ADVANCED SEARCH SPACE PRUNING WITH ACOUSTIC LOOK-AHEAD FOR WFST BASED LVCSR

David Nolden, Ralf Schlüter, Hermann Ney

Chair of Computer Science 6, RWTH Aachen University
Ahornstr. 55
D-52056 Aachen, Germany
{nolden, schlueter, ney}@cs.rwth-aachen.de

ABSTRACT

In this work we show how some concepts already known from dynamic network decoding can be used to improve the efficiency of WFST based decoders. First we apply the concept of acoustic look-ahead to a WFST based decoder, and then we analyze the applicability of LM state pruning, a well motivated pruning method which is fundamental to token-passing decoders. The structure of the composed WFST search network makes it difficult to motivate advanced pruning methods, and consequently it is difficult to achieve a real reduction in search space. Nonetheless, we show how LM state pruning can be applied to WFST based decoders to improve their efficiency.

The search space can be reduced by up to 50% at equal precision through acoustic look-ahead. Since our decoder follows a dynamic composition approach, the advantage in search space does not fully transfer to the RTF, which can be reduced by around 20% through acoustic look-ahead, and additional 5% through LM state pruning.

Index Terms— LVCSR, WFST, look-ahead, pruning

1. INTRODUCTION

Weighted finite state transducers (WFST) have become the dominant decoding approach in large vocabulary continuous speech recognition (LVCSR) research. While it has been shown that advanced dynamic network decoders perform similarly regarding runtime [1, 2], WFST based decoders allow an unprecedented level of flexibility, due to the abstract representation of the different knowledge sources, and due to the separation of the decoder implementation from that of the search network.

In WFST decoders, the n-gram language model (LM) transducer G, the lexicon transducer L, and the context dependency transducer C are composed to $C \circ L \circ G$ using standard finite-state operations to create a joint search network [3] (determinization and minimization yield a relatively compact network).

If a large LM and lexicon are used, then the static composition of the $C \circ L \circ G$ transducer can become very expensive, and the resulting transducer may require large amounts of memory. Therefore, on-the-fly composition has been proposed [4], where the $C \circ L$ and the G transducer are created and optimized in a preprocessing step, and then composed to $C \circ L \circ G$ on-the-fly during decoding, leading to a search strategy similar to dynamic network decoding [2] but based on the WFST terminology.

In dynamic network decoders, usually a wide range of pruning methods is applied, some of which are just tricks required in a specific decoder implementation, others of which allow a systematic reduction of the search space at equal precision [5]. Furthermore, acoustic look-ahead is a very effective method to improve the efficiency of dynamic network decoders [6].

In this work, we evaluate the effectiveness of acoustic look-ahead in the WFST decoding architecture with on-the-fly composition, and we use the theory of anticipated path recombination [7] to implement LM state pruning, which is a common pruning method in dynamic network decoders [1, 2], but does not fit into the WFST framework at first glance.

2. DECODER

Our decoder [8] is based on the OpenFst toolkit [9] using on-the-fly composition with weight- and label pushing [4]. The L and C transducers are determinized and minimized, and no further transducer operations are applied after composition. The final $C \circ L \circ G$ transducer has tied HMM input labels and word output labels. Each input label corresponds to a specific HMM state sequence, and the HMM state sequences are expanded on-demand during decoding.

The search space is cascaded into active states, active arcs, and active HMM states (each active HMM state belongs to an active arc, each active arc belongs to an active origin state).

The decoder uses *global beam pruning* to focus the search: All HMM state hypotheses with a score worse than the best one plus a specific threshold are discarded.

The decoder applies global beam pruning at many points: Prospectively while expanding HMM transitions, explicitly after computing acoustic scores, and while expanding cross-arc transitions and epsilon arcs. The beam applied at cross-arc transitions is tighter than the beam applied within arcs, to reduce the costs of on-the-fly composition.

2.1. Mapping into $C \circ L$

Each state in the composed transducer can be associated to one corresponding state in the $C \circ L$ transducer and one state in the G transducer. The alternative epsilon sequencing filter is used for composition [4], which matches the LM back-off arcs in G right after word labels, and avoids any back-off matches until the next word label has been matched. Thus, a subgraph of the $C \circ L$ transducer is spanned for each state in G leading to all word labels which have a match in the specific LM context. The context-dependent subgraph of $C \circ L$ is only *complete* for LM contexts that have a match for *every* word label, which is usually only the case in the unigram

state, which can be reached through LM back-off arcs following any word label (usually at least a unigram probability is available for each word in the lexicon).

