Large Vocabulary Continuous Speech Recognition of an Inflected Language using Stems and Endings

Tomaž Rotovnik, Mirjam Sepesy Maučec, Zdravko Kačič

▶ **To cite this version:**

Tomaž Rotovnik, Mirjam Sepesy Maučec, Zdravko Kačič. Large Vocabulary Continuous Speech Recognition of an Inflected Language using Stems and Endings. Speech Communication, 2007, 49 (6), pp.437. 10.1016/j.specom.2007.02.010 . hal-00499182

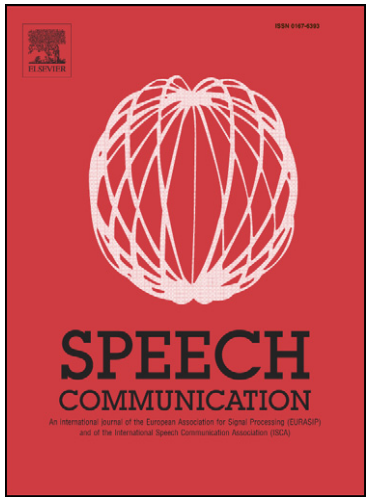
HAL Id: hal-00499182
https://hal.science/hal-00499182

Submitted on 9 Jul 2010

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Accepted Manuscript



Large Vocabulary Continuous Speech Recognition of an Inflected Language using Stems and Endings

Tomaž Rotovnik, Mirjam Sepesy Maučec, Zdravko Kačič

PII: S0167-6393(07)00042-8
DOI: 10.1016/j.specom.2007.02.010
Reference: SPECOM 1625

To appear in: *Speech Communication*

Received Date: 22 December 2005
Revised Date: 14 February 2007
Accepted Date: 19 February 2007

Please cite this article as: Rotovnik, T., Maučec, M.S., Kačič, Z., Large Vocabulary Continuous Speech Recognition of an Inflected Language using Stems and Endings, *Speech Communication* (2007), doi: 10.1016/j.specom.2007.02.010

This is a PDF file of an unedited manuscript that has been accepted for publication. As a service to our customers we are providing this early version of the manuscript. The manuscript will undergo copyediting, typesetting, and review of the resulting proof before it is published in its final form. Please note that during the production process errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.

ACCEPTED MANUSCRIPT

Title:
Large Vocabulary Continuous Speech Recognition of an Inflected Language using Stems and Endings

Authors:
Tomaž Rotovnik
Mirjam Sepesy Maučec
Zdravko Kačič

Affiliation:
Faculty of Electrical Engineering and Computer Science,
University of Maribor,
Smetanova 17,
2000 Maribor,
Slovenia

Corresponding author:
Tomaž Rotovnik
tomaz.rotovnik@uni-mb-si
tel: +386 2 220 7229
fax: +386 2 251 1178

Abstract:
In this article, we focus on creating a large vocabulary speech recognition system for the Slovenian language. Currently, state-of-the-art recognition systems are able to use vocabularies with sizes of 20,000 to 100,000 words. These systems have mostly been developed for English, which belongs to a group of uninflectional languages. Slovenian, as a Slavic language, belongs to a group of inflectional languages. Its rich morphology presents a major problem in large vocabulary speech recognition. Compared to English, the Slovenian language requires a vocabulary approx. ten times greater for the same degree of text coverage. Consequently, the difference in vocabulary size causes a high degree of OOV (out-of-vocabulary words). Therefore OOV words have a direct impact on recognizer efficiency. The characteristics of inflectional languages have been considered when developing a new search algorithm with a method for restricting the correct order of sub-word units, and to use separate language models based on sub-words. This search algorithm combines the properties of sub-word-based models (reduced OOV) and word-based models (the length of context). The algorithm also enables better search-space limitation for sub word models. Using sub-word models, we increase recognizer accuracy and achieve a comparable search space to that of a standard word-based recognizer. Our methods were evaluated in experiments on a SNABI speech database.

Keywords:
large vocabulary continuous speech recognition, sub word modeling, search algorithm, stem, ending

ACCEPTED MANUSCRIPT

ACCEPTED MANUSCRIPT

Main text:

1. Introduction

The natural development of language has caused its variability and ambiguity. The result is about 6 000 known languages today. They differ to a great extend in word formation rules. Homonym disambiguation is another big challenge today. From the speech recognition point of view, it would be logical to classify all languages by their common sources, the results of which could also be seen in various multilingual recognition experiments. The reason for such presumptions is the fact that similar languages share the same or, at least, similar grammatical and phonetical attributes as their sources. In general, development in speech recognition is moving towards the customisation of recognizers for use with different language groups. This article refers to large-vocabulary speech recognition with emphasis on inflectional languages, among which is the Slovenian language. Its rich morphology, therefore, represents a major problem in large vocabulary speech recognition, which is reflected in a high degree of OOV words and a much more varied word order, in comparison to the English language. A common word order in English would be SVO (subject, verb, object) structure, whereas this is not the case in Slavic languages; the only exceptions being Macedonian and Bulgarian. However, freer word order reduces the efficiency of the statistical language modeling commonly used in large vocabulary speech recognition. Besides the Upper Sorbian language, Slovenian is the only language to include additional word forms for dual, which causes an even greater number of words. Another feature of the Slovenian language is the category of verbal aspect and palatalization, where the next sound causes a change in the previous sound, creating even more new word forms.
The afore-mentioned features of inflectional languages prevent straightforward usage of the state of the art recognition systems technology developed for English.

This article is divided into six chapters. Sub-word modeling is covered in the following chapter. The main advantage of sub-word-based models compared to word-based is a smaller OOV ratio. The core part of this chapter describes the morphological structure of the Slovenian language. The third chapter discusses those problems encountered in the recognition of different sub-word units. A mathematical formula is presented for recognition process with word and sub-word units. The definition of a novel search algorithm follows, for the recognition of inflectional languages. We treat it from the point of acoustical and language modeling. We also present different improvements in the proposed search algorithm. The fourth chapter describes the experimental system setup and captures speech recognition results with different search algorithms and with different vocabulary units. Recognition error, recognition speed, and the size of search space, expressed as the average number of active instances are compared. Discussion on improvements in recognition results are presented in chapter five when using a new search algorithm with extended context in speech recognition of the Slovenian language. The last chapter provides a summary of the presented works, achievements, and ideas for future work.

