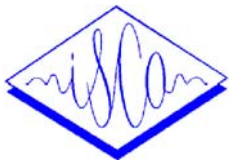

4th International Conference on Spoken Language Processing (ICSLP 96)
Philadelphia, PA, USA
October 3-6, 1996

INCREMENTAL GENERATION OF WORD GRAPHS*

G. Sagerer, H. Rautenstrauch†, G. A. Fink, B. Hildebrandt, A. Jusek, F. Kummert

University of Bielefeld, Technical Faculty, Postfach 100131, 33501 Bielefeld, Germany
e-mail: sagerer@techfak.uni-bielefeld.de

ABSTRACT

We present an algorithm for the incremental generation of word graphs. Incremental means that the speech signal is processed left-to-right by a time synchronous Viterbi algorithm and word hypotheses are generated with some delay to Viterbi decoding. The incrementally generated word hypotheses can be used for early interaction between linguistic analysis and acoustic recognition. Therefore, it is possible to derive acoustic constraints from linguistic restrictions dynamically.

1. INTRODUCTION

Speech recognition and speech understanding techniques have already achieved a point that they can provide tools for man-machine interaction as well as assist in man-man dialogs for translation purposes. Nearly all speech understanding systems divide the overall complex task at least into two main processing units. A word recognizer deals with the detection of most probable words and word chains which fit the incoming signal. These data are passed to a linguistic unit which make use of structural syntactic, semantic, pragmatic, and dialog constraints in order to extract the intended meaning of an utterance. Beside others, the following reasons support this strategy:

- It is possible to use adequate processing paradigms for each step. E.g., statistical approaches based on HMMs for recognition and knowledge based techniques for understanding.
- The search effort for the most expensive understanding units can be reduced by only applying them to the very best hypotheses provided by the recognition unit.

In order to achieve the latter goal, i.e. only creating as few word hypotheses with liable accuracy as possible, many recognition algorithms use two or three passes to optimize the word hypotheses [1, 16]. They also incorporate statistical knowledge about the domain of the system. A priori language models such as word, bigram, or trigram probabilities are estimated to reflect the structural properties of utterances. However, the main disadvantages are evident. Even if the recognition process is near to real time, in the sense, that, e.g., an utterance lasting 5 seconds only needs 5 seconds to be processed, it is only guaranteed that the hypotheses are available 10 seconds after the utterance started. The interpretation process can not be started before the recognition unit reached its end. Additionally, it is not possible to use intermediate interpretations of the understanding process to restrict the search for word hypotheses. The statistical language model does not necessarily coincide with the linguistic knowledge bases, although it covers a lot of the different aspects of linguistic evaluations. From human speech perception and cognition it is well known that recognition and understanding are coupled and restrict each other [7, 2]. Furthermore, man-man dialogs show that turn taking does not depend on the end of constituents or even sentences. Disconnections occur in order to establish mutual agreements on the meaning of previous words and constituents.

These remarks motivate our investigations on *incremental* word recognition algorithms. Fig. 1 outlines the general idea. Already while an utterance is being produced word hypotheses are generated. A short time delay has to be accepted. If a certain time t is reached, words ending in a time interval $[t - s - w, t - s]$ are established as hypotheses. In order to achieve robustness not only the optimal sub-chain but a word graph is constructed. This process will be explained in section 2. Currently, there are only a few speech understanding systems using incremental techniques for the recognition task, e.g. [8, 5, 4, 6]. Except the early transmission of hypotheses to the further processing modules, an closer interaction between recognition and understanding will be performed.

2. WORD GRAPHS

Several interfaces between the recognition and the understanding unit can be used: the optimal word chain with

*This work was funded by the German Federal Ministry of Education, Science, Research and Technology (BMBF) in the framework of the VERBMOBIL Project under Grant 01IV102G/7. The responsibility for the contents of this study lies with the authors.

