Pronunciation Modeling Using a Finite-State Transducer Representation *

Timothy J. Hazen, I. Lee Hetherington, Han Shu, and Karen Livescu

MIT Computer Science and Artificial Intelligence Laboratory
32 Vassar Street, Cambridge, Massachusetts 02139 USA

The MIT SUMMIT speech recognition system models pronunciation using a phonemic baseform dictionary along with rewrite rules for modeling phonological variation and multi-word reductions. Each pronunciation component is encoded within a finite-state transducer (FST) representation whose transition weights can be trained using an EM algorithm for finite-state networks. This paper explains the modeling approach we use and the details of its realization. We demonstrate the benefits and weaknesses of the approach both conceptually and empirically using the recognizer for our JUPITER weather information system. Our experiments demonstrate that the use of phonological rewrite rules within our system achieves word error rate reductions between 4% and 9% over different test sets when compared against a system using no phonological rewrite rules.

1. Introduction

Pronunciation variation has been identified as a major cause of errors for a variety of automatic speech recognition tasks (McAllaster *et al.*, 1998). In particular, pronunciation variation can be quite severe in spontaneous, conversational speech. To address this problem, this paper presents a pronunciation modeling approach that has been under development at MIT for more than a decade. Our approach systematically models pronunciation variants using information from a variety of levels in the linguistic hierarchy. Pronunciation variation can be influenced by the higher level linguistic features of a word (e.g., morphology, part of speech, tense, etc.) (Seneff, 2004), the lexical stress and syllable structure of a word (Greenberg, 1999), and the specific phonemic content of a word sequence (Riley *et al.*, 1999; Tajchman *et al.*, 1995). When all of the knowledge in the linguistic hierarchy is brought to bear upon the problem, it becomes easier to devise a consistent, generalized model that accurately describes the allowable pronunciation variants for particular words. This paper presents the pronunciation modeling approach that has been implemented and evaluated within the SUMMIT speech recognition system developed at MIT.

Pronunciation variation in today's speech recognition technology is typically encoded using some combination of a lexical pronunciation dictionary, a set of phonological rewrite rules, and a collection of context-dependent acoustic models. The component which models a particular type of pronunciation variation can be different from recognizer to recognizer. Some recognizers rely almost entirely on their context-dependent acoustic models to capture phonological effects (Hain, 2002), while other systems explicitly model phonological variation with a set of phonological rewrite rules (Hazen *et al.*, 2002). Some systems do not use an explicit set of phonological rules but account for a wide variety of phonological effects using (multiple) alternate pronunciations directly in the pronunciation dictionary (Lamel & Adda, 1996). In this paper we use the SUMMIT recognizer to examine the advantages and disadvantages of accounting for general phonological variation explicitly with phonological rules versus implicitly within context-dependent acoustic models. We also describe a pronunciation variation modeling approach which uses a cascade of finite-state transducers, each of which models different variations resulting from different underlying causes.

*This research was supported by DARPA under contract N66001-99-1-8904, monitored through Naval Command, Control and Ocean Surveillance Center. This manuscript is a preprint version of an article published in *Speech Communication*, vol. 46, no. 2, pp. 189-203, June, 2005.

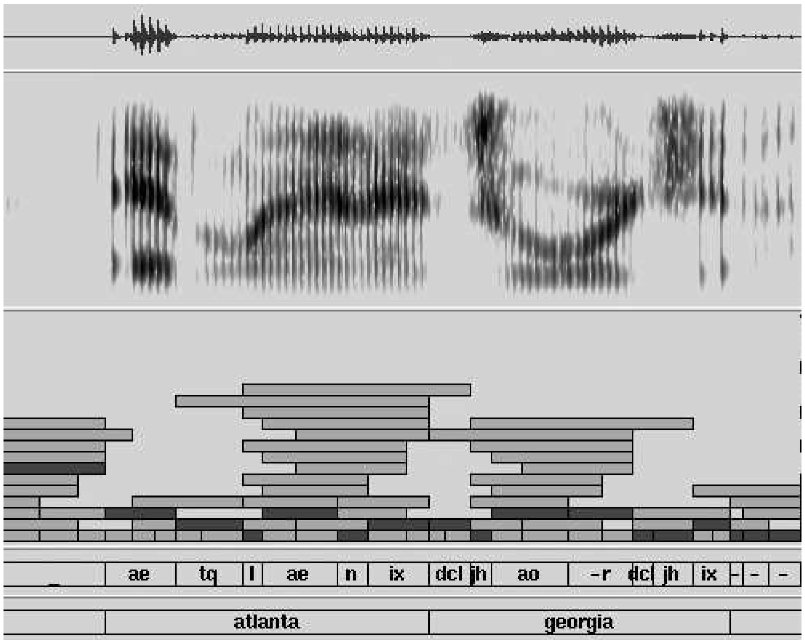


Figure 1. The output of a graphical interface displaying a sample waveform, its spectrogram, the hypothesized SUMMIT segment network with the best path segment sequence highlighted, the time-aligned phonetic transcription of the best path, and the time-aligned word transcription of the best path.

2. General Overview

2.1. Segment-Based Recognition

The experiments presented in this paper use the SUMMIT speech recognition system. SUMMIT uses a segment-based approach for acoustic modeling (Glass, 2003). This approach differs from the standard hidden Markov modeling (HMM) approach in that the acoustic-phonetic models are compared against pre-hypothesized variable-length segments instead of fixed-length frames. While HMM systems allow multiple frames to be absorbed by a single phoneme model via self-loops on the HMM states, our segment-based approach assumes a one-to-one mapping of hypothesized segments to phonetic events. This approach allows the multiple frames of a segment to be modeled jointly, removing the frame independence assumption used in the standard HMM. Details of SUMMIT's acoustic modeling technique can be found in (Ström *et al.*, 1999).

Figure 1 shows the recognizer's graphical display containing a segment graph (with the recognizer's best path highlighted) along with the corresponding phonetic transcription. It is important to note that SUMMIT pre-generates a segment network based on measures of local acoustic change before the search begins. The smallest hypothesized segments can be as short as a single 10 millisecond frame, but segments are typically longer in regions where the acoustic signal is relatively stationary.

