INTERFACING ACOUSTIC MODELS WITH NATURAL LANGUAGE PROCESSING SYSTEMS

Michael T. Johnson, Mary P. Harper, and Leah H. Jamieson

Purdue University, School of Electrical and Computer Engineering
West Lafayette, IN 47907
{mjohnson,lhj,harper}@ecn.purdue.edu

ABSTRACT

The research presented here focuses on implementation and efficiency issues associated with the use of word graphs for interfacing acoustic speech recognition systems with natural language processing systems. The effectiveness of various pruning methods for graph construction is examined, as well as techniques for word graph compression. In addition, the word graph representation is compared to another predominant interface method, the N-best sentence list.

1. INTRODUCTION

An important research topic in recent years has been the integration of speech recognition systems with language models [2, 8]. Many systems integrate stochastic language models directly into the speech recognizer. However, a structure in which a front-end acoustic recognizer is interfaced to a separate language processing module allows use of more sophisticated parsing techniques and additional semantic and contextual information to aid in speech understanding. The choice of data representations used to accomplish this interface is of great significance, because this choice determines how word and sentence hypotheses are evaluated in light of our understanding of language and grammar. The underlying goal is to identify the 'best' overall sentence candidate with respect to all available knowledge sources, as constrained by time and space considerations.

Recently, word graphs have begun to be used as an alternative to N-best sentence lists as an interface representation [1, 4, 5, 6]. N-best lists are a stream-based interface between acoustic and language components, where the system must work on alternatives one at a time. Word graphs, although they can be constrained to the stream-based view, are able to support an aggregate processing view as well, and therefore have flexibility which is important in examining integration alternatives.

This research concentrates on evaluating the strengths of the word graph representation. We systematically measure word graph effectiveness against a variety of recognition parameters, and for reference compare these measures against the traditional N-best model. Effectiveness is judged here in terms of the accuracy of representation, size of representation, and ease of interface to additional knowledge sources. Although prior work has been done on evaluation of the word graph representation [3, 5, 9], most of this work is theoretical in nature and has not included systematic experimentation and comparison to alternative methods of representation.

Careful study is made of whether significantly more information is contained in word graphs as compared to N-best lists. The gain in information is determined by tracking the number of sentences throughout a corpus for which the word graph representation contains the correct sentence but the N-best list does not. For our research the recognition task is Research Management, a mid-size corpus (approximately 1000 words) containing 5000 acoustic utterances of 3000 distinct sentences.

In addition, since word graphs can be made arbitrarily large by using lengthy acoustic processing with little pruning, experiments were done which tracked the average word graph size, average word graph accuracy, and information gain (versus N-best lists of 1-10 sentences) against a wide variety of pruning control parameters.

2. SYSTEM CONFIGURATION

The configuration for the system is shown in Figure 1 below. The acoustic portion of the system is based on a multiple-mixture triphone Hidden Markov Model (HMM) [7] with a simple integrated grammar (either wordpair or bi-gram models), based on HTK Version 2.1 by Entropic [10]. Recognition is achieved using a token-passing implementation of the Viterbi algorithm, the output of which is a large recognition lattice. The language model is a Constraint Dependency Grammar (CDG) [2] ; the CDG parser is designed to parse either word graphs or individual sentences.

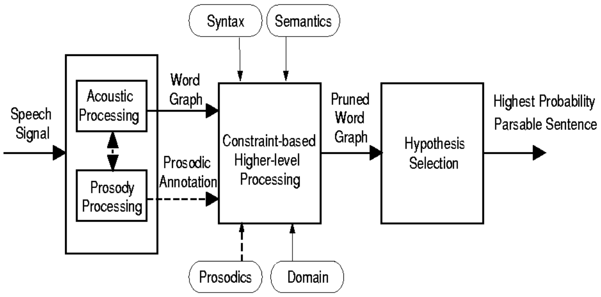

Figure 1: System Block Diagram (dashed lines are not implemented in the current system)

For clarity, we will define the following terms:

Lattice The Lattice represents the raw output of the acoustic recognizer, and is a complete record of all tokens which were not pruned during the recognition process. It may include many similar or identical paths with slight differences in word starting and ending times.

