Annual Report for Period: 08/2001 - 08/2002
Submitted on: 06/28/2002
Principal Investigator: Picone, Joseph .
Award ID: 0085940
Organization: Mississippi State Univ
Title:
ITR: Information Access to Spoken Documents

Project Participants

Senior Personnel

Name: Picone, Joseph
Worked for more than 160 Hours: Yes
Contribution to Project:
improved spontaneous speech recognition performance using Support Vector Machines

Name: Ostendorf, Mari
Worked for more than 160 Hours: No
Contribution to Project:
integrating prosodic information in parsing spoken language

Name: Charniak, Eugene
Worked for more than 160 Hours: Yes
Contribution to Project:
integration of parsing in the retrieval of spoken documents

Name: Jelinek, Frederick
Worked for more than 160 Hours: Yes
Contribution to Project:

Name: Johnson, Mark
Worked for more than 160 Hours: Yes
Contribution to Project:
integration of parsing in the retrieval of spoken documents

Name: Khudanpur, Sanjeev
Worked for more than 160 Hours: Yes
Contribution to Project:
incorporating uncertainty in parsing to handle speech recognition errors

Name: Byrne, William
Worked for more than 160 Hours: No
Contribution to Project:
incorporating uncertainty in parsing to handle speech recognition errorsyti

Name: Hoffman, Thomas
Worked for more than 160 Hours: No
Contribution to Project:
enhancing information retrieval of spoken documents through the use of prosodic and other non-segmental information

Post-doc

Graduate Student

Name: Hamaker, Jonathan
Worked for more than 160 Hours: Yes
Contribution to Project:
improve recognition performance of spontaneous speech using Support Vector Machines

Name: Xu, Peng
Worked for more than 160 Hours: Yes
Contribution to Project:

Name: Emami, Ahmad
Worked for more than 160 Hours: Yes
Contribution to Project:

Name: Shafran, Izhak
Worked for more than 160 Hours: Yes
Contribution to Project:
Research on spectral correlates of prosody and disfluencies

Name: Weinschenk, Jeff
Worked for more than 160 Hours: Yes
Contribution to Project:
Research on prosody and parsing

Name: Jelinek, Bohumir
Worked for more than 160 Hours: Yes
Contribution to Project:
Mr. Jelinek has been developing improved ways to apply Support Vector Machines (SVMs) to speech recognition by using more exhaustive search techniques.

Undergraduate Student

Technician, Programmer

Name: Damon, Lee
Worked for more than 160 Hours: No
Contribution to Project:
computer systems administration

Other Participant

Name: Carmichael, Lesley
Worked for more than 160 Hours: Yes
Contribution to Project:
Prosodic labeling

Research Experience for Undergraduates

Organizational Partners

Other Collaborators or Contacts

Michael Tipping, Microsoft (U.K.): consultations on convergence issues in his iterative solution for finding relevance vectors.

Joachim Kohler, IMK, Fraunhofer (Germany): discussions of applications of our public domain software and RVM technology to information extraction for digital libraries.

Activities and Findings

Research and Education Activities: (See PDF version submitted by PI at the end of the report)

- Interaction of Speech and Parsing: investigated the way in which language phenomena which are very common in speech, but relatively rare in the formal text that parsing technology typically deals with, effects the parsing process.

- Lattice Generation Technology: developed lattice cutting techniques that transforms traditional word lattices into a series of segment sets that contain confusable words and phrases, thereby simplifying the search process during rescoring.

- Prosody and Parsing: developed categorical prosodic break labels, building on linguistic notions of minor and major prosodic phrases and the hesitation phenomena.

- Relevance Vector Machines: developed new reestimation techniques to make this approach feasible for large-scale system evaluations.

- Investigated the use of a parsing model as a language model

- Developed a hybrid speech recognition system that integrates a relevance vector machine for acoustic modeling.

- Investigated the development of an enhanced version of the Penn Treebank for supporting our research into prosody.

- Conducted a one-day project review at Johns Hopkins University at which we discussed approaches to parsing, modeling of prosody, and spontaneous speech recognition.

Findings:

- Language phenomena that are common in speech but not in text, such as filled pauses, appear not to make parsing easier, as initially conjectured.

- Verified that a neural network approach to language modeling can result in a perplexity reduction.

- Confirmed that that punctuation and repair points are correlated with prosodic breaks and hesitations. Silences alone (though useful) are not a reliable predictor of prosodic phrases.

- Demonstrated that the Relevance Vector Machine approach can result

in improved performance and decreased complexity as compared to traditional HMM and Support Vector Machines approaches. Further refinements to the training process are needed to make this technology practical.

- Two immedidate-head language models significantly reduce perplexity as compared to trigram and previous state-of-the-art grammar-based language models.

- Support vector machines provide a modest improvement over conventional HMM techniques on a task consisting of letters and numbers spoken over the telephone.

Training and Development:

Members of the team with expertise in speech recognition (JHU, Washington, MS State) have received training on modern approaches to parsing.

Members of the team with expertise in parsing (Brown) have received training on speech recognition and search.

Outreach Activities:

Journal Publications

E. Charniak, "Immediate-Head Parsing for Language Models", Proceedings of the 39th Annual Meeting of the Association for Computational Linguistics, p. 1, vol. 1, (2001). Published

S. Kumar and W. Byrne, "Risk Based Lattice Cutting for Segmental Minimum Bayes-Risk Decoding", Proceedings of the International Conference on Spoken Language Processing, p. , vol. , ( ). Accepted

S. Geman and M. Johnson, "Dynamic programming for parsing and estimation of stochastic unification-based grammars", Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, p. , vol. , ( ). Accepted

M. Johnson, "A simple pattern-matching algorithm for recovering empty nodes and their antecedents", Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, p. , vol. , ( ). Accepted

M. Ostendorf, I. Shafran, S. Shattuck-Hufnagel, B. Byrne and L. Carmichael, "A prosodically labeled database of spontaneous speech", Proceedings of the ISCA Workshop on Prosody in Speech Recognition and Understanding, p. 119, vol. , (2001). Published

A. Ganapathiraju, J. Hamaker and J. Picone, "Continuous Speech Recognition Using Support Vector Machines", Computer Speech and Language, p. , vol. , ( ). Submitted

A. Ganapathiraju, J. Hamaker, and J. Picone, "Advances in Hybrid SVM/HMM Speech Recognition", Proceedings of the International Conference of Spoken Language Processing, p. , vol. , ( ). rejected, to be resubmitted to another conference

J. Hamaker, J. Picone, and A. Ganapathiraju, "A Sparse Modeling Approach to Speech Recognition Based on Relevance Vector Machines", Proceedings of the International Conference on Spoken Language Processing, p. , vol. , ( ). Accepted

Books or Other One-time Publications

Web/Internet Site

URL(s):

http://www.isip.msstate.edu/projects/nsf_itr

Description:

This web site is used to disseminate information about the project, including publications, presentations, software, and data.