The segment-based approach presents several modeling issues which are generally not present in frame-based HMM systems. For example, in HMM recognizers a single multi-state phoneme model can be used to implicitly learn the closure and burst regions of a plosive consonant. However, in our segment-based approach plosives must be explicitly modeled as two distinct phonetic events, a closure and a release. This is necessary because the segmentation algorithm will observe two distinct acoustic regions and may not hypothesize a single segment spanning both the closure and the burst regions.

Another issue faced by our segment-based approach is its difficulty in absorbing deleted or unrealized phonemic events which are required within its search path. An HMM need only absorb as little as one poorly scoring frame when a phonemic event in its search path is not realized, while SUMMIT must potentially absorb a whole multi-frame segment. As a result, accurate phonetic modeling that accounts for potentially deleted phonemic events is more crucial for segment-based approaches than for HMM approaches. It is our belief that accurate phonetic segmentation and classification is important for distinguishing between acoustically confusable words.

2.2. FST-Based Search

The SUMMIT recognizer utilizes a finite-state transducer (FST) representation for its lexical and language modeling components. The FST representation allows the various hierarchical components of the recognizer's search space to be represented within a single parsimonious network through the use of generic FST operations such as composition, determinization and minimization (Pereira & Riley, 1997). The full search network used by SUMMIT is illustrated in Figure 2. The figure shows the five primary hierarchical components of the search space: the language model (G), a set of word-level rewrite rules for reductions and contractions (R), the lexical pronunciation dictionary (L), the phonological rules (P), and the mapping from phonetic sequences to context-dependent (CD) model labels (C). Each of these components can be independently created and represented as an FST. By composing the FSTs such that the output labels of the lower-level components become the inputs for the higher-level components, a single FST network is created which encodes the constraints of all five individual components. The full network can be represented mathematically with the following expression:

$$N = C \circ P \circ L \circ R \circ G$$

This paper focuses on the reductions FST R, the lexicon FST L and the phonological rules FST P.

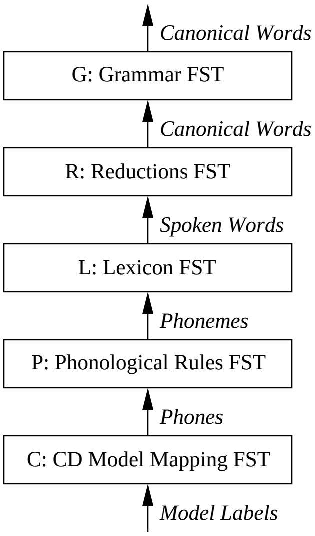


Figure 2. The set of distinct FST components which are composed to form the full FST search network within the SUMMIT recognizer.

2.3. Levels of Pronunciation Variation

In our pronunciation modeling approach we distinguish between four different levels of pronunciation variation: (1) variations that depend on word-level features of lexical items (such as part-of-speech, case, tense, etc.), (2) variations that are particular to specific lexical entries, (3) variations that depend on the stress and syllable position of phonemes, and (4) variations that depend only on local phonemic or phonetic context.

It is important to note that pronunciation variation can result from other sources as well, such as human error (i.e., mispronunciations), regional dialect, or foreign accent. We don't explicitly account for these types of variations within the framework presented in this paper. However, an FST-based approach for learning phonetic transformations due to foreign accents has been previously explored within the context of our recognizer (Livescu & Glass, 2000).

In the following paragraphs we provide examples of these variants specifically for English.

Type (1) variants include contractions (*what's*, *can't*, etc.), reductions (*gonna*, *wanna*, etc.), part-of-speech variants (as in the noun and verb versions of *record*), and tense variants (as in the past and present tense versions of *read*). In most speech recognition systems, these types of variants are handled in very superficial manners. Reductions and contractions are typically entered into the pronunciation lexicon as distinct entries independent of the entries of their constituent words. All alternate pronunciations due to part of speech or tense are typically entered into the pronunciation lexicon within a single entry without regard to their underlying syntactic properties. In our system reductions and contractions are handle by the reductions FST R, while all other type (1) variants are encoded as alternate pronunciations within lexical entries in the lexicon FST L. In future work we may investigate methods for explicitly delineating pronunciation variations caused by the part-of-speech, case or tense of a word.

Type (2) variants are simply word-dependent pronunciation variants which are not the result of any linguistic features of that word. A simple example of a word with a type (2) variant is *either*, which has two different phonemic pronunciations as shown here:

either: (iy | ay) th er

These variants are typically encoded manually by lexicographers. In our system these variants are all handled as alternate pronunciations in the lexicon FST L.

Variants of type (3) in English are typically related to the realization of stop (or plosive) consonants. The set of possible allophones of a stop consonant in English is heavily dependent on its position within a syllable and the stress associated with the syllables preceding and following the stop. For example, a stop in the suffix or coda position of a syllable can be unreleased, while stops in the prefix position of a stressed syllable must be released. An example is shown here using the word *laptop*:

laptop: l ae pd t aa pd

In this example, the label /pd/ is used to represent a /p/ within a syllable suffix or coda whose burst can be *deleted*. The /t/ in this example is in the onset position of the syllable and therefore must have a burst release. Type (3) variants are encoded using syllable-position-dependent phonemic labels directly in the lexicon FST L. The details of the creation of the pronunciation lexicon using these special labels are presented in Section 3.2.

Variants of type (4) can be entirely determined by local phonemic or phonetic context and are independent of any higher-level knowledge of lexical features, lexical stress, or syllabification. Examples of these effects are vowel fronting, place assimilation of stops and fricatives, gemination of nasals and fricatives, and the insertion of epenthetic silences. To account for type (4) variants we have developed our own FST mechanism for applying context-dependent phonological rules. The details of the syntax and application of the rules are described in (Hetherington, 2001). Examples of these rules will be presented in Section 3.3. In relation to Figure 2, type (4) variants are generated by the phonological rules FST P.

In some cases, it may be debatable which variant type describes a particular alternate pronunciation. For example, one could ask if the deletion of the third syllable schwa in the word *temperature* is a generalizable variant that can be expressed with a phonological rule or a variant specific to this word that must be encoded in its baseform pronunciation. In this work, we make no claims about how the specific decisions on the labeling of pronunciation variants into the four types listed above should be made. The framework we have developed is agnostic to these specific decisions. It is more important that these decisions be made consistently so that all expected pronunciation variations are accounted for within some FST component of the system (and preferably not accounted for redundantly within multiple FSTs).