Word Graph This is directed acyclic graph representing the possible word paths through the utterance, after compressing and post-processing the word lattice. There are several equivalent definitions for word graphs; in our research the graph nodes represent words and connecting arcs represent word transitions. The graph may be re-scored and pruned to incorporate additional knowledge sources, thus decreasing the total number of paths.

N-best Sentence List This is a list of the top N most likely sentence paths, produced by searching the lattice.

3. REPRESENTATION ISSUES

3.1. Pruning Mechanisms

Pruning is typically performed to control lattice growth during recognition. All pruning methods are applied to the lattice itself and therefore affect the N-best list and word graph in identical ways. The pruning variables include:

- **Beam Width:** As tokens pass through the recognition network, the total number of active word models is limited by a beam width mechanism. The difference between the log probability of each active model and the current maximum log probability is the determining factor in this pruning method.
- **Maximum Active Models:** Similar to beam width pruning in that it works by limiting the number of active word models, this method utilizes a hard ceiling on the number of models allowed to be active at any point in the utterance.
- **Word End Likelihood:** This is also a beam width mechanism, but one which considers only models labeled as word-end nodes within the recognition network, thus allowing pruning to happen at the word level rather than the phoneme level.
- **Number of Tokens:** Pruning with this method is implemented by starting the recognition process in each state with multiple tokens rather than just one, allowing for a higher branching factor in the lattice.

The measures by which the effect of these pruning variables can be determined include primarily lattice size (number of word nodes) and lattice accuracy (defined to be the percentage of lattices which contained the correct sentence as a possible path). Experiments were run which varied each of the pruning variables individually, while holding all other factors to an empirically established baseline point.

3.2. Post-processing Techniques

In addition to the pruning methods, some post-processing can be done which decreases the average size of the lattice while maintaining all possible lattice paths. This compression is possible because identical paths are represented more than once in the lattice due to differences in word starting and ending times. These paths may be combined in a post-processing step.

Our algorithm identifies all path-identical sub-graphs by finding and compressing node pairs which represent identical words and have either identical precursor lists or identical successor lists (or both). Recursive application of this technique ensures that identical sub-graphs of any size will be compressed, giving the smallest possible graph that still contains the same word-paths as the original lattice. Using this compression technique, the word graphs for our experiments were an average of 60% smaller than the original lattices.

Post-processing techniques may also be used to handle lexical issues between the recognizer and the parser, such as contractions and proper nouns. In our system, all contractions are identified in the word graph and split into multiple nodes, while proper nouns are identified and compressed (subject to suitable path constraints) into single nodes. Techniques such as these may either be implemented as separate processing tasks or incorporated directly into the parser.

4. RESULTS

The entire Resource Management corpus (roughly 5000 separate utterances of 3000 distinct sentences) was evaluated for each combination of parameters.

As stated earlier, since pruning methods have been applied directly to the recognition lattice, the N-best sentence list is a subset of the word graph. To quantify the amount of information gained by using the word graph representation, we compute the Information Gain as the number of sentences for which word graphs contain the correct utterance and N-best sentences do not, i.e.:

$$\text{Gain } G = W - N \text{ , where}$$

W = No. of word graphs containing correct utterance

N = No. of N-best lists containing correct utterance

Lattice accuracy, defined as the percentage of lattices which contain the correct sentence, and lattice size, defined as the number of lattice nodes, are also tracked for all cases. Both word-pair and bi-gram grammar models were considered.

4.1. Pruning Results

4.1.1. Beam Width Pruning

The beam width used in this set of experiments was adjusted from 50 (very tight pruning) to infinity (no pruning). The impact on average lattice size was significant, growing from 13.5 to 107.0 nodes with a word pair grammar and from 14.9 to 495.0 nodes with a bi-gram grammar. As the pruning was decreased, a small but increasing number of sentences in the set were contained in the word graph but not the N-best list. At most, G reached 20 sentences, representing 0.39% of the corpus. These results are shown below in Figure 2.

