ON THE USE OF LATTICES FOR THE AUTOMATIC GENERATION OF PRONUNCIATIONS

Sabine Deligne and Lidia Mangu

IBM T. J. Watson Research Center P. O. Box 218,
Yorktown Heights, NY 10598
{deligne,mangu}@us.ibm.com

ABSTRACT

In this paper, we explore the use of lattices to generate pronunciations for speech recognition based on the observation of a few (say one or two) speech utterances of a word. Various search strategies are investigated in combination with schemes where single or multiple pronunciations are generated for each speech utterance. In our experiments, a strategy that combines merging time-overlapping links in a context-dependent subphone lattice and generating multiple pronunciations provides the best recognition accuracy. This results in average relative gains of 30% over the generation of single pronunciations using a Viterbi search.

1. MOTIVATION

Speech recognition systems usually rely on a fixed lexicon where the pronunciations of the vocabulary words are given by hand-crafted phonetic baseforms, i.e. sequences of phones written by a phonetician. However, many applications require new words to be dynamically added to the recognition vocabulary, or new pronunciations of invocabulary words to be added to the lexicon. We consider situations where the spellings of the words are not available: the user is asked to utter once or twice the words to add to his/her personalized vocabulary, and phonetic baseforms for these words are derived from the acoustic data. In these situations, standard approaches ([1],[2]) usually rely on the combined use of: (i) an existing set of speaker-independent acoustic models of subphone units, and (ii) a model of transition between these subphone units. In [9], multiple baseforms are derived from each speech utterance by varying the relative weights of the acoustic and transition models. In this paper, we combine both single and multiple baseform generation schemes with lattice rescoring techniques.
The structure of this paper is as follows. Section 2 explains the procedure used for lattice generation and section 3 proposes three search strategies to retrieve pronunciations from the lattices. Section 4 describes two different schemes according to which the recognition lexicons are built. Section 5 reports on speech recognition experiments comparing various combinations of lattice generation, search strategies and lexicon schemes. Section 6 concludes this work.

2. LATTICE GENERATION

In this section, we describe our acoustic and transition models and the way these models are combined to produce search lattices at the level of either Context-Dependent (CD) subphone units, Context-Independent (CI) subphone units or phone units. In our acoustic modeling scheme, each phone is described as a sequence of three CI subphone units called arcs. Each arc is further modelled with a set of CD units, more precisely triphone units, which are called leaves as they are obtained by using a phonetic decision tree [5]. The cepstral distribution of each leaf is modeled with a mixture of Gaussians. The number of Gaussians in each mixture is optimized by using the BIC criterion [6].
A phone graph that allows any phone to follow any other phone is expanded into an arc graph and then into a leaf graph by integrating the contextual constraints given by each arc's phonetic decision tree. A leaf lattice containing the Viterbi path for each leaf sequence in the input graph is then generated for the data of each enrollment utterance. In our experiments, pronunciations are derived using either these leaf lattices or lattices at the arc or phone level. The arc lattices are obtained by converting each leaf label into the corresponding arc label, while preserving the acoustic score of each path. Similarly, the phone lattices are obtained by converting each sub-sequence of arc labels matching the same phone into the corresponding phone label, while preserving the acoustic score of each path.
The lattices are then rescored with a transition model, i.e. a Language Model (LM), between the leaves, arcs or phones the same way word lattices are rescored with an LM in speech recognition. In our experiments, the LM is a bigram model using Kneser-Ney-mod-fix smoothing technique [8]. The LM training data was obtained by aligning a large dataset of speech with a known transcription at the leaf level. The

arc and phone bigram models are estimated after converting the aligned corpus into a corpus of arc and phone labels respectively.

3. SEARCH STRATEGIES

Having generated the lattices at a leaf, arc or phone level, we can retrieve the most likely path in each lattice using the Viterbi algorithm, which we will refer to as the Viterbi strategy.

We can also compute link posterior probability as the sum of the posterior probabilities of all the sentences (paths) which contain (go through) that particular link (computed efficiently using the Forward-Backward algorithm [3]). We can then extract the path in the lattice with the highest combined link posterior probabilities, which we will refer to as the Forward-Backward strategy. By doing this, we accumulate path posterior probabilities for each link, but links with the same label which are hypothesized in similar time intervals are considered two different entities on competing paths.