†Now with SAP-AG, Neurotstr. 16, 69190 Walldorf, Germany

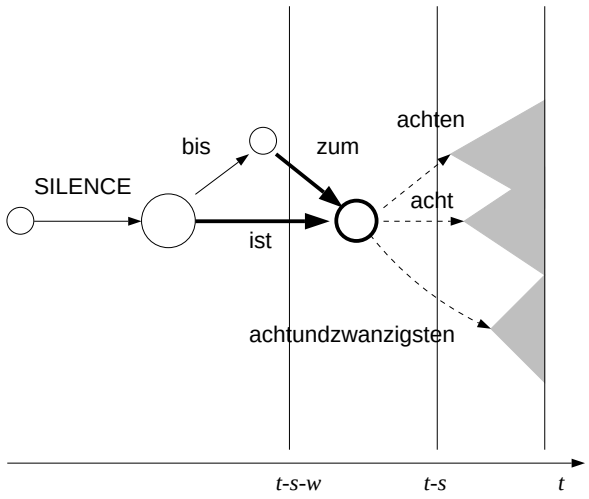

Figure 1: Incremental Recognition

respect to a given utterance and a HMM including a language model, the n-best word chains [14, 15], a set of word hypotheses [13], or a word graph [10]. While the best chain or the n-best ones are easy to parse and to interpret, errors are hard to avoid or to correct if only the best chain is used. Increasing robustness can be achieved by using the n-best strategy but here n must grow exponentially with the number of words in an utterance if the recognition rate should not decrease. A linear relationship between words spoken and word hypotheses is achieved by word graphs. They are labeled directed graphs. Each link represents a word hypothesis with an associated score. A node represents a unique time interval. The incoming and outgoing word hypotheses must end respectively start within this interval. Therefore, the duration of a node should be as short as possible to avoid larger overlaps and gaps but also cover enough time to establish a path through the graph which represents the correct sequence of words.

In the usual case of multi-pass recognition the word graph can be optimized as a whole but not during the first pass by using previous results of the n-best word chains. Contrarily, in an incremental approach the nodes must be determined dynamically according to both the actual hypotheses in the search space and the language model. Furthermore, because the continuation is unknown "dead-ends" must be taken into account.

3. AN INCREMENTAL SEARCH ALGORITHM

Using HMMs for speech and language modeling requires an efficient search algorithm which has to deal with a huge number of states and state transitions. A state transition reflects one step in time from some t to $t+1$. In order to reduce the overall complexity not all possible states at a certain time step but only the local best ones are continued. Although this beam search paradigm does not guarantee to achieve the optimal solution, it provides a good balance between search effort and the quality of the resulting word hypotheses. The search space can be constructed time synchronously. But immediate generation of word hypotheses would cause serious problems. Locally optimal paths, and only they could be used for decisions, may result in globally bad solutions or they can not be continued to the end of the utterance at all. Generating more hypotheses could overcome these problem but neglect the overall goal. Therefore, algorithms are widely investigated which process the speech signal forward and backward in at least two passes. nodes. Examples of such algorithms are presented in [11, 16].

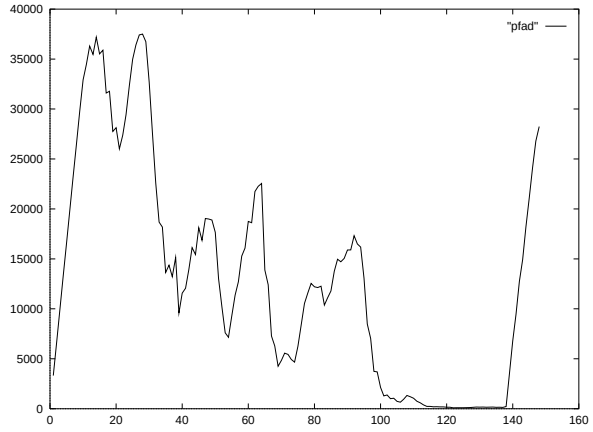

Figure 2: Size of a search space during the utterance *bis zum achtundzwanzigsten*

In our approach we combine the incremental ability of time synchronous recognition algorithms with a more global view on local optimization of word hypotheses. Similar to the technique suggested in [3] for training purposes a sliding window technique is used. To reduce the problem of immediate local decisions a delay time is accepted. The strategy is based on the observation that the density of word hypotheses in the search space increases at word boundaries [9] (see Fig. 2 for an example). This fact is turned into account for the incremental search:

- No hypotheses shall be generated at word boundaries, because important restrictions become evident after a short time delay.
- Within words search space is reduced. The quality of previous words can be checked more reliable. *Expressive* hypotheses are selected.

Therefore, the search space currently looked at is reduced to a short time interval, and hypotheses are generated for those time slots already processed and not in the *analysis window*.