Other Specific Products

Product Type: Software (or netware)

Product Description:

The ISIP public domain speech recognition toolkit, developed under partial funding from a previous NSF grant, provides an easy to use state of the art speech recognition system that has been designed to facilitate rapid evaluation of new research. This software has been in release for over four years now.

In this project, we have extended this software to include libraries that implement our research into support vector machines and relevance vector machines. We have also extended our basic decoder to support an alternate search algorithm based on stack decoding, that is required by the new support vector machine technology.

Sharing Information:

All ISIP software is freely available on the Internet at the following URL:

http://www.isip.msstate.edu/projects/speech

This software is unrestricted, and can be used for both research and commercial development. There is currently an active user base of over 150 sites, and several companies are using our software as a reference implementation for their products.

Contributions

Contributions within Discipline:

- provided increased motivation for integrating parsing technology into speech recognition systems

- demonstrated the viability of risk mininzation techniques in speech recognition at the acoustic modeling level.

Contributions to Other Disciplines:

Contributions to Human Resource Development:

Contributions to Resources for Research and Education:

- developed new iterative estimation techniques that improve the convergence of our relevance vector machine technology. This has been incorporated into our toolkit.

- Added a stack search algorithm to our search library to extend the functionality of our public domain toolkit.

- integrated the relevance vector machine and support vector machine tools into our public domain speech recognition system, which is widely used in research and education. Two annual training workshops are held based on this software.

Contributions Beyond Science and Engineering:

Special Requirements

Special reporting requirements: None
Change in Objectives or Scope: None
Unobligated funds: less than 20 percent of current funds
Animal, Human Subjects, Biohazards: None

Categories for which nothing is reported:

Organizational Partners
Activities and Findings: Any Outreach Activities
Any Book
Contributions: To Any Other Disciplines
Contributions: To Any Human Resource Development
Contributions: To Any Beyond Science and Engineering

08/15/01 — 08/14/02: RESEARCH AND EDUCATIONAL ACTIVITIES

In the second year of this project, we focused our efforts in four areas:

- **Interaction of Speech and Parsing**: investigated the way in which language phenomena which are very common in speech, but relatively rare in the formal text that parsing technology typically deals with, effects the parsing process.
- **Lattice Generation Technology**: developed lattice cutting techniques that transforms traditional word lattices into a series of segment sets that contain confusable words and phrases, thereby simplifying the search process during rescoring.
- **Prosody and Parsing**: developed categorical prosodic break labels, building on linguistic notions of minor and major prosodic phrases and the hesitation phenomena.
- **Relevance Vector Machines**: developed new reestimation techniques to make this approach feasible for large-scale system evaluations.

These developments are described in more detail in the sections below.

A project meeting was held at Brown University on June 13, 2002 to coordinate the work on this project. All organizations involved in this project were present at this meeting. Our next joint project meeting is planned for early June'2003.

A. Laboratory for Linguistic Information Processing, Brown University

One of the research activities this last year on the interaction between speech and parsing was an investigation into the way in which language phenomena which are very common in speech, but relatively rare in the formal text that parsing technology typically deals with, effects the parsing process. In particular, both "filled pauses" ("ums" and "ahs") and parentheticals ("you know") are common in speech, but not text. Previous work in the area has shown that both tend to occur more readily at clause boundaries than elsewhere in sentences, leading to the conjecture that rather than making parsing more difficult, they might make things easier. Unfortunately, some recent experiments at Brown, to be presented at the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP) seem to suggest that this is not the case. A standard statistical parser was trained on text with and without such phenomena, and its performance was measured. It seems that filled pauses make parsing harder, parentheticals make parsing harder, and both together make it harder still. This goes against the prevailing expectations, and some recent suggestions by the prosody researchers within this project are going to be followed up upon in an effort to refine this result. In particular there is some evidence that "ah" and "um" might behave differently in this regard, and it might be worth distinguishing the two, something not done in this last year's experiments.

B. Center for Language and Speech Processing, Johns Hopkins University

CLSP concentrated on three activities this year: lattice cutting, neural probabilistic language modeling, and the impact of richer syntactic dependencies on the performance of the structured language model.

B.1 Lattice Cutting

CLSP has provided lattices for use in developing parsers automatically transcribe speech. Lattices were generated on the RT-02 (Rich Transcription 2002 Evaluation) development test set using a

conversational speech ASR system trained on the SWITCHBOARD corpus. These were provided in determinized acoustic score form, in that the language model scores used in generating the lattices were removed so that only the acoustic score of each individual word hypotheses remained. These were determinized using the AT&T FSM toolkit so that each lattice is compact and easy to search using any left-to-right language model.

A segmented version of these lattices were also produced using lattice cutting techniques developed at CLSP. Lattice cutting pinches word lattices so that lattices are transformed to look like a series of segment sets that confusable words and phrases. Lattice rescoring is changed from searching over entire sentences found in the original lattice to resolving the small number of confused words and phrases in the segment sets. A research avenue to explore is whether parsers can be modified to search over these smaller, more constrained sets.

B.2 Neural Probabilistic Language Modeling

The problem of language modeling research for ASR is essentially the problem of sparseness of data. Conventionally, it has been treated by smoothing of various kinds and lately by utilization of sentence structure. However, Bengio and his associates [1] have come up with a novel approach based on artificial neural networks (ANNs). We have confirmed their results by independent experimentation, as shown in Table 1.

