Combining Linguistic Knowledge and Acoustic Information in Automatic Pronunciation Lexicon Generation

Grace Chung†, Chao Wang‡, Stephanie Seneff‡, Ed Filisko‡, Min Tang‡

†Corporation for National Research Initiatives
1895 Preston White Drive, Suite 100, Reston, VA 22209
gchung@cnri.reston.va.us

‡MIT Computer Science and Artificial Intelligence Laboratory
The Stata Center, 32 Vassar Street, Cambridge, MA 02139
{wangc,seneff,filisko,mtang}@csail.mit.edu

Abstract

This paper describes several experiments aimed at the long term goal of enabling a spoken conversational system to automatically improve its pronunciation lexicon over time through direct interactions with end users and from available Web sources. We selected a set of 200 rare words from the OGI corpus of spoken names, and performed several experiments combining spelling and pronunciation information to hypothesize phonemic baseforms for these words. We evaluated the quality of the resulting baseforms through a series of recognition experiments, using the 200 words in an isolated word recognition task. We also report here on a modification to our letter-to-sound system, utilizing a letter-phoneme n-gram language model, either alone or in combination with our original "column-bigram" model, for additional linguistic constraint and robustness. Our experiments confirm our expectation that acoustic information drawn from spoken examples of the words can greatly improve the quality of the baseforms, as measured by the recognition error rate. Our ultimate goal is to allow a spoken dialogue system to automatically expand and improve its baseforms over time as users introduce new words or supply spoken pronunciations of existing words.

1. Introduction

Over the years, our work has addressed the dynamic addition of new words into a spoken dialogue system via verbal entry. The overall objective is to develop systems that can intelligently handle the incidence of new words through accurate detection and deduction of their spellings and pronunciations, as well as their dynamic and seamless incorporation into the system lexicon. This is particularly pertinent to narrow domain systems that provide on-line information, where the database contains a large set of proper names that are likely to change frequently.

In the past [1], we reported on a system that can recognize spellings and pronunciations of open vocabularies of proper names using a unified framework that combines sublexical modeling with bi-directional letter-to-sound conversion. Previous work has mainly addressed spelling extraction, as opposed to phonemic extraction, in recognizing unknown words.

†The research at CNRI is sponsored in part by SPAWAR SSC-SD. The content of this paper does not necessarily reflect the position or policy of the Government, and no official endorsement should be inferred. ‡The research at MIT is supported in part by an industrial consortium supporting the MIT Oxygen Alliance.

In this paper, we further our research by examining the quality of phonemic baseforms derived using our letter-to-sound capabilities, and comparing performance for alternate ways of extracting pronunciations for unknown words, given the user waveforms. While many researchers are addressing the letter-to-sound problem [2, 3, 4, 5, 6], and some have reported on baseform generation from acoustic data alone [7, 8], this paper distinguishes itself from other work in that our letter-to-sound models are applied directly during recognition to generate baseforms for the acoustic data.

We envision a future system that is able to acquire new words automatically through interaction with the user and Web sources. For instance, if the user asks about book stores in Saint Louis, the system could immediately download a candidate list from the Web, and update the recognizer's dynamic vocabulary to reflect the immediacy of those words. Any words which are as yet unavailable in its large off-line lexicon could be entered with a pronunciation obtained via a letter-to-sound system. Further interaction with the user might involve a request for one of the originally missing words. A follow-up subdialogue might solicit a spoken spelling of the word, which would become a further resource to the system. Subsequent confirmation by the user would assure the validity of the user's spoken rendering. The system could then update its lexical entry for this rare word, based on the user's pronounced example. Over time, many words in the lexicon would become refined in this manner, and the quality of the pronunciation lexicon would steadily improve.

This scenario poses several research questions that we attempt to answer formally here. Our experiments focus on a set of around 200 words from the OGI names corpus [9], selected specifically because they do not appear in our pre-existing 100,000 word lexicon of people's first and last names. We ask several questions: (1) how well can the letter-to-sound system do by itself in proposing pronunciations for these words, (2a) how much can these pronunciations be improved by taking advantage of a previously spoken instance of this word, but by another speaker, (2b) how much difference does it make if the previous pronunciation is by the same speaker,[1] and (3) what happens if the spelling is not exact, but is instead available only via recognition from a spoken spelling? In all of our experiments, we evaluate by performing a recognition task using the SUMMIT speech recognizer [10], with a lexicon consisting of the 200 OGI words. This is an approximation to the situation that would exist with the dynamic vocabulary recognizer. Furthermore,

[1] Here we use the exact same word as a stand-in for a previous instance of the word, recognizing that this gives an upper-bound performance result on this scenario.

since we do not have available in our corpus the spoken spellings of the words, we synthesize the spellings using the Festival speech synthesizer.

