

論文 / 著書情報
Article / Book Information

Title	Finite-State Transducer Based Hungarian LVCSR with Explicit Modeling of Phonological Changes
Author	Mate Szarvas, Sadaoki Furui
Journal/Book name	7th International Conference on Spoken Language Processing (ICSLP-2002), Vol. , No. , pp. 1297-1300
発行日 / Issue date	2002, 9

FINITE-STATE TRANSDUCER BASED HUNGARIAN LVCSR WITH EXPLICIT MODELING OF PHONOLOGICAL CHANGES

Máté Szarvas Sadaoki Furui
mate@mateweb.net furui@furui.cs.titech.ac.jp

Department of Computer Science
Tokyo Institute of Technology
2-12-1, Ookayama, Meguro-ku, Tokyo, 152-8552 Japan

ABSTRACT

This article describes the design and the experimental evaluation of the first Hungarian large vocabulary continuous speech recognition (LVCSR) system. The architecture of the recognition system is based on the recently proposed weighted finite state transducer (WFST) paradigm. The task domain is the recognition of fluently read sentences selected from a major daily newspaper. Recognition performance is evaluated using both monophone and triphone gender independent acoustic models. The vocabulary units used in the system are morpheme based in order to provide sufficient coverage of the large number of word-forms resulting from affixation and compounding in Hungarian. The language model is a statistical morpheme bigram model. Besides the basic list style pronunciation dictionary model we evaluate a novel phonology modeling component that describes the phonological changes prevalent in fluent Hungarian. Thanks to the flexible transducer-based architecture of the system the phonological component is integrated seamlessly with the basic modules with no need to modify the decoder itself. The proposed phonological model decreases the error rate by 8.32% relatively compared to the baseline triphone system. The morpheme error rate of the best configuration is 17.74% in a 1200 morpheme task with test set perplexity 70.

1. INTRODUCTION

Hungarian is a Finno-Ugric language spoken by about 15 million people mainly in Hungary and in the neighbouring countries. There are 64 phonemes (14 vowels and 50 consonants) in Hungarian that can be divided into two groups of short/long pairs (length is a phonemically distinguishing feature both in the case of vowels and consonants). Similarly to the other members of the Finno-Ugric language family Hungarian is an agglutinating language, that is, it relies heavily on suffixes. Hungarian is using the Latin alphabet and the written and spoken forms of words have a relatively close correspondance. In most cases, the words are spoken as written but the consonant combinations that would be difficult to pronounce constitute an exception to this rule.

Speech research has a long tradition in Hungary [3] and there exist several research and commercial systems both for speech synthesis and automatic speech recognition (ASR). Previous ASR research efforts have been limited, however, to command and control tasks that have a limited vocabulary. Besides the shortage of resources the main obstacle that delayed the beginning of Hungarian LVCSR research is the size of the vocabulary and the complexity of the morphology. The number of different word forms is in the range of hundreds of millions according to an estimate by the authors of the best Hungarian spell-checking software [7] and the accurate modeling of this vocabulary is not easy even with morphological decomposition because the number of inflection classes is very large due to historic reasons. The other difficulty from an ASR point of view is the accurate computational representation of pronunciation. It is true that the spelling system is

phonemic and the written and spoken forms are strongly related but the pronunciation of most words starting or ending with a consonant depends on the adjacent words because difficult consonant combinations are replaced by simpler ones by a hierarchy of phonological rules.

In our previous work [8, 9] we proposed methods for treating both of these obstacles but these methods could not be evaluated that time due to the lack of suitable databases and the lack of an implementation. In Section 2 of this article we describe the architecture of our new weighted finite state transducer based recognition system that was designed to facilitate an efficient implementation of both our phonology and morphology modeling methods. Then we give an overview of the acoustic and language modeling components including a description of our recently collected speech database and the language model database. The details of our pronunciation and phonology modeling method are explained in Section 4 while the results of the experimental evaluation of the system are described in Section 5. Finally, we conclude our work in Section 6 with a summary and plans for future work.

2. SYSTEM OVERVIEW

The standard knowledge components in a state-of-the-art ASR system are the acoustic model, the pronunciation model and the language model. The usual practice is to represent each of these different types of knowledge in their specialized data structure and to use dedicated code in the decoder for combining and searching them. This practice has been motivated by the need for very efficient implementations and possibly also by the incremental development of the recognition systems.

