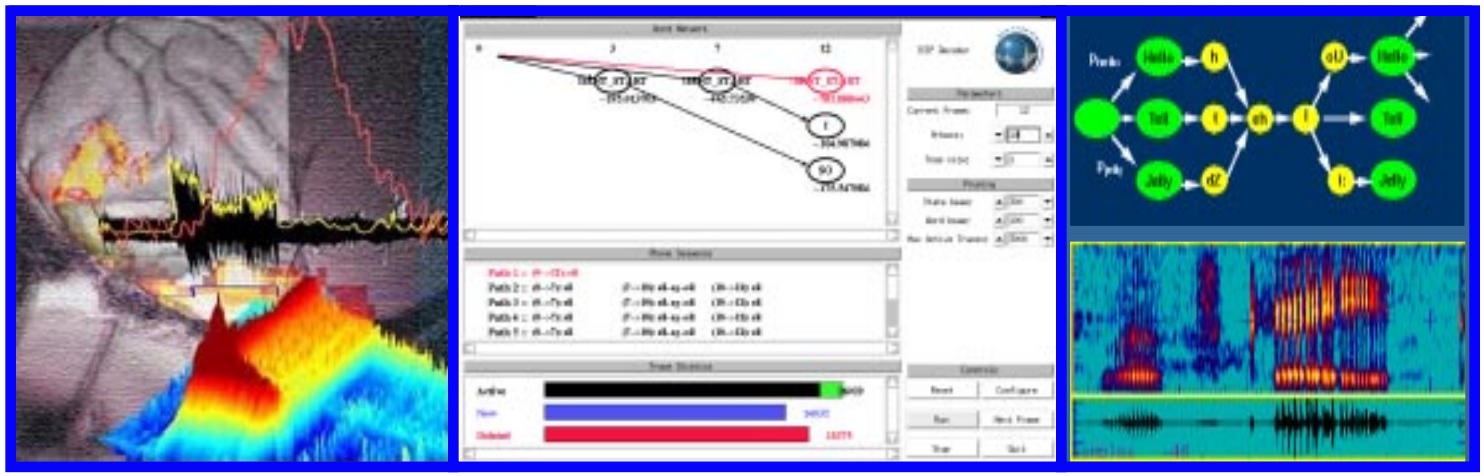
SPARSE BAYESIAN METHODS FOR CONTINUOUS SPEECH RECOGNITION

Jonathan E. Hamaker, Bohumir Jelinek and Joseph Picone

Institute for Signal and Information Processing
Mississippi State University

{hamaker, jelinek, picone} @isip.msstate.edu



RESEARCH GOALS

- Overarching goal: Build better, more robust speech systems

  Our belief is that better (more realistic - less assumptions) acoustic models will yield a more robust system.

  Take our cue from humans who are able to classify using both representational and discriminative knowledge.

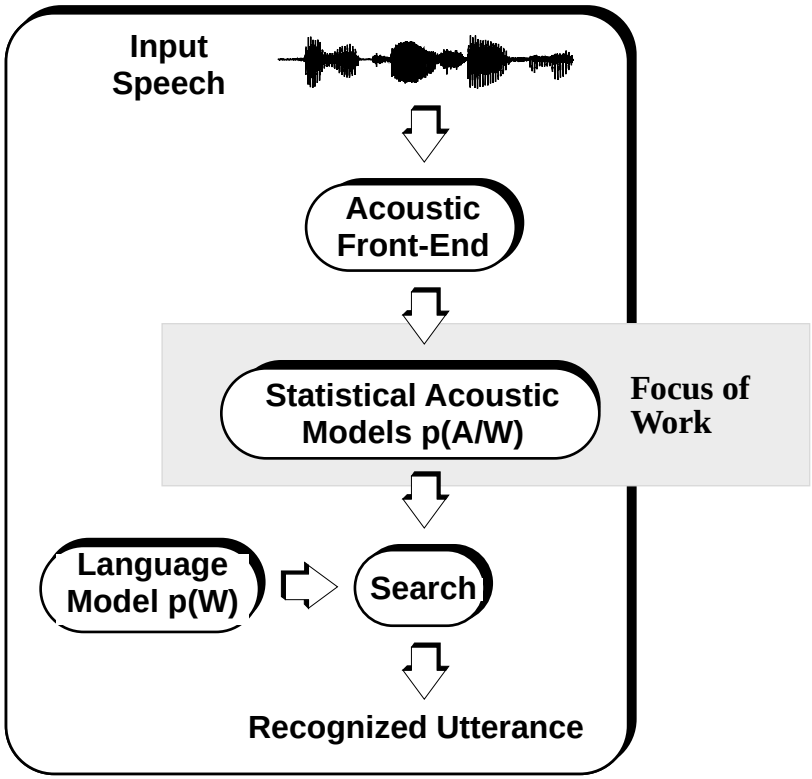
- ITR goal: Explore discriminative modeling in the context of LVCSR technology

  Primary focus: kernel-based methods including Support Vector Machines (SVMs) and Relevance Vector Machines (RVMs) - non-parametric, nonlinear discriminative learning paradigms

  Eliminate problems of practicality that exist when applying these methods to LVCSR - training time and memory are polynomial with the size of the training corpus.

  Apply current technology in LVCSR - EM style training and online

ASR PROBLEM



- The Front-end maintains information important for modeling in a reduced parameter set.

- The language model typically predicts a small set of next words based on knowledge of a finite number of previous words (N-grams) — leads to search space reduction.

Bayesian formulation for speech recognition:

$$P(W|A) = \frac{P(A|W)P(W)}{P(A)}$$

Objective: minimize the word error rate by maximizing $P(W|A)$

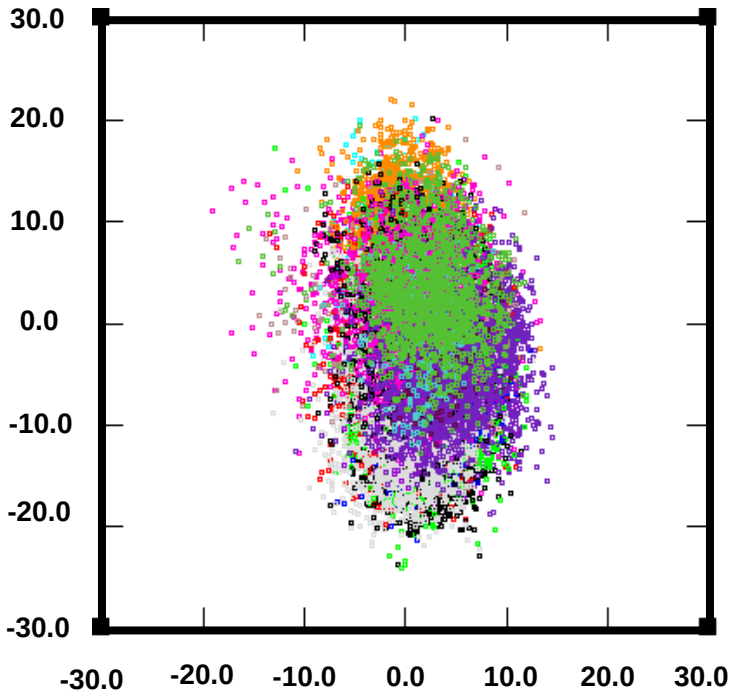
Approach: maximize $P(A|W)$ (training)

Components:

- $P(A|W)$: acoustic model (hidden Markov models, mixture of Gaussians)
- $P(W)$: language model (statistical, N-grams, finite state networks)
- $P(A)$: acoustics (ignore during maximization)

ACOUSTIC MODELING

Acoustic Confusability: Requires reasoning under uncertainty!