2. Sub-word Modeling

2.1 Current review of inflectional languages in the field of speech recognition

An extensive vocabulary presents the major problem in large vocabulary speech recognition of an inflectional language. Restricting the size of vocabulary to satisfy memory and speed requirements can

ACCEPTED MANUSCRIPT

cause additional recognition error. The solution, in the case of recognizing the speech of inflectional (Slavic), tonal (Chinese) and agglutinative (Japanese, Finnish, Korean, Turkish, Hungarian, and German which presents agglutinative features in its open lexicon but not in its case system) languages was shown when using sub-word units as basic speech recognition units. In (Geuntner, 1995) words were split into morphemes, which were then used as individual units. Using much shorter units than words in the cases of slightly inflectional languages (i.e. German) or highly inflectional languages (i.e. Serbian and Croatian) did not result in decreasing recognition error overall, because the positions of different types of morphemes (suffixes, prefixes, etc.) was not considered. Another suggestion was to use units larger than morphemes, such as stems, the stem being that part of the word which is common to all words belonging to the same word family or vocabulary entries (lemma). Lemma defines basic dictionary entry with different definitions for individual word forms. Stems were used to build language models, whereas the vocabulary still contains words. Another lacking feature of the new language model was information about the ending. The results of this scheme were again unsatisfactory, with no improvement in recognition error for German, Serbian and Croatian. In (Byrne et al., 2000) stems and endings were also used for building sub-word language models for speech recognition of Czech language. A common vocabulary was used with stems and endings marked with special characters. Stems with an empty ending (words) did not differ from stems with a non empty ending. By using sub-word units, the number of OOV words decreased and the recognition accuracy increased; however the evaluation of recognition error was performed at a sub-word level. In (Byrne et al., 2001) a two-pass strategy realized with finite state transducers was taken-up. In the first pass, a standard sub-word bigram language model was used to build the N-best list of sentences and in the second pass an interpolated sub-word trigram language model was used. Stems were predicted from previous stems, and the previous stem and ending were used for predicting endings. Despite its contribution to decreasing the amount of OOV words, a recognition process using sub-word units did not reduce total recognition error. They did not include information about which endings could follow a particular stem. In (Ircing and Psutka, 2002), besides the sub-word language model, also a language model based on word categories was used for speech recognition of Czech language. The sub-word model did not include endings and was only able to predict the sequences of stems. The received recognition error was decreased by 4% absolute, in comparison to word-based recognition. Similar procedures were also used on an agglutinative language, namely Hungarian. In (Szarvas and Furui, 2003), a finite state transducer was selected for speech recognition. In addition to these basic components described in (Mohri et al., 2002), they added two additional components: phonological rules and morphosyntactic rules. With the latter they filtered out the ungrammatical combinations (incorrect sub-word order), and with a basic trigram sub-word language model the error rate decreased by 18%, relatively. Similar methods were also applied on other agglutinative and tonal languages such as Korean (Choi et al., 2004, Kwon and Park, 2003), Japanese (Ohtsuki et al., 1999) or Turkish (Cilingir, 2003, Erdogan et al., 2005). All these languages have common characteristic of rapid growth in vocabulary and with it OOV words. The main difference between inflectional and agglutinative languages is in the number of morphemes per word. Agglutinative languages tend to have a high rate of morphemes per word, whereas in the case of inflectional languages, a word is typically composed by adding one inflectional morpheme to the base form.

In addition to sub-word modeling, there is another solution founded on the adaptation of vocabulary (Carki et al., 2000; Geuntner et al., 1998a, 1998b), but is only appropriate for processes (i.e. generating transcriptions) that are not limited by time scale.

ACCEPTED MANUSCRIPT

The first continuous speech recognition experiments for Slovenian were published in (Rotovnik et al., 2002). Word-based and sub-word-based recognition systems were reported. When using sub-word models, increased recognition performance only occurred if the comparison between word and sub-word units were executed on the same length of context. Experiments were performed with a HVite recognizer (Woodland et al., 1994). Later, in (Rotovnik et al., 2003), the recognizer was replaced by a trace_projector recognizer (Deshmukh et al., 1999), which is also used in this article. The results from a standard recognizer are comparable with those published in section 4.6 of this article.

In comparison to published work we would like to present the following points of this paper:

- distinguish between different types of sub-word units,
- how to deal with empty ending,
- restrict sets of endings for a particular stem,
- enlarge the context of language model history in search algorithms.

These terms will be discussed in detail in the following sections.

2.2 Morphological Structure of the Slovenian Language

This subsection presents the essential characteristics of the Slovenian language. Since most of the existing work and progress in the field of speech recognition has been done for the English language, we will compare the characteristics of Slovenian with those of the English language. The structure of language indirectly influences speech recognition efficiency. The Slovenian language shares its characteristics with many other inflectional languages, especially those of the Slavic family (Comrie and Corbett, 2001). Slavic languages are divided into three main groups:
• Southern: Slovenian, Serbian, Croatian, Bosnian, Macedonian and Bulgarian,
• Eastern: Russian, Ukrainian, Belarusian and Rusyn,
• Western: Czech, Slovak, Polish, Kashubian, Upper and Lower Sorbian.
In Slovenian, the parts of speech are divided into two classes, according to their inflectional characteristics:
• Inflectional category: nouns (substantive words), adjectives (adjectival words), verbs and adverbs.
• Non-inflectional category: prepositions, conjunctions, particles and interjections.
Slovenian words often exhibit clearer morphological patterns in comparison to English words. Morpheme is the smallest part of a word with its own meaning (or several meanings). In order to form different morphological patterns (declinations, conjugations, gender, number inflections), two parts of a word are distinguished: stem and ending. The stem is that part of the inflected word that carries its meaning; while an ending specifically denotes categories of case, person, gender and number, or the final part of a word, regardless of its morphemic structure. Stems can contain at least one morpheme, while endings usually contain one single item. The concept of grammatical categories will be introduced to outline the Slovenian inflectional morphology. In general, Slovenian shares its grammatical categories with other Slavic languages.

The Slovenian language distinguishes between three types of gender: masculine, feminine and neuter, whilst English does not. Slovenian nouns have six cases: nominative, genitive, dative, accusative, locative and instrumental. This multiple choice of cases enables a more flexible word order in

ACCEPTED MANUSCRIPT

Slovenian compared to English. Some Slavic languages distinguish all seven cases (Czech, Polish). The Slovenian word forms, not only differ in cases, but also in declination for all the three types of gender. The grammatical category of number is expressed in the ending and differs according to the quantity it expresses: one (singular), two (dual), and three or more (plural). Three types for the grammatical category of person (1st, 2nd, 3rd person) reflect the relationships between communication participants. The grammatical category of voice denotes the relationship between the object of the action and its executor. As with most European languages (derived from the Indo-European branch of languages), Slovenian has two voice categories: active and passive voice. Another grammatical category, mood, denotes the feeling of the speaker towards the act, state, course etc. which is defined by the verb. The three types of mood in Slovenian are: indicative, imperative and conditional. As in the English language, there are three degrees of comparison: positive, comparative and superlative. There are four tenses in the Slovenian language: present, past, past perfect and future. Table 1 shows different word forms for the word "nesti". For some words in Slovenian, it is possible to count up to 100 different word forms. These properties have already been successfully used in language modeling for Slovenian (Sepesy et al., 2003). There is one additional feature of the Slovenian language– morphologically speaking, some morphemes can alternate in consonants or vowels, and some in both simultaneously (table 2). Slovenian contains up to one thousand different combinations of morphological categories, while English only has about 30. Consequently, the word order in English is more rigid which means a greater contribution to building language models with lower perplexities. English words have less grammatical information encoded within a word. Grammatical features are evident from the relative order of words in a sentence. In the Slovenian language, the grammatical information is determined by a word's inflection. Consequently, word order in Slovenian is more relaxed. This characteristic of highly inflective languages causes high perplexity values, which could not be resolved by replacing a bigram model with higher order models. In this paper we do not address the problem of relaxed word order. The solution for decreasing very high OOV rate proves to be the use of sub-word or morphological units for language modeling. On the other hand, sub-word units introduce garbage words and the language model becomes less constrained but more robust.