2.4. Modeling Variation with Context-Dependent Models

When devising an approach for capturing phonological variation there is flexibility in the specific model in which certain types of phonological variation are captured. In particular, certain forms of phonological variation can easily be modeled either explicitly with phonological rules using symbolically distinct allophonic variants, or implicitly using context-dependent acoustic models which capture the acoustic variation from different allophones within their probability density functions (Jurafsky *et al.*, 2001). One example is the place assimilation effect, which allows the phoneme /d/ to be realized phonetically as the palatal affricate [jh] when followed by the phoneme /y/ (as in the word sequence *did you*). The effect could be modeled symbolically with a phonological rewrite rule allowing the phoneme /d/ to be optionally realized as [jh]. Alternately, it can be captured in a context-dependent acoustic model which implicitly learns the [jh] realization within the density function for the context-dependent model for the phoneme /d/ in the right context of the phoneme /y/.

Modeling effects such as place assimilation within the context-dependent acoustic model has several advantages. First, this type of model simplifies the search by utilizing fewer alternate pronunciation paths in the search space. The likelihoods of the alternate allophones are encoded directly into the observation density function of the acoustic models. Additionally, no hard decision about which allophone is used is ever made during either training or actual recognition.

Pushing the modeling of allophonic variation into the context-dependent acoustic model does have potential drawbacks as well. In particular, traditional context-dependent acoustic models may not accurately represent the true set of allophonic variants in cases where stress and syllable-boundary information is required for predicting the allowable set of allophones. For example, consider the two word sequences *"the speech"* and *"this peach"*. Both of these word sequences can be realized with the same phonetic sequence:

th ix s pcl p iy tcl ch

In this particular example, there are two acoustically distinct allophonic variants of /p/; the /p/ in *"the speech"* is unaspirated while the /p/ in *"this peach"* is aspirated. The exact variant of /p/ is determined by the location of the fricative /s/ in the syllable structure. In *"the speech"* the /s/ forms a syllable-initial consonant cluster with the /p/ thereby causing the /p/ to be unaspirated. In *"this peach"* the /s/ belongs to the preceding syllable thereby causing the /p/ to be aspirated. A standard context-dependent acoustic model will model these variants inexactly, allowing the /p/ to be either aspirated or unaspirated in either case. In essence, pushing the modeling of phonological variation into the context-dependent acoustic models runs the risk of creating models which *over-generate* the set of allowable realizations for specific phonemic sequences.

It should be noted that promising methods for adding stress and syllable information into the contextual information used by context-dependent acoustic models have been explored (Riley *et al.*, 1999; Shafran, 2001). These approaches can alleviate allophonic over-generation problems, like the one presented above, at the expense of an increase in the complexity of the conditioning context.

3. Pronunciation Modeling in SUMMIT

3.1. Deriving the Reduction FST

To handle reductions and contractions, a reduction FST is created which encodes rewrite rules that map contractions and other multi-word reductions to their underlying canonical form. Some examples of these rewrite rules are as follows:

gonna → going to
how's → how is
I'd → I would | I had
lemme → let me

In some cases, such as the contraction *I'd*, a contracted form could represent more than one canonical form. The output of the reduction FST R serves as the input to the grammar FST G, thus allowing/constraining the grammar FST G to operate on the intended sequence of canonical words, irrespective of their surface realization. In the JUPITER weather information domain, the reduction FST R contains 120 different contracted or reduced forms of word sequences.

3.2. Deriving the Lexicon FST

The lexicon FST represents the phonemic pronunciations of the words in the system's vocabulary (including contractions and reductions). This FST is created primarily by extracting pronunciations from a syllabified dictionary. The dictionary used in our experiments is a combination of the PronLex dictionary,[2] the Carnegie Mellon University Pronouncing Dictionary,[3] and manually crafted pronunciations derived by experts in our group. The full dictionary was automatically syllabified using rules originally derived by Church (Church, 1983). The syllabified dictionary expresses the pronunciations with a set of 41 basic phonemic labels.

As mentioned earlier the dictionary can contain alternate pronunciations for each entry. To provide an example about the typical number of alternate pronunciations in L, roughly 17% of the entries in our JUPITER weather information lexicon contain more than one pronunciation.

From the syllabified dictionary, a set of rewrite rules is used to generate special phonemic stop labels, which capture information about the allowable phonetic realizations of each stop based on stress and syllable position information. For example, stops in an onset position of a syllable retain their standard phonemic label (/b/, /d/, /k/, etc.) while stops in the suffix or coda of a syllable are converted to labels indicating that their closure can be unreleased with the burst being *deleted* (/bd/, /dd/, /kd/, etc.). In total, the set of 6 standard stop labels are converted into a set of 20 different stop labels for the purpose of encoding the allowable allophones for each stop.

One potential issue that arises is the potential harm that may be introduced by incorrect syllabification. This could result from inappropriate selections of the various stop labels. We did not find this to be a serious problem in our system for two reasons. First, the number of incorrect syllabifications was small and limited to three particular types of words: compound words, foreign words, and words with common prefixes and suffixes like *co-* and *-ed*. Within our full reference lexicon, we manually corrected all of the incorrect syllabifications contained in words with common suffixes and prefixes. We also manually checked the syllabification of every word in the vocabulary of the recognizer used in our experiments. Second, even without the manual corrections, the typical result of an improper syllabification is the production of a stop label that over-generates the potential allophones. While this may lead to increased confusions, it is not as serious a problem as failing to generate an expected alternate pronunciation for a word. We have not, however, examined the potential degradation that might have resulted without our manual corrections.

3.3. Deriving the Phonological FST

To encode the possible pronunciation variants caused by phonological effects, we have developed a syntax for specifying phonological rules and a mechanism for converting these rules into an FST representation. In this approach phonological rules are expressed as a set of context-dependent rewrite rules. All of the phonological rules in our system have been manually derived based upon acoustic-phonetic knowledge, and upon actual observation of phonological effects present within the spectrograms of the data collected by our systems. The full set of phonological rules contains 164 context-dependent rewrite rules (excluding canonical context-independent rules which map phonemes one-for-one to their equivalent phonetic units). A full description of the expressive capabilities of the phonological rule syntax and the mechanism for compiling the rules into an FST can be found in (Hetherington, 2001).