- First two cepstral coefficients for all vowels (based on a conversational speech corpus — SWITCHBOARD).
- Overlap represents a fundamental barrier for good classification.

Acoustic Models Must:

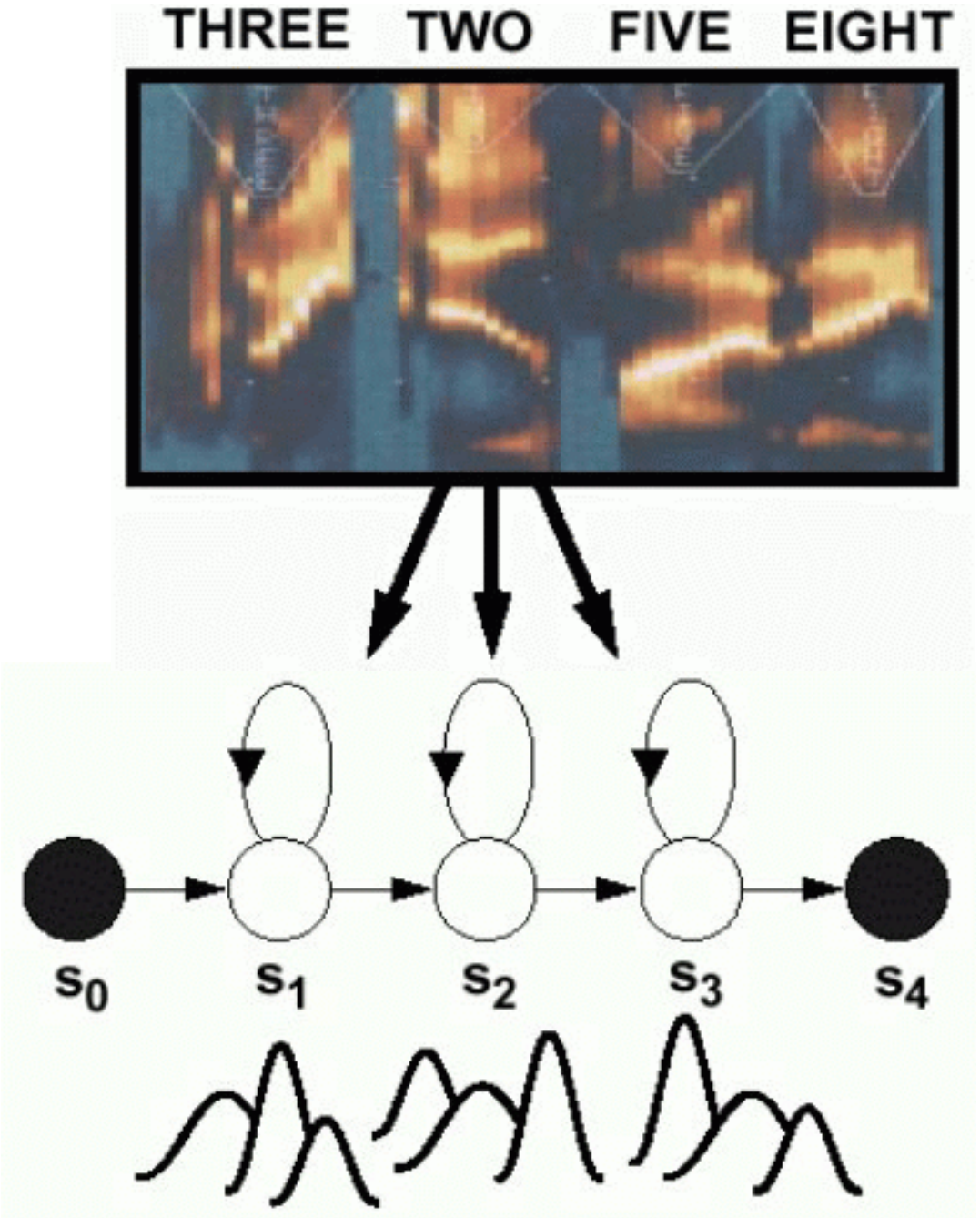
- Model the temporal progression of the speech signal
- Model the acoustic characteristics of sub-word units
- Account for variations in speaker characteristics (speaker-independent)

Acoustic Models Should:

- Optimally trade-off discrimination and representation
- Make efficient use of parameters
- Produce confidence measures of their predictions for higher-level decision processes

PRIOR ART: HMMs

- Acoustic models encode the temporal evolution of the features (spectrum).
- Gaussian mixture distributions are used to account for variations in speaker, accent, and pronunciation.
- Sharing model parameters is a common strategy to reduce complexity.
- The goal of our research is to replace the Gaussian likelihood computation at each state with a machine that incorporates notions of:
  - **discrimination (“one vs. all”)**
  - **Bayesian statistics (priors)**
  - **confidence**
  - **sparsity**
- Maintain computational efficiency?



PRIOR ART: HMMs

- Data-driven modeling supervised only from a word-level transcription.
- The expectation/maximization (EM) algorithm is used to improve our estimates:

$$\log P(Data|\bar{\lambda}) \geq \log P(Data|\lambda)$$

if:

$$Q(\lambda, \bar{\lambda}) \geq Q(\lambda, \lambda)$$

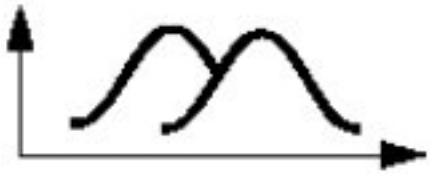
Approach: maximum likelihood estimation

- Computationally efficient training algorithms (Forward-Backward) have been crucial.
- Batch mode parameter updates are typically preferred.
- Decision trees are used to optimize sharing parameters, minimize system complexity, and integrate additional linguistic knowledge.

- Initialization

- Single Gaussian Estimation

- 2-Way Split

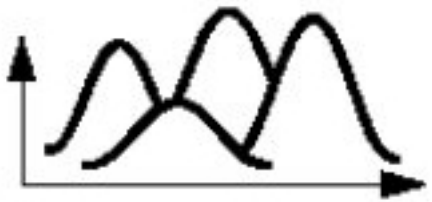


- Mixture Distribution Reestimation

- 4-Way Split

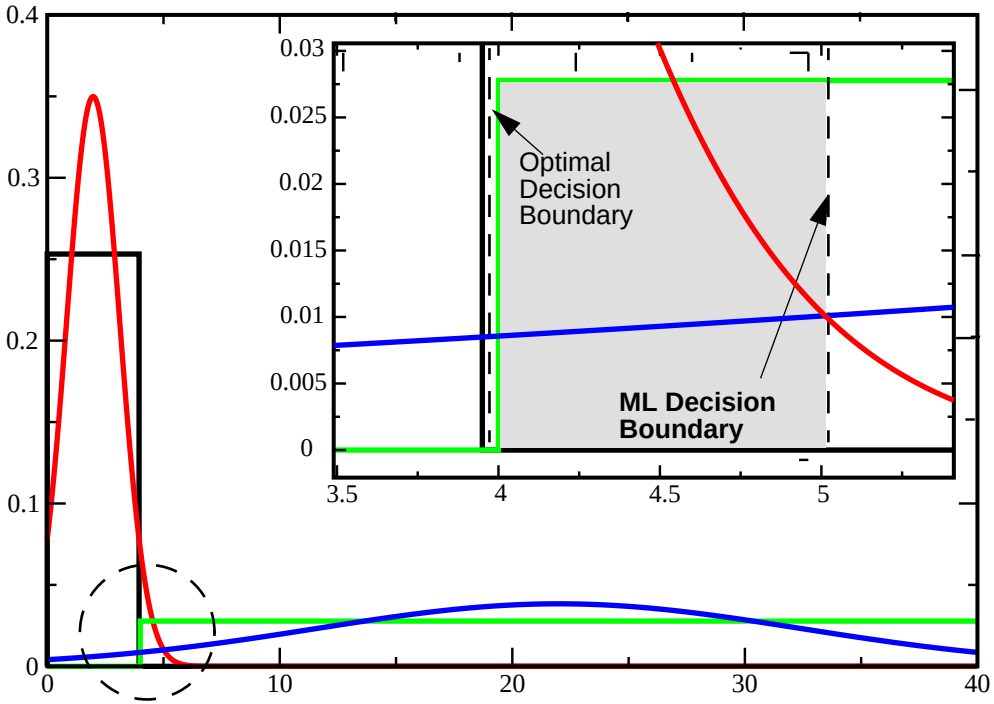
- Reestimation

...