The price of this highly optimized implementation is, however, the loss of flexibility for adding new knowledge sources to the system. The reason is that the specialized code gets increasingly complex and usually only the original developer of the decoder module would be able to add the new components. Even though it has been widely understood for a long while that all the usual knowledge sources (KS) are just different instantiations of the same basic mathematical data structure it has only recently been demonstrated [6] that a recognition system using a flat data representation and generic algorithms for all KSs can achieve, with affordable system resources, a performance similar to specialized systems.

This weighted finite-state transducer (WFST) based architecture [5, 6] is especially attractive for us because all the phonological and morphological dependencies described in our previous work [8, 9] can be easily converted into a WFST representation. Moreover, we believe that higher level linguistic dependencies, such as the agreement of the number and person of the subject and the predicate, can also be straightforwardly represented in this framework. Therefore we designed our recognition system from the beginning according to the flat-data WFST paradigm.

2.1. Review of WFST-based ASR

The main idea of WFST-based speech recognition [5, 6] is that each of the knowledge sources is represented as a weighted finite-state transducer. The search-space of the given task is obtained by combining the basic components using the composition operator of WFSTs. The main components of our current system besides the decoder itself are the acoustic models A , the context dependency mapping CD , the phonological rules P , the basic pronunciation dictionary D and the stochastic (bigram) language model LM . The planned sixth knowledge source in the system is the morphosyntactical rule set M outlined in [8, 9] but it is not yet implemented.

The recognition task is then defined as finding the path with the highest likelihood in the integrated recognition network

$$\text{ASR} = \text{best_path } A \circ CD \circ P \circ D \circ M \circ LM \quad (1)$$

where $\circ$ represents transducer composition.

3. ACOUSTIC AND LANGUAGE MODELING

3.1. Acoustic modeling

Training Database. The acoustic training database contains 10 hours of read speech from 15 male and 15 female speakers. However, currently only 2 hours (uniformly distributed over all speakers) can be used for training because the rest is not yet transcribed accurately. The contents of the training database is contemporary juvenile literature because its simple language helped to reduce the number of speaking errors and restarts. The speech was recorded with a portable DAT recorder using a Sony C 355 condenser microphone in the home of the speakers with no special sound-treatment of the recording environment. The sampling rate of the database is 44.1 kHz but it was band-limited to 8 kHz before parameterization.

Signal processing. The signal processing front-end computes a sequence of vectors that capture the spectral characteristics of the speech over a small window, or “frame,” of speech. Every 10 ms, a vector of 12 mel-warped cepstral coefficients and log-energy measure is computed using a 30 ms Hamming windowed speech segment resulting in 13 static spectral parameters.

Besides these “static” coefficients we also use the first two spectral derivatives, the so-called delta and delta-delta coefficients. These coefficients are calculated using the regression method with ± 2 frames of data [2].

Finally, we apply cepstral mean subtraction (CMS) in order to compensate for speaker and channel variations. CMS is applied on the utterance level; that is, after an utterance is parameterized, the mean vector is determined and subtracted from the parameter vector of each frame. Due to software limitations, both the “silence” and the “active” frames are used for computing the cepstral mean.

Phoneme models. Our system is using gender independent hidden Markov models (HMMs) for acoustic modeling. Each phoneme model has 3 states with 1 self-transition and 1 forward transition to the following state. There is also 1 single-state silence model that can be optionally inserted after any recognition unit. The observation likelihoods are modeled by a diagonal covariance matrix Gaussian mixture density function. It is possible to use both monophone and shared-state cross-word triphone models. In the case of monophone models the best error rate was achieved with 20 mixtures while in the case of triphone models the 9-mixture model gave the best error rate. The long and short consonants were modeled by the same models because they differ only in duration but

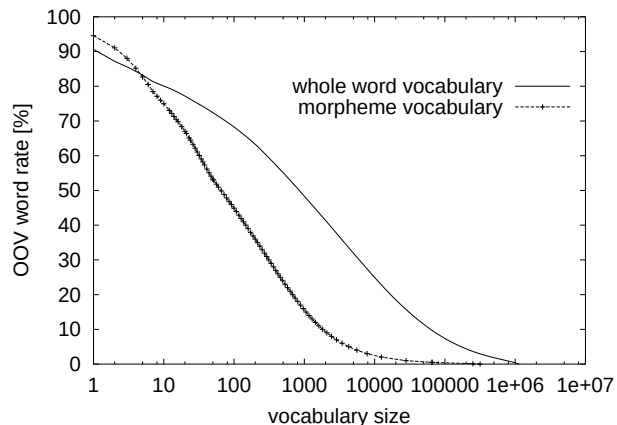


Figure 1. Out of vocabulary word rate as a function of vocabulary size when using whole words units and morpheme units. (The preprocessing for this figure was different from that for Table 1.)

duration is not well modeled by the current HMM topology. The number of different states is 98 in the monophone case and 1237 in the triphone case.

