

Pronunciation Modeling in Continuous Number Recognition

Péter Mihajlik, Tibor Fegyó, Péter Tatai, and Géza Gordos

Department of Telecommunications and Telematics

Budapest University of Technology and Economics

Pázmány P. sétány 1/D, Budapest, 1117, Hungary

Phone: (36) 1-463 3424 Fax: (36) 1-463 4189 E-mail: {mihajlik, fegyo}@alpha.ttt.bme.hu

Keywords: speech recognition, automatic pronunciation modeling, continuous number recognition

Abstract - In this article we introduce a continuous number recognizer for the Hungarian language. The special problem to be solved is the automatic pronunciation modeling with emphasis on coarticulation at word boundaries, and on alternative utterances. The effects of this approach on the recognition error rates are investigated experimentally.

I. INTRODUCTION

Continuous number recognition is a classical problem in the field of automatic speech recognition. The standard way of building such a recognizer is to construct first the proper phoneme acoustic models for the given language, then to collect the base stems for continuous numbers, and finally to build a grammar network that generates all possible numbers in a given range.

In our approach the base units of recognition are the phonemes, therefore the vocabulary items must be transcribed to phoneme sequences. This task can be solved by hand or more typically, automatically, based on rules or statistical observations. In our case the per-word transcription can be performed easily because of the shortness and the low number of required words. In the following section we introduce a method developed at our Department for the automatic, rule-based derivation of pronunciation networks of Hungarian words.

The second problem addressed is the coarticulation in the continuous number recognition tasks. Phonetic changes at word boundaries (inter-word coarticulation) needs special considerations, because one word can be followed by several alternative words allowed by the recognition network, resulting in different phoneme interactions. In the third section we introduce a method to handle this phenomenon.

II. AUTOMATIC PHONETIC TRANSCRIPTION OF ISOLATED WORDS

To transcribe the vocabulary items to phoneme sequences the following steps are executed:

- The first step is to segment the written form of the given words to letters. It can be solved unambiguously by rules with a few exceptions [1].

- Because in Hungarian the letter to phoneme conversion is unambiguous, the letter sequence can be transcribed to a sequence of phonemes.

- The third step is the phoneme to phoneme transcription. In Hungarian there is a relatively close correspondence between orthography and pronunciation, and the

relevant phonological phenomena can be described quite well by a compact set of rules. Although these rules are thoroughly described by linguists, they are not directly applicable, because they use such morphological and other higher-level information, which are not accessible for us. Therefore, all of our formalized rules below are based only on the phoneme sequence.

The rules are assorted into groups (Table I), and used in the following order:

$$(1,2,3,4) \rightarrow (5,6) \rightarrow 4 \rightarrow (7,8) \rightarrow 4$$

In isolated words these rules describe sufficiently well the pronunciation changes, so the resulted phonotypical transcriptions are, in most cases, correct. The rules may have multiple outputs (not represented in Table I) which means that the result is not always unique, because there might exist alternative pronunciations (e.g., *apátság* \rightarrow a p a: t S a: g or a p a: tS: a: g /abbey/). In most cases only one of them is used generally, but others could be correct as well.

Finally, the phonotypical transcription of a word is converted to a pronunciation network. This step is implemented by a single finite state automaton.

An illustrative example for a pronunciation network is shown in Fig. 1.

TABLE I
GROUPS OF PRONUNCIATION RULES

	Rule group	sample rules (using SAMPA symbols [2])	
1)	Insertion of consonants	i o:	\rightarrow i j o:
2)	Dropping of consonants	z d g	\rightarrow z g
3)	Lengthening of vowels and consonants	i j	\rightarrow i j:
4)	Shortening of vowels and consonants	b: r	\rightarrow b r
5)	Partial assimilation determined by the place of articulation	n b	\rightarrow m b
6)	Partial assimilation determined by voicedness	S d'	\rightarrow Z d'
7)	Full assimilation	l j	\rightarrow j:
8)	Melting of consonants	t s	\rightarrow ts:

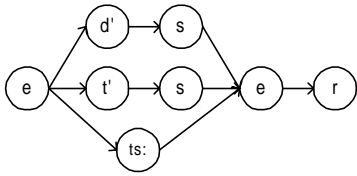


Figure 1: The automatically generated pronunciation network of the Hungarian word *egyszer* (once) with SAMPA symbols.

Summarized, the described method is applicable to the transcription of the written form of any Hungarian word to a pronunciation network.

III. INTER-WORD PHONETIC CHANGES AT CONTINUOUS NUMBER RECOGNITION

In the previous section we introduced a method to transcribe the orthographic form of isolated words to their pronunciation models. The difficulty in continuous number recognition – in contrary to the isolated task – is the fact that a word can be followed by alternative words, the continuation depends on the grammar network. Therefore, the previous method obviously cannot be applied at word boundaries. The problem can be treated only at a higher, i.e., grammar network level.