Perplexity experiments were carried out on the Brown corpus and the UPenn section of the WSJ corpus. We will next (a) ascertain the ASR error rate effects obtainable from these improvements (b) apply the approach to improving components (i.e., the predictor) of a structural language model.

B.3 The Impact Richer Syntax Dependencies on the Structured Language Model

We studied the impact of richer syntactic dependencies on the performance of the structured language model (SLM) along two dimensions: perplexity (PPL) and word-error-rate (WER, N-best rescoring).

Under the equivalence classification in the SLM, the conditional information available to the SLM model components is made up of the two most-recent exposed heads consisting of two NT tags and two headwords. In an attempt to extend the syntactic dependencies beyond this level, we enriched the non-terminal tag of a node in the binarized parse tree with the NT tag of the parent node (PA), or the NT tag of the child node from which the headword is not being percolated (OP), or we added the NT tag of the third most-recent exposed head to the history of the CONSTRUCTOR component (h-2).

Without interpolating with the 3-gram, the opposite (OP) scheme performed the best, reducing the PPL of the baseline SLM by almost 5% relative. When the SLM is interpolated with the 3-gram, the

Corpus Subset	Baseline	Neural Network	Combined
Brown	366	257	N/A
UPenn	141	157	121

Table 1. A comparison of a neural network (NN) based language modeling technique to traditional methods. The baselines 3-gram interpolated and 5-gram Knesser-Ney interpolated respectively. For the combined case, the NN model was interpolated with the baseline using a constant weight of 0.5023.

h-2+opposite+parent scheme performed the best, reducing the PPL of the baseline SLM by 3.2%.

The h-2+opposite scheme achieved the best WER result, with a 0.4% absolute reduction over the performance of the opposite scheme. Overall, the enriched SLM achieves 10% relative reduction in WER over the 3-gram model baseline result. This scheme outperformed the 3-gram used to generate the lattices and N-best lists, without interpolating it with the 3-gram model.

We will continue to study additional changes in the SLM parametrization schemes.

C. Signal, Speech, and Language Interpretation Lab, University of Washington

Prosody can be thought of as the "punctuation" in spoken language, particularly the indicators of phrasing and emphasis in speech. Most theories of prosody have a symbolic (phonological) representation for these events, but a relatively flat structure. In English, for example, two levels of prosodic phrases are usually distinguished: intermediate (minor) and intonational (major) phrases [2]. While there is evidence that both phrase-level emphasis (or, prominence) and prosodic phrases provide information for syntactic disambiguation [7], the most important cue seems to be phrase structure. The acoustic correlates of prosody are continuous-valued, including fundamental frequency (F0), energy, and duration cues. Particularly at phrase boundaries, cues include significant falls or rises in fundamental frequency accompanied by word-final duration lengthening, energy drops and optionally a silent pause.

C.1 Data Annotation and Development

The usefulness of speech databases for statistical analysis is substantially increased when the database is labeled for a range of linguistic phenomena, providing the opportunity to improve our understanding of the factors governing systematic suprasegmental and segmental variation in word forms. Prosodic labels and phonetic alignments are available for some read speech corpora, but only a few limited samples of spontaneous conversational speech have been prosodically labeled. To fill this gap, substantial samples of the Switchboard corpus of spontaneous telephone quality dialogs were labeled using a simplification of the ToBI system for transcribing pitch accents, boundary tones and prosodic constituent structure [6]. The aim was to cover phenomena that would be most useful for automatic transcription and linguistic analysis of conversational speech. This effort was initiated under another research grant, but completed with partial support from this NSF ITR grant.

The corpus is based on about 4.7 hours of hand-labeled conversational speech (excluding silence and noise) from 63 conversations of the Switchboard corpus and 1 conversation from the CallHome corpus. The orthographic transcriptions are time-aligned to the waveform using segmentations available from Mississippi State University (*http://www.isip.msstate.edu/projects/switchboard/index.html*) and a large vocabulary speech recognition system developed at the JHU Center for Language and Speech Processing for the 1997 Language Engineering Summer Workshop (www.clsp.jhu.edu/ws97) [3]. All conversations were analyzed using a high quality pitch tracker [9] to obtain F0 contours, then post-processed to eliminate errors due to crosstalk. The prosody transcription system included: i) breaks (0-4), which indicate the depth of the boundary after each word; ii) phrase prominence (none, *, *?); and iii) tones, which indicate syllable prominence and tonal boundary markers. It also provides a way to deal with the disfluencies that are common in spontaneous conversational speech (a p diacritic associated with the prosodic break), and to indicate labeller uncertainty about a particular transcription. The

annotation does not include accent tones, primarily to reduce transcription costs and because we hypothesized that the phrase tones would be relevant to dialog act representations which may be relevant for question-answering.

C.2 PROSODY AND PARSING

Our approach is to integrating prosody in parsing is to use categorical prosodic break labels, building on linguistic notions of minor and major prosodic phrases and the hesitation phenomena. An important reason for using categorical units rather than the acoustic correlates themselves is that the intermediate representation simplifies training with high-level (sparse) structures. Just as most ASR systems use a small inventory of phones as an intermediate level between words and acoustic feature to have robust word models (especially for unseen words), the prosodic breaks are also useful as a mechanism for generalizing the large number of continuous-valued acoustic features to different parse structures. In addition, the low-dimensional discrete representation is well suited to integration with current parsing frameworks, either as added “words” or as features on words. This approach is currently somewhat controversial because of the high cost of prosodic labeling, and to some extent because of the association with a particular linguistic theory. The the specific subset of labels used in this work, though founded in the ToBI system, collapse some of the detail of the system to simply represent minor and major phrases and disfluencies, so in fact the categories are relatively theory neutral (and language independent). Furthermore, a key objective of this work is to overcome the cost of prosodic labeling by using bootstrapping techniques.

Specifically, a small set of labeled data is used to train an automatic prosody annotation algorithm that has both text and acoustic cues. These cues are used in combination to automatically label the rest of the Switchboard data, and then new (separate) prosody-parse and prosody-acoustic models are designed for the final system, building on EM or co-training techniques. An alternative approach, as in [4], is to assign categorical “prosodic” labels defined in terms of syntactic structure and presence of a pause, without reference to human perception, and automatically learn the association of other prosodic cues with these labels. While this approach has been very successful in parsing speech from human-computer dialogs, we expect that it will be problematic for Switchboard because of the longer utterance and potential confusion between fluent and disfluent pauses.