PRIOR ART: HMMs

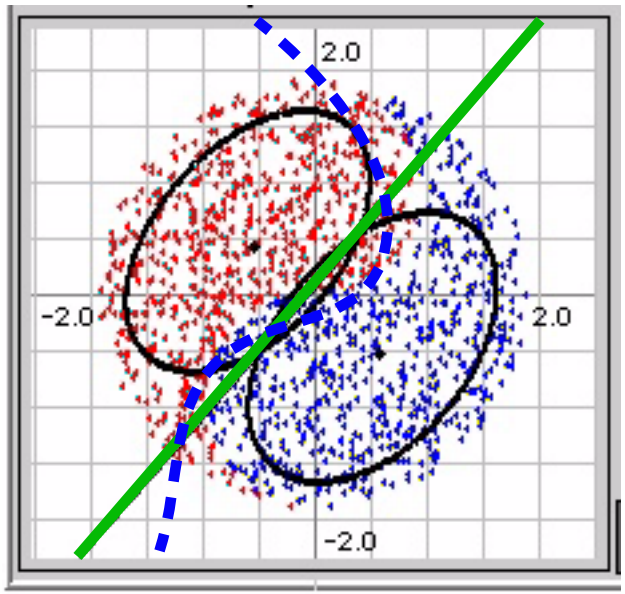
Convergence in maximum likelihood does not translate to optimal classification:



- Error results from fitting uniform distributions with Gaussians (and using an ML boundary).
- Since the classes are separable, finding the optimal decision surface is trivial.

Discrimination

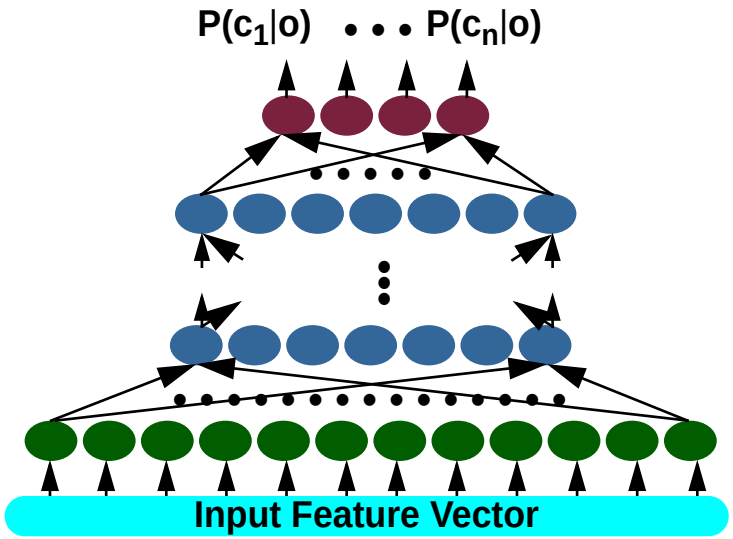
Class-Dependent PCA (MLE Gaussian)



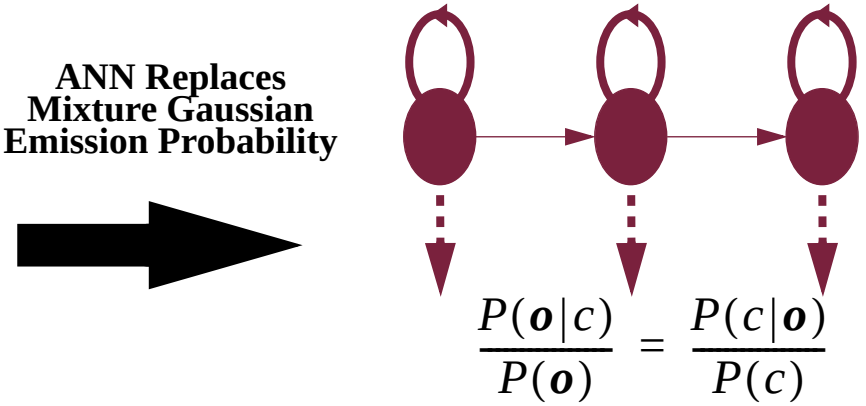
- Data not separable by a hyperplane (a nonlinear classifier is needed).
- Gaussian MLE models tend towards the center of mass (overtraining).

Solution: Nonlinear discriminative classifiers!
First Cut: Artificial Neural Networks

PRIOR ART: HMM/ANN HYBRIDS



ANN Replaces Mixture Gaussian Emission Probability



Architecture:

- ANN provides flexible, discriminative classifiers for emission probabilities that avoid the HMM independence assumptions (can use wider acoustic context).
- Trained using Viterbi iterative training (hard decision rule) or can be trained to learn Baum-Welch targets (soft decision rule).

Shortcomings:

- Prone to overfitting: require cross-validation to determine when to stop training. **Need a method for automatically penalizing overfitting!**
- No substantial recognition improvements over HMM/GMMs

RISK MINIMIZATION

- Expected Risk:

$$R(\alpha) = \int \frac{1}{2} |y - f(x, \alpha)| dP(x, y)$$

Not possible to estimate $P(x, y)$.

- Empirical Risk Minimization:

$$R_{emp} = \frac{1}{2l} \sum_{i=1}^{l} |y_i - f(x_i, \alpha)|$$

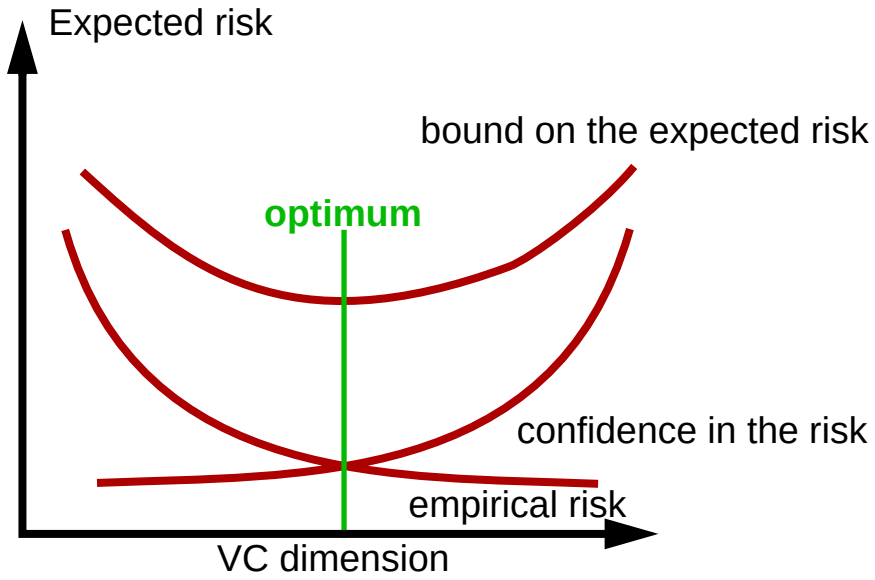
- Related by VC (Vapnik-Chervonenkis) dimension:

$$\boldsymbol{R(\alpha) \leq R_{emp}(\alpha) + f(h)}$$

$$f(h) = \sqrt{\frac{h(\log((2l/h) + 1)) - \log(\eta/4)}{l}}$$

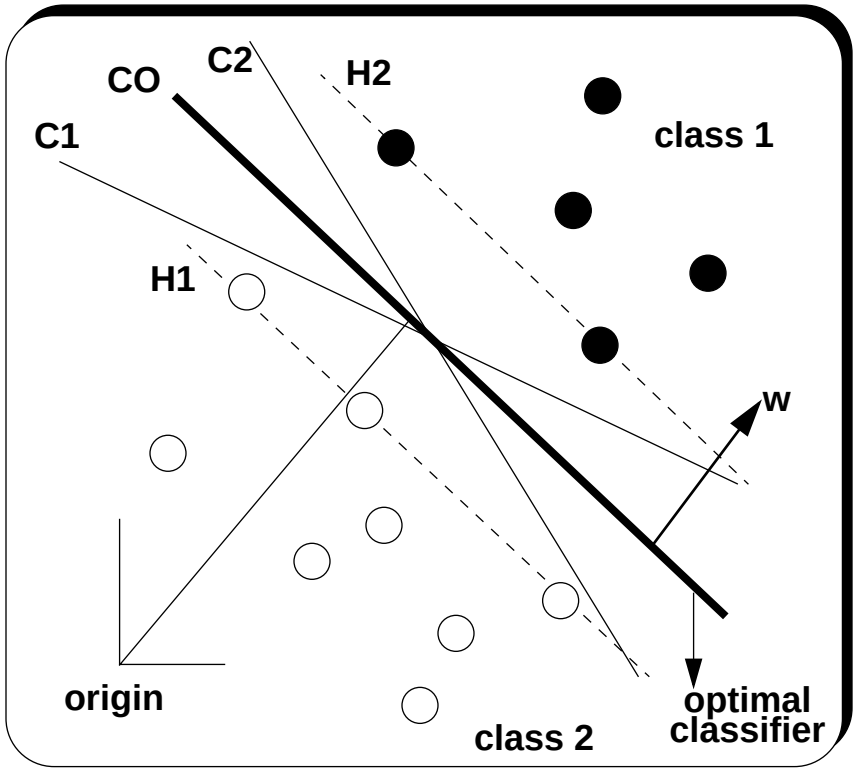
$f(h)$ is referred to as the VC confidence, η is a confidence measure $(0 \leq \eta \leq 1)$.

- Approach: **choose the machine that gives the least upper bound on actual risk**



- The VC dimension, h is a measure of the capacity of the learning machine.
- Principle of structural risk minimization (SRM) (Vapnik, 1979) involves finding the subset of functions that minimizes the bound on the actual risk.
- Optimal hyperplane classifiers achieve zero empirical risk for linearly separable data.

SUPPORT VECTOR MACHINES



- Hyperplanes C0-C2 achieve perfect classification — zero empirical risk.
- C0 is optimal in terms of generalization.
- The data points that define the boundary are called support vectors.

Optimization (Separable Data)

- Hyperplane: $\boldsymbol{x} \cdot \boldsymbol{w} + b$
- Constraints:

$$y_i(\boldsymbol{x}_i \cdot \boldsymbol{w} + b) - 1 \geq 0 \qquad \forall i$$

The data points that satisfy the equality are called support vectors.

- Optimize:

$$L_P = \frac{1}{2}\|\boldsymbol{w}\|^2 - \sum_{i=1}^{N} \alpha_i y_i (\boldsymbol{x}_i \cdot \boldsymbol{w} + b) + \sum_{i=1}^{N} \alpha_i$$

- Minimization of this Lagrange functional minimizes risk criterion (maximizes margin).
- Final classifier:

$$f(\boldsymbol{x}) = \sum_{i=1}^{numSVs} \alpha_i y_i \boldsymbol{x}_i \cdot \boldsymbol{x} + b$$

SUPPORT VECTOR MACHINES

- Data for practical applications typically not separable using a hyperplane in the original input feature space

- Transform data to higher dimension where hyperplane classifier is sufficient to model decision surface

$$\Phi : \Re^n \rightarrow \Re^N$$

- Kernels used for this transformation

$$K(x_i, x_j) = \Phi(x_i) \cdot \Phi(x_j)$$

- Final classifier:

$$f(\boldsymbol{x}) = \sum_{i=1}^{numSVs} \alpha_i y_i K(\boldsymbol{x}, x_i) + b$$

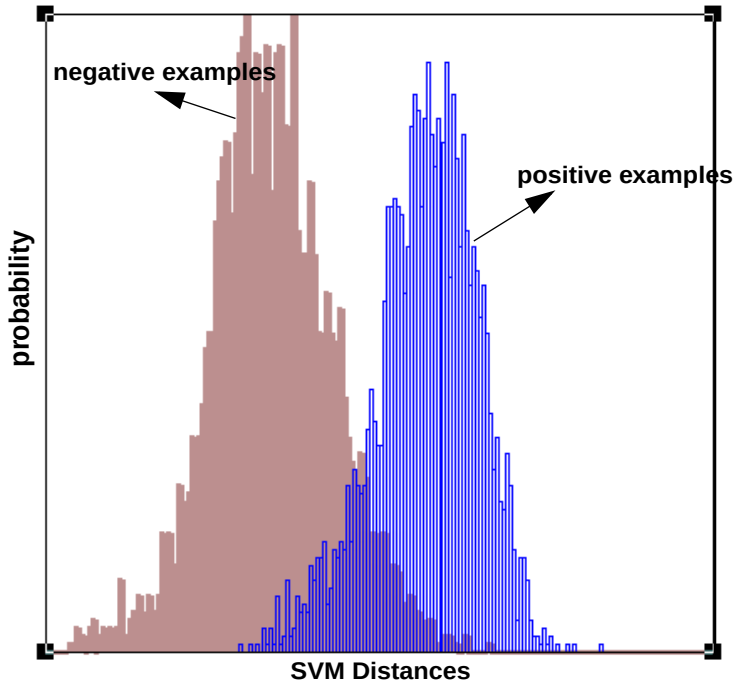
- Soft margin classifiers used in practice:

$$y_i(\boldsymbol{x}_i \cdot \boldsymbol{w} + b) \geq 1 - \xi_i \qquad \forall i$$

- SVMs do not generate likelihoods directly

- Posterior estimation required for speech

- Use a sigmoid function to map distances to posteriors:



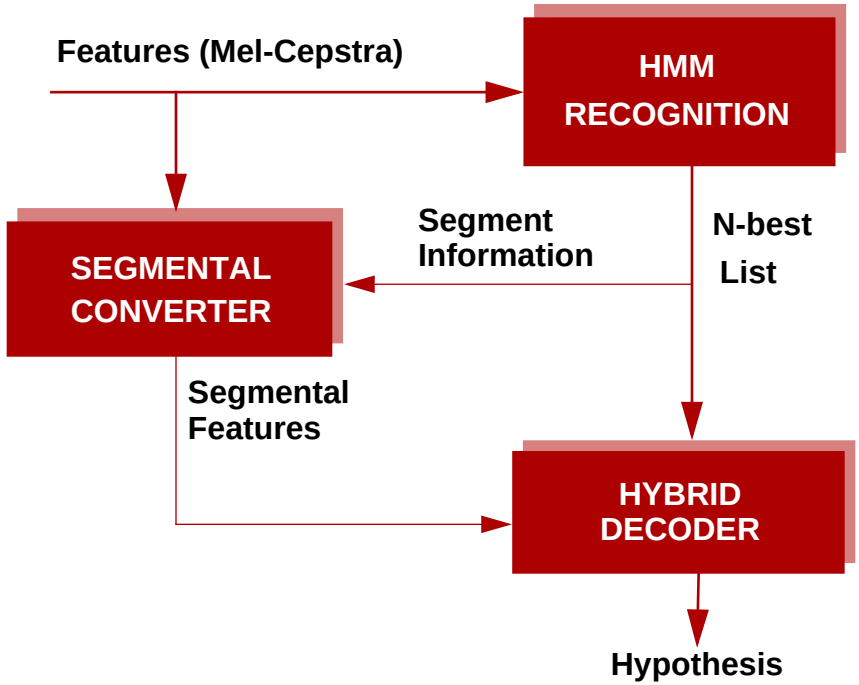
$$p(y = 1/f) = \frac{1}{1 + \exp(Af + B)}$$

SUPPORT VECTOR MACHINES

- Experimental Results: Deterding Vowel (11 vowels spoken in “h*d” context)

Approach	Error Rate
K-Nearest Neighbor	44%
Gaussian Node Network	44%
SVM: Polynomial Kernels	49%
SVM: RBF Kernels	35%
Separable Mixture Models	30%
RVM: RBF Kernels	30%

- A Hybrid Speech Recognition Framework



- Experimental Results: Continuous Speech

Information Source		HMM		Hybrid	
Transcription	Segmentation	AD	SWB	AD	SWB
N-best	Hypothesis	11.9	41.6	11.0	40.6
N-best	N-best	12.0	42.3	11.8	42.1
N-best + Ref.	Reference	—	—	3.3	5.8
N-best + Ref.	N-best + Ref.	11.9	38.6	9.1	38.1

- Rescore N-best lists using phone classifiers
- Use a segmental modeling approach for phone classifiers
- 10.6% on AD task using hybrid system that combines HMM and SVM scores

BAYESIAN MODELING

- First level of inference:

$$P(\boldsymbol{w}|D, H_i) = \frac{P(D|\boldsymbol{w}, H_i)P(\boldsymbol{w}|H_i)}{P(D|H_i)}$$

$\boldsymbol{w}$: the set of adjustable parameters

D: data from which we make inferences

H_i: overall model

- Second level of inference:

$$\frac{P(H_1|D)}{P(H_2|D)} = \frac{P(D|H_1)P(H_1)}{P(D|H_2)P(H_2)}$$

if $P(H_1) = P(H_2)$, best model chosen by evaluating evidence $P(D|H_i)$.

- Evidence marginalized across model parameters:

$$P(D|H_i) = \int P(D|\boldsymbol{w},H_i)P(\boldsymbol{w}|H_i)d\boldsymbol{w}$$

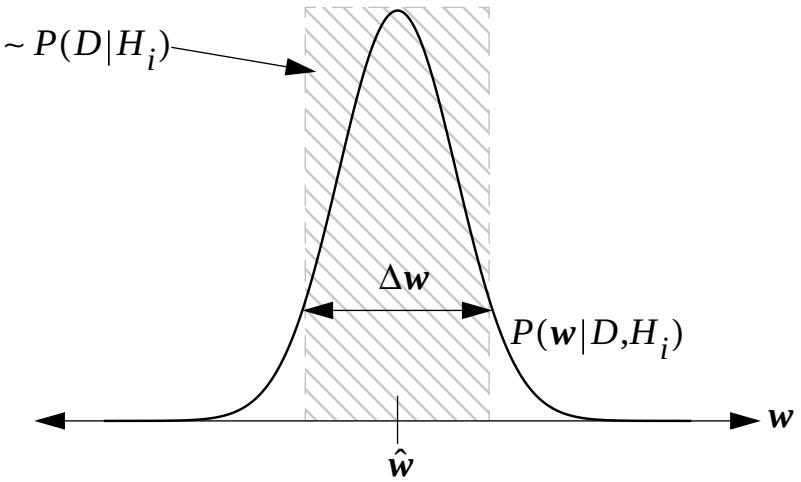
- It is impractical to compute this integral, so we need an approximation.

- Under the assumption that the posterior probability is Gaussian:

$$P(\boldsymbol{w}|D,H_i) \approx P(D|\boldsymbol{w},H_i)P(\boldsymbol{w}|H_i)$$

- The marginalization integral can be assumed to have a strong peak at the most probable value of the parameters, $\hat{\boldsymbol{w}}$.

- The evidence can then be approximated by multiplication of the height of the integrand and the width of the posterior, $\Delta\boldsymbol{w}$.



- Evidence approximation for a single model (Gaussian assumption)

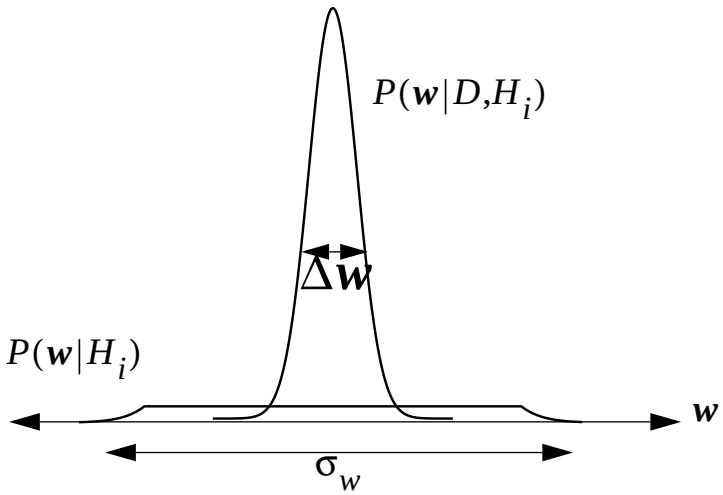
EVIDENCE FRAMEWORK

- The evidence is approximated by

$$P(D|H_i) \approx P(D|\hat{\boldsymbol{w}},H_i)P(\hat{\boldsymbol{w}}|H_i)\Delta \boldsymbol{w}$$

$P(D|\hat{\boldsymbol{w}},H_i)$ is the likelihood of the data given the best-fit parameter set

$P(\hat{\boldsymbol{w}}|H_i)\Delta \boldsymbol{w}$ is a penalty on the range of [0,1] which measures how well our posterior model fits our prior assumptions.



- The parameter's prior distribution and the posterior distribution width determine the model complexity

- The objective in training:

$$(\hat{\boldsymbol{w}}, \hat{\alpha}) = \underset{\boldsymbol{w},\, \alpha}{\text{argmax}} \; p(\boldsymbol{w}, \alpha|\boldsymbol{t}, \boldsymbol{O})$$

- Using Bayes' rule:

$$p(\boldsymbol{w}, \alpha|\boldsymbol{t}, \boldsymbol{O}) = \frac{p(\boldsymbol{t}|\boldsymbol{w}, \alpha, \boldsymbol{O})\, p(\boldsymbol{w}, \alpha|\boldsymbol{O})}{p(\boldsymbol{t}|\boldsymbol{O})}$$

- A closed form solution to this maximization is not possible.
- An iterative approximation has been developed by MacKay that has complexity $O(N^3)$ and is based on Gaussian assumptions. Not feasible for large speech recognition tasks.
- This approach is similar to Minimum Description Length (MDL) and Bayesian Information Criterion (BIC).