Our solution is the following: first the vocabulary transcriptions are generated with the algorithm described in the previous section. Then these transcriptions are substituted into the word-level grammar network resulting in a phoneme-level grammar network. Finally, inter-word phonetic changes, based on the same rules (Table I), are modeled and inserted as sub graphs of pronunciation alternatives into the network. Currently this step is partially manual, but we are working to fully automatize it.

There are other approaches to handle the inter-word phonetic changes, too, such as multiple transcriptions per word or simply ignoring the changes. In the following we introduce them and their effect on the recognition rates compared to our solution.

A. Basic network

In Hungarian, similarly to other languages, the number recognition network is constructed by connecting basic stems to each other according to the grammar rules. In our case the network contains about 200 (word-level) arcs, each one is matched to a stem, so the network can generate the numbers from 0 to 1 million. The network accepts only the morphemically correct forms. The phonetic changes at the stem boundaries, however, are not modeled, so it is called *basic network* (illustrated in Figure 2).

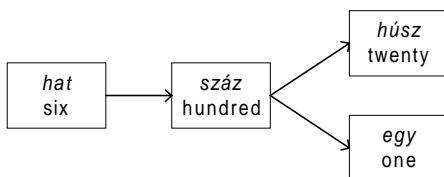


Figure 2: Part of the basic network

B. Extended vocabulary network

In the basic network every stem – which can either be a word or a morpheme – has only one phonetic transcription, in our concept it is the most probable (frequent) one. But similarly to the method commonly used for isolated recognition, the basic network can be extended by adding alternate parallel transcriptions to each stem. Although the word-level network remains unchanged it may still contain not only intra-word, but also cross-word alternative pronunciations by simply enumerating the possible pronunciation forms.

Actually, each (word-level) arc in the basic network is transformed to a branch of arcs (representing the parallel pronunciation variants), but considering this approach on a higher – more linguistic – level, we call it extended vocabulary network.

There are, however, disadvantages of this method. Firstly, at word boundaries all pronunciation alternatives must be listed in the extended vocabulary because the following word is unknown. This means that the network will generate many inaccurate forms like “ s a : s e d ’ ”, which may decrease the efficiency of the recognition. (In Figure 3 the dashed lines represent the incorrect links.)

The other drawback of this method is, that it wastes memory, because it stores slightly different pronunciation representations many times. Clearly, storing only the differences would be enough. But the growth of the recognition graph has an other, more severe consequence. Calculating the network-expansion factor, it is proportional with the average of the product of the word length and the number of alternative transcriptions per word, which is in our case about 4. As the computational load is roughly proportional with the recognition graph’s size, it grows also by this factor. So, this approach is not feasible for relatively large vocabulary tasks.

C. Extended Network

The root of the problem with the extended vocabulary network is that it handles the pronunciation alternatives on the word (or morpheme) level. The obvious solution would be to handle them on the phoneme level. At the beginning of the section we described the generation of this phoneme-level extended network, in the following it will simply be referred to as extended network (illustration in Figure 4.).

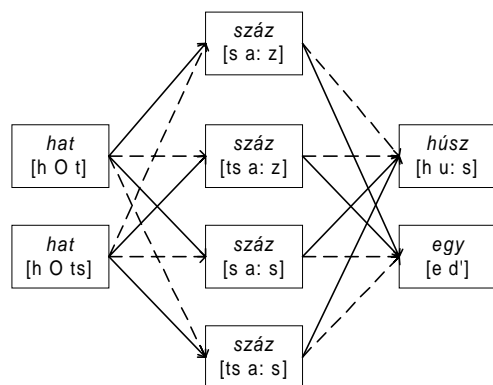


Figure 3: Part of the basic network using the extended vocabulary

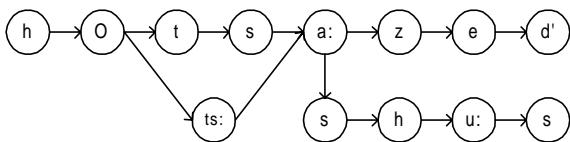


Figure 4: Part of the optimized network

The main advantage of the above solution is that it can model all types of phonetic changes with minimum network size. Although the automatic generation of this extended network from the basic network is theoretically possible on the basis of the phonetic rules, but it is a complex task. Therefore, for the experiments we used semiautomatic methods. As the results show it would be worth to fully automatize the process, and we are working on it.

IV. EXPERIMENTAL RESULTS

A set of experiments was carried out in order to assess the relevance of the modeling of the phonetic changes at word boundaries, and of the use of pronunciation variants in continuous number recognition. As a basis for comparison, isolated number recognition tests were performed.

A. Model Training

Currently BABEL is the largest high quality Hungarian speech database available for research purposes [3]. It consists of three different parts: isolated digit and continuous number utterances, consonant-vowel-consonant syllables, and continuously read speech. A fraction of the database is segmented and labeled by hand at the phoneme level. We had access to the labeling information for five speakers' voices, 400 seconds of speech in total. The number of speakers available is 20 (10 men and 10 women), and there are altogether about 900 sentences and 9700 continuous number strings in the database. The voice of 14 speakers composed the training set, and the rest of the data were used in the recognition tests. In the experiments the numbers and the paragraphs were used separately for training, resulting in 2 different acoustic model sets. For parametrization we used 10 MFCC + log energy with delta and acceleration coefficients altogether 33 components [4].