4.1.2. Maximum Active Model Pruning

The maximum number of active models was varied from 25 to infinity, yielding an average lattice size ranging from 23.3 to 58.6 in the word pair case and from 81.6 to 241.0 in the bi-gram case. With this method, G peaked at 46 (0.89% of the corpus). Overall, this approach yielded the strongest data for word graph usage, especially considering that the average lattice size was smaller than that obtained using beam width pruning. Results for this approach are shown below in Figure 3.

4.1.3. Word-end Pruning

The word-end beam width level was adjusted from 25 to infinity. This change, however, did not result in an increase in G, which flattened out at a level of 6 (0.12% of the corpus) and stayed there throughout the sequence of experimental runs. Impact on lattice size was negligible, moving from an average size of 22.5 up to 31.0.

4.1.4. Token Pruning

The number of tokens per state was increased from 2 up to 10; however, as with the word-end pruning, the changes had little impact on G, which stayed at 9 (0.18% of the corpus) through the experiment. Again, lattice sizes were fairly constant, averaging from 24.8 to 35.4.

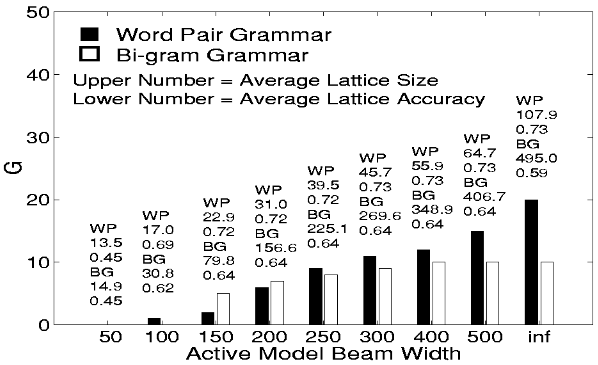

Figure 2: Information Gain vs. Beam Width

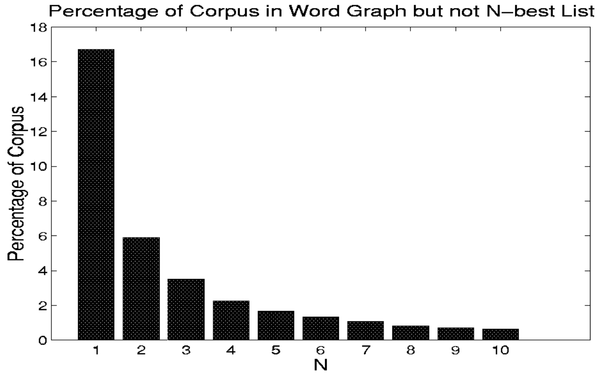

Figure 4: Representation capability

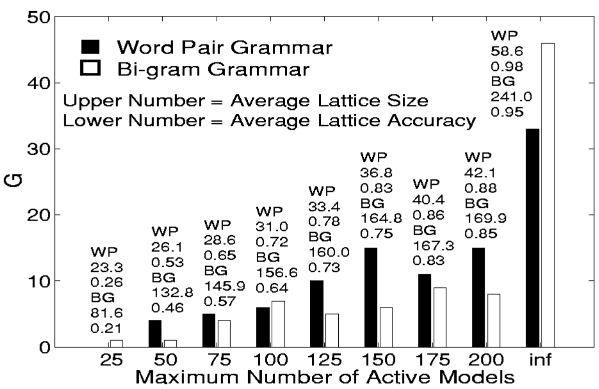

Figure 3: Information Gain vs. Number of Active Models

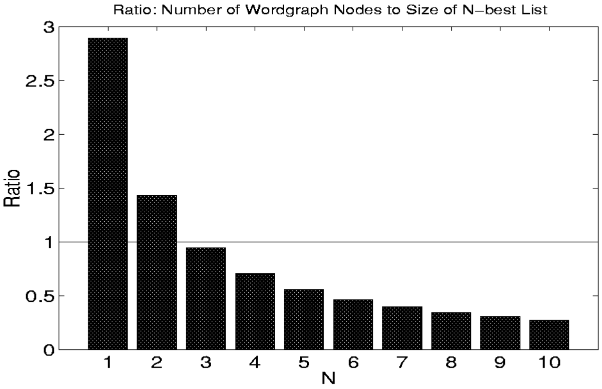

Figure 5: Representation size

4.2. Representation Issues

To summarize the overall impact of the information gain due to the word graph representation, Figure 4 displays the information gain (as a percentage of the sentences in the corpus) for increasing length N-best lists, for the best case experiment (word-pair grammar, infinite maximum active models). Figure 5 shows the relative size of the word graphs and N-best lists.