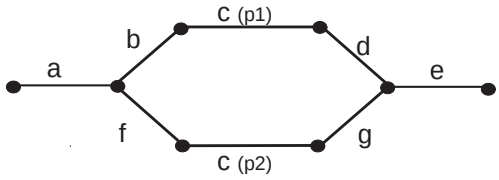
In order to move one step closer towards computing the posterior probability of a link, arc or phone in a particular time interval[1], we need to merge these overlapping links and create a new lattice. More precisely, we initialize clusters with all the links with the same label, start and end time. We then merge the clusters that bear the same labels and whose link components overlap in time[2]. This is similar to the intra-word stage of the clustering procedure described in[4]. At the end of the merging procedure we connect the resulting clusters based on the order relations existing in the original lattice, i.e. existing paths in the original lattice between the links components. Figure 1 illustrates the effect of merging on both the topology and the scores of the original lattices. By summing the posterior probabilities of all the links in each cluster we obtain a new score for each link in the new lattice. The path in the new lattice with the highest cumulative score is the newly proposed hypothesis, which we will refer to as the merging strategy.

4. LEXICON SCHEMES

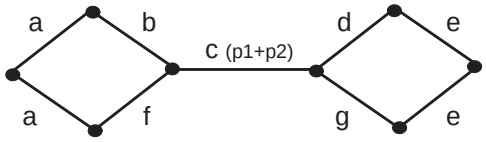
As explained in section 3, a sequence of either leaf, arc or phone labels is retrieved from the lattices generated for each enrollment utterance following the Viterbi strategy, the forward-backward strategy or the merging strategy. The sequences of labels output by the search procedure are converted into phonetic baseforms by replacing the subphone labels with their phone counterpart and by merging repeated

[1]The time interval is suggested by the start and end time of the overlapping links.

[2]We experimented with the degree of overlap as a constraint for merging and found that there is no need for such a constraint.



(a) Original lattice



(b) Lattice after merging

Fig. 1. Effect of merging on the topology and scores of the original lattices.

phones. Beginning and ending silence labels are filtered out. The influence of the LM weight λ used to rescore the lattices is investigated in two different schemes. In the "single baseform" scheme, all the distinct baseforms retrieved for a given speaker and a given LM weight are gathered (together with the hand-written baseforms of the fixed vocabulary) to form the recognition lexicon. In the "multiple baseform" scheme, all the distinct baseforms retrieved for a given speaker by rescoring the same lattice with different LM weights are gathered in the same lexicon as pronunciation variants as proposed in [9]. Each of the three search strategies is assessed in combination with either the single or multiple baseform schemes using either leaf, arc or phone lattices.

5. EXPERIMENTS

5.1. Enrollment data

We report on experiments with 2 different sets of enrolled words: (i) the enrollment set E_1 consists of 50 distinct words, each word being repeated twice by 10 speakers, (ii) the enrollment set E_2 consists of 35 distinct words, each word being repeated once by 20 speakers. All the data are recorded using a push-to-talk button in a quiet environment at 22kHz and downsampled to 11kHz. The front end computes 12 cepstra + the energy + delta and delta-delta coefficients from 15ms frames. Baseforms are generated using a reduced-size

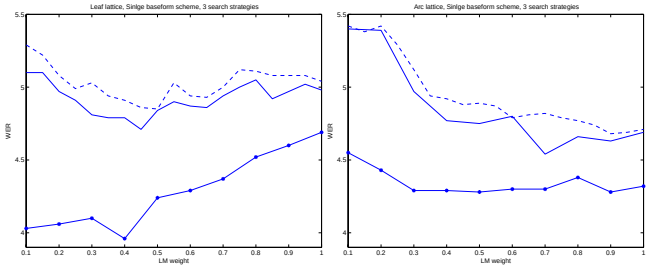
acoustic model especially designed to be used in portable devices or in automotive applications [7]. It consists of a set of speaker-independent acoustic models (156 subphones covering the phonetics of English) with about 5,000 context-dependent gaussians, trained on a few hundred hours of general English speech (about half of these training data has either digitally added car noise, or was recorded in a moving car at 30 and 60 mph). The bigram model of subphones was estimated off-line on an aligned corpus of about 17,000 sentences (names, addresses, digits). Speaker-dependent lexicons are formed for each speaker in respectively E_1 and E_2 following the schemes described in section 4.