3.2. Language modeling

As explained in the introduction, one of the difficulties in building a Hungarian LVCSR system is the modeling of the large vocabulary. For example, our language model (LM) database of 40 million words contains over 2 million different tokens before preprocessing. The number of different tokens remains over 1 million even after replacing all the number and punctuation characters with a white-space and converting all upper case characters to lower case. Therefore it is essential to use units smaller than words as the basic recognition unit in order to cover the vocabulary using system resources within practical limits. Similarly to other languages, the most convenient linguistically motivated such unit is the morpheme for Hungarian as well.

Training database and coverage. We used 40 months of text of a large Hungarian daily newspaper (“Magyar Hírlap” between September 1. 1994 and December 31. 1997) as the LM development data. There are 38.9 million white space separated tokens in the unprocessed database and the whole data size is 300 MByte. After normalizing the database by removing punctuation characters and splitting words into their constituent morphemes the total number of morpheme tokens is 74.1 million. We removed all digit characters and converted all upper-case characters to lower-case for the coverage tests that we conducted in order to assess the effectiveness of morpheme analysis in reducing the number of token types. The results of these tests are displayed in Figure 1. It is clear from the figure that the analysis significantly decreased the number of units necessary for a given coverage. For example to attain a coverage of 99% (OOV=1%) we need only 28k morpheme units while the number of necessary word units would be over 750k. We note that the tail of the curve for morpheme coverage was generated by words that our analyzer could not split, therefore the theoretical coverage would be much better for morpheme vocabulary sizes over 30k.

N-gram models and perplexities. The other important characteristic of a language model is its test set perplexity. The

Table 1. Language model statistics for different vocabulary sizes (morpheme units).

Vocabulary size	OOV [%]	PP 2GR	PP 3GR
1k	25.0	74.2	44.7
5k	16.3	85.6	49.1
10k	15.1	89.5	51.2
20k	14.6	97.9	52.9
65k	14.3	95.1	54.5

perplexity of 2- and 3-gram models for different vocabularies is displayed in Table 1. It is clear from the table that 3-gram models have a significantly smaller perplexity and their use is mandatory for high-accuracy recognition. The out of vocabulary word (OOV) rates displayed in the same table are larger than those in Figure 1 because the number tokens were not removed and the capital letters were also retained.

4. PRONUNCIATION MODELING

The mapping between the spoken and written form of words is represented in pronunciation dictionaries in ASR systems. For some languages the dictionary is compiled manually, while for some others it is possible to generate it automatically. We chose the latter alternative for developing the pronunciation model of our system because spelling is rather close to pronunciation and for this same reason there are no pronunciation dictionaries available. The two main steps in determining the pronunciation(s) of a word from its written form are to first determine the phoneme sequence corresponding to the character sequence and then to apply some phonological rules that change some phonemes when they occur next to some others.

4.1. Phoneme tokenization

In Hungarian each phoneme has its own written form. Most phonemes are denoted by a single character, such as *a, á, c, d, s, t, z*. Some phonemes (e.g., *cs, sz, zs*), however, are denoted in writing by multi-character combinations and splitting a word into phonemes becomes ambiguous when two such phonemes come next to each other within a word. For example, the word *százszor* (hundred times) could be split into phonemes either as *sz á z sz o r* or as *sz á z s z o r* or as *sz á z s z o r*, but only the first one corresponds to a meaningful morpheme combination (*száz+szor*).

We are using a non-deterministic WFST tokenizer for getting the phoneme sequence corresponding to a character sequence. The tokenizer has parallel arcs with character sequences on the input side and the corresponding phoneme sequences on the output side. Each arc has the same weight and the optimal segmentation to phonemes is defined by the shortest path in the graph of the composition of the input sequence and the tokenizer. This ensures that the smallest number of arcs would be used, that is, arcs covering a larger portion of the input sequence are chosen when there is an ambiguity. The word in the previous example will be properly tokenized if we have, besides the arc for each phoneme, the following arc in the tokenizer: *s z o r : sz o r*, where the colon is separating the input and the output sides.