The standard OpenFst on-the-fly composition algorithm allows identifying the corresponding state in $C \circ L$ for each state in the composed transducer. However a mapping from arcs in $C \circ L \circ G$ to arcs in $C \circ L$ is usually not possible, because the arcs are filtered during composition, and thus their indices are changed. For our LM state pruning and acoustic look-ahead approaches, we require to map each arc into $C \circ L$, therefore we have modified the OpenFst on-the-fly composition algorithm to store the $C \circ L$ arc index into every composed arc. The efficiency of composition is affected only insignificantly by this modification.

3. ACOUSTIC LOOK-AHEAD

Acoustic look-ahead denotes the approximative pre-evaluation of the acoustic model to improve the focus of beam search. In [6] two approximations of acoustic look-ahead have been proposed and compared to the perfect look-ahead: The *temporal approximation* and the *model approximation*, both of which can be combined to achieve about 70% of the reduction in search space achievable through perfect acoustic look-ahead, at negligible runtime costs.

Temporal approximation: At a specific time frame, due to specific attributes of the training procedures and speech signals, the local acoustic emission score of an HMM state hypothesis for that time frame can be considered an approximation of the expected emission scores for the next time frames. For pruning, the emission score of each HMM state hypothesis is scaled by the *temporal look-ahead scale*, and added to the overall score of the hypothesis.

Model approximation: A limited set of very simple look-ahead models are assigned to each HMM state in the search network and trained so that they represent the acoustic emission models of the successor HMM states as closely as possible. The assignment and training happens iteratively based on expectaction maximization with simple single-Gaussian models that are derived from the original acoustic models. For pruning, the simplified models are evaluated on future acoustic observations, scaled by the *model look-ahead scale*, and added to the HMM state hypothesis scores.

The advantage of the temporal approximation is that the complex original acoustic models are used, while the advantage of the model approximation is that both the models and the acoustic observations actually correspond to the *future*. Both methods can be combined, using individual scales for each.

To apply acoustic look-ahead in our WFST based decoder, we pre-compute the acoustic look-ahead models for model approximation on the $C \circ L$ transducer equivalently to the single-word search network used in [6]. During decoding, we then map each arc into the $C \circ L$ transducer to get the corresponding look-ahead model (see Subsection 2.1).

We incorporate the look-ahead scores at every pruning step possible to increase the precision of the pruning (eg. while expanding HMM transitions, after computing scores, while expanding cross-arc transitions, and while expanding epsilon arcs). Only model-approximated look-ahead is used while expanding HMM transitions, because acoustic scores for the current timeframe were not yet computed at that point.

4. LM STATE PRUNING

LM state pruning is a common pruning method in dynamic network decoders [1, 2, 5]. There, it helps to reduce the overall number of different LM contexts which need to be maintained, and thus reduces the number of LM look-ahead tables which need to be calculated. It also helps improving the relationship between the size of the search space and the precision [5], and it is one of the few well-motivated pruning methods.

Consider a state hypothesis (s, h, q) in a dynamic network decoder with LM history h, score q, and network-state s. The state hypothesis (s, h, q) is removed if there is another state hypothesis (s, h', q') on the same network state s with score q' better by a specific threshold.

Motivation: If two state hypotheses share a state s in the single-word search network, then the relative probabilities of all followup paths through the network leading to a sentence-end can only be discriminated by the LM (the acoustic model assigns equal probabilites to equal HMM state alignments). The acoustic model has a much stronger influence on the overall hypothesis probabilities than the LM, thus a majority of the variability that can discriminate the followup hypotheses has fallen away. Therefore the LM state pruning threshold can typically be much tighter than the global beam pruning threshold without introducing additional errors.

The minimized single-word search network of dynamic network decoders corresponds to the $C \circ L$ transducer in the WFST framework, and LM histories h correspond to states in the G transducer. Following this analogy, we can directly transfer LM state pruning into the WFST framework. However, composition using the alternative epsilon-sequencing filter (see Subsection 2.1) invalidates the motivation of LM state pruning: The set of acoustic followup paths through the $C \circ L$ transducer may be unequal for different network states corresponding to the same $C \circ L$ state, because the LM back-off is matched only after word labels, and then different filtered subgraphs of $C \circ L$ are spanned for different LM states. However, since the unigram back-off is reachable from every state in G, for every $C \circ L$ path which is filtered away by the epsilon-sequencing filter at a higher-order state in G, there is an equivalent path going through the unigram state of G which was reachable by backing-off right after the previously crossed word label. Thus, we can avoid problems by performing LM state pruning only relative to state hypotheses which correspond to the unigram state of G, consistently with the theory of asymmetric path anticipated path recombination introduced in [7].