5.2. Evaluation data

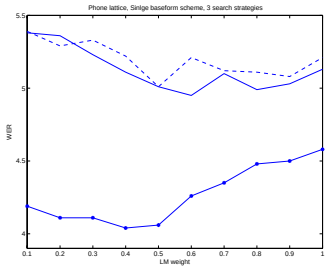
The recognition lexicons derived for each speaker in the enrollment set E_1 are evaluated on 2 test sets: (i) the test set $T1.1$ where each of the 50 words in E_1 are repeated in isolation 10 times by each of the same 10 speakers, (ii) the test set $T1.2$ where each of the 50 words in E_1 are repeated in 10 different short sentences (typically command sentences like "ADD $< name >$ TO THE LIST", where $< name >$ is an enrolled word) by each of the same 10 speakers. The recognition lexicons derived for each speaker in the enrollment set E_2 are evaluated on 3 test sets: (i) the test set $T2.1$ is recorded in a quiet environment, (ii) the test set $T2.2$ is recorded in a car moving at 30mph, (iii) the test set $T2.3$ is recorded in a car moving at 60mph. All 3 sets $T2.1$, $T2.2$ and $T2.3$ consist of the 35 words in E_2 uttered once and preceded by either the word "CALL", "DIAL" or "EMAIL", by each of the speakers in E_2. The baseforms of the command words "CALL", "DIAL"... in the test sets are linguist-written baseforms. In the following section, we show the overall Word Error Rate (WER) obtained on all five test sets.

5.3. Recognition scores

Figures 2(a), 2(b) and 2(c) compare the WER obtained with the different search strategies for respectively the leaf, arc and phone lattices in the single baseform scheme. The WER is plotted as a function of the LM weight λ (with $0.1 \leq \lambda \leq 1$ a multiple of 0.1, for an acoustic model weight set to 1). Regardless of the unit label, the forward-backward strategy provides an average relative WER reduction of about 2% (across all LM weights) over the Viterbi strategy. The merging strategy results in more significant WER reductions: respectively 13%, 12% and 17% relative for the leaf, arc and phone lattices.
Figures 3(a), 3(b) and 3(c) compare the WER obtained with the merging strategy combined with either the single or multiple baseform schemes for respectively the leaf, arc and phone lattices. In the multiple baseform scheme, baseforms are accumulated by scanning a set of LM weights



(a) **Leaf** lattices, (b) **Arc** lattices,



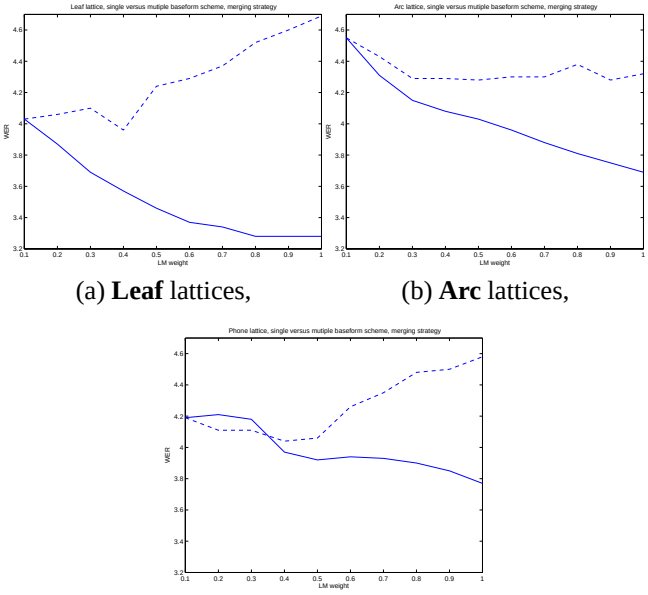
(c) **Phone** lattices.

Fig. 2. WER as a function of LM weight. Viterbi (- -) versus Forward-Backward (-) versus Merging (-*) search strategies in the single baseform scheme

$\{0.1; 0.2; ...; \lambda\}$. In both the single and multiple baseform schemes the WER is plotted as a function of λ. Accumulating baseforms when incrementing the LM weight decreases the WER by respectively 17%, 7% and 6% relative for the leaf, arc and phone lattices. As the single baseforms obtained with the leaf lattices are of better quality than the single baseforms obtained with the arc and phone lattices (the solid line on Figure 3(a) versus the solid lines on Figure 3(b) and Figure 3(c), cumulating them is all the more rewarding. These improvements confirm the ones obtained in [9].
Figure 4 compares the WER obtained with the leaf, arc and phone lattices for the optimal configuration where the merging strategy is combined with the multiple baseform scheme. The leaf lattices clearly outperform both the arc and phone lattices, with relative WER reduction of respectively 13% and 12% in average. The arc and phone lattices produce similar performances.

