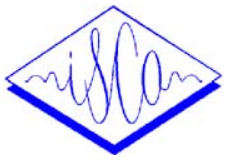
ISCA Archive
http://www.isca-speech.org/archive



6th European Conference on
Speech Communication and Technology
(EUROSPEECH'99)
Budapest, Hungary, September 5-9, 1999

TOWARDS MULTI-DOMAIN SPEECH UNDERSTANDING USING A TWO-STAGE RECOGNIZER[1]

Grace Chung, Stephanie Seneff and Lee Hetherington

Spoken Language Systems Group
Laboratory for Computer Science
Massachusetts Institute of Technology
Cambridge, Massachusetts 02139 USA
http://www.sls.lcs.mit.edu, mailto:{chung, seneff, hetherington}@sls.lcs.mit.edu

10.21437/Eurospeech.1999-586

ABSTRACT

This paper describes our efforts in designing a two-stage recognizer with the objective of developing a multi-domain speech understanding system. We envisage one first-stage recognition engine that is domain-independent, and multiple second-stage systems specializing in individual domains. A major novelty in our initial two-stage design is a front-end that incorporates ANGIE-based hierarchical sublexical probability models encapsulated within a finite-state transducer (FST) paradigm. This first stage is a context-dependent syllable-level recognizer which outputs acoustic-phonetic networks to be processed in a second pass. The second stage incorporates higher order linguistic knowledge, from phonological to syntactic and semantic, in a tightly coupled search. This system has yielded up to a 28.5% reduction in understanding error, compared with a single stage context-dependent recognizer which does not use ANGIE-based probabilities.

1. INTRODUCTION

Today, telephone-based conversational systems are beginning to emerge as human-computer interfaces for information access and interactive problem-solving tasks. This has placed growing demands on the usability of these interfaces. In the future, a realistic application will need to retrieve information from a broad range of sources, such as on-line databases, and will allow users to switch seamlessly and transparently among several topic domains. That is, a user will be able to pursue several topics of interest, from flight status or world-wide weather information to traffic conditions, within a single telephone call and within the same dialogue. Under such circumstances, many challenging research problems arise, one of which is the ability to cope with unknown words. In real-world tasks with multiple users, a speech recognizer will inevitably encounter out-of-vocabulary words. This is because a fixed recognizer vocabulary cannot anticipate all words employed by all potential users or even entirely cover the information content whose nature is on-demand and could change frequently and unpredictably.

Our objectives are motivated by the vision of an architecture that supports multiple topic domains in parallel, while a dynamic and flexible vocabulary resides within each domain. With conventional systems, developed over single domains, vocabularies tend to be closed or fixed. When unknown words emerge at the input, these systems can only propose errorful hypotheses or reject the utterance altogether. On the other hand, we foresee a future system that handles unknown words both at the spoken input as well as from the information source. For instance, this system would be able to detect the presence of a previously unseen word and subsequently deduce the acoustic, phonological and linguistic properties of this word. This can be performed automatically, without extensive retraining and thus the recognizer vocabulary is expanded dynamically.

In this paper, we introduce the conceptual design for a flexible vocabulary multi-domain speech understanding system. We propose a two-stage architecture where the front-end is a domain-independent recognition engine, and this is interfaced via a subword network to a back-end which incorporates constraints from higher order knowledge sources tailored to several individual domains. This paper will describe our initial step towards realizing our objectives. Here, we have developed a preliminary first-stage core recognizer which utilizes only syllable-level linguistic information, using the ANGIE framework [5]. An acoustic-phonetic network is output for the second stage processing. We would like the first stage to evolve into a generic speech recognizer independent of domain-specific vocabularies and be ultimately capable of supporting a flexible vocabulary. The second stage includes a single search that tightly couples natural language (NL) processing, provided by TINA [4], with hierarchical language models, also derived from ANGIE.

A major novelty in this work is the folding of hierarchical linguistic knowledge into an FST representation in our first-stage recognizer. We have transformed ANGIE, our trainable sublexical framework, into an FST, which has enabled us to incorporate powerful linguistic constraints into a state-of-the-art near real-time speech recognizer. Although experiments to date are conducted exclusively over the JUPITER weather domain, ANGIE's sublexical models could conceivably be trained over multiple corpora and effectively capture generic language information for a domain-independent recognizer. The next sections will elaborate on our conceptual two-stage design and our current, interim solution. Then we will detail our efforts in encapsulating ANGIE into an FST and the performance improvements in doing so.