We define the LM state pruning in the WFST framework as follows:

The state hypothesis (s, q) is removed if there is another state hypothesis (s', q') with $G(s') = G_{unigram}$, $CL(s) = CL(s')$ and $q > q' + f_{LM}$.

Where s is an HMM state in the composed network, q is a score, $G_{unigram}$ is the unigram grammar state, f_{LM} is the LM state pruning threshold, $G(s)$ is the state in the G transducer corresponding to the composed state s, and $CL(s)$ is the corresponding state in the $C \circ L$ transducer.

We expand HMM states dynamically from the arcs during decoding, therefore we need arc-mapping (see Subsection 2.1) to map HMM states into $C \circ L$.

To further improve the effectiveness of LM state pruning, we once statically expand the $C \circ L$ transducer up to HMM state level, then apply a minimization transformation to that

HMM network, and use the generated minimizing mapping for LM state pruning to potentially tie HMM states together which belong to different arcs.

5. EXPERIMENTAL RESULTS

We perform our experiments on the first speaker-independent pass of the RWTH Aachen Quaero English ASR system [10]. The lexicon is comprised of 158k words with 180k pronunciations, modeled by 45 phonemes and 6 non-speech phones, and the 4-gram LM is composed of 50M n-grams. The acoustic model is comprised of 4501 Gaussian mixture models with a globally tied covariance matrix and 1M mixture densities. The test corpus consists of 1482 segments with a duration of 3.4h and about 36k spoken words.

The acoustic scores are computed efficiently using quantized features, temporal batching, and Gaussian preselection. The Gaussian densities are clustered into 256 clusters and at each time frame only the closest 32 clusters are considered (batching and preselection together reduce the effort of acoustic scoring to approximately one third at equal error rate).

Real time factors (RTF) were measured on a 4-core AMD Opteron 2220 machine with 2.8Ghz and 16GB of memory (without parallelization).

5.1. LM state pruning unigram approximation

We have verified that our unigram approximation of LM state pruning (see Section 4) does not negatively impact the effectiveness of the pruning, by modifying the WFST composition algorithm to make the unigram approximation unnecessary.

The modified composition algorithm inserts additional epsilon LM back-off arcs in all states of the composed search network which do not correspond to the unigram state of the G transducer. The additional back-off arcs only marginally affect the actual search space, but they make the reachability of word labels in the composed WFST equivalent to the reachability in the $C \circ L$ transducer, thus LM state pruning without the unigram approximation becomes well-motivated.

However we observed no significant difference in the effectiveness of LM state pruning with the modified composition algorithm and without unigram approximation vs. LM state pruning with unigram approximation and the standard WFST composition algorithm, which shows that the unigram approximation of LM state pruning is as effective as the original LM state pruning. The additional back-off arcs inserted by the modified composition algorithm make the decoding process slightly less efficient due to the required expansion of additional epsilon arcs during decoding, thus we use the standard composition algorithm for all further experiments.

5.2. Method evaluation

Figure 1 shows the relationships between WER and search space (measured in active HMM state hypotheses per time frame) at varied global beam pruning thresholds, using previously optimized look-ahead scales and LM state pruning thresholds. By adding temporally approximated acoustic look-ahead, the relationship is improved by nearly 50% for the higher error rates, but the relationship is close to the baseline for lower error rates. When adding acoustic look-ahead based on the model approximation on top of the temporal approximation, the relationship is improved for the better error rates too, and an improvement of nearly 50% is achieved on all error rates.

LM state pruning further slightly improves the relationship, but the difference is only significant for the better error rates. Generally the impact of LM state pruning on search space is much lower than previously observed on our dynamic network decoder [5], where LM state pruning achieved a reduction of 10 to 20% at equal precision. The most likely reasons for the underperformance of LM state pruning on the WFST based decoder are: 1. The backing-off recombination, which is present in the WFST-based decoder but not in the dynamic network decoder, recombines paths with different LM contexts which would otherwise be affected by LM state pruning, and 2. The $C \circ L$ transducer is not minimized regarding the number of HMM states, but regarding network arcs, which leads to increased redundancy regarding the HMM states (this redundancy can be cirvumvented by inserting epsilon arcs, but that would lead to less efficient decoding and composition). We alleviate the second problem by locally minimizing the HMM network for LM state pruning (see Section 4), but we cannot completely compensate it.