To demonstrate some of the expressive capabilities of our phonological rule syntax, we now provide some examples of the phonological rules used in our system. Two example phonological rules for the phoneme /s/ are:

$$\{\text{l m n ng}\}\ \text{s}\ \{\text{l m n w}\} \rightarrow [\text{epi}]\ \text{s}\ [\text{epi}]$$
$$\{\}\ \text{s}\ \{\text{y}\} \rightarrow (\ \text{s} \mid \text{sh}\ )$$

The first rule expresses the allowed phonetic realizations of the phoneme /s/ when the preceding phoneme is an /l/, /m/, /n/, or /ng/ and the following phoneme is an /l/, /m/, /n/, or /w/.

[2] Available from the Linguistic Data Consortium: http://www.ldc.upenn.edu

[3] Available from the Speech at CMU web page: http://www.speech.cs.cmu.edu/speech/

In these phonemic contexts, the phoneme /s/ can have an epenthetic silence optionally inserted before and/or after its phonetic realization of [s]. In the second rule the phoneme /s/ can be realized as either the phone [s] or the phone [sh] when followed by the phoneme /y/ (i.e., the /s/ can be palatalized).

To provide another example, the following rule accounts for the optional deletion of /t/ in a syllable suffix position when it is preceded by an /f/ or /s/ (as in the words *west* and *crafts*):

{f s} td {} → [tcl [t]]

In this example the /t/ can contain a closure and a release, can be realized with an unreleased closure, or can be completely deleted.

To provide one more example, the following rule can be used to optionally insert a transitional [y] unit following an /iy/ when the /iy/ is followed by another vowel or semivowel:

{} iy {VOWEL r l w hh} → iy [y]

While this specific type of phonological effect is typically handled within the context-dependent acoustic models of a recognizer, this type of rule can be effective for providing additional detail to time-aligned phonetic segmentations. This can be especially helpful when utilizing automatically derived time-alignments for corpus-based concatenative synthesis.

3.4. Training the Pronunciation FSTs

As the number of rules introducing alternate pronunciations increases, the problem of confusibility between acoustically similar words increases. In particular, the additional rules could lead to the generation of many alternate pronunciations which are incorrect or, at the very least, highly improbable. By taking the likelihood of the various alternate pronunciations into account within the pronunciation model, the potential for the recognizer to select a highly unlikely alternate pronunciation within an incorrectly hypothesized word is reduced.

To incorporate knowledge about the likelihoods of the alternate pronunciations encoded within the various component FSTs, we have implemented a maximum likelihood (ML) EM training algorithm for arbitrary FST networks (Dempster *et al.*, 1977; Eisner, 2002). The goal of the training is to produce the conditional probability of an input sequence given an output sequence, for example $\Pr(\mathit{phones}|\mathit{phonemes})$ for P or $\Pr(\mathit{phonemes}|\mathit{words})$ for L.

When using the training algorithm it is important to note that the size of the trained FSTs can be larger than that of the untrained FSTs. In general, a given FST topology might not support a conditional probability of an input sequence given an output sequence. We train a joint probability model and convert this to a conditional probability model, and this conversion generally results in a topology change and increased size. More details are available in (Shu & Hetherington, 2002).

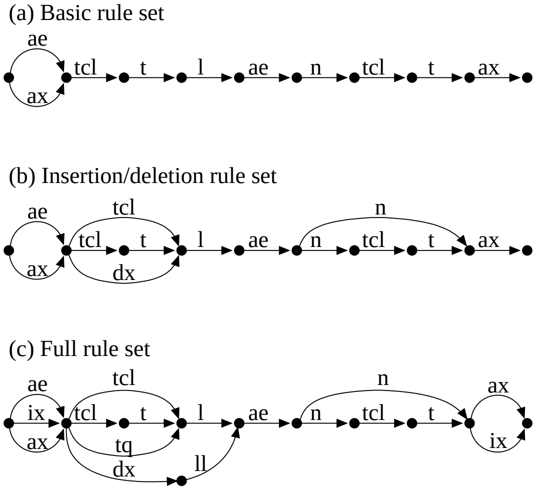
The training algorithm produces a weighted FST which models the conditional probability of an input sequence to the FST for a given output sequence. For example, an untrained lexical FST L can be trained to produce a trained lexical FST $tr(L)$ which encodes the likelihood of a phoneme sequence given a spoken word sequence, i.e., $\Pr(\mathit{phonemes}|\mathit{words})$. If each of the FST components is trained independently, then the composition of trained FSTs $tr(P) \circ tr(L) \circ tr(R)$ encodes the probability of a phone sequence given a sequence of canonical words via a probability chain rule.

The training algorithm can be used to train the individual component FSTs independently or jointly. When training the components independently (i.e., $tr(P) \circ tr(L) \circ tr(R)$) the likelihoods of specific phonological rules can be generalized across all words sharing these rules. When training the components jointly (i.e., $tr(P \circ L \circ R)$) the phonological rule probabilities are not shared across words and the likelihood of a particular realization of a phonological rule becomes dependent on the word in which it is applied. Because of the word-dependency introduced by joint training, the number of FST weights that must be trained can be substantially larger. In previous experiments we found that joint training dramatically increased the size of the final static FST without improving the recognizer's accuracy (Shu & Hetherington, 2002).

For our training data, phone strings are generated automatically from word strings using forced alignment. In this case, the intermediate phoneme layer is *hidden.* Thus, when P and L are trained independently, the EM algorithm is used to learn the probabilities of alternate phoneme strings in the hidden layer. In most cases, though, the phoneme string is uniquely determined from the word and phone string. When $P \circ L$ is trained jointly, the joint FST has no hidden layer because it maps directly from phones to words. In this case, the EM algorithm need not learn any hidden probabilities and the EM training reduces to trivial count-based ML estimation.