3. Recognition using Sub-word Units

3.1 Statistical Speech Recognition

State-of-the-art recognition systems (Beyerlein et al., 2002; Evermann and Woodland, 2003; Kanthak et al., 2002; Mohri et al., 2002) use a statistical approach for speech recognition, based on the Bayes decision rule. The basic structure of such a system is presented in figure 1. It includes four components: acoustic analyzer, search algorithm, acoustic model, and language model. An input module called an acoustic analyzer transforms an analog speech signal into a sequence of acoustic features, which includes information about the spoken elements. The second module is the recognizer which, together with the stochastic models, represents the core of the recognition system. Stochastic models, acoustic, and language models present a source of information for the search algorithm in the recognizer. Most of the current state-of-the-art systems use Hidden Markov Models (HMM) to model acoustic production process (Rabiner, 1989). HMM's are stochastic finite automata consisting of states and transitions with attached probabilities (emission and transition probability respectively). The search algorithm uses the information provided by the acoustic model and the language model to determine the best word sequence:

ACCEPTED MANUSCRIPT

$$\left[w_1^N \right]_{opt} = \arg\max_{w_1^N, N} \left\{ p\left(w_1^N \right) \cdot p\left(x_1^T \mid w_1^N \right) \right\}$$

$$\approx \arg\max_{w_1^N, N} \left\{ \prod_{n=1}^{N} p\left(w_n \mid w_{n-m+1}^{n-1} \right) \cdot \max_{s_1^T} \prod_{t=1}^{T} \left\{ p\left(x_t \mid s_t, w_1^N \right) \cdot p\left(s_t \mid s_{t-1}, w_1^N \right) \right\} \right\} \quad \text{,where} \qquad \text{(Eq. 1)}$$

$N \rightarrow$ number of words $\qquad T \rightarrow$ number of acoustic features

$\mathrm{w}_n \rightarrow$ word n $\qquad x_t \rightarrow$ acoustic feature t

$s_t \rightarrow$ state t of HMM $\qquad w_1^N = w_1, ..., w_N$ (word sequence)

$x_1^T = x_1, ..., x_T$ (sequence of acoustic features)

The search problem described in Equation (1) can be efficiently solved after applying Viterbi approximation and using dynamic programming (Bellman, 1957). This so-called Bayes decision rule contains two types of stochastic models:

- The *m*-gram language model represents the a-priori probability of a given word set (first part of Eq. (1)).
- The acoustic model presents the conditional probability for the observed sequence of acoustic feature vectors, when the speaker has spoken a word sequence. Probability $p\left(x_t \mid s_t, w_1^N \right)$ is emission probability distribution attached to state s_t. $p\left(s_t \mid s_{t-1}, w_1^N \right)$ represents the transition probability attached to the transition between s_{t-1} and s_t.

In developing a search algorithm for a large vocabulary speech recognition system, Equation (1) is decomposed into the contributions of the individual recognition units (i.e. words, stems, endings, syllables, etc.) of the word sequence w_1^N. A word-based language model is used in recognition processes, when using words as recognition units. We will be assuming that recognition units are represented by a string of phonemes, which are defined by a pronunciation dictionary and modeled on the basis of tri-phone context. During the search, the recognition units are organized in a tree structure which combines equal unit prefixes. For different language model history a separate tree copy is generated. In this article we will restrict our explorations to recognition units internal context only, due to the complexity of the subject matter. The development of search algorithms of sub-word units has been limited by standard word-based recognizer.

3.2 Recognition problems with different sub-word units

3.2.1 Search space problem

Recognition process is, time-wise, very wasteful, because probability for every possible sequence of states should be calculated and the most probable one selected. But in large vocabulary continuous speech recognition, the number of possible sequences is immense, even for a very short speech segment. Different methods have, therefore, been developed to limit this search space, which has also led to the development of different recognition algorithm schemes. Selecting basic units of recognition,

ACCEPTED MANUSCRIPT

therefore, additionally complicates the selection of a recognition algorithm, and search-space restraining techniques.

3.2.2 Selection of basic recognition units

There are different techniques for limiting search space, such as using a dictionary of the most common words or using sub-word units: syllables, morphemes, lemmas, stems, endings etc. or anything that is shorter than the word and is in the vocabulary. If we use units shorter than words for recognition, differences may arise that can interfere with the results of the search algorithm. Here are the most important differences:

• In a word-structured dictionary, every unit (word) is followed by a silence or an unit boundary. With sub-word units, when the context is not allowed to extend across unit boundaries (search algorithm limitation), words will contain more than one unit boundary. From an acoustic point of view, the boundaries between words are defined by longer silence sections, but continuous speech has almost no instances of complete silence which can be distinguished easily. On the other hand, people use the logical meanings of words to define the boundaries between them. One of the possible solutions to this problem, therefore, might be in using a special dictionary unit, which marks the unit boundary that would allow the recognition system to use its language-model probability for discerning the boundaries between sub-word units. Combined sub-word units between the boundaries will then present a whole word. The second solution, which we have used in our experiments, is dividing words into two units at the most. The second unit will present the word ending and also the boundary between sub-word units. When we mentioned dividing words into two units at the most, we also added non-splitable words with an empty ending into the dictionary. This, consequently, enabled the existence of two identical sub-word units, where the first one would end with a silence, and the second one continues with the appropriate ending. Since it is not possible to distinguish these two sub-word units on the basis of acoustic information (they have the same transcription), we looked for the information in a language model. When building a language model, we used different notations to separate distinctive units, which resulted in a better performance of the language model.

• In general, recognition systems can use different sub-word recognition units, and their selection presents a very important factor in recognition process. The search space is directly limited by the unit set in the dictionary, because it only allows limited state sequences. By defining the size of the dictionary and by selecting the proper units we can, therefore, influence the size of search space. Acoustic and language models, on the other hand, indirectly determine the search space, assigning different probabilities to different units. Selecting basic recognition units is a compromise between two contrary features:

- how successfully the sub-word-based set will model words (from the qualitative and quantitative point-of-view), and
- how successfully the sub-word-based set will limit search space.

With shorter sub-word units we can obtain a better coverage of the whole word corpus whilst simultaneously enabling grammatically incorrect, yet similar, words. On the other hand, effective language models are difficult to build using very short units (for example phonemes). At the same time, using a word-based dictionary will not enable complete word coverage, but will still efficiently restrict search space.