Fig. 3 illustrates the algorithm. The analysis window is the time interval $[t-s, t]$. Words ending before $t-s$ which have

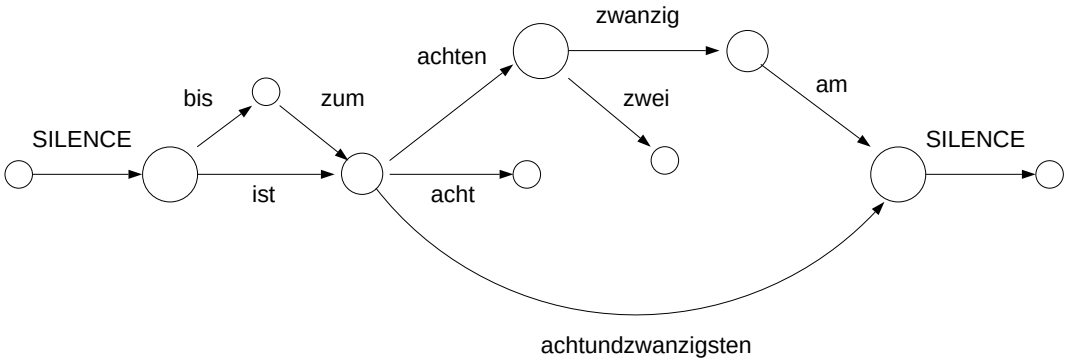

Figure 4: Word Graph at the End of an Utterance

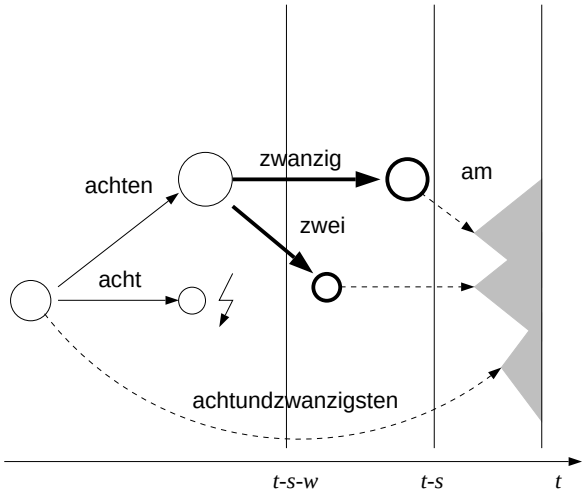

Figure 3: Illustration for the Search Algorithm

a path extension to t form new hypotheses, e.g., *zwanzig, zwei*. Although most of the "dead ends" can be omitted, the overall continuation of intermediate word chains to the end of an utterance is not guaranteed, in the example the word *acht*. The word *achtundzwanzigsten* does not form a hypothesis at time t because it was not finished at least at time $t - s$. The delay time is at least s frames. This can be slightly increased due to the construction of a word graph node (see next section).

4. CONSTRUCTION OF WORD GRAPH NODES AND LINKS

The presented search algorithm enables the incremental generation of a tree of word chains (see Fig 3). In order to achieve a word graph, start and end times of hypotheses must be connected to logical nodes. Due to the goal to produce a word graph as compactly as possible, gaps and overlaps between words which are connected by one node of the graph are accepted. Again, it is not possible to fix the time interval of any node with knowledge about the entire utterance. But either an a priori definition of the intervals is not suitable, because incremental processing requires a dynamic adaptation of the width of word graph nodes.

This is provided by the following steps. First, an initial hypotheses list is constructed:

- Each word hypothesis h_j which ends at time t is assigned to a list l_t.
- If hypotheses in l_t represent the same word and start in the same node, only the best scored one is further processed.

Second, the currently considered time t is compared to already fixed word graph nodes. The list l_t forms new incoming links for a node n if t fits the time interval of this node. In addition, if one of the following conditions is fulfilled the start respectively end time of the interval for a word graph node n will be adjusted and the members of l_t will be allocated as incoming links to n.

$$e(n) - a(n) + (t - e(n)) < \Delta t \quad \text{if} \quad t > e(n) \qquad (1)$$

$$e(n) - a(n) + (a(n) - t) < \Delta t \quad \text{if} \quad t < a(n) \qquad (2)$$

In the equations, $e(i)$ is the end time and $a(i)$ the start time of the time interval of node i and Δt a trained threshold. In all other cases a new node with time interval $[t, t]$ is introduced for the members of l_t.

Fig. 4 depicts a word graph at the end of an utterance. Each word hypothesis ending in node n is available for further processing with a time delay of $s + \Delta t$. Therefore, top down restrictions for expansions of a word chain can be taken into account in the recognition process if they enable predictions after this time intervall.

5. FIRST RESULTS AND CONCLUSION

The incremental speech recognition unit as described above has been tested within the VERBMOBIL evaluation. The test set consists of spontaneous human-human dialogs. A bigram statistical language model has been provided. For the training of the HMM parameters we used the ISADORA system [12]. A maximum of 5 competing links at time t within a word graph has been determined. The recognition vocabulary covered 3300 words, the language model reduced perplexity to 107. The results of our incremental algorithm — word accuracy 60% — still differ from multi-pass algorithms — the best one achieved 85%. But different aspects must be taken into account. The algorithmic time delay of 200ms requires decisions with a very short look ahead. The incremental algorithm produces "dead ends" in the word graph which can be omitted by multi-pass algorithms. However, one of the main advantages, the more flexible interaction with other modules, was not enabled in test.