It is important to note that the specific topology of a network controls the probability space that is trained. Networks which are minimized will represent a different probability space than networks which are fully expanded. For example, illustration (c) of Figure 3 shows a determinized and minimized graph of possible phonetic pronunciations of the word *Atlanta* as it could be configured within the joint $P \circ L$ FST. There are 48 unique paths through this network. If this minimal graph were expanded into a deterministic tree, each of the 48 unique paths could receive its own independent probability. In the minimal graph, context information about the phonetic realization of previous phonemes is not retained as one moves through the graph. In essence, the minimized graph represents a context-independent pronunciation model, while a fully expanded tree represents a (within-word) context-dependent pronunciation model. In the case of the word *Atlanta*, minimizing the graph reduces the number of probabilities to be learned from 48 to 11.

In practice, minimal graphs are more robust to sparse data training issues, require less memory, and use less computation during search. As discussed earlier, our previous experiments in jointly training a $P \circ L$ FST have shown that the use of word-dependent phonological rule weight training did not improve accuracy over word-independent phonological rule weight training (Shu & Hetherington, 2002). As a result, we have elected not to pursue the use of a probabilistic model that also incorporates phonetic context (via the use of deterministic trees instead of minimized graphs). This added context would only further increase the size of the search space and the data sparseness issues encountered in our previous experiments. As such, the experiments in this paper only focus on the training of minimized graphs.


(a) Basic rule set

(b) Insertion/deletion rule set

(c) Full rule set

Figure 3. Phonetic pronunciation networks for the word *Atlanta* generated from three different phonological rule sets.

4. Experiments & Results

4.1. Phonological Rule Sets

To investigate the effectiveness of using phonological rules, we evaluated three different sets of rules. These rule sets can be described as follows:

- Basic phoneme rules: This set of rules generates a one-to-one mapping of phonemes to phones. This is essentially the same as applying no rules except for the fact that we split stop and affricate phonemes into two phonetic segments to represent the closure and release portions of the phones with different models.

- Insertion and deletion rules: This set of rules augments the basic set with a collection of rules for inserting or deleting phonetic segments in certain contexts. This primarily includes the deletion of stop bursts or entire stop consonants, the reduction of stops to flaps, the insertion of epenthetic silences near strong fricatives, and the replacement of schwa-nasal or schwa-liquid combinations with syllabic nasal or syllabic liquid units. This set adds an additional 65 context-dependent rules to the basic phoneme rules.

- Full rule set: This set augments the insertion and deletion rules with a large set of rules for allophonic variation. This includes the introduction of new allophonic labels for stops and semivowels as well as rules for place assimilation and gemination. This set contains 164 context-dependent rules beyond the basic phoneme rules.

To illustrate the types of phonological variation that these rule sets can cover, consider the three pronunciation networks for the word *Atlanta* in Figure 3. The baseform pronunciation of *Atlanta* in the lexicon is expressed as:

Atlanta: (ae | ax) td l ae n tn ax

In this pronunciation, the special label /td/ represents a /t/ in the suffix of a syllable, which can be unreleased, and the /tn/ represents a word internal /t/ following an /n/, which can be deleted. These special labels were automatically generated when the lexicon FST was created from the syllabified dictionary.

In (a), the basic rule set produces a single phonetic representation for each phoneme in the baseform pronunciation. In the case of the /t/ stop consonants, the phonetic representation contains two phonetic units: the closure [tcl] and the burst [t]. In (b), the insertion/deletion rule set allows the first /t/ to be alternately realized with an unreleased burst or as a flap, while the second /t/ can be completely deleted. In (c), the full rule set introduces several new allophonic variants including a fronted schwa and a glottal stop /t/.

By creating these three distinct sets of phonological rules we can examine the effect of modeling different phonological variants either within the phonological rules or within the acoustic models. We first examine the effectiveness of introducing rules that account for phonetic insertions and deletions against the basic set of rules which do not allow insertions and deletions. Figure 4 shows the phonetic alignment obtained by the SUMMIT recognizer using only the basic set of phonological rules on the same utterance presented earlier in Figure 1. An examination of the phonetic alignment in Figure 4 presents anecdotal evidence that the recognizer is not able to model the true sequence of phonetic events with the minimal set of phonological rules. This is particularly obvious in the word *Atlanta* where the recognizer was forced to insert [t] releases for both /t/ phonemes despite the fact that the speaker actually used the glottal stop allophone for the first /t/ and completely deleted the second /t/. Despite the poor phonetic transcription, the recognizer was still able to recognize this utterance correctly.

By augmenting the insertion/deletion rule set with rules which cover substitutional allophonic variation, we can investigate the effectiveness of modeling allophonic variation implicitly using context-dependent acoustic models versus explicitly using context-dependent phonetic rewrite rules. Anecdotal evidence of the effectiveness of utilizing explicit rewrite rules to capture allophonic variation can be seen in the example in Figure 1 (in Section 2.1). By examining the phonetic transcription in this figure, it can be observed that the recognizer successfully identified the use of the glottal stop variant of /t/ at the beginning of *Atlanta* and the use of fronted schwas at the end of both *Atlanta* and *Georgia*.

4.2. Experimental Details

Our experiments were conducted using the SUMMIT recognizer trained specifically for the JUPITER weather information system, a conversational interface for retrieving weather reports and information for over 500 cities around the world (Glass *et al.*, 1999; Zue *et al.*, 2000). This recognizer has a vocabulary of 1915 words (excluding contracted or reduced forms) and includes 5 noise models for modeling non-speech artifacts and 3 models for filled pauses. The recognizer's acoustic model uses diphone landmark modeling and segment duration modeling. Diagonal Gaussian mixture models are used for the system's acoustic models. Details of the acoustic modeling component of the recognizer are available