From the above data it is clear that the two first two pruning methods, which affected the number of active models, were the predominant factors in causing a change in the word graph efficacy. These were also the methods which had the greatest impact on total lattice size. Since smaller lattices are desireable for time complexity reasons, it is important to know whether small lattices and high accuracy can by achieved simultaneously.

Figure 6 shows graph accuracy versus pruning levels, while Figure 7 shows the average word graph size. Following these, Figures 8 and 9 show the correlation scatterplots between word graph size and information gain and between word graph accuracy and information gain, with correlation coefficients of 0.7953 and 0.6745, respectively. Together, these figures show that experiments giving the largest word graphs are not necessarily those giving the highest accuracies or largest information gains.

5. CONCLUSION

Results indicate that word graphs do offer a clear representation advantage. The degree of this advantage is tied to the type of recognition model, its accuracy, and the degree of pruning. Word graph sizes were manageable even at lower pruning levels, and information gain at these levels varied from 17% for the 1-best case to a little under 1% for the 10-best case. Although information gain certainly correlates with word graph size, the graphs which had the highest overall accuracies (peaking at a sentence-level accuracy of 98%) were not in fact the largest ones. This suggests that high accuracy and tractable word graph sizes are mutually achievable. Results also suggest that the importance of word graph representations will likely grow with higher vocabulary and higher complexity tasks.

The experiments to date have examined the effectiveness of N-best and word graph representations in the context of an overall language processing system. Word graphs clearly have an advantage in compactness, since graphs are smaller than lists (in terms of number of words), yet N-best lists are retrievable directly from the graph for any N. This allows both aggregate and stream processing approaches to be supported.

Future work will include similar experiments on larger vocabulary corpora with varying acoustic and language parsing models. In addition, the interface mechanism can be tightened by incorporating feedback from the language model directly into the acoustic recognition. The benefits of a word graph approach are likely to increase with a tighter interface, since the word graph parser could work together with the recognizer to prune illegal sentences from the lattice during the recognition process.