The automatic prosody annotation effort is described further in the next section; here we briefly outline the key research issues for designing and integrating the different model components for the parsing application. Building on the two-stage approach introduced in [10], the planned architecture involves prosodic break detection and generation of an augmented word transcription, followed by detection of edit points and disfluent regions, and finally parsing. Key research issues include whether the prosodic breaks should be treated as “words” or as features on words, whether disfluencies should be represented as an independent component, and how to represent uncertainty of the prosodic classifier.

C.3 AUTOMATIC LABELING OF PROSODIC STRUCTURE

An important part of the past year’s effort was development of a prosodic labeling system in order to increase the effective training data, and for analysis of the dependence between prosodic and parse structures in conversational speech. Experiments were conducted on the Switchboard

corpus, specifically the prosodically annotated subset described previously. We used decision tree classifiers with a combination of acoustic, punctuation, parse, and disfluency cues. In this initial study, the only acoustic cue was silence duration; work with F0, energy and duration cues is in progress. The punctuation, parse and disfluency cues were taken from the annotations available from the Linguistic Data Consortium, aligned to the word transcriptions as described above. Speaker turn and incomplete sentence markers were among these cues. The disfluency file included repair points as well as markers of filled pauses and coordinating conjunctions used in utterance initial position. The parse features included right-to-left and left-to-right relative depth, part-of-speech, label of the left and right syntactic constituents, and parenthetical markers.

The baseline system assigns the most likely class (default word boundary) in all cases except for assigning a major phrase boundary at pause locations. This strategy gives 74% accuracy. Using disfluency, parse features and silence duration features improves performance to 86% accuracy. The most important features are punctuation, silence duration, disfluency edit point markers, left-to-right depth of the parse tree, and part-of-speech tags. The tree was quite simple (8 nodes). We anticipate further performance gains with the use of duration lengthening and intonation cues.

From analysis of the resulting tree, we find that punctuation and repair points are correlated with prosodic breaks and hesitations, as expected. Silences alone (though useful) are not a reliable predictor of prosodic phrases, since there are many prosodic phrases that do not occur at silences and because silences are frequently associated with hesitations. The depth (or possibly the length) of the left constituent is a useful predictor, but labels of the neighboring constituents do not. Further investigation of representations of syntactic constituent labels is ongoing, since other studies have shown association of clause boundaries (e.g. constituent labels) with major prosodic breaks. We find that minor phrase boundaries are never predicted by the decision trees designed in these experiments. Although this is not entirely surprising, since minor phrases are rare, we think that it may be possible to distinguish these structures given duration lengthening and/or intonation cues, in which case analysis of the full Switchboard corpus could show some relationship between minor phrases and particular syntactic constituents.

C.4 PROSODY AND ACOUSTIC MODELING

Most research on the use of prosody in automatic speech processing has focused on F0, energy and duration correlates to prosodic structure. However, there is evidence from long standing acoustic, articulatory and perceptual studies of speech suggesting that there are spectral correlates as well. For that reason, we conducted an analysis of our prosodically labeled conversational speech data using acoustic parameters and clustering techniques that are standard in speech recognition. We found that prosodic factors are associated with acoustic differences that can be learned in standard speech recognition systems. Both prosodic phrase structure and phrasal prominence seem to provide distinguishing cues, with some phones being affected much more than others (as one would expect from the linguistics literature). We hypothesized that we would find that constituent onsets were important at all levels (syllable, word and prosodic phrase). Instead, we found that onset is more important for syllables, but constituent-final position is more important at higher levels. Prosodic prominence had a smaller affect than phrase structure in terms of increasing likelihood of the training data, but seemed to result in more separable models when it did play a role. Finally, we found evidence that segmental cues can help distinguish fluent from disfluent phrase boundaries, in that segments associated with these categories are frequently

placed in different clusters. These differences can be leveraged in a "multiple pronunciation" acoustic model to aid in detecting fluent vs. disfluent prosodic boundaries, though additional prosodic cues are necessary to separate these from unmarked word boundaries. A limitation of this work was that it was based on hand-labeled data, and therefore did not take advantage of the full training data set needed for designing a state-of-the-art recognition system. However, with our recent developments in prosodic annotation, we will be able to assess the usefulness on a much larger corpus in the future.

D. Institute for Signal and Information Processing, Mississippi State University

The work at Mississippi State University this year has centered on extending last year's progress in acoustic modeling robustness through kernel-based discriminative modeling. At the close of the last fiscal year, we had developed a hybrid speech recognition system that combined the temporal modeling benefits of hidden Markov models (HMMs) and the discriminative modeling capabilities of the support vector machine (SVM) paradigm. This hybrid system used a segmental modeling approach to phone classification, building a set of one-vs-all binary classifiers. To integrate the SVM into the HMM framework, a sigmoidal posterior probability function was used to convert the SVM distances to probabilities. From this work, we identified two major limitations of the hybrid HMM/SVM framework that have become the core of this year's work:

- **HMM-derived segmentations:** In the hybrid system, the SVM is dependent on the HMM core to provide good segmentations. It would be preferable to have the SVM determine for itself an optimal segmentation and hypothesis set.
- **Ad-hoc probability estimator:** The sigmoid posterior estimate incorporated into the hybrid HMM/SVM system was found to be ineffectual — a follow-up experiment indicated that simply using a step function yielded only a negligible loss in accuracy. The relevance vector machine (RVM), a completely probabilistic model that retains many of the discrimination and sparsity properties of the SVM, was identified as a potential solution to this problem.

An initial method for removing the dependency of the SVM on the HMM segmentations was built upon a time-synchronous Viterbi decoder. The SVM in this system is presented with all possible phone segmentations for all possible hypotheses. It scores those according to the one-vs-all binary classifiers, and the search process chooses the best sequence of words given those scores. However, in this framework, we found that the computational resources required were too large. To achieve a reasonable resource requirement, pruning thresholds needed to be tuned to the point where overpruning frequently occurred. This resulted in search errors and very poor word error rates. For instance, on an alphadigits task where state-of-the-art error rates are in the range of 10-15%, the SVM system could only achieve 85% error.

