A HYBRIDE APPROACH FOR GRAPHEME-TO-PHONEME CONVERSION BASED ON A COMBINATION OF PARTIAL STRING MATCHING AND A NEURAL NETWORK

Horst-Udo Hain

SIEMENS AG, Corporate Research, 81730 Munich, Germany
horst-udo.hain@mchp.siemens.de

ABSTRACT
The quality of a text-to-speech (TTS) system heavily depends on the transcription quality of the words to be spoken. Obviously the best transcription can be found in a phonetic dictionary. But for out of vocabulary (OOV) words fall back routines have to be developed.

This paper proposes a fall back routine that combines the correctness of a phonetic dictionary with the flexibility of a neural network. In the first step parts of the OOV word are looked up in the dictionary. They are then connected with the additional feature that the last phoneme of the first part is re-estimated using a neural network and a special phonetic dictionary. In the second step the word stress is determined either from the dictionary or using a second neural network.

1. INTRODUCTION
A common problem for every TTS system is the quality of the phonetic transcription. Errors in phonemes, syllable or stress marks may lead to an unintelligible synthetic speech or in some cases to a misunderstanding of the content.

One possibility for handling OOV words is to learn the grapheme-to-phoneme conversion rules from a phonetic dictionary for a given language. To allow a system to be multi-lingual and easily adaptable, data driven approaches like neural networks are used. But such learning algorithms show problems handling rare cases that do not fit the most likely rules [1]. Therefore it is recommendable to use as much information of the phonetic dictionary as possible.

This paper proposes an architecture for grapheme-to-phoneme conversion that combines the correctness of a phonetic dictionary with the flexibility of a neural network. The conversion is performed in two major steps. The word first is looked up in the phonetic dictionary. If it cannot be found there, a partial string matching algorithm tries to find parts of the word. The transcription of the parts are then connected to get the complete phoneme string with additional feature that the last phoneme of the first part is determined using a special phonetic dictionary and a neural network. Remaining gaps between the parts are also filled by the neural network.

In the second step stress marks are inserted to the phoneme string. If the word or at least the first part of it is found in the dictionary, the stress mark contained in the phoneme string is used without modification. Otherwise, the position of the stress mark is obtained by a second neural network.

2. ALIGNED PHONETIC DICTIONARY
The preparation of the special aligned dictionary is presented in [2] and [3]. This preparation originally is performed for the training pattern generation for the neural network. In this dictionary the graphemes are aligned to their corresponding phonemes. Common dictionaries look like

shadow S { d @U
The aligned dictionary has the form
sh a d ow
S { d @U

It contains the information which phoneme was generated from which grapheme clusters. This is used for the re-estimation of the last phoneme (cf. section 4.1). From this dictionary also a phoneme-grapheme-conversion table can be derived. It contains for a phoneme every grapheme cluster that it can be generated from. It looks like
t$\mathcal{S}$, ch $\mathcal{C}$ z $\mathcal{C}$ tch $\mathcal{C}$$\mathcal{C}$ t che

This table is used for the analysis of the neural network's output to reject impossible phonemes for a given grapheme input (cf. section 3.2).
3. NEURAL NETWORK FOR
GRAPHEME-TO-PHONEME CONVERSION

The original task for this neural network was the determination of the phoneme string for words not contained in the phonetic dictionary. The input is the grapheme string of the word plus some additional information, and the output is the information about the derived phoneme, the grouping and the syllable break. The architecture is shown in figure 1.

![Neural network diagram](image)

Figure 1: neural network for grapheme-to-phoneme conversion

3.1. Input and Output Information

The grapheme input is divided into three parts: the center grapheme, the left and the right grapheme context. The additional input is the output of the last decision of the neural network for this word, and the information whether the previous phoneme was a syllable nucleus or not. This is important for the syllable break output. The output consists of the generated phoneme, the grouping and the decision whether to set a syllable break in front of the phoneme or not. The grouping is the number of graphemes that were used to generate the phoneme. It is used as the step size for the next input. If for instance for the word *shadow* the grouping for center grapheme <s> is 2, then the <h> would be skipped and <a> would be next center grapheme.