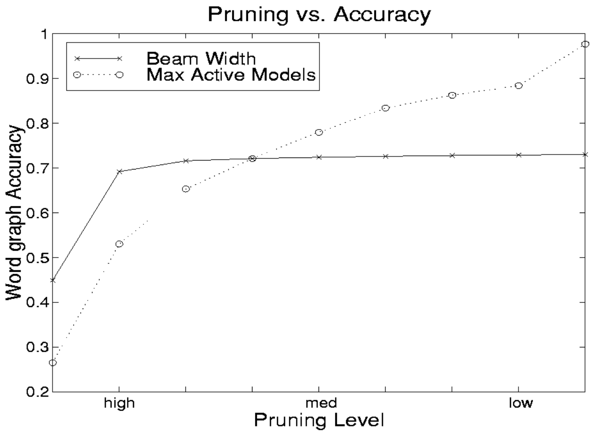

Figure 6: Pruning Level vs. Word Graph Accuracy

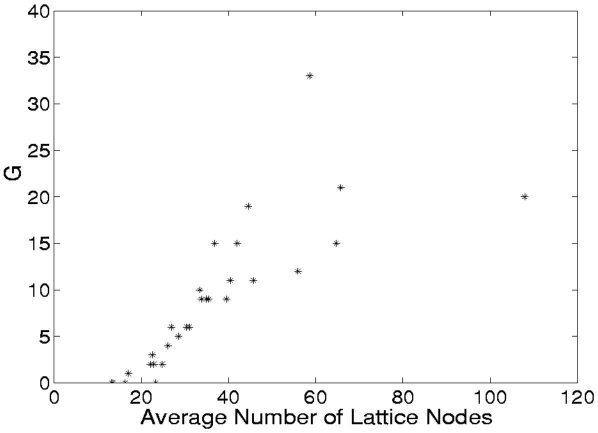

Figure 8: Information Gain vs. Lattice Size

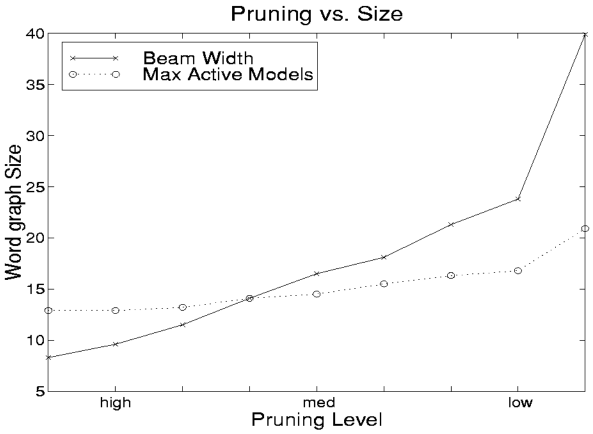

Figure 7: Pruning Level vs. Word Graph Size

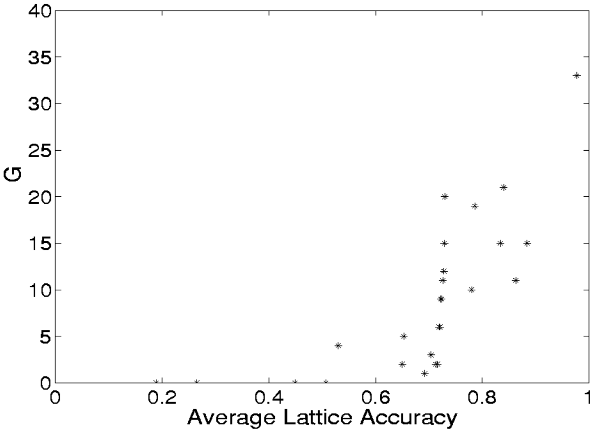

Figure 9: Information Gain vs. Lattice Accuracy

6. ACKNOWLEDGMENTS

This research is supported by the National Science Foundation under Grant No. IRI-9704358.

7. REFERENCES

[1] Xavier Aubert and Hermann Ney. Large vocabulary continuous speech recognition using word graphs. *Proc. ICASSP '95*, pages 49–52, 1995.

[2] Mary P. Harper and Randall A. Helzerman. Extensions to constraint dependency parsing for spoken language processing. *Computer Speech and Language*, pages 187–234, 1995.

[3] Thomas Kuhn, Pablo Fetter, Alfred Kaltenmeier, and Peter Regel-Brietzmann. Dp-based wordgraph pruning. *Proc. ICASSP '96*, pages 861–864, 1996.

[4] H. Ney, S. Ortmanns, and I. Lindam. Extensions to the word graph method for large-vocabulary continuous speech recognition. *Proc. ICASSP '97*, pages 1791–1794, 1997.

[5] Martin Oerder and Hermann Ney. Word graphs: An efficient interface between continuous-speech recognition and language understanding. *Proc. ICASSP '93*, pages II–119–122, 1993.

[6] S. Ortmanns and H. Ney. A word graph algorithm for large vocabulary continuous speech recognition. *Computer Speech and Language*, pages 43–72, 1997.

[7] Lawrence R. Rabiner. Tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77:257–286, 1989.

[8] Ludwig A. Schmid. Parsing word graphs using a linguistic grammar and a statistical language model. *Proc. ICASSP '94*, pages II–41–44, 1994.

[9] Tohru Shimizu, Hirofumi Yamamoto, Hirokazu Masataki, Shoichi Matsunaga, and Yoshinori Sagisaka. Spontaneous dialogue speech recognition using cross-word context constrained word graphs. *Proc. ICASSP '96*, pages 145–148, 1996.

[10] Steve Young, Julian Odell, Dave Ollason, Valtcho Valtchev, and Phil Woodland. *The HTK Book*. Entropic Cambridge Research Laboratory Ltd., 2.1 edition, 1997.