2. DESIGN CONSIDERATIONS

Our conceptual multi-domain system is illustrated in Figure 1. The initial stage consists of a domain-independent core recognition engine which only utilizes acoustic and general linguistic knowledge to produce hypotheses. Trained on several large corpora, this recognizer codifies general English morphological and syllabic information. We postulate that within the linguistic hier-

[1]This material is based upon work supported by the National Science Foundation under Grant No. IRI-9618731.

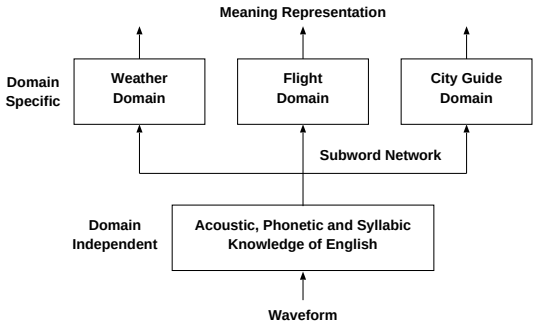


Figure 1: *A Two-Stage Multi-Domain Speech Understanding System.*

archy, information up to and including the syllable level can be valuable in enforcing constraints in recognition while maintaining generality without confining the system to any fixed vocabulary items. By training on a large number of syllables, in order to maximize coverage, and thereby accumulating general knowledge of English sublexical structures, this recognizer serves as a first pass whose function is to prune away a large portion of the search space. Moreover, it is capable of providing probabilistic support for novel words (that are consistent with English word structure), false starts and partial words. A recognizer with only phonemic-level information would not constrain the space sufficiently, giving rise to a large number of hypotheses. On the other hand, using word-level units soon becomes unwieldy because no word-level recognizer can cover all novel constructions and partial word possibilities. The output of the front-end is a subword network which is then processed by a suite of domain-dependent speech understanding modules. With a reduced search space and thereby more manageable computational requirements, each of these modules utilizes higher level linguistic information such as domain-specific natural language (NL) models which account for dialogue context. The final decision for the best meaning representation is mediated by a top-level decision algorithm.

Our current system is a natural extension of the preliminary two-stage system introduced in [1]. As before, the first stage is a syllable-level recognizer, supported by a morph-based n-gram language model. Morphs are syllable-level units with distinct spellings and augmented by positional markers. For example, the word *prediction* is composed of 3 morphs: *pre−*, *dic+* and *−tion*. The first stage employs SUMMIT [2], a segment-based recognizer that utilizes context-dependent acoustic models. Recently, SUMMIT was modified to use a single FST [3] representing all search constraints. This single FST maps from phonetic labels to word or syllable labels, and is generally the composition of lexicon and language model FSTs. This composition can be either precomputed and optimized or performed on-the-fly. This feature has made possible the folding of ANGIE's hierarchical language models into a computationally tractable paradigm. The first stage outputs an acoustic-phonetic graph (also represented as an FST) which is input to a search incorporating ANGIE and TINA. This will be discussed in Section 5.

3. THE ANGIE FRAMEWORK

Introduced in [5], ANGIE is a system for speech analysis which characterizes word substructure via a multi-layered hierarchical representation. It combines a trainable probabilistic framework with a hand-written context-free grammar. From bottom to top, the layers capture phonetics, phonemics, syllabification and morphology. Stress information is embedded explicitly throughout sublexical nodes of the hierarchy, and the phoneme-to-phone layers govern phonological events.

ANGIE's parser proceeds in a bottom-up, left-to-right manner, advancing column[2] by column. Upon the completion of a word parse, ANGIE yields a linguistic score that comprises log probabilities for each column which, in themselves, are sums of trigram bottom-up probabilities and conditional probabilities for advancing columns. Training is conducted on automatically generated phonetic alignments for a large set of training utterances. It has been shown that ANGIE's phone perplexity is lower than that of a phone trigram, and we have successfully employed ANGIE's linguistic model in aid of a variety of recognition tasks.