4.2. Phonology modeling

The actual pronunciation of the words is, however, modified by phonological changes. These changes are natural and automatic for the speakers but modeling them in an ASR system is not straightforward because the pronunciation (real-

ized phoneme sequence) of the words *depends* on the adjacent words. This dependence is different from allophonic variation because allophonic alteration leaves the phonemic identity of the sound intact. For example, the first sound of both “cup” and “kiss” is /k/ even though they sound rather different. Phonological alterations, in contrast, make more marked changes that result in the change of the phonemic identity of phonemes.

But this phenomenon cannot be modeled by simply registering each vocabulary unit with multiple pronunciations because the changes can occur only in certain contexts. In other words, one pronunciation is valid only in one context while the others are valid in other contexts.

Context dependent rewrite rules. Phonological variations can be described by a set of compact rules for most languages. A popular technique in the linguistic literature for describing phonological regularities is the use of context dependent rewrite rules [4] of the form

$$a \rightarrow b/c_d \quad (2)$$

The interpretation of such a rule is as follows. Each of the expressions *a, b, c* and *d* denote a regular expression. The meaning of the rule is that anything that matches *a* is to be replaced by *b* whenever it is preceded by *c* and followed by *d*.

The following example illustrates a rule for modeling voice harmony: $t \rightarrow d / _ \text{VOICED}$, that is, the phoneme *t* is replaced by *d* whenever it is followed by a voiced consonant.

Such rewrite rules are easy to apply to a linear string of phoneme symbols to determine the pronunciation of any particular word sequence, but it is not straightforward to apply them to a recognition network because different rules need to be activated depending on the successor word.

Implementing rewrite rules by transducers. It has been shown that these rewrite rules can be converted into finite state transducers [4]. And once they are in a transducer representation we can integrate them into our recognition network by the composition operation.

Let $P_1, P_2, \dots, P_k$ denote the specific phonological rules of the form in Eq. (2) and let $P = P_1 \circ P_2 \circ \dots \circ P_k$ represent the transducer for the sequential application of all of the simple rules. The composition of the phonological transducer P with the dictionary transducer D will result in a phoneme to word transducer that is consistent with the phonological rules of the language.

5. EXPERIMENTAL EVALUATION

Testing database. The testing database contains 600 sentences read by 20 speakers (12 male, 8 female), all different from those in the training database. The recording conditions were similar to that of the training database. The contents of the testing speech database was designed to permit the development of a good language model (LM) and the possibility to conduct recognition tests of varying difficulty. All the sentences were selected from the newspaper that was used for developing the language model (but there was no overlap between the LM training data and the test sentences). The test sentences were selected so that none of them was longer than 100 characters in order to exclude difficult to read sentences. Each speaker read 30 sentences of gradually increasing complexity. The simplest sentence of any speaker can be covered with under 1000 vocabulary units while the most complex sentences require a vocabulary of 20k units.

Phoneme recognition. First we conducted phoneme recognition experiments in order to assess the quality of our acoustic

Table 2. Phoneme error rates using monophone and triphone models.

	Monophone models	Triphone models
Error rate	42.24%	39.13%

models. Short and long consonants were considered equivalent during these tests because the models have no duration modeling component. Short and long vowel pairs, however, differ not only in length but also spectrally, therefore all vowels were treated as different. There was also a silence model in the vocabulary besides the 14 vowels and 25 consonants, therefore the vocabulary size was 40 for this task. The recognition grammar was a zero-gram (all phonemes equally likely) loop of all models with an optimized insertion penalty. Proper treatment of context dependency was not ensured in the case of triphone models, that is, any triphone model could follow any other.

The error rates (defined as $100 \frac{S+D+I}{N}\%$, where S , D and I denotes the number of substitutions, deletions and insertions and N denotes the total number of phonemes in the test-set) are displayed in Table 2 for both the monophone and triphone case. The use of triphone models instead of monophone models reduced the error rate only by 7% relatively, suggesting that our triphone models are not yet optimized entirely.

Continuous word recognition. In the second set of experiments we used our recognition system for transcribing the easiest subset of our testing database. The morpheme vocabulary size was 1200 and the test set perplexity of the bigram language model (LM) was 70. The size of the vocabulary was limited by computational reasons because the application of the phonological rules increased the size of the precompiled recognition network considerably. The use of a trigram LM was prevented by the same reason. The recognition error rates (defined similarly to the phoneme recognition case) under 4 different conditions are displayed in Table 3. The number of Gaussian mixtures and the language model scaling factor and word insertion penalty were optimized separately for each of the 4 conditions. Moreover, the phoneme transcription of the acoustic model training data was also adjusted to the given conditions. For example, phonological changes were modeled during the conversion of the word transcription to a phoneme sequence only if the recognition network was also built modeling these changes but not for the other case.

