Comparison of Two Tree-Structured Approaches for Grapheme-to-Phoneme Conversion

Ove Andersen[1], Roland Kuhn[2], Ariane Lazaridès[2], Paul Dalsgaard[1], Jürgen Haas[3], Elmar Nöth[3]

[1]Center for PersonKommunikation, Aalborg University, Denmark
[2]Centre de recherche informatique de Montréal, Canada
[3]University of Erlangen-Nürnberg, Germany

ABSTRACT

Recently, we described a two-step self-learning approach for grapheme-to-phoneme (G2P) conversion [1]. In the first step, grapheme and phoneme strings in the training data are aligned via an iterative Viterbi procedure that may insert graphemic and phonemic nulls where required. In the second step, a Trie structure encoding pronunciation rules is generated. In this paper we describe the alignment module, and give alignment accuracies on the NETtalk database. We also compare transcription accuracies for two approaches to the second step on three databases: the NETtalk database, the CMU dictionary and the French part of the ONOMASTICA lexicon. The two transcription approaches applied in this research are a Trie approach [1] and an approach based on binary decision trees grown by means of the Gelfand-Ravishankar-Delp algorithm [2,3,4]. We discuss the choice of questions for these decision trees - it may be possible to formulate questions about groups of characters (*e.g.*, "is the next letter a vowel?") that yield better trees than those that only use questions about individual characters (*e.g.*, "is the next letter an 'A' ?"). Finally, we discuss the implications of our work for G2P conversion.

1. INTRODUCTION

An important prerequisite for services involving speech recognition and/or speech synthesis is information about the correspondence between the orthography and the pronunciation(s). Many applications involve using a dynamic vocabulary for which it would be impractical (read impossible) to establish a dictionary with complete coverage, and therefore they call for automatic grapheme-to-phoneme (G2P) conversion.

A traditional way of handling words not present in a dictionary is to apply a rule-based system for transcription; such systems demonstrate impressive performance for some tasks [5]. However, rule-based systems have an inherent problem with maintenance. It is difficult to change some of the rules without introducing unwanted side effects. Furthermore, porting such systems to new tasks and especially to new languages is extremely time consuming and requires expert phonetic knowledge. Instead, we propose a two-step self-learning approach which automatically derives rules for G2P conversion from training data. In the first step, corresponding grapheme and phoneme strings in the training data are aligned; in the second step, either a binary decision tree or a Trie lookup data structure learns and stores G2P conversion rules from the aligned strings.

2. ALIGNING THE DATABASES

The training data consist of many matching pairs of grapheme and phoneme strings

```
graphemes:  a f f i x -
            | | | | | |
phonemes:   a f - i k s
```

Figure 1: Alignment of graphemes and phonemes ('-' = graphemic / phonemic null)

The alignment of the two strings within a pair is carried out by an iterative Viterbi algorithm that may insert graphemic and phonemic nulls in order to ensure that the grapheme string has the same length as its phoneme string counterpart. The basis for this alignment is the set of probabilities $Pr(\textit{ grapheme } i \mid \textit{phoneme } j)$, which for all but the first iteration are estimated from the output of the previous iteration. For the first iteration, these probabilities are estimated from the grapheme and phoneme strings having equal length (before nulls are inserted). More details are given in [1].

The next step is to employ this aligned database for training the decision tree or the Trie structure. This will be described further in the next two sections.

3. BINARY DECISION TREE APPROACH

Binary decision trees are well-known pattern recognition tools that have been applied to a variety of problems in speech recognition and understanding, as well as in other fields [2,3,4]. They are remarkable for their robustness and their ability to combine diverse information sources. To grow a decision tree on a set of labelled training data items, one must supply three elements:

- a set of possible yes-no questions;
- a rule for selecting the best question at a node;
- a method for pruning trees to prevent over-training.

The choice of the question set depends entirely on the application (and on the ingenuity of the researcher), while the other two elements are application-independent. For the experiments reported here, we employed the well-known Gini criterion ([2], pp.

103-104) to pick the best question at a node, and used the Gelfand-Ravishankar-Delp algorithm [3] to carry out pruning.