RELEVANCE VECTOR MACHINES

Drawbacks of SVMs:

- Complexity scales linearly with the training data for nontrivial problems (prohibitive for large speech recognition tasks).
- Sparsity of the model should be explicit in the optimization of the model.
- Need a posterior probability, not distance.
- The sigmoid approximation tends to overestimate confidence (Tipping).

Relevance Vector Machines:

- A kernel-based learning technique.
- A Bayesian approach (MacKay) that incorporates an automatic relevance determination (ARD) prior over each model parameter.
- RVMs typically require an order of magnitude less parameters than SVMs, but require significantly more training time.

- As with SVMs, the RVMs are formed by defining a vector-to-scalar mapping:

$$y(\boldsymbol{o};\boldsymbol{w}) = w_o + \sum_{i=1}^{M} w_i \phi_i(\boldsymbol{o}) = \boldsymbol{w}^T \phi(\boldsymbol{o})$$

- RVMs take a Bayesian approach and explicitly define an ARD prior distribution over the weights:

$$p(\boldsymbol{w}|\alpha) = \prod_{i=0}^{N} N\left(w_i|0, \frac{1}{\alpha_i}\right) = \frac{1}{\sqrt{(2\pi)^{N+1}|\boldsymbol{A}^{-1}|}} e^{-\frac{1}{2}\boldsymbol{w}^{\mathrm{T}}\boldsymbol{A}\boldsymbol{w}}$$

- To complete the Bayesian specification of the model, we use a non-informative (flat) prior for α_i.
- The likelihood of the training data set can be written as:

$$P(\boldsymbol{t}|\boldsymbol{w}, \boldsymbol{O}) = \prod_{n=1}^{N} \sigma_n^{t_n} (1-\sigma_n)^{1-t_n}$$

where $\sigma_n = \sigma\{y(\boldsymbol{o}_n;\boldsymbol{w})\}$.

SVM / RVM COMPARISON

Support Vector Machines

Data:

Class labels: {-1,+1}; “one vs. all”

Goal:

Find decision surface that maximizes the margin between two classes

Training:

Adjust parameters under constraint:

$$y_i(\boldsymbol{x}_i \cdot \boldsymbol{w} + b) \geq 1 - \xi_i \qquad \forall i$$

Optimize:

$$L_P = \frac{1}{2}\|\boldsymbol{w}\|^2 - \sum_{i=1}^{N} \alpha_i y_i(\boldsymbol{x}_i \cdot \boldsymbol{w} + b) + \sum_{i=1}^{N} \alpha_i$$

Training Complexity: $O(N^2)$

Classification: Threshold decoding (0.0)

Decoding:

- Rescoring N-best lists
- Segmental models

Relevance Vector Machines

Data:

Class labels: {0,1}; “one vs. all”

Goal: Learn posterior, $P(t|\boldsymbol{x})$.

Training:

$$y(\boldsymbol{x}) = \boldsymbol{w} \cdot \boldsymbol{x} + b$$

$$P(t|\boldsymbol{x}) = \frac{1}{1 + e^{-y(\boldsymbol{x})}}$$

$$p(\boldsymbol{w}|\alpha) = \prod_{i=0}^{N} N\left(w_i|(\mu_i = 0), \frac{1}{\alpha_i}\right)$$

find: $\underset{\bar{\boldsymbol{w}},\, \bar{\alpha}}{\operatorname{argmax}} P(\boldsymbol{w},\alpha|[t],[x])$

iteratively find $\hat{\boldsymbol{w}}|\alpha$ then $\hat{\alpha}|\hat{\boldsymbol{w}}$.

Training Complexity: $O(N^3)$

Classification: Threshold decoding (0.5)

Decoding: Integrated likelihood computation

RESEARCH PLAN

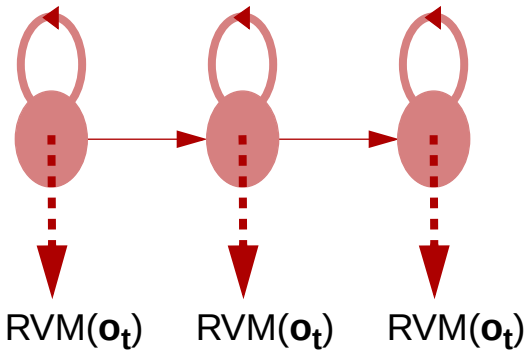
Current:

- Two-pass decoding methodology
- Ad hoc method for determining optimal segmentation
- Ad hoc method for determining posterior probability
- Lacks iterative training

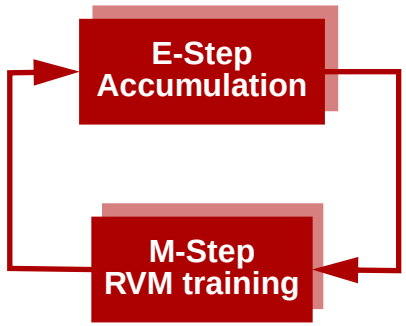
Proposed:

- Integrated HMM/RVM solution
- Frame-based modeling: eliminates need for segmental models
- RVM is naturally probabilistic: eliminates need for sigmoid posterior fit

Proposed Architecture:



HMMs with RVM Emission Distributions



Iterative Parameter Estimation Similar to Baum-Welch Training of HMMs

- Convergence properties and efficient training methods are critical.
- Bootstrapping or incremental training
- Available as part of the ISIP speech recognition toolkit.

EXPERIMENTS

- Experimental Results: Deterding Vowel (11 vowels spoken in “h*d” context)

Approach	Error Rate
K-Nearest Neighbor	44%
Gaussian Node Network	44%
SVM: Polynomial Kernels	49%
SVM: RBF Kernels	35%
Separable Mixture Models	30%
RVM: RBF Kernels	30%

Approach	Avg. Parameter Count
SVM: RBF Kernels	83 SVs
RVM: RBF Kernels	13 RVs

- RVMs yield superior sparsity with comparable generalization.

- Experimental Results: OGI Alphadigits

Approach	Error Rate	Avg. Parameter Count	Training Time	Testing Time
SVM	16.4%	257 SVs	1/2 hour	30 mins
RVM	16.2%	12 RVs	1 month	1 min

- Experimental Results: SWB

Approach	Error Rate	Avg. Parameter Count
HMM	41.6%	---
SVM	40.8%	1213 SVs
RVM	41.2%	178 RVs

- Reduced training set size

- Computational cost mainly in training, but is still prohibitive for large data sets.

RESEARCH PLAN

Practical optimization methods

- Currently $O(N^2)$ in memory and $O(N^3)$ in time - prohibitive for large data sets.
- Explore methods for incremental learning: Active learning (MacKay) or decomposition (Tipping)

Integrated, iterative HMM/RVM training

- RVM replaces Gaussian
- E-M style training paradigm
- Need to address issues such as convergence and parameter tying

Integrated HMM/RVM decoder

- Single-pass decoding
- Parameter tuning necessary

Experimental Progression

Task	System	Data	Date
Static Classification	Set of 1-vs-All classifiers	Deterding Vowel	March
Pilot ASR	Hybrid HMM/RVM (same as SVM) 2000 training	Alphadigit	March
Practical Optimization	Set of 1-vs-All classifiers	Deterding Vowel	April
	Hybrid HMM/RVM Full training	Alphadigit	June
HMM/RVM Training and Decoding	Frame-based models Full training. First with single frame feature vectors, then with extended feature set	TIDigits 13000 train, 13000 test	July
		Alphadigits 60k train 3300 test	Aug
		SWB 114k train 2400 test	Sep

PUBLICATIONS

Accepted:

J. Hamaker, A. Ganapathiraju and J. Picone, "A Sparse Modeling Approach to Speech Recognition Based on Relevance Vector Machines," *International Conference of Spoken Language Processing*, Denver, Colorado, USA, September 2002.