Because only a small fraction of the database was segmented at the phoneme level, the model training was carried out in two steps. In the first step initial models were Viterbi-trained using the available data, and the rest of the database was labeled automatically by forced alignment with the FlexiScribe tool[1]. In the second step the entire training set was used for training with the labels generated previously.

The models were context independent left to right ones with three states per phoneme. A 10-mixture diagonal covariance matrix Gaussian distribution has been used in all states. From among the 64 phonemes of Hungarian, only the vowels and the short consonants were used [1].

B. Isolated Number Recognition

During the isolated number recognition tests all 140 numbers occurring in the test database were listed in the

vocabulary. The numbers were transcribed to phoneme sequences automatically. In the experiment the effect of the presence or the absence of pronunciation alternatives were investigated (Table II). In the first case the most frequent pronunciation was used while in the second case all alternatives were listed in the vocabulary.

The error rate decreased slightly for both acoustic model sets confirming the importance of pronunciation alternatives. Moreover, in most of the cases only one item in the number was misrecognized, so the digit error rate is even lower.

C. Continuous Number Recognition

In the second group of experiments the vocabulary contained about 50 number stems (e.g., *egy* (1), *két* (2), *kettő*(2)... , *tíz* (10), *tíz*(10) ...), and a network generated all possible numbers. The three different networks described in section III were investigated. The basic network generated the numbers from 0 to 999,999 and only intra-word phonetic changes were considered. In the extended vocabulary the base morphemes were the same, but all possible pronunciation alternatives were listed that occur in this task. In the last case the network was phoneme based, and it allowed much less improper pronunciation possibilities. The experiments were carried out using both training sets (Table III).

The continuous recognition mode means that compound numbers were continuously read with short pauses between them and no explicit word boundary detection was performed.

TABLE II

Isolated number recognition error rates using two different pronunciation models. Acoustic models were trained by numbers (a) and by general speech (b).

(a)		
Vocabulary representation	Error rate	Relative improvement
Most frequently used form	0.48%	6.3%
Pronunciation alternatives	0.45%	
(b)		
Vocabulary representation	Error rate	Relative improvement
Most frequently used form	2.69%	4.1%
Pronunciation alternatives	2.58%	

TABLE III

Continuous number recognition error rates using various pronunciation models. Acoustic models were trained by numbers (a) by general speech (b).

(a)		
Network type	Digit string error rate	Relative improvement
Basic network	5.86%	–
Extended vocabulary	3.96%	33%
Extended network	3.48%	41%
(b)		
Network type	Digit string error rate	Relative improvement
Basic network	16.10%	–
Extended vocabulary	13.45%	16%
Extended network	13.29%	17%

If cross-word changes were not modeled, the error rate became the highest, but modeling the phonetic changes improved the system. With a brute force solution (extended vocabulary) the relative improvement was lower, while with the optimal network, a higher improvement could be achieved. The results indicated that the best acoustic models resulted from using the training set containing only numbers. The highest relative change between the basic and the revised pronunciation models was achieved using the best acoustic model, but the improvement is significant in the other cases, too.

The error rates are larger even in the best case than they were in the isolated number tests, but it can be noted that the effective vocabulary size grew from about 200 up to more than 2 000 000, and the word boundaries were unknown at the continuous recognition.

V. CONCLUSION

In this article we presented a rule-based method, which is capable to produce automatically the various pronunciation forms of Hungarian words. Also, we introduced cross-word pronunciation modeling techniques and applied them for the continuous number recognition task. In the first case a basic vocabulary was extended to represent all the pronunciation alternatives of the morphemes. In the second, more elaborated modeling technique, a phoneme-level network was generated which provided an optimal solution.

Isolated and continuous number recognition test results were presented. Only the introduction of pronunciation alternatives did not affect significantly the recognition in the isolated task. But the application of the phonological rules at morpheme boundaries resulted in a substantial improvement in the continuous recognition tasks. A noticeable phenomenon was that the higher the baseline error rate was, the smaller the relative improvement caused by the pronunciation modeling became, which means that the quality of the acoustic model influences the effectiveness of the pronunciation modeling.

REFERENCES

- [1] M. Szarvas, T. Fegyó, P. Mihajlik, P. Tatai, "Automatic Recognition of Hungarian: Theory and Practice", *International Journal of Speech Technology*, vol. 3, pp. 237-251, 2000.
- [2] SAMPA—Computer Readable Phonetic Alphabet. Available: www.phon.ucl.ac.uk/home/sampa, 1996.
- [3] K. Vicsi and A. Vig, Babel—A multi-lingual database, Technical report, György Békésy Acoustics Research Laboratory of the Budapest University of Technology and Economics, 1997.
- [4] Steve Young, et al. *The HTK Book*, Version 3.0. Available: <http://htk.eng.cam.ac.uk/>, 2000.