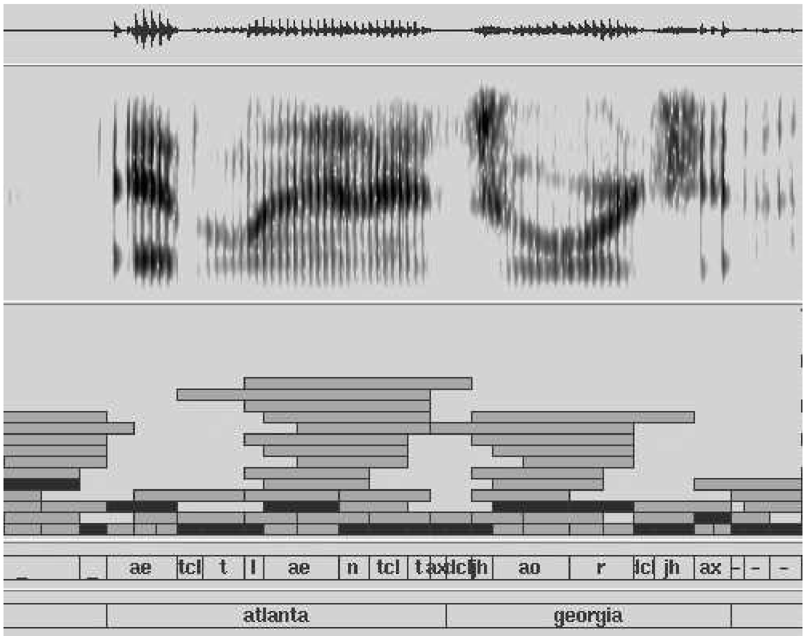


Figure 4. The output of SUMMIT's graphical interface on the word sequence *Atlanta Georgia* when the recognizer uses only a basic set of phonological rules which do not generate any phonological variants.

in (Ström *et al.*, 1999).

The system models were trained using 126,966 utterances collected over the telephone network by publicly available dialogue system maintained by our group. Approximately 75% of this data was collected by the JUPITER system. The system was tested on a randomly selected set of 1888 utterances from calls made to JUPITER's toll-free telephone line (we call this the *full test set*). Results are also reported for a 1303 utterance subset of the test data containing only in-vocabulary utterances with no non-speech artifacts (we call this the *clean test set*). The evaluation on the clean test set allows us to examine the performance of the modeling techniques independent of the confounding factors contributed by unknown words and non-speech artifacts.

4.3. Results with Untrained FSTs

Table 1 contains the results of our experiments when using untrained versions of the component FSTs. As can be observed in the table, incorporating phonological rules for handling insertions and deletions of phonetic events resulted in a relative word error rate reduction of 9% (from 12.1% to 11.0%) on the clean test set.[4] Over the full test set the error rate reduction was a more modest 4% (from 19.1% to 18.4%). Using the matched pairs sentence-segment word error (MAPSSWE) significance test (Gillick & Cox, 1989), the improve-

Table 1
Performance of JUPITER recognizer on the full test set and on the clean test set using three different sets of phonological rules and untrained FSTs.

Phonological	Word Error Rate (%)	
Rule Set	Full Test Set	Clean Test Set
Basic Set	19.1	12.1
Ins./Del. Set	18.4	11.0
Full Rule Set	19.0	11.7

[4] These results are slightly different than results presented in (Hazen *et al.*, 2002) because the clean test set now contains ten additional utterances that were inadvertently excluded from this set in our earlier experiments.

ment is statistically significant to a 0.005 probability level. These results demonstrate that standard context-dependent models by themselves are not sufficient for modeling contextual effects that cause the number of realized phonetic events to be different from the underlying canonical form.

Table 1 also shows that the additional rules in the full rule set actually degrade performance. However, a MAPSSWE significance test finds this degradation to be statistically insignificant at a 0.005 probability level. These additional rules explicitly model allophonic variations which do not alter the number of phonetic events (such as palatalization, vowel fronting, etc.). This suggests that the context-dependent acoustic models are sufficient for modeling allophonic variation caused by phonetic context, and that the added complexity required to explicitly model these effects does not provide any benefit (and may actually hinder the recognizer's performance).

It is important to note that the increase in the error rate of the system using the full rule set does not result from increasing the complexity of the search space without increasing the search's pruning thresholds. The accuracy using the full rule set does not improve when the pruning thresholds are relaxed. Thus, the accuracy degradation is purely a result of the discriminative capabilities of the models. One might hypothesize, based on these results, that increasing the number of allowable phonetic realizations for each word increases the likelihood of its confusion with other words (as has also been suggested by others (Hain, 2002)).

4.4. Model Complexity Issues

To further demonstrate the effect that adding phonological rules has on the recognizer's complexity, Table 2 shows the size of the recognizer for each of the three different rule sets in terms of the number of states and arcs in the pre-compiled untrained FST network. The table shows a dramatic increase in the complexity of the search space as additional phonological rules are added to the system. The full rule set causes a 70% increase in the size (in megabytes) of the lexical search network compared to the basic rule set and a 30% increase compared to the insertion/deletion rule set.

Table 2
Effect of phonological rules on the size of the untrained static FST search network (i.e., $C \circ P \circ L \circ R \circ G$) in terms of the FST states, FST arcs, and size.

Phonological	Full Static FST		
Rule Set	# States	# Arcs	Size
Basic Set	39389	215633	5.0 MB
Ins./Del. Set	45263	282603	6.5 MB
Full Set	54641	386500	8.7 MB

The addition of new phonological rules to a system requires the creation of a new set of acoustic models. The number of acoustic models is determined for each phonological rule set automatically based on phonetic-context decision-tree clustering. The number of Gaussians per context-dependent model is determined via an empirically optimized heuristic which is based on the number of training samples available for each model. Specifically, a model contains one Gaussian component for every N training tokens (where N is the number of dimensions in the input feature vector, which is 50 in this system). The maximum number of Gaussians per model is capped at 75.

Table 3 shows a dramatic increase in the number of acoustic models and Gaussian components used as the size of the phonological rule sets increases. This is a result of the new allophonic variants introduced by the rule sets and the new contexts they produce. As the number of new allophonic variants and their contexts increases, the potential number of acoustically dissimilar context-dependent models also increases. Table 3 shows that the full rule set produces 66% more symbolically distinct diphones (i.e., adjacent phone pairs) than the basic rule set and 29% more than the insertion/deletion rule set.

4.5. Analysis of Acoustic Model Size

In examining Table 3, one could argue that the experiments presented in Table 1 are inherently unfair because each system uses a different number of Gaussian components. To demonstrate

Table 3
Effect of phonological rules on the size of the context-dependent acoustic models and the number of unique diphone pairs.