3.2. Analysis of the Network Output

In some cases the output node for a phoneme had the highest value that was no possible candidate for the given grapheme input. Therefore the phoneme-grapheme-conversion table described in section 2 is used to check all possible phonemes for that grapheme context. Now the maximum value of the possible phonemes determines the generated phoneme. This procedure significantly increased the result on correct words.

4. GRAPHEME-TO-PHONEME
CONVERSION

To get the transcription of a word it first is looked up in a phonetic dictionary containing the phonemes, syllable breaks and stress marks. If the word cannot be found there, a partial string matching approach tries to find parts of the word with a minimal length (see section 6). If no part was found, then the grapheme-to-phoneme conversion completely is done by the neural network. If at least one part was found, then the following two steps are applied.

4.1. Re-Estimation of the Last Phoneme

In first tests the phoneme strings found in the dictionary were just connected to get the complete phoneme string. This approach was error prone in some cases especially for German. This is due to the different conversion of some graphemes to different phonemes in dependency whether they occur at a syllable start or end.

For example, the phoneme string for the German word *Tag* (day) is

Tag
  t aː k

The <g> is here at the end of the syllable and is therefore converted to phoneme [k]. The plural of this word is

Ta-ge
  t aː ə ː ɡ ə

In this case, the <g> is the beginning of the syllable and is converted to phoneme [ɡ]. To overcome this problem, a new decision about the last phoneme of the first part in dependency of the context is performed.

This decision is made possible because of the aligned phonetic dictionary that was described in section 2. This dictionary contains the mapping between the corresponding graphemes and phonemes. This information is then used to decide which graphemes are necessary for the decision about the last phoneme.

For the re-estimation, the center grapheme for the input is the first grapheme of the grapheme cluster mapped to the last phoneme. The left context is filled with the graphemes preceding this grapheme cluster. For the right context the rest of the grapheme cluster...
and the following graphemes are used. The input nodes for the previous output of the network are filled with the information about the phoneme preceding the last phoneme. The syllable break output is used to set a syllable break in front of the last phoneme or not.

4.2. Filling Remaining Gaps

Gaps at the beginning or the end of the grapheme string or between parts are filled with the same neural network as in the previous step. The left and right contexts are derived from the surrounding graphemes and phonemes. For the last decision, only the grapheme context is allowed that ensures no overlapping with the right part.

5. STRESS MARKS

As mentioned in section 4, the phonetic dictionary contains stress marks. If the word or at least the first part can be found in the dictionary, then the stress mark is used without modification. Otherwise, the position of the stress mark is obtained by the second neural network shown in figure 2.

![Stress Mark Network Diagram]

The input is divided into three parts for every syllable. Node \( v \) contains a value for the vowel, and \( s \) and \( e \) are flags that are set if the syllable starts or ends with a consonant. The number of syllables is language dependent and is derived from the dictionary. The number of output nodes depends on the regarded syllables. The node with the maximum output value determines the stressed syllable.

6. RESULTS

6.1. Word Stress

The results of the neural network for word stress determination are shown in table 1. For every language the context has a length of five syllables. The first line contains the number of unique words in the dictionary. The second line shows the number of words with only one syllable. The third line contains the number of words that have the stressed syllable behind the fifth syllable. For those words, the decision of the network is always wrong. The fourth line contains the remaining words to test. In the following line the amount of correct decisions of the network is shown. The last line contains the total result. This value includes the words that cannot be handled correct because the stressed syllable is out of the scope.