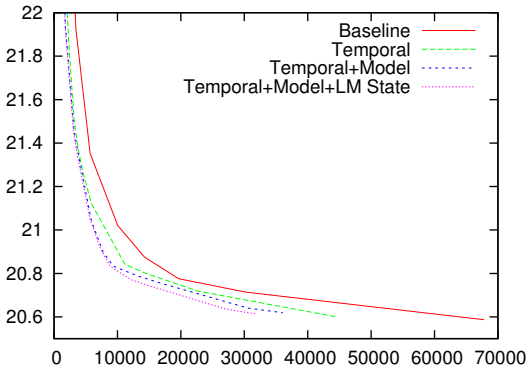

Fig. 1. WER vs. search space.

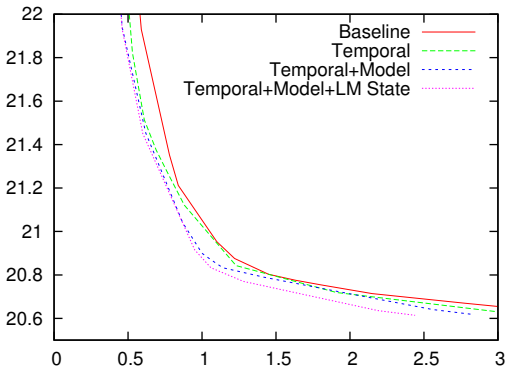

Fig. 2. WER vs. RTF.

Figure 2 shows the relationship between WER and RTF. By adding temporally approximated acoustic look-ahead to the baseline, the RTF can be improved by approximately 10% at equal precision for the higher error rates. By further adding model-approximated acoustic look-ahead, an improvement of around 20% can be achieved relative to the baseline for most error rates. Adding LM state pruning further improves the relationship by another 5% for the better error rates. LM state pruning is more effective regarding RTF than it is regarding the search space, because it positively affects the dynamic composition of the search network (see Table 2).

Table 1. Search space statistics.

	Baseline	+Temporal	+Model	+LM state
Global beam	365	370	385	385
WER [%]	20.80	20.84	20.83	20.83
RTF	1.45	1.23	1.14	1.06
# States	584	692	636	523
# Arcs	3.5k	4.1k	3.7k	3.2k
# HMM states	18.2k	11.2k	9.4k	8.8k
# Comp. states	1.9M	1.7M	1.7M	1.5M

Table 1 shows the most important decoder statistics for a set of decoder configurations with similar word error rates (based on individual samples from the previous graphs). Acoustic look-ahead increases the scores used for pruning, thus larger beams are required to achieve similar error rates as without acoustic look-ahead. In addition to the beam size, the word error rate and the RTF, the table shows the number of composed network states per segment, the average number of active network states per time frame, the average number of active network arcs per time frame, and the average number of active HMM states per time frame. Since our decoder expands the HMM state sequences dynamically from arcs, active network states and arcs induce a specific overhead. A network arc is active if at least one of its corresponding HMM states is active, and a network state is active if at least one of its outgoing arcs is active. The most important factors regarding efficiency are the number of active HMM states and the number of composed states though (dynamic composition with weight pushing is very expensive when a large vocabulary and LM are used). Adding temporal acoustic look-ahead to the baseline significantly reduces the number of active HMM states, reduces the number of composed network states, and reduces the RTF. Interestingly, at the same time, temporal look-ahead increases the number of active network states and arcs, presumably because seemingly likely HMM states are kept active for a longer time. Both model-based acoustic look-ahead and LM state pruning significantly improve all search space statistics.

Table 2 shows a profiling of the decoder at the same operating points as used for Table 1. Overall the on-the-fly composition of $C \circ L$ with G accounts for the largest individual portion with real-time costs of 0.45. Acoustic look-ahead makes all components of the search process more efficient, specifically it helps reducing the costs of acoustic score calculation. The per-time-frame calculation of the model-based look-ahead scores accounts to merely 0.01. LM state pruning is implemented in the scorer component, where the best scores are recorded, and in the pruning component, where the pruning is applied. LM state pruning costs 0.02 in each, however those costs are compensated by the accelerated composition which is improved by 20% from 0.45 to 0.36. Due to weight pushing, the composition of states which correspond to the *early* parts of the $C \circ L$ transducer is more expensive than other parts, because more word ends are reachable from there, and thus the pushing algorithm needs to sum over more word probabilities. LM state pruning shows most of its effect in exactly this part of the search network.

6. CONCLUSIONS

We have shown that acoustic look-ahead can significantly improve the efficiency of WFST based LVCSR decoders.

Table 2. Profiling.