6. CONCLUSIONS AND PERSPECTIVES

We investigated the use of lattices for the automatic generation of pronunciations when only one or two enrollment speech utterances are available. Various search strategies are used to rescore context-independent and dependent subphone lattices and context-independent phone lattices. Our best strategy involves merging links that both overlap in time and have identical labels in a context-dependent sub-



(a) **Leaf** lattices, (b) **Arc** lattices,

(c) **Phone** lattices.

Fig. 3. WER as a function of LM weight. Single (- -) versus multiple (-) baseform schemes combined with the merging search strategy.

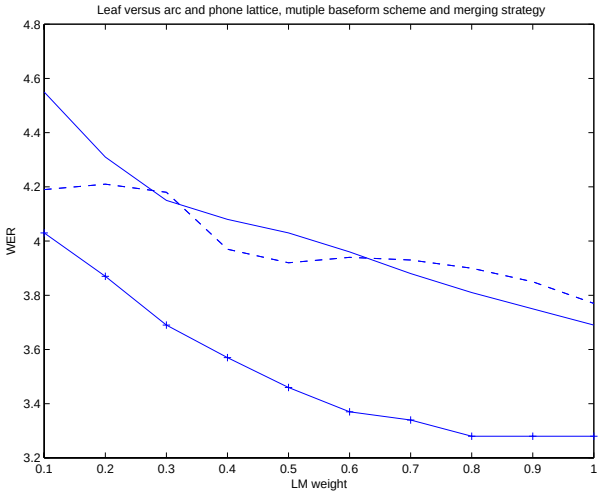


Fig. 4. WER as a function of LM weight. Leaf (-+) versus arc (-) and versus phone (- -) lattices using the merging search strategy combined with the multiple baseform scheme

phone lattice. This strategy consistently improves the quality of the baseforms, resulting in more than 10% relative reduction in WER. Moreover, accumulating the pronunciations across LM weights improves the robustness of the lexicons, as reflected by the 30% relative WER reduction. We expect further improvements can be obtained from taking into account the acoustic confusability between units.

7. REFERENCES

[1] R. C. Rose and E. Lleida, "Speech Recognition using Automatically Derived Baseforms", ICASSP 1997, pp 1271-1274.

[2] B. Ramabhadran, L.R. Bahl, P.V. DeSouza and M. Padmanabhan, "Acoustics-Only Based Automatic Phonetic Baseform Generation", ICASSP 1998.

[3] L. E. Baum, T. Petrie, G. Soules, and N. Weiss. "A maximization technique occuring in the statistical analysis of probabilistic function of Markov chains", *Annals of Mathematical Statistics*, 41(1):164-171, 1970.

[4] L. Mangu, E. Brill, and A. Stolcke. "Finding Consensus in Speech Recognition: Word Error Minimization and Other Applications of Confusion Networks", *Computer, Speech and Language*, 14(4):373-400, 2000.

[5] L.R. Bahl and P.V. de Souza and P.S. Gopalakrishnan and D. Nahamoo and M.A. Picheny, "Decision Trees for Phonological Rules in Continuous Speech," ICASSP 1991.

[6] S. S. Chen, E. Eide, M. J. F. Gales, R. A. Gopinath, D. Kanevsky and P. Olsen, "Automatic Transcription of Broadcast News," *Speech Communication*, May 2002, 37(1-2), pp69-87.

[7] S. Deligne, E. Eide, R. Gopinath, D. Kanevsky, B. Maison, P. Olsen, H. Printz, J. Sedivy: "*Low-resource speech recognition of 500-word vocabularies*", EUROSPEECH 2001.

[8] S. Chen and J. Goodman, "An empirical study of smoothing techniques for language modeling", Computer, Speech and Language, 1999, 13(4), pp359-393.

[9] S. Deligne, B. Maison and R. Gopinath: "Automatic generation and selection of multiple pronunciations for dynamic vocabularies", ICASSP 2001.