An analysis of the search paths at runtime indicated that the problem was in the combined use of synchronous Viterbi search and segmental models. The segmental models require that a complete phone segment be hypothesized before the phone is actually hypothesized and scored. This results in a large number of hypotheses that exist in the same model at the same time but which can not be compared for pruning purposes. In other words, Viterbi pruning can not be carried out at the sub-phone level. Contrast this to standard HMM systems where the predominate pruning is the Viterbi pruning carried out a the sub-phone level. A potential solution to this problem is the implementation of a stack-based decoding approach. With the removal of the time-synchrony limitation, phone hypotheses can be pursued and pruned without the accumulation of many non-

viable hypotheses.

A second line of research pursued this year was replacement of the SVM by an RVM model. The RVM is a Bayesian model which takes the same form as the SVM model and provides a fully probabilistic alternative to the SVMs which use the ad-hoc sigmoid posterior estimate. The RVMs have been found to provide generalization performance on par with SVMs while typically using nearly an order of magnitude fewer parameters as indicated for a vowel classification task in Table 2. Sparseness of the model is automatic using MacKay's automatic relevance determination methods.

Our initial attempts to incorporate the RVM technology used an approach identical to the hybrid HMM/SVM system. A set of one-vs-all RVM phone classifiers were trained on segmental data. Unlike the SVM, there was no need for a posterior estimator function since the RVM is, itself, a posterior estimator. As with SVMs, the process to train an RVM classifier is computationally expensive even for small problems. For the RVM, though, the computational complexity is $O(M^3)$ in run-time and $O(M^2)$ memory, where M is the number of basis functions and is initially set to the size of the training corpus. Thus, our initial attempts were limited to relatively small training sets as indicated in Tables 3 and 4. Since our aim is to replace the HMM emission distribution by an RVM, the RVM would be exposed to every frame of data in the training corpus. For even small speech corpora the number of frames in the training set is on the order of 10^6. The usual RVM training methods are, thus, rendered impractical.

We are currently researching a number of alternative training schemes. Most of these incorporate a reduced set methodology where the optimization problem is solved for a small portion of the training data. The solution for that portion is then used to select other interesting portions of the training set that need to be examined. Eventually, an optimum on the entire training set is achieved through the optimization of many smaller sets.

The results of the HMM/SVM hybrid system indicate a need to automatically incorporate

Classifier	Error Rate	Average Parameter Count
SVM	35.0%	82.8
RVM	30.3%	12.6

Table 2. Comparison of SVM and RVM classifiers on Deterding vowel data. Each classifier type was trained as a set of 11 1-vs-all classifiers. The best performance reported thus far on this data is 30.4% using a speaker adaptation scheme called Separable Mixture Models.

Classifier	Error Rate	Average Param. Count	Train Time	Test Time
SVM	16.4%	257	1/2 hour	30 min
RVM	16.2%	12	1 month	1 min

Table 3. Comparison of SVM and RVM classifiers on alphadigit recognition tasks. Both systems used a segmental hybrid architecture. Note that the RVM has over an order of magnitude fewer parameters but requires significantly longer to train. A reduced training set of 2000 segments was used.

segmentation variation into the training process. HMMs offer a principled approach to this problem via the EM-based Baum-Welch algorithm. Our continued research aims to create a similar algorithm for training HMM/ RVM systems. The RVM will replace the Gaussian as the frame-level emission distribution in the HMM state. Iterative reestimation formulae which describe cycles of Baum-Welch statistical accumulation (the expectation step) followed by Bayesian RVM training (maximization step) will be derived. In building this training algorithm we must address issues of iterative and monotonic convergence and stopping criteria. Similar work that has been developed for connectionist HMM/ ANN systems will serve as reference.

Classifier	Error Rate	Average Parameter Count
SVM	40.8%	1213
RVM	41.2%	178

Table 4.. Comparison of SVM and RVM classifiers on alphadigit recognition tasks. Both systems used a segmental hybrid architecture. Note that the RVM has over an order of magnitude fewer parameters but requires significantly longer to train. A reduced training set of 2000 segments was used.

08/15/00 — 08/14/01: RESEARCH AND EDUCATIONAL ACTIVITIES

In the first year of this project, we focused our efforts in two core areas:

- **Parsing Technology**: intimately coupling parsing technology with speech recognition technology and evaluating performance on conversational speech.
- **Risk Minimization in Acoustic Modeling**: developed a new acoustical modeling framework based on the principle of risk minimization using relevance vector machines; developed baseline recognition results for a related approach based on support vector machines.

We also began work on the integration of prosodic information into speech recognition and parsing. We developed a format for interfacing prosody output with our parser. We reviewed and cleaned up transcriptions of a prosodically labeled subset of the Switchboard corpus. We also discussed strategies for incorporating prosody into the search process in speech recognition.

A project kickoff meeting was held at Johns Hopkins University in June to coordinate the work on this project. All organizations involved in this project were present at this meeting. Discussions focused on three major topics: parsing, integration of prosody, and the development of resources to support this research. Plans were developed to begin evaluating the impact of parsing using a lattice rescoring approach, and to investigate the resources required to develop a time-aligned version of the Penn Treebank corpus that will be used for prosodic modeling. Other topics of discussion included some preliminary results on a hybrid speech recognition system using Support Vector Machines. Our next joint project meeting is planned for early June'2002.

E. Parsing Technology

We have begun research into applying parsing technology to speech. While our ultimate goal is to intimately couple parsing technology with speech recognition technology, clearly a first step is to demonstrate that current parsing technology is in fact compatible with the kind of language that occurs in naturally-occurring speech, and demonstrating that current parsing technology can do a reasonable job of parsing speech transcripts is an important first step. State-of-the-art statistical parsers are invariably trained on Treebank training data, and the recent release of a treebanked portion of the Switchboard corpus by the LDC permitted us to train such a parser on spoken language transcripts. We have two papers that have already appeared in prestigious conferences, and one new result which we expect to submit to a 2002 conference. Charniak and Johnson [10] investigated the performance of state-of-the-art parser technology when applied to speech transcripts. Current parsing technology has been primarily developed using written material; indeed, the best high-performance statistical parsers available today are based on Wall Street Journal newspaper texts, and it was an open question whether this technology is applicable to spoken language.