Another aspect of this paper is the introduction of a modification to our statistical letter-to-sound model to use grapheme-phoneme trigrams in addition to the original "column-bigrams," [1] and a demonstration that this improves overall letter-to-sound results in our experiments.

Over the next sections, the two letter-to-sound models, the column-bigram and a joint grapheme-phoneme n-gram model, are briefly described. Following this, a series of experiments for deriving pronunciations using letter-to-sound alone and in combination with user waveforms are presented.

2. Approach

2.1. ANGIE Column-Bigram Method

In previous work [11, 12], a hierarchical framework known as ANGIE, capturing subword structure, has been used to predict phoneme-grapheme mappings. This framework combines corpus-based statistical methods with explicit linguistic information to generalize from the observation space to unseen words. In the previously developed paradigm, the hierarchical models are converted to a finite-state transducer (FST) representation, providing sound-to-letter mappings. The FST configuration captures bigram statistics on units identified as vertical columns of a parse table, which we refer to as a "column-bigram." The grapheme and phoneme units are enriched with morph-syllabic properties and lexical stress.

In this work, the probability model is trained on a lexicon of proper nouns, containing both first and last names. About 100,000 proper nouns are used via a semi-automatic procedure, described in [12]. In total, there are 214 unique graphemes (some of which are doubletons such as "th") and 116 unique phoneme units.

2.2. Letter-Phoneme N-gram Models

Recently, many researchers have employed the joint modeling of phonemic and graphemic units to address the letter-to-sound problem [4, 3, 2]; each method employs various means for finding alternate alignments of phonemes with graphemes. In this work, we can take advantage of ANGIE parse trees to align graphemes with phonemes. The n-gram model can then be trained on alignments derived automatically by parsing a large training corpus. The pre-terminal and terminal units of each column of the parse chart are concatenated together to form each grapheme-phoneme unit, thus creating a grapheme-phoneme baseform for each training datum. Hence, the joint probability distribution for letters (l) and phoneme units (p), $p(l,p)$ is modeled using standard n-gram models:

$$p(LP_1 \cdots LP_m) = \prod_{i=1}^{m} p(LP_i | LP_{i-n+1} \cdots LP_{i-1}) \qquad (1)$$

where LP stands for "letter-phoneme" and represents a grapheme-phoneme unit[2], and m is the number of letter-phonemes in a word. This unit is determined strictly from the vertical columns of the ANGIE parse table. Table 1 illustrates some example names and their baseforms in our training. There are in total 592 letter-phonemes. Both the underlying phoneme and letter sequences are easily extracted from the baseform.

[2]For clarity, we will henceforth refer to these units as "letter-phoneme" (LP) units, recognizing however that such a unit may contain more than one letter in some circumstances.

latoria *l_l! a_ah t_t! o_aor+ r_null i_iy a_ah*
middleton *m_m! i_ih+ d_d d2_d! le_el t_t! o_en n_null*
streetman *st_st! r_r ee2_iy+ t_t m_m! a_ae n_n*

Table 1: Example baseforms for some names in the training set. Letter-phoneme units each comprise a grapheme component and a phoneme component. The phoneme component may be "null" or one of a set of special units marked for stress and syllable onset.

The LP n-gram model is trained on the same 100,000 word training set as the column-bigram method. Given the spelling of a word, the highest scoring letter-phoneme sequence, according to the n-gram model, is computed, and the subsequent phonemic sequence is extracted. These computations are achieved via FST operations.

2.3. Extraction of pronunciation from waveforms

As both of the above letter-to-sound systems are encapsulated within an FST framework, as is our speech recognizer, SUMMIT [10], when the spelling of a word is given, it is possible to create an FST supporting the various phonemic sequences as specified by the model for the corresponding input letter sequence. This is achieved via FST composition of a spelling FST to constrain the letter-to-sound FST to the input spelling. The resulting FST is also accompanied by scores from the probability model. This will be used as the path constraint in a forced alignment of the waveform.

When using the SUMMIT recognizer, phonemic labels are expanded by phonological rules, and then mapped to context-dependent labels. These phonemic labels are also projected to the FST output to be extracted at the output of the alignment search.