ACCEPTED MANUSCRIPT

• In addition to limiting search space, the purpose of the dictionary is also to define transcriptions and pronunciations of basic recognition units – sequences of HMM states. In Slovenian, the pronunciation of basic units can be fairly accurately determined from their written forms. This is completely opposite to English, where you would have to know the whole word to conclude the pronunciation of its parts. So in some languages, pronunciation will determine the set of basic sub-word recognition units.

• Selecting phonemic models is closely connected to the sub-word units. If we decide to take context into consideration, then every phoneme will form several models, depending on its phonemic context. The lengths of basic units will also influence the number of acoustic models, especially when the recognition system does not use cross-unit acoustic models. It is obvious that, when using context dependent models, longer units will contain more information compared to shorter ones, because they include fewer boundaries, which disable the use of context. The use of cross-unit models will cause a surplus of computations, which depend on the length of the basic units. Usually extending context over the boundaries of basic units will demand the use of additional tree copies of the dictionary and, if the units are short, this will often mean including more new tree copies, as opposed to using longer units. In those cases where the word is split into no more than two parts, we also have to consider the lengths of the sub-word unit parts. The longer part will have better acoustic differentiation, while the differentiation of the shorter second part will, consequently, be nontrivial. For this reason it is very important to find a compromise between word decompositions and the length of sub-word units.

As can be seen, choosing the right sub-word unit for large vocabulary continuous speech recognition depends on more than one parameter (its length, context, pronunciation, coverage), in addition to the search algorithm used. In the following subsections, we will present an algorithm for specific types of sub-word units, which will successfully replace words as basic recognition units in the speech recognition process.

3.3 Recognition using word-based models

In recognition with word internal acoustic models, the contextual dependency between phonemes is only present within the word. Figure 2 shows a chain of stochastic models for the string "jaz sem" (meaning "I am"). HMM for the given word string is composed of three-state HMM triphones for the individual words. Only part of the phonetic context between word boundaries is taken into consideration. In our case, this would mean that the right-hand context of the last phoneme in »jaz« and the left-hand context of the second word "sem" will be marked as unknown. When a word is comprised of only one phoneme, the left-hand and the right-hand contexts are both marked as unknown. We use a special phoneme symbol "/" for marking all the boundaries between words. By eliminating contextual dependency at word borders, the HMM of a given word depends solely on the word itself, and can be directly determined from a pronunciation vocabulary. With the Viterbi approximation, the acoustical model contribution in the Bayes decision rule (Equation 1) can be broken down into the contributions of individual words for a word sequence w_1^N by using optimization over finite times t_1^N of individual words (Sixtus and Ney, 2002):

ACCEPTED MANUSCRIPT

$$p\left(x_1^T \mid w_1^N\right) \approx \max_{t_1^N} \prod_{n=1}^{N} \left\{ \max_{s_{t_{n-1}+1}^{t_n}} \prod_{t=t_{n-1}+1}^{t_n} \left\{ p\left(x_t \mid s_t, w_n\right) \cdot p\left(s_t \mid s_{t-1}, w_n\right) \right\} \right\} \quad \text{(Eq. 2)}$$

In this case, according to the definition $t_0 = 0$ and $t_N = T$. The sequence of states $s_{t_{n-1}+1}^{t_n}$ is composed of HMM states for the word hypothesis w_n. The contribution of the word w_n, which begins at the time $t_{n-1}+1$ and ends at time t_n, is given in the outer brackets of Eq. (2). This part of the equation determines the probability that a part of the sentence (the word w_n) has generated acoustical features $x_{t_{n-1}+1}, ..., x_{t_n}$. As can be seen in Eq. (2), the previously-mentioned equation-part depends only on the current word w_n and its starting and ending times. The decision rule for recognition with word internal models can be formed into the following equation:

$$\left[w_1^N \right]_{opt} \approx \underset{w_1^N, N}{arg\max} \left\{ \max_{t_1^N} \prod_{n=1}^{N} \left\{ p\left(w_n \mid w_1^{n-1}\right) \cdot \max_{s_{t_{n-1}+1}^{t_n}} \prod_{t=t_{n-1}+1}^{t_n} \left\{ p\left(x_t \mid s_t, w_n\right) \cdot p\left(s_t \mid s_{t-1}, w_n\right) \right\} \right\} \right\} \quad \text{(Eq. 3)}$$

3.4 Sub-word recognition with sub-word models

3.4.1 Acoustical modeling

Sub-word acoustic models can be defined similarly to those of word-based. The main difference between them is in the emergence of new unit boundaries. As can be seen in Figure 3, the use of sub-word units can indirectly generate a larger number of monophones and biphones. Adding the mark "0" at the beginning of word "jaz" means a stem with an empty ending (word "jaz" was not decomposed). In our case, the tri-phone was broken down into a biphone and a monophone. This diminishes the quality and the differentiation of the acoustic model, but retains the complexity of the search space in the sense of independent recognition units. Because of this, it is not so complicated to include acoustic models with sub-word recognition units into those commonly accessible recognition systems which use word units. As can be seen in the following subsection, this process will also create some redundancy. Word sequence w_1^N will be replaced with stems and endings in a sub-word model. To simplify the mathematical formulation, we will assume that the decomposition is known and each word w_n is decomposed into a stem o_n and an ending k_n (we will not discern between empty and non-empty-ending stems):

$$w_1^N = \left(w_1, w_2, ..., w_N\right) = \left(o_1, k_1, o_2, k_2, ..., o_N, k_N\right) \quad \text{(Eq. 4)}$$

If we additionally consider transformations $o_n \rightarrow u_{2n-1}$ and $k_n \rightarrow u_{2n}$, we can form a new Eq. (5) :

$$w_1^N = u_1^{2N} = \left(u_1, u_2, ..., u_N, ..., u_{2N}\right) \quad \text{(Eq. 5)}$$

ACCEPTED MANUSCRIPT

Because the stem o_n and the ending k_n follow each other directly, we replaced them with a sequence u_1^{2N}, which will double the number of units in comparison to word sequence w_1^N. In the case of the word-based acoustic model, individual word contributions were optimized over their finite times t_1^N, while in sub-word acoustic models they were optimized over finite times t_1^{2N} of sub-word units. By taking into consideration the transformation above, we obtain the following equation for the acoustical model contribution with sub-word recognition units:

$$p\left(x_1^T \mid w_1^N\right) = p\left(x_1^T \mid u_1^{2N}\right) \approx \max_{t_1^{2N}} \prod_{n=1}^{2N} \left\{ \max_{s_{t_{n-1}+1}^{t_n}} \prod_{t=t_{n-1}+1}^{t_n} \left\{ p\left(x_t \mid s_t, u_n\right) \cdot p\left(s_t \mid s_{t-1}, u_n\right) \right\} \right\} \quad \text{(Eq. 6)}$$