Transcribed speech differs from edited written text in that it contains disfluencies of various kinds. The two major types of disfluencies we considered in this work are interjections (e.g., "ugh"), parentheticals (e.g., "Sam is, I think, insane") and speech repairs (e.g., "I told my brother, ugh, my sister I'd be late"). Interjections are extremely easy to recognize using standard part-of-speech tagging techniques, and there has been speculation in the literature that interjections provide valuable clues to phrase boundaries (we describe empirical an evaluation of this hypothesis below). Written text also contains parentheticals, and these do not seem to cause current parsing technology any particular problems. However, in a pilot experiment we determined our standard

statistical parser, even when trained from a Switchboard speech transcript treebank that identifies speech repairs, fails to identify any speech repairs in the test corpus. This is not too surprising, since modern statistical parsers function by modeling the tree-structured head-to-head dependencies in a normal natural language sentence, but speech repairs do not seem to be included in such dependencies. Charniak and Johnson [10] present a simple architecture for parsing transcribed speech in which an edited-word detector first removes such words from the sentence string, and then a standard statistical parser trained on transcribed speech parses the remaining words. The edit detector achieves a misclassification rate on edited words of 2.2%. (The **NULL**-model, which marks everything as not edited, has an error rate of 5.9 %.) To evaluate our parsing results we introduce a new evaluation metric, the purpose of which is to make evaluation of a parse tree relatively indifferent to the exact tree position of **EDITED** nodes. By this metric the parser achieves 85.3\% precision and 86.5\% recall; results which are comparable with the best written text parsing results of just a few years ago.

In [11], we investigated the use of our parsing model as a language model. Language models, of course, are used in speech recognition systems to distinguish between likely and unlikely word strings proposed by the speech recognizer's acoustic model. Most speech recognition systems use the very simple trigram language model, but recently there has been increased interest in using parsing for this task. However, the previous parsers used for this purpose have not performed parsing tasks at state-of-the-art levels. This is because the researchers assumed that any language model would have to work in a strict left-to-right fashion. Unfortunately, the best statistical parsers are "immediate-head" parser — our name for a parser that conditions all events below a constituent *c* upon the head of *c*. Because the head of a constituent may appear in the middle or at the end (e.g., the head of a noun-phrase is typically the right-most noun) immediate head parsers cannot work in a strict left-to-right fashion. However the reasons for preferring strict-left-to-right are not iron-clad and we were interested in determining if better parsing performance of immediate-head parsers would lead to a better language model. In the paper we presented two immediate-head language models. The perplexity for both of these models significantly improve upon the trigram model base-line as well as the best previous grammar-based language model. For the better of our two models these improvements are 24% and 14% respectively. We also found evidence that suggests that improvement of the underlying parser should significantly improve the model's perplexity. Since these models do not use prosodic information that most assume should help in parsing, we believe that even in the near term there is a lot of potential for improvement in immediate-head language models. Finally we note that this paper received the "Best Paper" award at ACL2001.

We now turn to our current research in the area of parsing speech data. As reported above, it is widely believed that punctuation, interjections and parentheticals all provide useful cues to phrase boundaries, and therefore their presence ought to improve parser performance. Previous experimentation with written texts had shown that removing punctuation from written texts decreases parser performance significantly, and indeed, finding prosodic cues that convey much the same information as punctuation is one of the goals of our future research. However, as a preliminary step we decided to empirically evaluate the usefulness of punctuation, interjections and parentheticals in parsing of speech transcripts. Our method of evaluation is to selectively remove each of these in turn from the training corpus, and then evaluate the accuracy of the parser's recovery of linguistically important structural details from a version of the test corpus from which the same elements were removed. Together with Donald Engel (a student at Brown),

Charniak and Johnson are systematically investigating the effect that punctuation, interjections and parethenticals have on parsing speech transcripts. As expected from the written text studies, punctuation supplies useful information for parsing spoken language transcripts, i.e., systematically removing punctuation from the training and test corpora reduces parse quality. However, contrary to the accepted wisdom, interjections and parenthetical seem not to supply useful information for parsing spoken language transcripts, i.e., systematically removing either of these elements improves parse quality, at least for our current parser. At this stage we can only speculate as to why; perhaps parentheticals are integrated into the rest of the sentence involving a structure different to the head-to-head dependency structure used in the parser, and perhaps interjections interrupt the sequences of dependencies tracked by the parser, in effect splitting the parser's internal state structure and leading to sparse data problems.

One of the central goals of this project is to integrate natural language parsing (which has been largely developed with respect to written texts) with speech recognition. As described above, we have demonstrated that parsing technology can be successfully applied to speech transcripts, and we have shown that the kinds of syntactic structures posited by a statistical parser can form the basis for a high-performance language model. These results suggest that a combined speech recognition/parsing system should perform extremely well. There is still a substantial amount of engineering and scientific work to be performed before we have achieved that integration. Currently we are investigating just what the interface between the speech recognition and parsing components should be in a combined system. It turns out that the basic data structures in each component — lattices in speech recognition, charts in parsing — are in principle quite compatible; theoretically at least one could imagine running a parser in parallel with an acoustic model (i.e., the parser would be the language model). This is a bold and attractive architecture, but we suspect that at the current stage it is impractical; the number of word hypotheses would simply overwhelm the parser. We are thus investigating ways of pruning the hypothesis space (perhaps by using a standard trigram language model) and of compacting the set of hypotheses (perhaps by using sausages instead of lattices); probably some combination of the two will turn out to be viable.

Other speech/parsing work we anticipate for this coming year will include looking at features that have been found to improve trigram language models that are not included in our language models to see if, as one might anticipate, they improve our parsing language models as well. This would include word clustering, caching, and simply training on more data (This last is not as easy for parsing models as we do not have more hand-parsed data, and thus would have to use machine-parsed data.) We also hope to start work on the integration of prosody with parsing, though this is a more ambitious project.