ANGIE was designed with speech recognition tasks in mind in a number of respects. Firstly, the probabilistic framework will cope with errorful input phone hypotheses without parse failure, due to the over-generalizing ability of the rules. Yet they are discouraged by the low probabilities, as is required by a recognition algorithm. It also promotes sharing of probability models afforded by words with common substructures. ANGIE has been envisaged in facilitating the acquisition of new words without lexical retraining where probabilities for phonological rules can be leveraged from the trained models for existing words.

4. THE ANGIE-BASED FST STRUCTURE

There are several ways in which the hierarchical knowledge of ANGIE can be folded into an FST. Here we will delve into one in particular which can easily be implemented and has yielded positive results. It involves precompiling the trained probability model into a left-branching network structure that accounts for all possible phone transitions for all syllables in the lexicon. The recognizer begins search from the left and finishes on the right when a phone sequence completes a syllable; a self-looping arc brings the path back to the start of another syllable or to a final state indicating end of sentence. As analogous with a standard pronunciation graph, the input alphabet of the transducer will be the phone terminals of an ANGIE tree while the output strings comprise syllables from the lexicon. Along the arcs of the FST will be probability estimates as arc weights computed from ANGIE. The allowable transitions on this FST would be determined by the rules stipulated in ANGIE. In this right-to-left branching tree structure, syllable labels are emitted at the beginning of the phone sequence (on the left-most end of a branch) and branches with common phone sequences (from the right) are successively collapsed together to share the same arcs.

The challenge was to choose a strategy for representing the rich probability space of ANGIE compactly and efficiently as well as preserving its flexibility. A procedure was developed to construct the FST from a set of training utterances with automatically computed phonetic alignments. It is set out as follows:

[2]This refers to the nodes along a given path from the root node to the terminal node.

1. *Train grammar.* An ANGIE grammar is trained from a set of utterances.
2. *Compute* ANGIE *scores and construct tree structure.* A second pass through this training set is performed to compile a structure that accounts for all phonetic sequences that occurred within training. For each syllable, an ANGIE parse tree is obtained for the respective phonetic realization and an algorithm runs backwards from the rightmost column, successively compiling a left-branching structure.

   At the same time, the score for each partial phone sequence as we traverse from the right across the graph is recorded in an array. The score for any partial phone sequence is computed by summing the corresponding column scores. Identical partial phone sequences may differ in score because, where the corresponding syllables differ, the associated ANGIE parse trees are also different, and therefore the respective columns score differently.
3. *Compute FST arc weights.* Successively compute the arc scores from right to left in a recursive algorithm:

```
Initialization.
Compute_scores_for_left_arcs_of(R):
   foreach (A = left_arc_of(R)):
      arc_score(A) = max_score(A) - arc_score(R)
      Compute_scores_for_left_arcs_of(A)
```

   where

$$max\_score(A) = max\{s_j, j = 1, 2, ..\} \qquad (1)$$

   and s_j is the jth score for a particular phone sequence that is found by traversing the FST backwards from the end phone up to arc A. Essentially, this algorithm computes the best incremental score, among all the different ANGIE partial parses, for choosing the arc transition as we traverse from right to left. As we complete a phone sequence of a particular syllable, the total path score will sum towards the total syllable score from an ANGIE parse tree.
4. *Compute inter-syllable arcs transitions.* The transition from the end of a syllable to the first phone of the next contains an arc weight that is a simple phone bigram probability computed at syllable boundaries. This is identical to ANGIE's treatment of syllable boundary probabilities.

After the construction of the ANGIE-FST, it is composed with the language model FST and standard manipulation algorithms are used to optimize the size of the FST. The resultant FST combines both the language model and ANGIE probabilities and is uploaded by the recognizer at recognition time. Alternatively, the composition can be performed on-the-fly.

In this implementation, we have essentially transformed the richly-embedded hierarchy of ANGIE into a flattened representation. Due to computing memory limitations, we were required to fold this large probability space as compactly as possible, and we have done so via this tree-like network structure which collapses common phone sequences together. However, one significant difference is that paths are accounted for in the FST only if the phonetic sequences have occurred in the training data, unlike ANGIE which has more powerful generalizing abilities. Therefore, in order to ensure adequate coverage and counter possible sparse data problems, we have used ANGIE in generation mode. This produces additional phonetic manifestations for each syllable, to artificially boost the training data. These are phonetic sequences that the dynamic ANGIE parser would allow. We expect this will enhance robustness to our FST.