3. Experiments

The experiments are formulated such that the quality of extracted baseforms is evaluated by the recognition performance for a set of names that are not present in training. Two waveform data sets with the same 198 words are drawn from the OGI Names Corpus [9]. One will be used for testing recognition accuracy, while the other will be used for deriving pronunciation baseforms.

Note that the test data were not artificially excluded from the training corpus. Rather, the test set was defined by selecting words in the OGI corpus that had not shown up in our previous collections of proper nouns from on-line sources. Hence we expect that many of the names are quite rare and dissimilar to the ANGIE training data. When, in a real dialogue application, a name is introduced, we assume that the system consults a dictionary first to search for a possible pronunciation prior to invoking a letter-to-sound module.

It can be argued that measuring recognition performance on a test set can be a better performance indicator than comparison with a human transcribed reference because name pronunciations are sometimes difficult to determine and are particular to the individual owner. Hence the task will consist of recognizing 198 isolated name waveforms, given the derived pronunciations. The speech recognizer [10] has no additional language models.

The first two of the following experiments will address the scenario where the spelling of a word is known. Hence the framework will use the letter-to-sound capability to extract phonemic baseforms from the correct spelling. The third experiment addresses the situation where the word in question is unknown – it is assumed that the user has also provided a spelled waveform. Therefore, spelling

Algorithm	WER (%)
1. CB + LP	33.8
2. LP Only	30.8

Table 2: Recognition error rate on a 198 isolated word task: each word was out-of-vocabulary, with baseforms derived from (1) a column-bigram model (CB) with a letter-phoneme trigram (LP) as back-off, and (2) a letter-phoneme trigram alone. Two alternate pronunciations are used for each word in the baseform. Note that ANGIE parsing was used to align the letters with the phonemes in the training set used by both systems.

hypotheses from a letter recognizer are combined with the letter-to-sound capability to propose the pronunciation baseforms.

3.1. Letter-to-Sound Evaluation

In the first experiment, the pronunciations for each word are derived solely from the letter-to-sound mappings on the FSTs. That is, no information from additional waveforms is used.

While ANGIE's probability model can generalize its observation space for some unseen words, the column-bigram FST, as a more compact representation, does not capture all of ANGIE's generalization capability. Rare sequences that have not been seen in training may occasionally result in hard failures. For our test set, there are 14 hard failures using the column-bigram. Hence, for a fair comparison on the recognition task, we examine the baseforms from (1) where the column-bigram FST (CB) performs letter-to-sound conversion, with failed utterances resorting to the letter-phoneme n-gram (LP) for back-off, and (2) where letter-to-sound conversion is performed entirely by the LP n-gram FST.

Results, tabulated in Table 2, show that the LP model alone in fact performs better than the hybrid CB-LP model. Note that the letter-phoneme alignments were derived from the ANGIE parse tree, and a trigram LP model with deleted interpolation for smoothing was used in the experiments. Further investigations with higher order n-grams did not yield any gains.

In the first system, the 198 word test set causes 14 hard failures in ANGIE, whose baseforms are subsequently obtained from the letter-phoneme model.

We also consider varying the number of alternate pronunciations. However, for all the experiments, using two variants appear to be ideal. Increasing the number of alternates rapidly deteriorates word accuracy. Under further examination, the correct pronunciation variant may often occur at the second hypothesis. Using the letter-phoneme model, with only one baseform per word, the word error rate (WER) is 35.9%. As using two variants generally afford superior performance, this parameter is held constant for the remainder of the experiments.

As an independent benchmark, we compare the letter-to-sound performance with a decision-tree-based framework as described in [6]. The decision tree is trained on our letter-phoneme lexicon[3], and achieve 35.4% word error rate when the derived baseforms are used in the recognition experiment.

3.2. Pronunciation Extraction from Waveforms

In the second experiment, we examine the extraction of phonemic baseforms from user waveforms by incorporating the letter-to-sound FSTs during forced alignment, as described in Section 2.3. Two conditions are examined: (1) using an alternate waveform to extract baseforms, then testing on our test corpus, and (2) using the same waveforms for baseform extraction and testing. The speakers for the two test sets did overlap some. However, about 87% of the speakers for these names differed for the two waveform sets.

[3]The tools were downloaded from http://www-2.cs.cmu.edu/ lenzo/t2p/.