F. Risk Minimization in Acoustic Modeling

An important goal in making speech recognition technology more pervasive is to improve the robustness of the acoustic models. Language models, for example, tend to port across domains much better than acoustic models. Learning paradigms for language models can fairly easily extract the domain-independent information, and don't have to deal with difficult problems such as the separation of the underlying speech spectrum from channel and ambient conditions. Though one might argue that even language models are susceptible to overtraining and a lack of generalization, the degree to which this corrupts system performance in a new domain is much

less severe. Acoustic models often require extensive training or adaptation, and this, in turn, requires the development of extensive application-specific data collection. The net effect is that the cost of developing new applications is very high.

A guiding principle we have in acoustic modeling is that of Occam's Razor: a model that makes less assumptions about the data will prove to be more robust. Further, we believe that we must gracefully mix representation and discrimination in our models. Intelligent machine learning seems to be a crucial issue as acoustic models can easily learn details of the acoustic channel from the training data, making them less portable to new applications where the channel, microphone, or ambient environment are different. A promising new framework for machine learning in which a balance between generalization and discrimination can be struck is based on the principle of risk minimization [12], and is known as a Support Vector Machine [13]. A summary of the benefits of the SVM approach is shown in Figure 1.

The goal in the first year of this project was to explore these models in the context of a realistic LVCSR task. Our primary focus has been kernel-based methods, which include two important related techniques: the Support Vector Machine (SVM) and the Relevance Vector Machine (RVM) [14]. On preliminary experiments involving phone classification, SVMs performed significantly better than HMMs [15]. These results are summarized in Table 1. For the six most confused phone pairs of an Alphadigit task, SVMs nearly halved the error rate, which is a significant reduction for this type of experiment.

Our initial experiments were constructed using a hybrid HMM/SVM system as shown in Figure 2. In this system, we generate N-best lists using a conventional HMM speech recognizer. We then use the same system to generate time alignments. The segments identified in these time

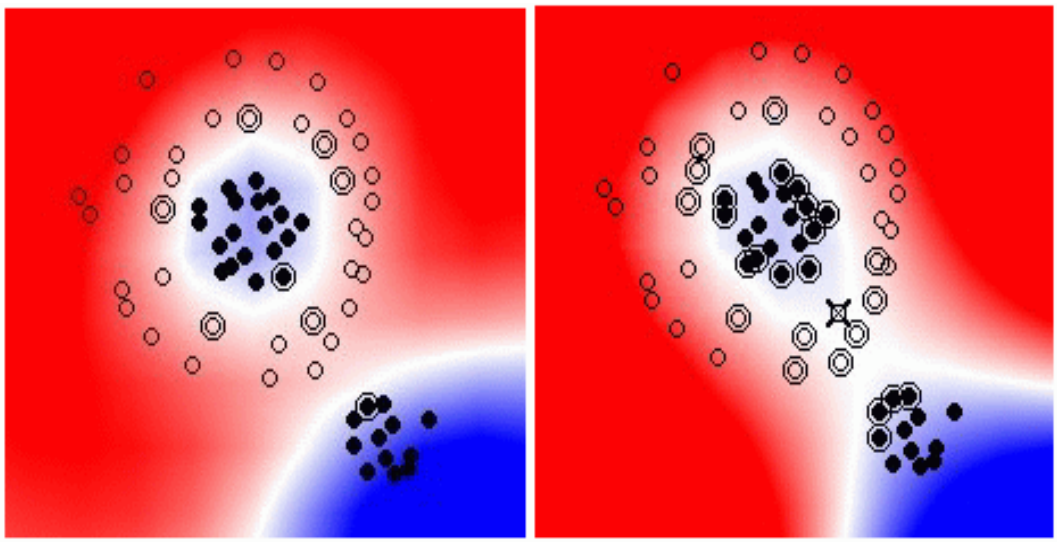

Figure 1. An SVM balances the ability to model a particular training set with generalization to other data. A feature of this machine is an ability to gracefully trade-off knowledge about the training data and the probability of error for unseen data. SVMs have proven to be very successful on several tasks including handwriting recognition, speaker identification, and vowel classification. SVMs have the ability to learn nonlinear decision regions using principles of discrimination. No assumptions about the underlying distributions are made — no parametric forms are used to build the decision surfaces.

phone pair	SVM misclassification rate	HMM misclassification rate
f <=> sil	14.6	13.1
r <=> l	11.9	17.8
s <=> sil	37.5	42.4
s <=> z	9.7	17.8
t <=> p	8.7	18.1
t <=> d	9.6	22.2

Table 1. A summary of performance of an SVM-based hybrid system on the most common phone confusions for Alphadigits. In some cases, the reduction in error rate is over 50%.

alignments are then rescored using likelihoods generated by SVM phone classifiers. The standard Gaussian statistical models are replaced with discrimination-based SVM models.

One problem in constructing this system was how to map distances computed by the SVM classifier to posterior probabilities, which are needed by the HMM speech recognition system (more precisely, the Viterbi search engine used in the HMM-based speech recognition system). A typical solution to this problem that has been used extensively in the neural network literature is to fit a sigmoid function to the distribution of distances. However, we have recently observed that this process tends to overestimate confidence in classification. We are revisiting this issue in subsequent research described below.

We have also had to overcome a number of other mundane but important problems related to the recognition system to make these experiments possible. Because of the computational complexity of the approach, we also needed to develop an iterative training scheme in which we build classifiers on small subsets of the data and combine these classifiers (rather than training across the larger data set). We use an approach known as “chunking” [16,17] which has been shown to provide good convergence while significantly reducing computational requirements.