As the basic units of the recognizer are morphs, the ANGIE grammar has been simplified to only carry syllable-level information. That is, the layer below the WORD node no longer specifies morphology but only distinguishes between stressed and unstressed syllables. This reduces the size of our grammar and subsequently our FST so that our sparse data problem is further alleviated. It implies that most of the probabilistic knowledge captured is associated with the phonological processes modelled by the lower layers of the hierarchy.

5. ANGIE-TINA INTEGRATION

Currently the second stage comprises an integrated ANGIE-TINA search. Analogous to ANGIE, TINA is a hierarchical NL framework based on a context-free grammar. It is augmented by (1) a set of features that enforce syntactic and semantic constraints, and (2) a trace mechanism that handles movement phenomena. More details are provided in [4]. The integrated ANGIE-TINA search is based on one described in [1]. A stack decoder search coordinates partial theories proposed by the ANGIE and TINA parsers in parallel, in a tightly integrated strategy. This second stage utilizes a word-level ANGIE grammar which contains additional morphological information, mainly encoding syllable-position, compared with the ANGIE grammar employed in the first stage. A fundamental difference from the system in [1] is the nature of the acoustic-phonetic network from recognition output. These networks are now in FST form and devoid of timing information, so that the stack search proceeds from left to right along topologically sorted nodes. Moreover, we have used standard FST manipulation algorithms to optimize the graph size, and this has contributed to the efficiency of our stack search.

6. EXPERIMENTAL METHOD

All experiments are conducted on the JUPITER world-wide weather information domain, comprising a 1341-sized word lexicon and a 1603-sized morph lexicon. The ANGIE probabilistic grammar is trained on 11677 utterances while the TINA word grammar is trained on 6531 utterances. The ANGIE-FST contained 7873 arcs and 1540 states. There are approximately 2.3 alternate pronunciations per word. The phonetic networks from the first stage were restricted to a maximum of 1000 arcs.

To evaluate our results, we use an understanding measure developed in [1] as well as a word error rate. This understanding measure is based on comparing key-value pairs computed from a TINA-based meaning representation. As in our previous work, we use a single stage SUMMIT word recognizer for baseline comparison. This uses the same acoustic models as our system in conjunction with a word bigram in a forward pass and a reverse word trigram in a backwards pass. There are two modes of operation: (1) SUMMIT Top 1: the highest scoring candidate is chosen and (2) SUMMIT 10-best: a rudimentary algorithm is used to choose, from the 10-best list, the most likely utterance where a meaning representation can be obtained, according to TINA, the NL parser.

We will report on results ascertained for the same development set that was quoted in [1] and also repeat these results on a previously unseen test set. In the morph-based first pass, we tried composing the ANGIE-FST with

System	WER (%)	UER (%)
1. SUMMIT Top 1	12.3	19.4
2. SUMMIT 10 Best	13.4	17.0
3. ANGIE (Bigram)	12.3	17.0
4. ANGIE-TINA (Bigram)	12.6	14.5
5. ANGIE (Trigram)	10.4	13.4
6. ANGIE-TINA (Trigram)	10.7	12.2

Table 1: *Word and Understanding Error Rates for the Development Set.*

System	MER (%)
1. SUMMIT Top 1	10.8
5 & 6 Trigram first-pass Top 1	9.7
5 ANGIE only	9.3

Table 2: *Morph Error Rates for the Development Set.*

both a bigram (Systems 3 and 4 in Table 1) and a trigram (Systems 5 and 6). For both of these cases, the ANGIE-FST was precomposed and optimized with the language model, resulting in faster computation at run-time. The composed FST with a bigram contained about 450,000 arcs and the one with a trigram contained about 2 million arcs. For the second stage, we compare the successive gains from augmenting with ANGIE only (Systems 3 and 5) and with ANGIE-TINA fully deployed (Systems 4 and 6).