For the G2P problem, we grow one decision tree per grapheme. Consider the grapheme 'G'. The training data items for the 'G' tree consist of all the aligned grapheme-phoneme pairs in which the grapheme string contains a 'G'. The questions asked were about the graphemic context. For instance, if we denote the position of the grapheme of interest ('G') as 0, the preceding grapheme position as -1, the succeeding position as +1, and so on, we can generate questions like: "Is +1 'H'?" "Is -1 'U'?" "Is -2 'O'?", and so on.

We allowed questions about positions -5, -4, ..., -1, +1, ..., +4, +5. Each leaf of the resulting tree will assign probabilities to possible phonemic realizations of the grapheme. For instance, if there is a leaf of the 'G' tree corresponding to [-2='O' & -1='U' & +1='H'], this leaf should assign high probability to phoneme 'f'. Note that in the Trie approach (see next section) the order of grapheme positions referred to is the same over all graphemes; in the decision tree approach, it is possible that, for instance, the question at the root of the 'G' tree will be about position +1, while the question at the root of the 'R' tree concerns position -2.

In speech recognition, there is a well-known problem concerning the modelling of a phoneme in context [4]. In the questions considered by decision trees grown to solve this problem, it has proved useful to group similar phonemes together: for instance, a decision tree may contain questions like "Is -1 a diphthong?" Although the questions in the G2P tree concern graphemes rather than phonemes, it is of interest to see whether questions about grapheme groups yield better trees. For our experiments, we allowed questions about 10 classes of graphemes, which were more or less a transposition of the phonetic classes used in [4]: vowels, consonants, nasals, plosives, fricatives, affricatives, glicals, liquids, glicals *or* liquids, and diphthongs (*i.e.*, letters that can participate in a diphthong).

The trees using classes of questions gave better results than those allowing only questions on single graphemes, as expected. Note that the first column in Table 1 gives the context span on both sides, *e.g.* "1" means that questions about -1 and +1 are allowed. All subsequent results for decision trees will thus be for trees using question classes.

NETtalk	Without class questions (%)		With class questions (%)	
	Phoneme	Word	Phoneme	Word
1	82.1	27.2	82.2	27.6
2	88.5	47.4	88.8	48.4
3	89.7	51.6	89.9	52.3

Table 1: Transcription accuracy obtained with decision trees with and without class questions as a function of context span.

4. TRIE STRUCTURED APPROACH

The Trie consists of leaves and branches. Each leaf records statistics about one grapheme in a well defined context. These statistics are in the form of a list which, for a given grapheme Gj, contains the number of occurrences n_i of each possible phoneme ϕ_i as found in the specific context in the entire training database. Furthermore, each leaf has a pointer to the following leaf. The path into the tree structure is defined by an order obtained from a calculation of mutual information; *i.e.*, graphemes with highest mutual information are considered first. For details, see [5].

After the training, each node (which represents a specific graphemic context) will have a list of possible phonemes and their corresponding probabilities.

During the use of the Trie structure for transcribing new words each grapheme in its context is looked up in the Trie. When the optimal match has been found one of the phonemes available in the list on this leaf is output. In the most simple case this is the most probable phoneme.

5. DATABASES

The databases used in this research comprise three corpora: the NETtalk database [6], the CMU dictionary [7], and the ONOMASTICA database for French [8]. The NETtalk database consists of 20,000 American-English words and their pronunciation. This database has the advantage that the graphemes and phonemes have been aligned manually, allowing comparison between automatic and manual alignment.

The CMU database does not contain alignments, but permits more comprehensive testing of transcription accuracy, since it includes more than 100,000 words, each accompanied by its American-English transcription.

The third database is useful for assessing the portability of these approaches across languages: it gives the French pronunciation of 100,000 surnames found in a French telephone directory. These data are part of the ONOMASTICA[1] database covering 11 European languages and a total of 8.5 million proper names.

Table 2 summarizes the sizes of the three databases.

		Phonemes	Words
NETtalk	Train	96606	15000
	Test	31174	5000
CMU dict.	Train	469074	75069
	Test	187354	29999
ONOMAS-TICA	Train	334166	63927
	Test	167225	31964

Table 2: Number of phonemes and words in the training and testing sections of the three databases

1. The ONOMASTICA database is only available to academic partners in the ONOMASTICA project and only for research purposes.

6. RESULTS

In this section, we first calculate the accuracy of the alignment component, then compare the transcription accuracy of the decision tree with that of the Trie.