A. Ganapathiraju, *Support Vector Machines for Speech Recognition*, Ph.D. Dissertation, Department of Electrical and Computer Engineering, Mississippi State University, January 2002.

Pending:

J. Hamaker, *Sparse Bayesian Methods for Continuous Speech Recognition*, Ph.D. Dissertation, Department of Electrical and Computer Engineering, Mississippi State University, December 2002.

B. Jelinek, *Support Vector Machine-Based Speech Recognition*, M.S. Thesis, Department of Electrical and Computer Engineering, Mississippi State University, August 2002.

A. Ganapathiraju, J. Hamaker and J. Picone, "Continuous Speech Recognition Using Support Vector Machines," submitted to Computer Speech and Language, October 2001.

REFERENCES

ASR: General

[1] L.R. Rabiner and B.H. Juang, *Fundamentals of Speech Recognition*, Prentice Hall, Englewood Cliffs, New Jersey, USA, 1993.

[2] R. Cole, J. Mariani, H. Uszkoreit, G. B. Varile, A. Zaenen, A. Zampolli, and V. Zue, eds., *Survey of the State of the Art in Human Language Technology*, Chapter 9, Cambridge University Press, Cambridge, Massachusetts, USA, March 1998.

[3] J. Picone, "Signal Modeling Techniques in Speech Recognition," *IEEE Proceedings*, vol. 81, no. 9, pp. 1215-1247, September 1993.

[4] J. Picone, "Continuous Speech Recognition Using Hidden Markov Models", *IEEE Acoustics, Speech, and Signal Processing Magazine*, vol. 7, no. 3, pp. 26-41, July 1990.

[5] L. R. Rabiner, "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257-285, February 1989.

[6] N. Deshmukh, A. Ganapathiraju, J. Hamaker, M. Ordowski and J. Picone, "A Public Domain Speech-to-Text System," *Proceedings of. Eurospeech*, vol. 5, Budapest, Hungary, September 1999.

[7] N. Deshmukh, A. Ganapathiraju and J. Picone, "Hierarchical Search for Large Vocabulary Conversational Speech Recognition," *IEEE Signal Processing Magazine*, vol. 16, no. 5, pp. 84-107, September 1999.

ASR: ANN

[8] S. Renals, *Speech and Neural Network Dynamics*, Ph. D. dissertation, University of Edinburgh, UK, 1990.

[9] M.D. Richard and R.P. Lippmann, "Neural Network Classifiers Estimate Bayesian a Posteriori Probabilities", *Neural Computation*, vol. 3, no. 4, pp. 461-483, 1991.

[10] H.A. Bourlard and N. Morgan, *Connectionist Speech Recognition — A Hybrid Approach*, Kluwer Academic Publishers, Boston, USA, 1994.

Support Vector Machine

[11] V.N. Vapnik, *Statistical Learning Theory*, John Wiley, New York, NY, USA, 1998.

[12] C. Cortes, V. Vapnik. Support Vector Networks, *Machine Learning*, vol. 20, pp. 293-297, 1995.

[13] C.J.C. Burges, A Tutorial on Support Vector Machines for Pattern Recognition, http://svm.research.bell-labs.com/SVMdoc.html, AT&T Bell Labs, November 1999.

Hybrid HMM/SVM

[14] A. Ganapathiraju, *Support Vector Machines for Speech Recognition*, Ph.D. Dissertation, Mississippi State University, Mississippi State, Mississippi, USA, 2002.

[15] A. Ganapathiraju, J. Hamaker and J. Picone, "Support Vector Machines for Speech Recognition," *Proceedings of the International Conference on Spoken Language Processing*, Sydney, Australia, November 1998.

[16] A. Ganapathiraju, J. Hamaker and J. Picone, "A Hybrid ASR System Using Support Vector Machines," submitted to the *International Conference of Spoken Language Processing*, Beijing, China, October, 2000.

REFERENCES

[17] A. Ganapathiraju and J. Picone, "Hybrid SVM/HMM architectures for speech recognition," submitted to *Neural Information Processing Systems - 2000*, Denver, Colorado, USA, November 2000.

[18] A. Ganapathiraju, J. Hamaker and J. Picone, "Hybrid HMM/SVM Architectures for Speech Recognition," *Proceedings of the Department of Defense Hub 5 Workshop*, College Park, Maryland, USA, May 2000.

[19] A. Ganapathiraju, J. Hamaker and J. Picone, "Continuous Speech Recognition Using Support Vector Machines," submitted to *Computer, Speech and Language*, November 2001.

[20] Ostendorf, M., Digalakis, V. and Kimball, O (1996). From HMM's to Segment Models: A Unified View of Stochastic Modeling for Speech Recognition. *IEEE Transactions on Speech and Audio Processing*, vol. 4, no. 5, pp. 360-378.

[21] J. Kwok, "Moderating the Outputs of Support Vector Machine Classifiers," *IEEE Transactions on Neural Networks*, vol. 10, no. 5, 1999.

Bayesian Methods

[22] E. T. Jaynes, "Bayesian Methods: General Background," *Maximum Entropy and Bayesian Methods in Applied Statistics*, J. H. Justice, ed., pp. 1-25, Cambridge University Press, Cambridge, UK, 1986.

[23] H. Jeffreys, *Theory of Probability*, Oxford University Press, 1939.

[24] T. J. Loredo, "From Laplace to Supernova SN 1987A: Bayesian Inference in Astophysics," *Maximum Entropy and Bayesian Methods*, P. Fougere, ed, Kluwer Publishing, 1989.

[25] S. F. Gull, "Bayesian Inductive Inference and Maximum Entropy," *Maximum Entropy and Bayesian Methods in Science and Engineering, vol. 1, Foundations*, G. J. Erickson and C. R. Smith, eds., Kluwer Publishing, 1988.

[26] D. J. C. MacKay, *Bayesian Methods for Adaptive Models*, Ph. D. thesis, California Institute of Technology, Pasadena, California, USA, 1991.

[27] D. J. C. MacKay, "Probable networks and plausible predictions --- a review of practical Bayesian methods for supervised neural networks," *Network: Computation in Neural Systems*, 6, pp. 469-505, 1995.

Relevance Vector Machine

[28] M. Tipping, "Sparse Bayesian Learning and the Relevance Vector Machine," *Journal of Machine Learning*, vol. 1, pp. 211-244, June 2001.

[29] M. Tipping, "The Relevance Vector Machine," in *Advances in Neural Information Processing Systems 12*, S. Solla, T. Leen and K.-R. Muller, eds., pp. 652-658, MIT Press, 2000.

[30] C. Bishop and M. Tipping, "Variational Relevance Vector Machines," *Proceedings of the 16th Conference in Uncertainty in Artificial Intellignece*, pp. 46-53, Morgan Kaufmann Publishers, 2000.

[31] A. Faul and M. Tipping, "Analysis of Sparse Bayesian Learning," *Proceedings of the Conference on Neural Information Processing Systems*, preprint, 2001.

[32] S. Chen, S. R. Gunn and C. J. Harris, "The Relevance Vector Machine Technique for Channel Equalization Applications," *IEEE Transactions on Neural Networks*, vol. 12, pp. 1529-1532, 2001.

REFERENCES

[33] J. B. Gao, S. R. Gunn, C. J. Harris and M. Brown, "Regression with Input-dependent Noise: a Relevance Vector Machine Treatment," *IEEE Transactions on Neural Networks*, March 2001.