Phonolog.	CD Acoustic Models		
Rule Set	Diphones	Models	Gaussians
Basic Set	3668	1173	38349
Ins/Del Set	4734	1388	41677
Full Set	6105	1630	45976

Table 4
Performance of JUPITER recognizer on the full test set when training the phonological FST P and the reductions FST R.

Training	Word Error Rate (%)	
Condition	Ins./Del. Set	Full Rule Set
$P \circ L \circ R$	18.4	19.0
$tr(P) \circ L \circ R$	18.2	18.6
$P \circ L \circ tr(R)$	18.2	18.8

that the differences in accuracy are not the result of differences in the number of parameters in the acoustic model sets, a second set of models, with roughly the same number of Gaussian components as the model set for the insertion/deletion rules, was trained for both the basic rule set and the full rule set.

For the basic rule set, the maximum number of Gaussian components per class was increased from 75 to 90. This resulted in a new model set with a total of 41,572 Gaussian components (just shy of the 41,677 components in the insertion/deletion rule set). This increase in the number of parameters resulted in an insignificant degradation in word error rate from 19.1% to 19.2%.

For the full rule rule set, the maximum number of Gaussian components per class was decreased from 75 to 67. This resulted in a new model set containing 41,712 components (just slightly larger than the insertion/deletion rule set). This decrease in the number of Gaussian components degraded the word error rate slightly from 19.0% to 19.2%.

These results confirm that the difference in performance between the three rule sets is not the result of a difference in the number of parameters provided to the acoustic model. The insertion/deletion rule set maintains superiority over the other two model sets even when their acoustic model sets are adjusted to use roughly the same number of parameters as the models of the insertion/deletion rule set.

4.6. Results with Trained FSTs

Table 4 shows the results on the full test set when various component FSTs are trained. By examining the first and second lines of Table 4, we see that training the phonological FST P improves the performance of the system using the full rule set (from 19.0% to 18.6%). This improvement is similar to past results we have obtained (Shu & Hetherington, 2002). The system exhibits a smaller improvement (from 18.4% to 18.2%) when training the P FST for the insertion/deletion rule set.[5] One can note that the insertion/deletion rule set with an untrained P still achieves a lower error rate than the full rule set using a trained P.

A comparison of the first and third lines of Table 4 shows that training the reductions FST R provides modest improvements to both systems. We also attempted to train the lexical FST L but did not achieve any performance improvement for either system from this training. We are also unable to report results for any system that combines a trained P with a trained R because the memory requirements for computing the composition of the individual component FSTs were prohibitively large. In past results using a slightly different pronunciation approach, where reductions were encoded directly within L, we were able to build a system which used both a trained P and a trained L within the final static FST to achieve a modest performance improvement (Shu

[5] This result differs slightly from our result in (Hazen *et al.*, 2002), where no improvement was observed when training the P FST for the insertion/deletion rule set. The new result was obtained after the correction of an error in our original evaluation of this rule set.

& Hetherington, 2002). We are currently investigating approximation methods to help reduce the size of the trained FSTs (and hence the memory requirements for building the final static FST).

5. Pronunciation Variation for Synthesis

Although this paper has focused on speech recognition, we have also utilized the same pronunciation framework in our group's concatenative speech synthesis system ENVOICE (Yi *et al.*, 2000; Yi & Glass, 2002). When applying the framework for synthesis, the FST network is given a sequence of words and is searched in the reverse direction (i.e., in generative mode) to find an appropriate sequence of waveform segments from a speech corpus to concatenate. In generative mode the phonological rules can also be weighted in order to provide preferences for specific types of phonological variation. For example, the synthesizer can be coerced into generating casual, highly-reduced speech, by weighting the FST networks to prefer reduced words, flaps and unreleased or deleted plosives. To generate well articulated speech the FST networks can be weighted to prefer unreduced words and fully articulated plosives.

6. Summary

This paper has presented the phonological modeling approach developed at MIT for use in the segment-based SUMMIT speech recognition system. We have evaluated the approach in the context of the JUPITER weather information domain, a publicly-available conversational system for providing weather information. Results show that the explicit modeling of phonological effects that cause the deletion or insertion of phonetic events reduced word error rates by 9% on our clean, in-vocabulary test set and by 4% over our full test set. Our results also demonstrated that phonological effects which cause allophonic variation without altering the number of phonetic events can be modeled implicitly with context-dependent models to achieve better accuracy and less search space complexity than a system which models these effects explicitly within phonological rewrite rules.

Anecdotal visual examinations of the phonetic transcriptions generated using a full set of phonological rules demonstrate a dramatic improvement in phonetic segmentation and classification accuracy during forced path recognition over a system using no phonological rules. This may not be of great consequence for word recognition, but it is vitally important for corpus-based concatenative synthesizers that rely on accurate automatically-derived time-aligned phonetic transcriptions in order to generate natural-sounding synthesized waveforms.

7. Future Work

While our work in this paper has been evaluated on spontaneous speech collected within a conversational system, we have found that human-human conversations tend to have even greater phonological variation than the human-machine data we have collected. Thus, we hope to evaluate our phonological modeling techniques on human-human corpora such as Switchboard or SPINE. We believe accurate modeling of phonological variation will have even greater benefits for these tasks.

While our paper has focused on modeling phonological variation within a sequence of independent FST layers, our group is also pursuing an approach which integrates the multiple layers within a single probabilistic hierarchical tree structure. This approach, called ANGIE, has the potential advantage of learning generalizations across the layers of the hierarchy which are currently modeled independently in our FST approach (Seneff & Wang, 2002).

Acknowledgements

The authors would like to acknowledge the efforts of both Jim Glass, who developed the initial versions of the JUPITER recognizer and lexicon used in this paper, and Jon Yi, who wrote the code to syllabify our various dictionaries. Jim and Jon are also the primary developers of the ENVOICE synthesizer discussed in this paper.

References

K. Church, *Phrase structure parsing: A method for taking advantage of allophonic constraints.* Ph.D. Thesis, Massachusetts Institute of Technology, Cambridge, Massachusetts, 1983.

A. Dempster, N. Laird and D. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society, Series B*, vol. 39, pp. 1–38, June 1977.