6.1. Alignment

The NETtalk database is employed to evaluate the alignment accuracy (since the NETtalk data have been aligned manually). In Table 3, the percentage of correctly aligned phonemes and words is shown after the first four iterations. A correctly aligned phoneme is one located in the same position as the manually aligned phoneme, while a correctly aligned word is a word without any alignment errors at the phoneme level.

Acc.	1 Iter.	2 Iter.	3 Iter.	4 Iter.
Phon.	84.7%	94.1%	93.2%	93.2%
Word	78.1%	85.1%	83.7%	83.7%

Table 3: Alignment performance on NETtalk database

It is seen that the overall alignment accuracy saturates after only two iterations.

To measure the effect of the automatic alignment on the performance, we have carried out experiments on NETtalk with the manually aligned data, then with the automatically aligned data.

	Manual alignment		Automatic alignment	
Context span	Decision tree (%)	Trie (%)	Decision tree (%)	Trie (%)
1	82.5	82.8	82.2	82.4
2	89.2	89.1	88.8	88.9
3	90.6	89.8	89.9	89.6

Table 4: Phoneme transcription performances for Nettalk

The results in Table 4 indicate that the use of automatic rather than manual alignment does not cause any significant drop in performance for the two approaches.

6.2. Transcription Accuracies Obtained using the Two G2P Approaches

The two transcription approaches are evaluated and compared to each other. The results for NETtalk data are shown in Table 5.

The results for NETtalk suggest that given the training data available, trees with a context span of 5 cannot be properly trained. They also show that for this training set size (15,000 words), the two approaches are roughly equivalent.

On CMU, with much more training data (75,000 words vs. 15,000 for NETtalk) the two approaches still have similar performance, but less so than on NETtalk (see Table 6). They both seem to reach a ceiling around a context span of 3 (wider spans don't yield better performance).

NETtalk	Decision tree (%)		Trie (%)	
	Phoneme	Word	Phoneme	Word
1	82.2	27.6	82.4	27.9
2	88.8	48.4	88.9	47.8
3	89.9	52.3	89.6	50.6
4	89.9	53.0	89.7	51.4
5	89.8	52.6	89.8	51.7

Table 5: Transcription accuracy obtained on the NETtalk database as a function of context span

CMU	Decision tree (%)		Trie (%)	
	Phoneme	Word	Phoneme	Word
1	83.3	32.7	83.7	31.6
2	89.3	49.1	88.6	46.8
3	90.8	56.9	89.0	48.1
4	91.1	58.0	89.1	48.4
5	91.1	57.9	89.1	48.5

Table 6: Transcription accuracy obtained on CMU

The French ONOMASTICA tests allowed us to show that without any modifications to the software, the Trie and the decision tree approaches both perform well when trained and tested on a new language (see Table 7). Note that because of the large amount of training data, the level of accuracy is very high.

ONOMASTICA	Decision tree (%)		Trie (%)	
	Phoneme	Word	Phoneme	Word
1	92.3	64.1	92.5	58.0
2	97.4	88.0	96.6	74.8
3	97.8	90.2	96.6	75.2
4	97.9	90.5	96.6	75.1
5	98.0	90.8	96.6	75.2

Table 7: Transcription accuracy on French ONOMASTICA

To gain insight about the strengths and weaknesses of the two approaches, we tried to run experiments using only a fraction of the original NETtalk training data. We kept a context span of 3 (which generally gives good results), and increased the size of the training data from 1,000 words to 15,000 words. The results are shown in Table 8. The performance difference between the two approaches when there are few training data available is probably due to the better generalization capability of the decision trees.

We also examined the questions chosen for the decision trees grown on CMU data. Questions about individual graphemes (*e.g.*, 'B') predominated over questions about classes, though the latter made up about 1/3 of the questions. Among the class-based

Size of training db (#words)	Decision tree (%)		Trie (%)	
	Phoneme	Word	Phoneme	Word
1000	83.2	29.8	80.5	21.5
2000	84.9	35.7	82.9	27.9
3000	86.3	40.7	84.7	32.4
4000	86.7	42.2	85.6	36.6
5000	87.2	44.4	86.5	39.3
10000	89.0	49.3	88.5	46.9
15000	89.9	52.3	89.6	50.6

Table 8: Transcription accuracy for various amounts of training data from NETtalk (context span fixed to 3)

questions, the classes vowel, consonant, and diphthong were mentioned most frequently.