	Baseline	+Temporal	+Model	+LM state
WER	20.80%	20.84%	20.83%	20.83%
RTF	1.45	1.23	1.14	1.06
Acoustic Scorer	0.35	0.26	0.22	0.24
Composition	0.45	0.46	0.45	0.36
Expand HMM	0.12	0.11	0.10	0.09
Expand cross-arc	0.08	0.07	0.06	0.05
Pruning	0.04	0.02	0.01	0.03
Lookahead	0.00	0.00	0.01	0.01

Temporal acoustic look-ahead can be implemented easily in any kind of decoder, while model-based acoustic look-ahead might require some deeper changes.

For WFST decoders with a dynamically composed search network, model-based acoustic look-ahead requires slight changes to the composition algorithm, where a mapping of arcs into the $C \circ L$ transducer needs to be preserved.

For WFST decoders with statically composed search network the acoustic look-ahead information could be precomputed right on the composed graph.

While LM state pruning has a minor positive effect of only around 5% on the search space, we have shown that even WFST based decoders can profit from advanced pruning methods. Since LM state pruning reduces the costs of the dynamic WFST composition, the improvement of 5% fully transfers to the RTF.

Overall the search space was reduced by up to 50% and the RTF by up to 30% at equal precision through acoustic look-ahead and LM state pruning.

7. RELATION TO PREVIOUS WORK

To the best of our knowledege this is the first work to transfer acoustic look-ahead and LM state pruning to a WFST based LVCSR decoder.

In [6] we have introduced generalized acoustic look-ahead and have shown that it can be used to significantly improve the efficiency of a dynamic network decoder. In this work we show how the same techniques can be used to significantly improve the efficiency of a WFST based decoder. Our WFST based decoder does on-demand expansion of HMM states from network arcs, thus there is some overhead on a higher level than the HMM state hypothesis level, therefore the reduction of the search space achieved through acoustic look-ahead shows a less significant effect on the RTF in this work than it did on the dynamic network decoder, where a 30 to 50% improvement in RTF was achieved.

8. ACKNOWLEDGEMENTS

This work was partly realized under the Quaero Programme, funded by OSEO, French State agency for innovation.

Supported by the Intelligence Advanced Research Projects Activity (IARPA) via Department of Defense U.S. Army Research Laboratory (DoD / ARL) contract number W911NF-12-C-0012.[1]

[1]The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon. Disclaimer: The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of IARPA, DoD/ARL, or the U.S. Government.

9. REFERENCES

[1] H. Soltau and G. Saon, "Dynamic Network Decoding Revisited," in *ASRU*, 2009.

[2] D. Nolden, D. Rybach, R. Schlüter, and H. Ney, "Joining Advantages of Word-Conditioned and Token-Passing Decoding," 2012, ICASSP.

[3] M. Mohri, F. Pereira, and M. Riley, "Speech Recognition with Weighted Finite State Transducers," in *Handbook of Speech Processing*. 2008, pp. 559–582, Springer.

[4] C. Allauzen, M. Riley, and M. Mohri, "A Generalized Composition Algorithm for Weighted Finite-State Transducers," in *Interspeech*, Brighton, U.K., September 2009, pp. 1203 – 1206.

[5] D. Nolden, R. Schlüter, and H. Ney, "Extended Search Space Pruning in LVCSR," 2012, ICASSP.

[6] D. Nolden, R. Schlüter, and H. Ney, "Acoustic Look-Ahead for More Efficient Decoding in LVCSR," in *Interspeech*, Florence, Italy, August 2011, pp. 893–896.

[7] D. Nolden, R. Schlüter, and H. Ney, "Search Space Pruning Based on Anticipated Path Recombination in LVCSR," in *Interspeech*, Portland, OR, USA, September 2012.

[8] D. Rybach, R. Schüter, and H. Ney, "A comparative analysis of dynamic network decoding," in *ICASSP*, Prague, Czech Republic, May 2011, pp. 5184–5187.

[9] C. Allauzen, M. Riley, J. Schalkwyk, W. Skut, and M. Mohri, "OpenFst: a general and efficient weighted finite-state transducer library," in *CIAA*, Prague, Czech Republic, July 2007, pp. 11–23.

[10] M. Sundermeyer, M. Nußbaum-Thom, S. Wiesler, C. Plahl, A. El-Desoky Mousa, S. Hahn, D. Nolden, R. Schlüter, and H. Ney, "The RWTH 2010 Quaero ASR Evaluation System for English, French, and German," in *ICASSP*, Prague, Czech Republic, May 2011, pp. 2212–2215.