3.4.2 Language modeling

When we exchange the sequence of words with that of sub-words, we get the following record of the language model:

$$p\left(w_1^N\right) = \prod_{n=1}^{N} p\left(w_n \mid w_1^{n-1}\right) = p\left(u_1^{2N}\right) = \prod_{n=1}^{N} p\left(\left(u_{2n-1}, u_{2n}\right) \mid \left(u_{2k-1}, u_{2k}\right)_{k=1}^{n-1}\right) \quad \text{(Eq. 7)}$$

When the recognition process is based on sub-word units, a word-based language model is of limited use. In the case of (Eq. (7)) language model probabilities can be applied only on transitions between words. The result is less accurate beam pruning and a much larger search space. Using word-based language models we can compose only those stems and endings which constitute words already in the language model. Consequently, the problem of OOV words is unsolved. Consequently we use a sub-word-based language model for recognition with sub-word units (Eq. (8)). In this language model, the probabilities of some OOV words are also captured (if the word consists of known sub-words), and language model probabilities can easily be integrated into the search network.

$$p\left(u_1^{2N}\right) \approx \prod_{n=1}^{2N} p\left(u_n \mid u_1^{n-1}\right) \neq p\left(w_1^N\right) \quad \text{(Eq. 8)}$$

On the basis of this, we can present the corresponding sub-word bigram and trigram models:

$$\prod_{n=1}^{2N} p\left(u_n \mid u_{n-1}\right) = \prod_{n=1}^{N} p\left(u_{2n} \mid u_{2n-1}\right) \cdot \prod_{n=1}^{N} p\left(u_{2n-1} \mid u_{2n-2}\right) =$$

$$= \prod_{n=1}^{N} p\left(k_n \mid o_n\right) \cdot \prod_{n=1}^{N} p\left(o_n \mid k_{n-1}\right) \quad \text{(Eq. 9)}$$

$$\prod_{n=1}^{2N} p\left(u_n \mid u_{n-1}, u_{n-2}\right) = \prod_{n=1}^{N} p\left(u_{2n} \mid u_{2n-1}, u_{2n-2}\right) \cdot \prod_{n=1}^{N} p\left(u_{2n-1} \mid u_{2n-2}, u_{2n-3}\right) =$$

ACCEPTED MANUSCRIPT

$$= \prod_{n=1}^{N} p\left(k_n \mid o_n, k_{n-1}\right) \cdot \prod_{n=1}^{N} p\left(o_n \mid k_{n-1}, o_{n-1}\right) \qquad \text{(Eq. 10)}$$

Eq. (9) shows that, in the case of a bigram model, the ending k_n depends on the previous stem o_n and stem o_{n+1} is predicted from the previous ending k_n. In (Sepesy, 2002), it was proven that the connection between the stem and the ending $\left(o_n \mid k_{n-1}\right)$ is very weak and only makes a minor contribution to the success of a sub-word language model. Using a trigram sub-word model will predict the current unit (stem or ending) from previous consecutive units. Predicting stem o_n from the previous stem o_{n-1} and the previous ending k_{n-1} will, in this case, present a similar contribution to that of the bigram word-based language model, whereas the part $\left(k_n \mid o_n, k_{n-1}\right)$ equals the contribution of a bigram sub-word model. In this way we can establish that, in the case of using sub-word language models, its order should be twice as high when compared to the order of a word-based language model, to cover the same amount of information.

3.4.3 Bayes decision rule

When we combine the contribution of an a priori probability for a sub-word language model (Eq. 8) and the contribution of a conditional probability for a sub-word unit internal acoustical model (Eq. 6) into a Bayes decision rule of the optimal word sequence, the following equation can be given:

$$\left[w_1^N\right]_{opt} = \left[u_1^{2N}\right]_{opt} \approx$$

$$\approx \underset{u_1^{2N}, N}{arg\max} \left\{ \max_{t_1^{2N}} \prod_{n=1}^{2N} \left\{ p\left(u_n \mid u_1^{n-1}\right) \cdot \max_{s_{t_{n-1}+1}^{t_n}} \prod_{t=t_{n-1}+1}^{t_n} \left\{ p\left(x_t \mid s_t, u_n\right) \cdot p\left(s_t \mid s_{t-1}, u_n\right) \right\} \right\} \right\}. \qquad \text{(Eq. 11)}$$

As can be seen, when compared to the Bayes decision rule for word-based models, twice as many time limits must be optimized. Since basic recognition units are consequently shorter, the probabilities of partial hypotheses are of greater similarity, which, in turn, will reduce the efficiency of beam pruning, and increase the search space. The result is a demand for a more optimal search algorithm for sub-word models. The following section proposes a novel extended search algorithm, which will limit the search space and stimulate recognition times.

3.5 Sub-word recognition with stem-ending models and correct sub-word order

In the previous section, we did not limit the order of the recognized units. This problem is partly reduced by the language model, which gives nonsensical pairs (stem-stem or ending-ending) a very small probability (on the basis of smoothing technique), and even then the search network will contain all combinations until they are removed from it, with the help of pruning techniques. Increasing search space will have a negative effect on memory usage and the speed of evaluating the best hypothesis. Figure 4 shows the additional parts of trees, which are combined to represent incorrect pairs in the search network. By considering the correct sequence of units in the search network, we can claim with some certainty that the search space will decrease, however its positive contribution to the final result

ACCEPTED MANUSCRIPT

will lessen due to the use of smoothing techniques. If we sum up Eq. (4) and include the correct order of stem o_n and ending k_n, we can divide the total contribution of the acoustic model into contributions of individual stems o_n and endings k_n in word order w_1^N. Here, the contributions are optimized over finite times of stems and endings t_1^{2N}:

$$p\left(x_1^T \mid w_1^N\right) = p\left(x_1^T \mid (o,k)_1^N\right) \approx \max_{t_1^{2N}} \prod_{n=1}^{N} \left\{ \begin{array}{l} \max\limits_{s_{t_{2n-2}+1}^{t_{2n-1}}} \prod\limits_{t=t_{2n-2}+1}^{t_{2n-1}} \left\{ p\left(x_t \mid s_t, o_n\right) \cdot p\left(s_t \mid s_{t-1}, o_n\right) \right\} \cdot \\ \cdot \max\limits_{s_{t_{2n-1}+1}^{t_{2n}}} \prod\limits_{t=t_{2n-1}+1}^{t_{2n}} \left\{ p\left(x_t \mid s_t, k_n\right) \cdot p\left(s_t \mid s_{t-1}, k_n\right) \right\} \end{array} \right\} \quad \text{(Eq. 12)}$$