6.3. Computational Requirements

The size of the trees generated by the Trie and the decision tree for the ONOMASTICA data range from 12Mb and 5 Mb for a context span of 5 to 66kb and 337kb for a context span of 1, respectively. However, both structures can be encoded much more compactly if necessary.

Once generated, both structures can be used to generate phoneme strings from grapheme strings very quickly (*e.g.*, a few minutes for 30,000 grapheme strings). However, the processing time required for training the decision tree is much greater than that required for training the Trie. Using a SUN Sparc 20 for training with a context span of 5 on the ONOMASTICA data, it took approximately 5 minutes to generate the Trie and 60 hours to generate the decision tree (two expansions and two prunings). This difference is due to the question-picking process that must be executed at each node of the decision trees. However, the training is done off-line and only once, so time needed for training the two approaches might not be of a crucial importance. Note that these figures were obtained without optimizing the software for speed.

7. CONCLUSIONS AND FUTURE WORK

Both approaches presented in this paper are viable alternatives to traditional rule-based systems for grapheme-to-phoneme conversion. In a comparative study of the Trie approach and a system based on rewrite rules, it was concluded that they yielded approximately equal performance [9]. However, the Trie and the decision tree approaches have an important advantage: they are easily updated on new data and easily ported to new languages. For rule-based systems new data calling for an extension of the rules, or a new language, may create complications. It is well-known that new rules can cause unwanted side-effects which again require patches.

Of course, both the Trie and the decision tree approaches require a training database of grapheme/phoneme pairs. If class-based questions are to be used, the decision tree approach also requires a specification of grapheme classes. In practice, this is unlikely to pose a problem: the decision tree approach is remarkably robust with respect to the definitions of the grapheme classes.

Our future work will be on the search algorithms used to generate phoneme strings from the Trie or tree structures. In earlier work [1], some of us have already studied search algorithms that combine bigram phoneme information with the information in the Trie to generate a phoneme string from a given grapheme. Apart from extending the span of this phonotactic information (*e.g.*, to phoneme trigrams) we are also considering an approach in which decision trees could combine graphemic and phonotactic information to score phoneme sequences. *I.e.*, a decision tree might contain questions both about the grapheme string, and about the previously generated phonemes. We are seeking an algorithm that generates pronunciations that cover the pronunciation space of a word as much as possible, rather than obtaining two or three very likely but very similar pronunciations.

8. REFERENCES

[1] Andersen, O., and Dalsgaard, P., "Multi-lingual testing of a self-learning approach to phonemic transcription of orthography". In *EUROSPEECH95*, pages 1117–1120, September 1995.

[2] Breiman, Friedman, Olshen, and Stone, "Classification and Regression Trees". Wadsworth Inc., 1984.

[3] Gelfand, S., Ravishankar, C., and Delp, E., "An Iterative Growing and Pruning Algorithm for Classification Tree Design". In *IEEE Pattern Analysis and Machine Intelligence*, pages 163-174, Feb. 1991.

[4] Kuhn, R., Lazaridès, A., Normandin, Y., and Brousseau, J., "Improved decision trees for phonetic modeling". In *ICASSP95*, pages 552–555, 1995.

[5] Andersen, O., and Dalsgaard, P., "A self-learning approach to transcription of Danish Proper Names". In *ICSLP94*, pages 1627-1630, Sep. 1994.

[6] Sejnowski, T.J., and Rosenberg, C.R., "Parallel networks that learn to pronounce English text". *Complex Systems*, pages 145–168, 1987.

[7] The CMU Pronouncing dictionary, URL: http://www.speech.cs.cmu.edu/cgi-bin/cmudict.

[8] Schmidt, M., Fitt, S., Scott, C., and Jack, M., "Phonetic transcription standards for European Names (ONOMASTICA)". In *EUROSPEECH93*, pages 279-282, Sep. 1993.

[9] Ottesen, G., Horvei, B., and Stensby, S., "Predicting the Pronunciation of Norwegian names by self-learning software". In proceedings of *Onomastica Research Colloquium*, pages 17-24, Dec. 1994.