<table>
<thead>
<tr>
<th></th>
<th>German</th>
<th>English</th>
<th>Dutch</th>
</tr>
</thead>
<tbody>
<tr>
<td>words</td>
<td>311360</td>
<td>61223</td>
<td>309668</td>
</tr>
<tr>
<td>single syllable</td>
<td>5250</td>
<td>7985</td>
<td>6125</td>
</tr>
<tr>
<td>out of context</td>
<td>1100</td>
<td>21</td>
<td>1316</td>
</tr>
<tr>
<td>pattern</td>
<td>305010</td>
<td>53217</td>
<td>302227</td>
</tr>
<tr>
<td>correct</td>
<td>90.5 %</td>
<td>88.4 %</td>
<td>84.9 %</td>
</tr>
<tr>
<td>total</td>
<td>90.2 %</td>
<td>88.4 %</td>
<td>84.5 %</td>
</tr>
</tbody>
</table>

Table 1: results for word stress

6.2. Complete Transcription

The results achieved by the neural networks only are compared with the hybrid approach using partial string matching and the networks. For the tests all words of the dictionaries with at least eight graphemes were used. Results for the 274,000 remaining German are shown in table 2.

<table>
<thead>
<tr>
<th>length</th>
<th>total</th>
<th>phonemes and stress</th>
<th>phonemes and breaks</th>
<th>only phonemes</th>
</tr>
</thead>
<tbody>
<tr>
<td>NN</td>
<td>79.9</td>
<td>81.6</td>
<td>88.6</td>
<td>91.0</td>
</tr>
<tr>
<td>3</td>
<td>83.2</td>
<td>85.5</td>
<td>87.8</td>
<td>90.3</td>
</tr>
<tr>
<td>4</td>
<td>87.2</td>
<td>89.5</td>
<td>91.6</td>
<td>94.2</td>
</tr>
<tr>
<td>5</td>
<td>91.1</td>
<td>93.1</td>
<td>94.9</td>
<td>97.2</td>
</tr>
<tr>
<td>6</td>
<td><strong>91.6</strong></td>
<td><strong>93.5</strong></td>
<td><strong>95.3</strong></td>
<td><strong>97.4</strong></td>
</tr>
<tr>
<td>7</td>
<td>91.3</td>
<td>93.0</td>
<td>95.3</td>
<td>97.2</td>
</tr>
<tr>
<td>8</td>
<td>90.3</td>
<td>92.0</td>
<td>94.8</td>
<td>96.7</td>
</tr>
</tbody>
</table>

Table 2: results for German
The first line contains the results achieved with the two networks. Combining the two neural networks 79.9 percent of the words are completely correct (cf. second column). In this cases all phonemes, syllable breaks and stress marks are correct. Without syllable breaks 81.6 percent of the words are correct, and without stress marks 88.6 percent are correct. For 91 percent of the words all phonemes are correct.

The table also shows that the results for the hybrid approach strongly depend on the minimal length of the string. In the beginning the results improve with an increasing length. There are two reasons for this improvement. First, using a short minimal length leads to many possible segmentations, and second, the transcription found for this short grapheme string is not necessarily representative for the grapheme string in this context.

With a minimal length of six graphemes the best results are achieved. In this case the word error rate is decreased by 11.7 percent, and for more than 97 percent of the words at least all phonemes are correct. A further increasing of the minimal length worsens the results. This is because now less and less strings are found in the dictionary and longer grapheme strings have to be converted by the neural network.

Some of the errors in phoneme stress occur because the stress is used without modification if the first part of the word is found in the dictionary. But for some foreign words the stress position moves if a suffix is appended to a word. This is the case for instance for Musik/Musiker or Motor/Motoren:

m u: - z "i": k
m "u: - z I - k 6
m "o: - t o: R
m o: - t "o: - R 0 n

Appending the derivation suffix -er to Musik, the word stress moves from the second to the first syllable. In opposition to that, appending the plural suffix -en to Motor, the word stress moves from the first to the second syllable.

The results for English (37,000 words) and Dutch (266,000) are shown in tables 3 and 4. For English the word error rate can be decreased by 11.7 percent, and for Dutch by 10.6 percent.

7. CONCLUSION

In this paper an architecture for a grapheme-to-phoneme conversion was proposed that combines the correctness of a phonetic dictionary with the flexibility of a neural network. The usage of the dictionary ensures a high quality of the generated phoneme string, and the neural networks enables the system to handle words that are not or not completely contained in the dictionary. The word error rate can be substantially decreased with this approach.

REFERENCES