Previous work in the construction of more restrictive and dynamical language models which make intensive use of the linguistic knowledge base, has already shown that the improvement of recognition rates by top down restrictions [4]. Therefore, a combination of the presented algorithm with a dynamic construction of language models can enable the development of cognitive adequate speech understanding systems.

6. REFERENCES

1. F. Alleval, X. Huang, and M.-Y. Hwang. An Improved Search Algorithm Using Incremental Knowledge fro Continuous Speech Recognition. In *Proc. Int. Conf. on Acoustics, Speech and Signal Processing*, volume 2, pages 307–310, Minneapolis, 1993.
2. G. Altmann and M. Steedman. Interaction with context during human sentence processing. *Cognition*, 30:191–238, 1988.
3. G. Boulianne, P. Kenny, L. Lenning, D. O'Shaughnessy, and P. Mermelstein. HMM Training on Unconstrained Speech for Large Vocabulary Continuous Speech Recognition. In *International Conference on Spoken Language Processing*, pages 229–232, Banff, Canada, 1992.
4. G. A. Fink, F. Kummert, and G. Sagerer. Speech Recognition using Semantic Hidden Markov Models. In *Proc. European Conf. on Speech Communication and Technology*, pages 1571–1574, Berlin, 1993.
5. D. Goodine, S. Seneff, L. Hirschmann, and M. Philips. Full Integration of Speech and Language Understanding in the MIT Spoken Language System. In *Proc. European Conf. on Speech Communication and Technology*, pages 845–848, 1991.
6. A. Hauenstein and H. Weber. An Investigation of Tightly Coupled Time Synchronous Speech Language Interfaces Using a Unification Grammar. In P. McKevitt, editor, *AAAI-94 Workshop Program: Integration of Natural Language and Speech Processing*, pages 42–49, Seattle, Washington, 1994.
7. W. Marslen-Wilson and A. Welsh. Processing interactions and lexical access during word recognition in continous speech. *Cognitive Psychology*, 10:29–63, 1978.
8. R. Moore, F. Pereira, and H. Murveit. Integrating Speech and Natural-Language Processing. In *Speech and Natural Language Workshop*, pages 243–247, Philadelphia, 1989.
9. H. Ney. Modeling and Search in Continuous Speech Recognition. In *Proc. European Conf. on Speech Communication and Technology*, pages 491–498, Berlin, 1993.
10. M. Oerder and H. Ney. Word Graphs: An Efficient Interface Between Continuous-Speech Recognition and Language Understanding. In *Proc. Int. Conf. on Acoustics, Speech, and Signal Processing*, volume 2, pages 119–122, Minneapolis, 1993.
11. D. B. Paul. An Efficient A* Stack Decoder Algorithm for Continuous Speech Recognition with a Stochastic Language Model. In *Proc. Int. Conf. on Acoustics, Speech and Signal Processing*, pages 25–28, San Francisco, 1992.
12. E. G. Schukat-Talamazzini. *Automatische Spracherkennung*. Vieweg, Wiesbaden, 1995.
13. G. Schukat-Talamazzini and H. Niemann. Generating Word Hypotheses in Continuous Speech. In *Proc. Int. Conf. on Acoustics, Speech, and Signal Processing*, pages 1565–1568, Tokyo, 1986.
14. R. Schwartz and S. Austin. Efficient, High-Performance Algorithms for N-Best Search. In *Speech and Natural Language Workshop*, pages 6–11, Hidden Valley, Pennsylvania, 1990. Morgan Kaufmann.
15. R. Schwartz, S. Austin, F. Kubala, J. Makhoul, L. Nguyen, P. Placeway, and G. Zavaliagkos. New Uses for the N-Best Sentence Hypotheses within the BYBLOS Speech Recognition System. In *Proc. Int. Conf. on Acoustics, Speech, and Signal Processing*, volume 1, pages 1–4, San Francisco, 1992.
16. M. Woszczyna, N. Coccaro, A. Eisele, A. Lavie, A. McNair, T. Polzin, I. Rogina, C. P. Rose, T. Sloboda, M. Tomita, J. Tsutsumi, N. Aoki-Waibel, A. Waibel, and W. Ward. Recent Advances in JANUS: A Speech Translation System. In *Proc. European Conf. on Speech Communication and Technology*, pages 1295–1298, Berlin, 1993.