Eq. (12) indicates that the contribution of a conditional probability for a sub-word unit internal acoustic model does not consider stem-stem and ending-ending pairs, while the number of optimizations after ending times is still twice the size, as in the case of word-based models. The use of sub-word language models will remain the same despite the limitations. Nonsensical sequences in the language model only appear through back-off weights. By joining the contributions of acoustic and language model probabilities, we get the following equation of Bayes decision rule for using bigram language models:

$$\begin{aligned} \left[w_1^N\right]_{opt} &= \left[(o,k)_1^N\right]_{opt} \approx \\ &\approx \arg\max_{(o,k)_1^N, N} \left\{ \max_{t_1^{2N}} \prod_{n=1}^{N} \left\{ \begin{array}{l} p\left(o_n \mid k_{n-1}\right) \cdot \max\limits_{s_{t_{2n-2}+1}^{t_{2n-1}}} \prod\limits_{t=t_{2n-2}+1}^{t_{2n-1}} \left\{ p\left(x_t \mid s_t, o_n\right) \cdot p\left(s_t \mid s_{t-1}, o_n\right) \right\} \cdot \\ \cdot p\left(k_n \mid o_n\right) \cdot \max\limits_{s_{t_{2n-1}+1}^{t_{2n}}} \prod\limits_{t=t_{2n-1}+1}^{t_{2n}} \left\{ p\left(x_t \mid s_t, k_n\right) \cdot p\left(s_t \mid s_{t-1}, k_n\right) \right\} \end{array} \right\} \right\}. \end{aligned} \quad \text{(Eq. 13)}$$

3.5.1 Sub-word recognition with stem-ending models with correct sub-word order and a limited set of endings for separate stems

When decomposing words into sub-word units it is possible to define a finite set of endings for a given stem (based on a training corpus), with the purpose of limiting the expansion of the recognized stem into a limited tree of endings. We suggest using a tree list to build separate trees of endings for individual stems (Figure 5). Although realization using a tree list increases static search space for the size of all trees of endings, we decided to use it, because of its simplicity. The conditional probability now additionally includes the existence of sequence $o_n k_n$, which affects the design of Eq. (12):

ACCEPTED MANUSCRIPT

$$p\left(x_1^T \mid w_1^N\right) = p\left(x_1^T \mid (o,k)_1^N\right) \approx$$

$$\approx \max_{t_1^{2N}} \prod_{n=1}^{N} \left\{ \begin{array}{l} \max\limits_{s_{t_{2n-2}+1}^{t_{2n-1}}} \prod\limits_{t=t_{2n-2}+1}^{t_{2n-1}} \left\{ p\left(x_t \mid s_t, o_n\right) \cdot p\left(s_t \mid s_{t-1}, o_n\right) \right\} \cdot \\ \cdot \max\limits_{s_{t_{2n-1}+1}^{t_{2n}}} \prod\limits_{t=t_{2n-1}+1}^{t_{2n}} \left\{ p\left(x_t \mid s_t, k_n\right) \cdot p\left(s_t \mid s_{t-1}, k_n\right) \cdot p\left(k_n, o_n\right) \right\} \end{array} \right\}. \qquad \text{(Eq. 14)}$$

The $p\left(k_n, o_n\right)$ represents the probability of a correct possible ending k_n, which can follow stem o_n, and is defined as:

$$p\left(k_n, o_n\right) = \left\{ \begin{array}{ll} 1 & \text{sequence } o_n k_n \text{ exist,} \\ 0 & \text{else} \end{array} \right\}. \qquad \text{(Eq. 15)}$$

As we can see, conditional probability is only calculated for certain predefined $o_n k_n$ pairs. Since we are still using the same language models, the Bayes decision rule is the same as defined by Eq. (13), except that we also have to include Eq. (15).

The idea to limit the set of endings for each individual stem is not used to improve the recognition accuracy, but to speed up the recognizer. Although, when using the additional knowledge source (morphological lexicon) to define all possible pairs $o_n k_n$, accuracy improvement could be expected as well. In this case we would be able to distinguish between linguistically correct (but in training corpus unobserved) and linguistically incorrect $o_n k_n$ sequences.

3.6 Sub-word Recognition using Stem-ending Models with Correct Sub-word Order, and Separate Sub-word Language Models (stem-stem, stem-ending)

The weakness of sub-word language models, when we compare them to word-based models, is in the length of context covered at the same language model order. As we have already mentioned, search space, in the case of sub-word models, is increased despite the same order. The reason for this increase is the larger number of time limits, over which conditional probabilities are calculated. At the same time shorter sub-word units become acoustically similar, which will additionally reduce the efficiency of search-space pruning techniques. The idea behind the following algorithm was, therefore, to preserve the same context length, as with word-based models, by combining sub-word stem-stem and stem-ending language models. Figure 6 illustrates changes in the sequence of probabilities for sub-word models. In basic search algorithm probability $p\left(k_n \mid o_n\right)$ is followed by probability $p\left(o_{n+1} \mid k_n\right)$ but in our new search algorithm, instead of the latter, we have used probability $p\left(o_{n+1} \mid o_n\right)$. The design of conditional probability for the acoustic model is the same as in Eq. (12). The a-priori probability of the separate sub-word language model for predicting stems and endings is defined by the following equation:

ACCEPTED MANUSCRIPT

$$p\left(u_1^{2N}\right) \approx \prod_{n=1}^{2N} p\left(u_n \mid u_1^{n-1}\right) = \prod_{n=1}^{N} p\left(o_n \mid o_{n-1}\right) \cdot \prod_{n=1}^{N} p\left(k_n \mid o_n\right) \text{ ,where} \qquad \text{(Eq. 16)}$$

$$\sum_{n=1}^{O_M} p\left(o_n \mid o_{n-1}\right) = 1 \text{ and } \sum_{n=1}^{K_M} p\left(k_n \mid o_n\right) = 1, \text{ with}$$

$O_M \rightarrow$ number of different stems

$K_M \rightarrow$ number of different endings

Transition from Eq. (8) to Eq. (16) uses the decomposition of each word into exactly one stem and one ending (Eq. (4)). Here, we have used an equation for a bigram model. If we compare Eq. (16) with Eq. (9) and Eq. (10), we can see that, compared to previous sub-word models, the latter has retained the context length of a trigram-gram sub-word model and is, therefore, comparable to a bigram word-based model. This will increase search space, when compared to previous sub-word models. Order expansion from bigram to trigram language model is straight forward. In case of trigram language model stem context covers two previously stems and the prediction of ending remains the same. By considering Eq. (16), we can define Bayes decision rule for recognition using separate sub-word models – for bigram language models as:

$$\left[w_1^N\right]_{opt} = \left[(o,k)_1^N\right]_{opt} \approx$$

$$\approx \arg\max_{(o,k)_1^N, N} \left\{ \max_{t_1^{2N}} \prod_{n=1}^{N} \left\{ \begin{array}{l} p\left(o_n \mid o_{n-1}\right) \cdot \max\limits_{s_{t_{2n-2}+1}^{t_{2n-1}}} \prod\limits_{t=t_{2n-2}+1}^{t_{2n-1}} \left\{ p\left(x_t \mid s_t, o_n\right) \cdot p\left(s_t \mid s_{t-1}, o_n\right) \right\} \cdot \\ \cdot p\left(k_n \mid o_n\right) \cdot \max\limits_{s_{t_{2n-1}+1}^{t_{2n}}} \prod\limits_{t=t_{2n-1}+1}^{t_{2n}} \left\{ p\left(x_t \mid s_t, k_n\right) \cdot p\left(s_t \mid s_{t-1}, k_n\right) \right\} \end{array} \right\} \right\}. \qquad \text{(Eq. 17)}$$