Algorithm	WER (%)
Using A 2nd Waveform to Derive Pronunciation	
3. LP FST with CB L2S	18.7
4. LP FST with LP L2S	18.7
5. CB + LP FST with CB L2S	17.2
6. CB + LP FST with LP L2S	16.7
Using the Test Waveform to Derive Pronunciation	
7. LP FST with CB L2S	15.2
8. LP FST with LP L2S	14.1
9. CB + LP FST with CB L2S	11.6
10. CB + LP FST with LP L2S	10.6

Table 3: Recognition error rates for various configurations using pronunciations derived from waveform data, assuming exact spellings of the words are available. See text for details.

Table 3 shows both sets of results. Systems 3–6 refer to extraction from a second waveform. Systems 7–10 refer to baseform extraction from the same waveform. It can be seen that all results deriving phoneme sequences from the waveform yield improved recognition compared to the letter-to-sound algorithms by wide margins. That is, acoustic data contributes greatly to the generation of phonemic pronunciations, as might be anticipated.

In considering the extraction of pronunciation from waveforms, several ways of combining the two methods are explored. This is important because forced alignments on the waveforms sometimes failed, as a result of all paths being pruned away, having fallen below the score thresholds during alignment. This was applicable to both the column-bigram and letter-phoneme methods. In the case of failed forced alignment, the phoneme baseform has to be derived from direct letter-to-sound conversion (without any contribution from the acoustic waveform), using either the column-bigram FST (CB L2S in Systems 3, 5, 7 and 9), or the letter-phoneme FST (LP L2S in Systems 4, 6, 8 and 10). Systems 3 and 4 use only the letter-phoneme FST for the forced alignment phase. Three utterances failed the forced alignment procedure here. Systems 5 and 6 use the column-bigram FST, where possible; for the 14 utterances that fail the column-bigram letter-to-sound, the phonemic baseform could not be derived using the column-bigram FST, and so, when performing the forced alignment, the letter-phoneme FST is used. This is denoted by CB + LP FST in Table 3.

Although for the letter-to-sound-only experiments, the letter-phoneme trigram outperformed the column-bigram method, it was found that, when dealing with waveform data, the hybrid approach combining the column-bigram and the letter-phoneme trigram, which acts as a smoothing mechanism for the column-bigram FST, yields better results. The optimal result here is 16.7% WER - we resort to the letter-phonemes trigram to perform letter-to-sound conversion when forced alignment fails for both the column-bigram and the letter-phoneme FSTs.

In the above two waveform sets, since the speakers vary, the pronunciations of names also differ. When pronunciation is extracted from the test waveform itself, results improve markedly. This is expected because the pronunciation extraction is self-referential. Again, the optimal condition occurs for a hybrid method combining the column-bigram and the letter-phoneme trigram, yielding 10.6% WER. Upon examination, it is found that the pronunciations of the proper nouns for the two waveform sets are quite variable, because even though the names were spelled the

Algorithm	WER (%)
Using A 2nd Waveform to Derive Pronunciation	
11. CB	24.7
12. CB concatenated	22.2
Using the Test Waveform to Derive Pronunciation	
13. CB	14.1
14. CB concatenated	13.1

Table 4: Recognition error rate on an isolated word task, using baseforms derived from spoken and synthesized spelling waveforms.

same, they often belonged to different people who pronounced them differently. Therefore, baseforms derived from one waveform were naturally not well matched to the test waveform pronunciation.

3.3. Incorporating Spoken Spellings

In the third experiment, we consider a situation whereby a speech interface is required to elicit not only the pronunciation of an unknown word but also its spelling. This would be applicable to a verbally entered unknown word. The test corpus used here did not contain spelled versions of these words, and therefore we simulated the conditions by using a speech synthesizer [13] to generate spelled version of the names. It turned out that the recognizer performance of the spelled synthetic data produced a letter error rate of 17.7%. In previous work, letter error rates, using the same speech recognizer, were much lower, when using real user spelled data. Our conjecture is that the intelligibility and naturalness of the synthesized waveforms are quite poor, thus causing poorer performance.

Instead of a single spelling, a spelling graph of the recognition hypotheses was used as input to the FST operations. The first method will use the ANGIE column-bigram FST only to create the phoneme baseforms. In an additional experiment, a method similar to one described earlier in [12] is used. The spoken waveform and the synthesized spelling waveform are concatenated, and a simultaneous search constrained that, for each hypothesis, the spellings of the spelled and spoken parts are identical. As explained above, the output of the alignment phase is a set of phonemic baseforms, which will be used for the name recognizer for evaluation.