J. Eisner, "Parameter estimation for probabilistic finite-state transducers," In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, Philadelphia, Pennsylvania, July 2002. ACL, East Stroudsburg, Pennsylvania, pp. 1–8.

L. Gillick and S. Cox, "Some statistical issues in the comparison of speech recognition algorithms", In *Proceedings of the 1989 IEEE International Conference on Acoustics, Speech, and Signal Processing*, Glasgow, Scotland, May 1989. IEEE, Piscataway, New Jersey, pp. 532–535.

J. Glass, "A probabilistic framework for segment-based speech recognition," *Computer Speech and Language*, vol. 17, no. 2–3, pp. 137–152, April/July 2003.

J. Glass, T. J. Hazen and I. L. Hetherington, "Real-time telephone-based speech recognition in the JUPITER domain," In *Proceedings of the 1999 IEEE International Conference on Acoustics, Speech, and Signal Processing*, Phoenix, Arizona, March 1999. IEEE, Piscataway, New Jersey, pp. 61–64.

S. Greenberg, "Speaking in shorthand - A syllable-centric perspective for understanding pronunciation variation," *Speech Communication*, vol. 29, no. 2–4, pp. 159–176, November 1999.

T. Hain, "Implicit Pronunciation Modelling in ASR," In *Proceedings of the ISCA Tutorial and Research Workshop on Pronunciation Modeling and Lexicon Adaptation for Spoken Language*, Estes Park, Colorado, September 2002. ISCA, Bonn, pp. 129–134.

T. J. Hazen, I. L. Hetherington, H. Shu and K. Livescu, "Pronunciation Modeling Using a Finite-State Transducer Representation," *Proceedings of the ISCA Tutorial and Research Workshop on Pronunciation Modeling and Lexicon Adaptation for Spoken Language*, Estes Park, Colorado, September 2002. ISCA, Bonn, pp. 99–104.

I. L. Hetherington, "An efficient implementation of phonological rules using finite-state transducers," In *Proceedings of the 7th European Conference on Speech Communication and Technology (EUROSPEECH)*, Aalborg, Denmark, September 2001. Kommunik Grafiske Løsniger A/S, Aalborg, pp. 1599–1602.

D. Jurafsky *et al.*, "What kind of pronunciation variation is hard for triphones to model?" In *Proceedings of the 2001 IEEE International Conference on Acoustics, Speech, and Signal Processing*, Salt Lake City, Utah, May 2001. IEEE, Piscataway, New Jersey, pp. 577–580.

L. Lamel and G. Adda, "On designing pronunciation lexicons for large vocabulary, continuous speech recognition," In *Proceedings of the Fourth International Conference on Spoken Language Processing*, Philadelphia, Pennsylvania, October 1996. Citation Delaware, New Castle, Delaware, pp. 6–9.

K. Livescu and J. Glass, "Lexical modeling of non-native speech for automatic speech recognition," In *Proceedings of the 2000 IEEE International Conference on Acoustics, Speech, and Signal Processing*, Istanbul, Turkey, June 2000. IEEE, Piscataway, New Jersey, pp. 1842–1845.

D. McAllaster, L. Gillick, F. Scattone and M. Newman, "Fabricating conversational speech data with acoustic models: A program to examine model-data mismatch," In *Proceedings of the 5th International Conference on Spoken Language Processing*, Sydney, Australia, December 1998. Causal Productions Pty Ltd, Adelaide, Australia, pp. 1847–1850.

F. Pereira and M. Riley, "Speech recognition by composition of weighted finite automata," in *Finite-State Language Processing* (E. Roche and Y. Schabes, eds.), pp. 431–453, Cambridge, MA, MIT Press, 1997.

M. Riley *et al.*, "Stochastic pronunciation modelling from hand-labelled phonetic corpora," *Speech Communication*, vol. 29, no. 2–4, pp. 209–224, November 1999.

S. Seneff, "The use of subword linguistic modelling for multiple tasks in speech recognition," *Speech Communication*, Vol. 42, pp. 373-390, April 2004.

S. Seneff and C. Wang, "Modelling phonological rules through linguistic hierarchies," In *Proceedings of the ISCA Tutorial and Research Workshop on Pronunciation Modeling and Lexicon Adaptation for Spoken Language*, Estes Park, Colorado, September 2002. ISCA, Bonn, pp. 71–76.

I. Shafran, *Clustering wide contexts and HMM topologies for spontaneous speech recognition.* Ph.D. Thesis, University of Washington, Seattle, Washington, 2001.

H. Shu and I. L. Hetherington, "EM training of finite-state transducers and its application to pronunciation modeling," In *Proceedings of the 7th International Conference on Spoken Language Processing*, Denver, Colorado, September 2002. Causal Productions Pty Ltd, Adelaide, Australia, pp. 1293–1296.

N. Ström, I. L. Hetherington, T. J. Hazen, E. Sandness and J. Glass, "Acoustic modeling improvements in a segment-based speech recognizer," In *Proceedings of the IEEE 1999 Automatic Speech Recognition and Understanding Workshop*, Keystone, Colorado, December 1999. IEEE, Piscataway, New Jersey, pp. 139–142.

G. Tajchman, E. Fosler and D. Jurafsky, "Building multiple pronunciation models for novel words using exploratory computational phonology," In *Proceedings of the 4th European Conference on Speech Communication and Technology (EUROSPEECH)*, Madrid, Spain, September 1995. Gráficas Brens, Madrid, pp. 2247–2250.

J. Yi, J. Glass and L. Hetherington, "A flexible, scalable finite-state transducer architecture for corpus-based concatenative speech synthesis," In *Proceedings of the 6th International Conference on Spoken Language Processing*, Beijing, China, October 2000. China Military Friendship Publishers, Beijing, pp. 322–325.

J. Yi and J. Glass, "Information-theoretic criteria for unit selection synthesis," In *Proceedings of the 7th International Conference on Spoken Language Processing*, Denver, Colorado, September 2002. Causal Productions Pty Ltd, Adelaide, Australia, pp. 2617–2620.

V. Zue *et al.*, "JUPITER: A telephone-based conversational interface for weather information," *IEEE Transactions on Speech and Audio Processing*, vol. 8, no. 1, pp. 85–96, January 2000.