3.6.1 Sub-word recognition with stem-ending models with correct sub-word order, limited set of endings for separate stems, and separate sub-word language models (stem-stem, stem-ending)

In the previous section we presented an extended search algorithm, which will increase context length and, consequently, search space. One of the upgrades in the new search algorithm is the idea of limiting search space using a finite set of endings for an individual stem (subsection 3.5.1). Mathematical integration of sub-word models into Bayes decision rule for extended search algorithm with limiting sets of endings, is very similar to Eq. (13), plus the addition of Eq. (15):

ACCEPTED MANUSCRIPT

$$\left[w_1^N \right]_{opt} = \left[(o,k)_1^N \right]_{opt} \approx$$

$$\approx \arg\max_{(o,k)_1^N, N} \left\{ \max_{t_1^{2N}} \prod_{n=1}^{N} \left\{ \begin{array}{l} p(o_n \mid o_{n-1}) \cdot \max\limits_{s_{t_{2n-2}+1}^{t_{2n-1}}} \prod\limits_{t=t_{2n-2}+1}^{t_{2n-1}} \{ p(x_t \mid s_t, o_n) \cdot p(s_t \mid s_{t-1}, o_n) \} \cdot \\ \cdot p(k_n \mid o_n) \cdot \max\limits_{s_{t_{2n-1}+1}^{t_{2n}}} \prod\limits_{t=t_{2n-1}+1}^{t_{2n}} \{ p(x_t \mid s_t, k_n) \cdot p(s_t \mid s_{t-1}, k_n) \cdot p(k_n, o_n) \} \end{array} \right\} \right\}. \quad \text{(Eq. 18)}$$

3.6.2 Search space improvement

The drawback of the new algorithm with extended context is the fact that it increases search space and slows down recognition speed. We prevented rapid growth in search space by joining those stem trees, which originate in the identical previous tree of endings, into a common tree (Figure 7), which will then only hold the current best partial hypothesis in every timeframe. This reduced the size of search space to that of the standard word-based search algorithm. Figure 7 illustrates that only one tree (the beginning of the next word) extends from the recognized word (stem + ending), while in the previous version of the algorithm with extended context, every ending was followed by another tree. If we compare this new search algorithm with the basic sub-word algorithm, the major differences are in the way they handle context. A basic sub-word algorithm, which does not distinguish between different types of sub-word units, will concatenate basic units regardless of whether they were stems or endings, whereas the new algorithm with extended context and limited set of endings will always perform a composition on stems. This algorithm will define the optimal ending for every stem in a search space, merging of stems, and predict stems from previous stems. If the stems maintain the same amount of linguistic information as the words, the presented algorithm would be very similar to a word-based search algorithm, regarding the size of search space.

4. Experiments

4.1. Speech database

Algorithms were evaluated using the studio part of the SNABI speech database (Kačič et al., 2000). The database was composed of 6 subcorpuses, which contained 1,530 different sentences. The database contained the speech of 52 speakers, where each speaker read more than 200 sentences, while 21 speakers also read a text passage of 91 sentences. The complete database consists of approx. 14 hours of speech. To increase the training set, we also used the telephone part of SNABI speech database, which has the same structure as the studio part, except that it is larger. It contains the speech of 82 speakers and, together with the studio part, contains approximately 40 hours of speech.
For the test set, we used 80 minutes of speech material from the studio part of SNABI speech database, which was speaker and domain-independent. The set was divided into two parts:
• Development set of approx. 15 minutes (195 sentences) was used for finding the optimum scaling factors,
• Evaluation set (from now on referred to as test set) of approx. 65 minutes (779 sentences) for evaluating the system's performance, representing approx. 10 % of the studio part of the SNABI speech database.

ACCEPTED MANUSCRIPT

4.2 Text database

For training language models, we used a corpus of newspaper articles. It was obtained from the archives of the Slovenian newspaper VEČER, spanning the period from 1998 to 2003. The corpus size is 105 million words, 660,000 of them different. Speech source and text source differ in their content, since the speech database includes speech that was read, while the text corpus captures daily news. Currently, the two databases are the only ones appropriate for large vocabulary Slovenian speech recognition. All language models used for the evaluation of speech systems were built on the basis of text corpus VEČER.

4.3 Vocabulary statistics

Experiments were performed on two different vocabulary sizes: 20,000 and 60,000 basic units. Words were transcribed on the basis of the morphological lexicon and transcriptions for those entries lacking one were generated automatically from morphological and phonological rules. For word-based models, we used the text corpus to select an appropriate amount of the most common words. With sub-word models we first used a data-driven method to split the vocabulary of words and then added the most common sub-word units from the text corpus to expand the new vocabulary. In this way, 660,000 different words were split into 327,000 different stems and 2,943 different endings.

4.3.1 Sub-word generation

Word decomposition (Sepesy, 2002) was based on a predefined list of endings. Words are decomposed using the longest-match principle. The list of endings is searched for the longest ending that could be mapped to the finishing part of the word. These algorithms often exhibit over-stemming – producing stems that are too short. Restriction was added to determine that the remaining stem should be of a predefined minimum length. An empty ending is added if a word cannot be decomposed. Automatic generation of endings is based on a method called stemming (Popovič, 1992) and includes three steps:

1 A list is created of all words, which were written in reversed character order.
2 Words are arranged alphabetically; thus words, sharing a common ending, appear together on the list.
3 Initial characters of adjacent words on the list are compared, to find a maximum match.
There are two restrictions to avoid over-stemming. The first restriction limits the minimum length of the stem, while the second restriction says that the first character of an ending match must be a vowel, because consonants carry more information about the meaning of the word than vowels do (Dimec et al., 1999). As a consequence of the second restriction, words are decomposed at a consonant-vowel pair in most cases.

4.3.2 Unknown words in test set

As we have already mentioned, the advantage of sub-word-based models in recognition is a much more extensive coverage of a test set, which results in a lower number of unknown words (OOV words). Figure 8 shows the correlation between the number of OOV words and the number of units in the vocabulary, for the vocabulary of words and the vocabulary of sub-word units. At a vocabulary size of 20,000 most common units from the training set, and with word-based models, the OOV rate on the

ACCEPTED MANUSCRIPT

test set is 17,5 %, but is much smaller (2,7 %) when using sub-word vocabulary. By increasing the size of vocabulary, this distinction is decreased and with 60,000 units is reduced to 8,7 %. It takes 660,000 words or 330,000 distinct sub-word units to cover the complete training set.