Corpora

[34] D. Deterding, M. Niranjan and A. J. Robinson, "Vowel Recognition (Deterding data)," Available at *http://www.ics.uci.edu/pub/machine-learning-databases/undocumented/connectionist-bench/vowel, 2000.*

[35] R. G. Leonard, "A Database for Speaker Independent Digit Recognition," *Proceedings of the International Conference for Acoustics, Speech and Signal Processing*, vol. 3, pp. 42-45, San Diego, California, USA, 1984.

[36] R. Cole, "Alphadigit Corpus v1.0". *http://www.cse.ogi.edu/CSLU/corpora/alphadigit*, Center for Spoken Language Understanding, Oregon Graduate Institute, USA, 1998.

[37] J. Hamaker, A. Ganapathiraju, J. Picone and J. Godfrey, “Advances in Alphadigit Recognition Using Syllables,” *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 421-424, Seattle, Washington, USA, May 1998.

[38] J. Godfrey, E. Holliman and J. McDaniel, "SWITCHBOARD: Telephone Speech Corpus for Research and Development," *Proceedings of the International Conference on Acoustics, Speech and Signal Processing*, vol. 1, pp. 517-520, San Francisco, California, USA, 1992.

[39] A. Ganapathiraju et. al., “WS97 Syllable Team Final Report,” P*roceedings of the 1997 LVCSR Summer Research Workshop*, Center for Language and Speech Processing, Johns Hopkins University, Baltimore, Maryland, USA, December 1997.

A COMPLETE SVM-BASED DECODER

Previous Experiments

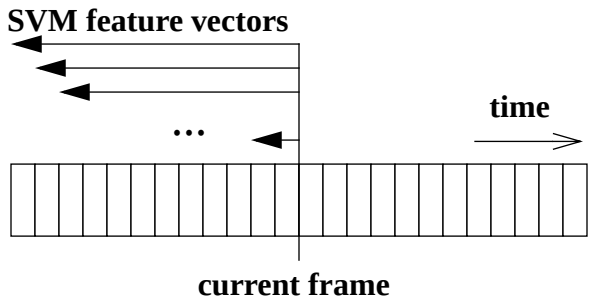
- Hybrid HMM/SVM decoder needs HMM to provide time alignment of phone hypotheses
- We prefer a method where the HMM is taken out of the loop

SVM Decoder Paradigm

- SVM is presented with all possible phone segmentations and must choose the most likely one
- Trained SVM models are able to generate estimates of posterior phone probabilities given the segmental features
- Initial cut attempted to build on a time-synchronous search paradigm (i.e. Viterbi Search)

SVM Decoder Implementation

- Direct replacement of HMM model by SVM model in ISIP prototype time-synchronous decoder
- Segmental features computed at the beginning of each time frame for all possible segment start frames



- Problem: Viterbi decision can only be made once the phone ending has been hypothesized — paths can exist in the search space even though they will eventually have very poor scores

FIRST EXPERIMENTS

ALPHADIGITS: Full Grammar

System	WER
SVM	85.5%
HMM/SVM	12.6%
HMM	11.0%

- SVM error rate primarily due to deletions - all valid paths were pruned
- Required very narrow pruning thresholds due to computational requirements:
  ~1 Gigabyte of RAM for many utterances
  ~200 xRT on 1GHz processor
  ~perplexity = ~40
- Gave up after completing only ~2% of the data

ALPHADIGITS: 10-Best Lists

System	WER
SVM	14.2%
HMM/SVM	11.0%
HMM	11.1%

- Much smaller task: perplexity on the order of 3-4
- However, the memory and CPU time requirements are still too high.
  ~800 Megabytes of RAM
  ~12xRT on 800MHz processor
- Again, only completed small portion of the the data: ~30% of data

PRACTICAL CONSIDERATIONS

Conclusions:

- SVM decoder is too inefficient even on small perplexity tasks
- We need to be able to prune early and often in the search without pruning away good paths
- The posterior phone evaluations prevent efficient pruning in a time-synchronous search

Possible Solutions:

- Stack Search: Provides a method to estimate how likely it is that a path will eventually be a poor-scorer so that we may prune it early in the search process

Best-first (A* Search):

- Ordered-search algorithm utilizing evaluation function of the form

$$\hat{f}(n) = \hat{g}(n) + \hat{h}(n)$$

$\hat{f}(n)$: evaluation function for the node n, it estimates true $f(n)$

$f(n)$: actual cost of the optimal path from the start node to the goal node through the node n

$\hat{g}(n)$: cost of the path to the node n followed so far, therefore $\hat{g}(n) \geq g(n)$

$\hat{h}(n)$: estimates the cost of remaining path from the node n to the goal node

- Best-first search is admissible (able to find optimal path if it exists) if the condition $\hat{h}(n) \leq h(n)$ is true [Nilsson; Hart]

STACK DECODING

Stack Decoding

- Tree based time-asynchronous A* search
- Evaluation function

$$f(H_N^t) = g(H_N^t) + h(H_N^{t,N})$$

is based on the forward probability [Jelinek; Huang]

$g(H_N^t)$: evaluation function for the partial path of H_N up to time t

$h(H_N^{t,N})$: heuristic function of the remaining path from $t+1$ to T for the path H_N

Heuristic Estimates:

$h(*)$ can be estimated from the training set data as the lower bound on the frame cost

- Almost admissible A* criterion [Paul]:

$$\Lambda_i(t) = L_i(t) - lubL(t)$$

$\Lambda_i(t)$: *A** evaluation function

$L_i(t)$: log likelihood of hypothesis

$lubL(t)$: least upper bound on $L_i(t)$

$lubsfL(t)$: least upper bound found so far is often used to approximate $lubL(t)$

RESEARCH PLAN

Stack search implementation:

- Multistack search maintains a separate stack for each time *t.* Computes one word extension hypotheses of the best path

- Fast match gives rapid computation of a list of candidate hypotheses. Expansive detailed match can be done after the fast match [Bahl]

Iterative SVM training

- With a working SVM decoder, we can improve performance using a Viterbi-style trainer

Experimental Progression:

System	Data	Date
Viterbi-style SVM decoder	**Alphadigits**	**May**
Stack-based SVM Decoder	**Alphadigits**	**June**
	SWB	**July**
SVM trainer	**Alphadigits**	**July**
	SWB	**August**

Culminates in publication of an M.S. Thesis:

B. Jelinek, *Support Vector Machine-Based Speech Recognition*, M.S. Thesis, Department of Electrical and Computer Engineering, Mississippi State University, August 2002.

REFERENCES

Search

[1] N. J. Nilsson, *Problem Solving Methods in Artificial Intelligence,* McGraw-Hill, New York, USA, 1971.

[2] P. E. Hart, N. J. Nilsson, B, Raphael: “A Formal Basis for the Heuristic Determination of Minimum Cost Paths”, *IEEE Transactions of Systems Science and Cybernetics,* vol. SSC-4, no. 2, pp. 100-107, 1968.

[3] F. Jelinek, *Statistical Methods for Speech Recognition,* MIT, Cambridge, MA, USA, 1998.

[4] X. Huang, A. Acero, and H.W. Hon, *Spoken Language Processing - A Guide to Theory, Algorithm, and System Development,* Prentice Hall, NJ, USA, 2001.

[5] L. R. Bahl, “Obtaining Candidate Words by Polling in Large Vocabulary Speech Recognition System,” *Proceedings of the IEEE International Conference on Acoustic, Speech and Signal Processing*, New York, USA, April 1988.

[6] D. B. Paul, “An efficient A* stack decoder algorithm for continuous speech recognition with a stochastic language model”, *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing,* San Francisco, CA, USA, March 1992.