Results are tabulated in Table 4. Systems 11 and 12 used one waveform set to extract pronunciations and the standard test set to compute recognition accuracy. Systems 13 and 14 used the same waveforms for deriving pronunciations and testing recognition. Systems 11 and 13 composed the letter graph from the letter spelling recognizer with the column-bigram FST to conduct the forced alignments. Systems 12 and 14 were augmented with the concatenation and simultaneous constraint that the spellings of the spelled and spoken parts would be identical.

In general, it can be seen that, even though letter recognition accuracies are quite poor, phoneme extraction can still be performed. And the quality of the phonemes is still much better than what is obtained when using a single letter-to-sound converter, that is, the true spelling is known but no waveforms are available. WER ranged from 13.1% to 24.7% compared to the best WER of 30.8% in letter-to-sound derivation only in Table 2. Again using the same waveform for pronunciation extraction and testing gives better recognition accuracy.

Concatenating the spoken waveform with the spelled one produces further improvement, as is consistent with previous results on letter recognition (from 24.7% to 22.2% for different waveforms and from 14.1% to 13.1% for the same waveform).

4. Conclusions and Future work

This paper has examined the performance of a combined knowledge-based and data-driven letter-to-sound framework, applied to automatically generate pronunciation baseforms from acoustic information and full or partial spelling knowledge.

One extension is to explore the migration to other domains concerning proper nouns for applications such as restaurants, hotels, shopping, city guides and so forth. Currently, we are investigating semi-automatic methods such as co-training using the ANGIE framework with the joint letter-phoneme modeling, in an effort to reduce the time to port to new domains, and to pool new data sets for training.

In the future, the optimized letter-to-sound capabilities will be integrated with a spoken dialogue system that can splice out spoken unknown words embedded within sentences, and elicit spellings of new words from users. These features will also be integrated with the ability to dynamically update the recognizer lexicon, during an ongoing conversation [14].

5. References

[1] G. Chung, S. Seneff, and C. Wang. Automatic acquisition of names using speak and spell mode in spoken dialogue systems. In *Proc. of HLT-NAACL*, Edmonton, Canada, 2003.

[2] L. Galescu and J. Allen. Name pronunciation with a joint n-gram model for bi-directional grapheme-to-phoneme conversion. In *Proc. ICSLP*, Denver, Colorado, 2002.

[3] M. Bisani and H. Ney. Multigram-based grapheme-to-phoneme conversion for lvcsr. In *Proc. Eurospeech*, Geneva, Switzerland, 2003.

[4] S. Chen. Conditional and joint models for grapheme-to-phoneme conversion. In *Proc. Eurospeech*, Geneva, Switzerland, 2003.

[5] R. I. Damper, Y. Marchand, M. J. Adamson, and K. Gustafson. Comparative evaluation of letter-to-sound conversion techniques for english text-to-speech synthesis. In *Proc. Third ESCA/COCOSDA Workshop on Speech Synthesis*, pages 53–58, Blue Mountains, NSW, Australia, 1998.

[6] A. Font Llitjos and A. Black. Knowledge of language origin improves pronunciation accuracy of proper names. In *Proc. Eurospeech*, Aalborg, Denmark, 2001.

[7] S. Deligne and L. Mangu. On the use of lattices for the automatic generation of pronunciations. In *Proc. ICASSP*, Hong Kong, China, 2003.

[8] B. Maison. Automatic baseform generation from acoustic data. In *Proc. Eurospeech*, Geneva, Switzerland, 2003.

[9] Names 1.3, the cslu ogi names corpus. http://cslu.cse.ogi.edu/corpora/name/.

[10] J. Glass. A probabilistic framework for segment-based speech recognition. *Computer Speech and Language*, 2003.

[11] G. Chung. A three stage solution for flexible vocabulary speech understanding. In *Proc. ICSLP*, Beijing, China, 2000.

[12] S. Seneff, G. Chung, and C. Wang. Empowering end users to personalize dialogue systems through spoken interaction. In *Proc. Eurospeech*, Geneva, Switzerland, 2003.

[13] The festival speech synthesis system. http://www.cstr.ed.ac.uk/projects/festival/.

[14] G. Chung. A dynamic vocabulary spoken dialogue interface. In *These proceedings.*