In these tests the use of triphone models improved the error rate more significantly (by 27.6% relatively) than in the phoneme recognition tests. Moreover, the explicit modeling of phonological changes within words and across word boundaries decreased the error rate by another 8.3% relatively when triphone acoustic models were used. However, somewhat surprisingly, explicit phonology modeling did not decrease the error rate at all in the case of monophone models. This is in contrast with the findings in [1] where a significant improvement was reported even in the case of monophone models. We attribute this difference to the fact that [1] used a phonologically modified transcription for both testing conditions, while in our case the acoustic models were optimized for each of the test conditions separately. The results are, however, consistent with those of [1] in that the improvement due to phonology modeling is more marked when the baseline acoustic models are of higher quality.

6. CONCLUSION AND FUTURE WORK

In this paper we introduced our weighted finite-state transducer based Hungarian LVCSR system. The system is using morpheme units instead of whole words in order to ensure

Table 3. Comparison of speaker independent morpheme error rates with and without phonology modeling.

	No phonology	With phonology
Monophone	26.73%	26.73%
Triphone	19.35%	17.74%

a good coverage of the huge vocabulary resulting from the rich morphology of Hungarian. Besides describing the design of the whole system and of our databases we introduced a completely automatic pronunciation modeling method that is capable of modeling phonological laws both within and across word boundaries. Thanks to the flexible finite state transducer based design of the system, the phonological module could be integrated into the system with no need for modifying the decoder itself. The use of the phonological rules could reduce the baseline morpheme error rate in the triphone system by 8.3% in accordance with theoretical expectations and earlier experimental results in a simpler system [1]. The exact reason for the inability of the phonological rules to decrease the error rate in the monophone case is still unanswered but taking into account the findings in [1] it is likely due to the inferior quality of the monophone models.

Future improvement of the phonology modeling method could be expected from estimating likelihoods for the rules using an algorithm similar to the one proposed in [10]. The acoustic modeling component could be improved by modeling phoneme duration explicitly because the 25 long consonants differ exclusively in duration from their short counterpart and without duration modeling the current system is unable to distinguish these pairs. Finally, we expect improvement of the language model by implementing the stochastic morphosyntactic model that we proposed in our previous work [8, 9].

7. REFERENCES

- [1] T. Fegyő, P. Mihajlik, P. Tatai, G. Gordos. Pronunciation Modeling in Continuous Number Recognition. *Proc. Eurospeech 2001.*, Vol. 3., pp. 1465–1468.
- [2] S. Furui. Speaker-independent isolated word recognition using dynamic features of speech spectrum. *IEEE Trans. ASSP* ASSP-34, 52–59, 1986.
- [3] M. Gősy. On the Early History of Hungarian Speech Research. *International Journal of Speech Technology*, 3(3/4):155–164, 2000.
- [4] R. Kaplan, M. Kay. Regular Models of Phonological Rule Systems. *Computational Linguistics*, 20(3):331–379, 1994.
- [5] M. Mohri. Finite State Transducers in Language and Speech Processing. *Computational Linguistics*, 23:2, 1997.
- [6] M. Mohri, F. Pereira, M. Riley. Weighted Finite-State Transducers in Speech Recognition. In *Proc. ISCA Automatic Speech Recognition 2000.*, pp. 97–106.
- [7] G. Prőszék, B. Kis. Computer Interaction—in a Human Language [in Hungarian]. Bicske, Szak kiadó. 1999.
- [8] M. Szarvas, T. Fegyő, P. Mihajlik, P. Tatai. Automatic Recognition of Hungarian: Theory and Practice. *International Journal of Speech Technology*, 3(3/4):237–251, 2000.
- [9] M. Szarvas, S. Furui. The use of finite-state transducers for modeling phonological and morphological constraints in automatic speech recognition. *Proc. Autumn Meeting of the Acoustical Society of Japan.*, 2-1-20, pp. 87–88, 2001.
- [10] H. Tsukada. Automatic Learning of Weighted Finite-State Transducers based on Context Trees. In *Proc. Workshop on Multi-Lingual Speech Communication.*, Kyoto, 2000.