The SVM system overall delivered a 1% absolute (10% relative) reduction in word error rate (WER) on the Alphadigits task described above, reducing the absolute error rate from 12% to

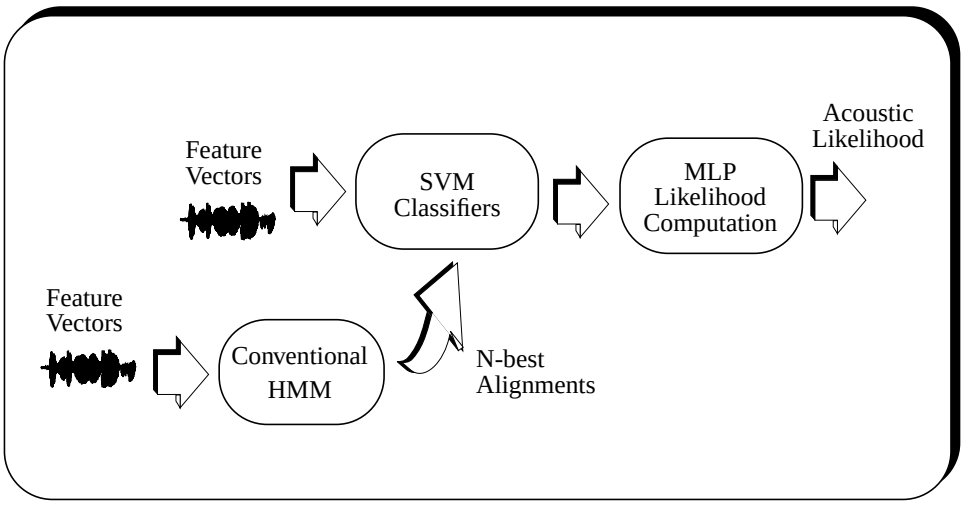

Figure 2. An overview of a hybrid HMM/SVM system being developed to improve the robustness of a speech recognition system.

11%. Such a small improvement is somewhat discouraging given the computational complexity of this approach. We believe a major limitation of this system is the dependence on the HMM-based N-best lists and segmentations. Hence, we are developing approaches in which the SVM-based classifier is integrated into the training process.

A natural way to do this is to modify the concept of an SVM to incorporate probabilistic models directly. The Relevance Vector Machine (RVM) [14] attempts to overcome the deficiencies of the SVM by incorporating a probabilistic model directly into the classifier rather than using a large margin classifier [14]. The principle attraction of the RVM is that it delivers comparable performance as an SVM, but uses much fewer parameters. It is also much more computationally efficient.

A major challenge in incorporating RVM models directly into the recognition training process is the development of practical and efficient closed-loop training techniques based on EM principles that demonstrate good convergence properties. Many of these discrimination-based techniques involve some form of nonlinear optimization that is unwieldy and prone to divergence problems. We are currently developing the RVM optimization process in a Baum-Welch training framework so that the parameters of these models can be estimated in a closed-loop process on large amounts of data. We expect to complete this work in early fall of 2001.

Finally, the software being developed on this part of the project is being implemented within our public domain speech recognition system [18]. Pieces of this system will be included in our upcoming release. The core of the system consists of two new classes, SupportVectorMachine and RelevanceVectorMachine, that are part of our pattern recognition classes. We also expect to release an application note shortly describing the use of the core pattern recognition engine, and will release the hybrid system by the end of 2001. We also expect to have completed large-scale pilot experiments on spontaneous speech data at that time.

G. REFERENCES

[1] R.D.Y. Bengio and P. Vincent, "A neural probabilistic language model," Tech. Rep. 1178, University of Montreal, 2000.

[2] M. Beckman and J. Pierrehumbert, "Intonational structure in Japanese and English," *Phonology Yearbook*, pp. 255--309, 1986.

[3] W. Byrne et al., "Pronunciation Modelling Using a Hand-Labelled Corpus for Conversational Speech Recognition," *IEEE International Conference on Acoustics, Speech and Signal Processing*, Seattle, Washington, USA, 1998.

[4] E. N'oth, A. Batliner, A. Kiebling, R. Kompe, and H. Niemann, "Verbmobil: The Use of Prosody in the Linguistic Components of a Speech Understanding System," *IEEE Trans. on Speech and Audio Processing*, 8(5):519-532, 2000.

[5] M. Ostendorf, "Linking Speech Recognition and Language Processing Through Prosody," *CC-AI*, vol. 15, no. 3, pp. 279-303, 1998.

[6] J. Pitrelli, M. Beckman, and J. Hirschberg, "Evaluation of prosodic transcription labeling reliability in the ToBI framework," in *Proc. of the International Conference on Spoken Language Processing*, pp. 123-126, 1994.

[7] P. J. Price, M. Ostendorf, S. Shattuck-Hufnagel and C. Fong, "The Use of Prosody in Syntactic Disambiguation," *Journal of the Acoustical Society of America*, vol. 90, no. 6, December 1991, pp. 2956-2970.

[8] I. Shafran, M. Ostendorf, and R. Wright, "Prosody and phonetic variability: lessons learned from acoustic model clustering," *Proc. of the ISCA Workshop on Prosody in Speech Recognition and Understanding*, pp. 127-131, October 2001.

[9] D. Talkin, "Pitch Tracking," in *Speech Coding and Synthesis*, ed. W.~B. Kleijn and K.~K. Paliwal, Elsevier Science B.V., 1995.

[10] E. Charniak and M. Johnson, "Edit Detection and Parsing for Transcribed Speech," *Proceedings of the 2nd Meeting of the North American Chapter of the Association for Computational Linguistics*, Pittsburgh, Pennsylvania, USA, pp. 118-126, June 2001.

[11] E. Charniak, "Immediate-Head Parsing for Language Models," *Proceedings of the 39th Annual Meeting of the Association for Computational Linguistics*, Toulouse, France, June 2001.

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

[13] C. J. C. Burges, "A Tutorial on Support Vector Machines for Pattern Recognition," *Data*

Mining and Knowledge Discovery, vol. 2, no. 2, pp. 121-167, 1998.

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

[15] A. Ganapathiraju, *Support Vector Machines for Speech Recognition*, Ph.D. Dissertation, Department of Electrical and Computer Engineering, Mississippi State University, Mississippi State, Mississippi, USA, December 2001 (in preparation).

[16] E. Osuna, R. Freund, and F. Girosi, "An Improved Training Algorithm for Support Vector Machines," *Proceedings of the IEEE Workshop on Neural Networks and Signal Processing*, Amelia Island, FL, September 1997.

[17] G. Zoutendijk, *Methods in Feasible Directions — A Study in Linear and Non-linear Programming*, Elsevier Publishing Company, New York, NY, USA, 1960.

[18] M. Ordowski, N. Deshmukh, A. Ganapathiraju, J. Hamaker, and J. Picone, "A Public Domain Speech-To-Text System," *Proceedings of Eurospeech'99*, pp. 2127-2130, Budapest, Hungary, 1999.