7. RESULTS AND ANALYSIS

Table 1 depicts results for the development set. In general, augmenting with NL degrades word error rate (WER) but reduces understanding error rate (UER) which is of greater importance to system performance. In our previous work [1] on two-stage systems, it was shown that a second ANGIE-TINA stage can enhance understanding performance on a word-level first pass recognizer as well as recover performance losses incurred when the first stage was stripped of word-level constraints. Our current results also reflect the same trends. The ANGIE-only system (3), using a morph bigram, already recovers performance comparable to the baseline, in spite of a relatively unconstrained n-gram model in the first pass. Adding TINA, compared with an ANGIE-only second stage, consistently decreases UER, accompanied by a small (WER) degradation. When a morph trigram is used in the first stage with an ANGIE-TINA second stage, results exceed all previously quoted ones. The final UER is reduced by 28.5% (from 17.0% to 12.2%) relative to the SUMMIT 10-best. The corresponding relative reduction in UER for a bigram in the first stage is 14.1% (from 17.0% to 14.5%).

These performance gains can largely be attributed to the ANGIE scores incorporated into stage one. This is further substantiated when we consider the morph error rates (MER)[3] in Table 2. We report the MER for the baseline system 1 and for the best scoring hypothesis in the first pass when a morph trigram is used. There is a 10.2% improvement (from 10.8% to 9.7%.) This suggests that considering the first stage alone, a morph-based ANGIE-FST recognizer is superior to a word-based baseline recognizer, a reflection on the power of ANGIE's sublexical constraints when combined with a morph trigram. When we consider the MER of the second stage output (in ANGIE only) at 9.3%, we can directly infer that the word-level ANGIE grammar still contributes additional knowledge and enhances performance through its use of higher level information, even though much performance improvement was already reaped in the first stage by the morph-based ANGIE.

Our experiments were also repeated on an independent test set and the same trends were ascertained. When comparing with a SUMMIT 10-best baseline, a 23.6% reduction in UER was achieved using a morph trigram ANGIE-FST first pass with ANGIE-TINA second pass.

These experiments were conducted on a subset of the entire data available and a subset of the entire vocabulary of the most recent JUPITER-based recognizer. Currently, we have a five-fold increase in training data available and our baseline results have benefited substantially from this. We plan to repeat the above experiments using the larger training set and expect to observe the same trends.

8. CONCLUSION AND FUTURE WORK

We have attained very positive results which demonstrate the power of ANGIE to model pronunciation variability effectively. This has been achieved by folding ANGIE's sophisticated sublexical models into an FST representation so that it is now amenable to operation with near real-time recognition. We are encouraged to pursue a further reliance on generic language models and sublexical probabilities in the first stage. We would like to continue work towards developing a multi-domain system with a domain-independent front-end. Many avenues remain to be explored. For instance, the current ANGIE-FST structure can be further improved. The full generalizing ability of this paradigm is partly diminished because the FST relies on training data examples in order to license phonetic sequences, even though ANGIE's probabilities give rare instances good support through common sharing. However, examples absent from training data are never instantiated in the FST, but they are permitted in a dynamic ANGIE parse. We are currently addressing this issue with an alternative FST implementation. Our goals are to develop a more flexible FST which will offer full probability support for novel word constructs, one that is independent of a fixed lexicon.

9. REFERENCES

[1] G. Chung and S. Seneff, "Improvements in Speech Understanding Accuracy Through the Integration of Hierarchical Linguistic, Prosodic, and Phonological Constraints in the Jupiter Domain," in *Proc. ICSLP '99*, Sydney, Australia, pp. 935–939, Dec. 1998.

[2] J. Glass, T. J. Hazen and I. L. Hetherington, "Real-time Telephone-based Speech Recognition in the Jupiter Domain," in *Proc. ICASSP '99*, Phoenix, pp. 61–64, Mar. 1999.

[3] M. Mohri, M. Riley, D. Hindle, A. Ljolie and F. Pereira, "Full Expansion of Context-Dependent Networks in Large Vocabulary Speech Recognition," in *Proc. ICASSP '98*, Seattle, pp. 665–668, May 1998.

[4] S. Seneff, "TINA: A Natural Language System for Spoken Language Applications," in *Computational Linguistics*, 18(1):61–86, March 1992.

[5] S. Seneff, R. Lau, and H. Meng, "ANGIE: A new framework for speech analysis based on morpho-phonological modelling," in *Proc. ICSLP '96*, Philadelphia, PA, vol. 1, pp. 110–113, Oct. 1996. URL http://www.sls.lcs.mit.edu/raylau/icslp96_angie.pdf

[3]The MER is computed in the same way as WER. In cases where the recognizer outputs words, these are decomposed to their respective morph units.