4.4 Acoustic models

Word internal triphone acoustic models with sixteen Gaussian mixtures were used for all recognition experiments. Table 3 shows the statistics of acoustic models. The models were trained over the SNABI speech database. The table includes the number of trained acoustic models, total number of acoustic models and the total number of states. The number of all acoustic models depends on the structure and size of the vocabulary and presents the number of models, needed for complete vocabulary coverage. Word-based models contain 5,389 states after state-tying. Sub-word models and word models share some common triphones, the difference only arises at the end of the stem and the beginning of the ending. For sub-word models, two biphones are used in the transition from stem to ending, whereas word models use two triphones in that position – the reason lies in using sub-word unit internal triphone models. If we wanted to keep the context, we would get cross sub-word unit acoustic models in this position, which, however, is beyond the scope of this article. Decomposing words into stems and endings, and using unit internal triphone models (also containing biphones and monophones) can, therefore, create new biphones, as seen in this table. If we compare the number of acoustic models, we can see that their number is greater in case of sub-word-based models for both vocabulary sizes. This is caused by an additional set of words in the sub-word vocabulary, because we first split the word-based vocabulary and then complemented the sub-word vocabulary with the most frequent sub-word units from the word corpus. This increased the number of different words and triphones but decreased the number of states for sub-word acoustic models, when compared to word-based ones. The reason for the decline was a different set of acoustic units for the training set (substituting triphones with biphones at decomposition point), which causes different tying of states.

4.5 Language models

We used SRILM V-1.3 toolkit (Stolcke, 2002) to build and evaluate the language models referred to in this article. Table 4 shows the perplexity and the size of separate types of language models. If we use the same procedure to calculate the perplexity of sub-word-based models, we obtain the value for perplexity at the sub-word level, however the results are not intercomparable. Perplexity depends on the vocabulary. Although both vocabularies are of the same size, their contents differ to a great extend. Sub-word perplexity would have been much smaller than the one for word level, mostly due to excellent predictions of probability for endings, which is the reason why we have also calculated perplexity at the word level for sub-word-based models. The overall high perplexity values of language models are partially a result of poor coverage of the target language (determined by the recognition test set) by the training corpus of the language model. When the perplexity of the sub-word-based language models is compared to the word-based model, the values were relatively higher, because as the units become fewer and smaller the language model becomes less constrained. If we compare basic sub-word models to word-based ones, the weakness of bigram sub-word models comes to the surface when calculating the perplexity: smaller context coverage causes a rise in perplexity. By restricting the order of sub-word units (New_SB) the perplexity was reduced. The number of bigrams for these models also increased, due to an increase in context when compared to basic sub-word models. Extending the

ACCEPTED MANUSCRIPT

context to trigram modelling improved the perplexity. Consequently the complexity of models increased (comparing the No. of 3-grams against the No. of 2-grams). As in the case of bigram models restriction of sub-word units order improved the results.

4.6 Recognition results

Firstly, we conducted recognition experiments on a vocabulary size of 20,000 units for different vocabulary types and versions of search algorithms. Using a trace_projector recognizer, we performed experiments on word-based models and basic sub-word models. We evaluated recognition error for words and sub-word units, recognition speed and the size of search space, expressed in the form of the average number of active models.

4.6.1 Recognition results for a vocabulary of 20,000 recognition units

Using word-based models (Standard_WB), bigram language model and a vocabulary with 20,000 units, we achieved a recognition error of 53, 3 % (Table 5). One source for error was found to be OOV words. Another source of errors is different word forms, derived from common lemma, which are phonetically very similar. By restricting search space we managed to influence the speed of recognition and optimize it according to the best recognition results. Speed values in other models and search algorithms will be presented relative to the speed of the standard recognition system, achieved with a word-based model. In this case, the recognition speed was 24.9-times the value of real time. We must also mention that we did not directly focus on the problem of reducing search space for word-based models and that we used a standard Viterbi search algorithm with beam pruning and restricting the number of active models. The same recognizer was used for the first part of the experiments with sub-word models (Basic_SB). By using these, we decreased the extent of OOV words and, consequently, total word error rate by 3 % absolute. Due to the increase in search space, recognition times were also increased by 14,1 % relative. Table 5 shows the increase in the average number of active models for relative was 11,6 %. As we have already mentioned, using a basic search algorithm with sub-word models was not optimal, in the sense of finding the best path, because it also includes incorrect combinations (sequence of endings or sequence of stems with a non-empty ending). That is why we additionally integrated techniques for restricting the order of sub-word units (New_SB+Order) into the search algorithm. This, however, had no greater impact on recognition accuracy. A slight degradation in recognition error (0,1 %) is due to the elimination of correct partial hypotheses, which influence the beam-pruning procedure and limit the number of active models with their partial results. Alternatively, recognition speed was lower by only 2,8 % relative, compared to word-based models, at an almost identical average number of active models. The next experiments used the new search algorithm with extended context, which includes a longer context at the sub-word level. By increasing context, the basic version of the new algorithm (New_SB+ExtContext) increased the search space and reduced recognition speed, while recognition error was decreased absolutely by 3,2%, when compared to a basic search algorithm with sub-word models, and 6,2 % when compared to word-based models. Search space was efficiently reduced by restricting the number of endings per stem (New_SB+ExtContext+LimEnding). Compared to a basic search algorithm with extended context (New_SB+ExtContext), the new one reduced the number of active models by 2,8 % relative. By restricting the number of endings, we had to rearrange the source code of the new search algorithm. This also resulted in a recognition speed, which increased by 8,3 %. Since the search algorithm with

ACCEPTED MANUSCRIPT

restricting the number of endings (New_SB+ExtContext+LimEnding) was still slower than the standard algorithm with word-based models (36,1 %), we additionally reduced the search space for the new algorithm with extended context by grouping trees of stems, originating from the same previous tree of endings, into a common tree (New_SB+ExtContext+LimEnding+Group). Recognition error did not increase, but search space decreased, which enabled the same recognition speed than with word-based models. We can see that the best version of this new search algorithm with extended context (New_SB+ExtContext+LimEnding+Group) has decreased the number of active models when compared to the search algorithm with order limitation (New_SB+Order), and come very close to the standard search algorithm with word-based models.

In all afore-mentioned experiments bigram language models were applied. Next recognition experiments include trigram language models. Results were reported only for standard search algorithms with word-based (Standard_WB) and sub-word-based (Basic_SB) models and for the best new search algorithm with extended context (New_SB+ExtContext+LimEnding+Group). When comparing recognition results obtained with word-based trigram language model against bigram language model recognition error decreased by 3.1% absolute, but recognition speed was more than two-times higher. It was caused by increased context of partial hypothesis which needs to be separately stored before they were merged in search process. When comparing standard search algorithms with word-based and sub-word-based models the last one decreased recognition error by 2.3% absolute. As in the case of bigram language model recognition time increased for 17.1% relative. The new search algorithm with extended context achieved smallest recognition error (absolutely 44.7%) with almost the same recognition time compared to standard search algorithm with word-based models.

4.6.2 Recognition results for a vocabulary of 60,000 recognition units

Enlarging the vocabulary to 60,000 units we achieved 45.7-times the real-time speed with word-based models (and bigram language model) and we decreased recognition error by 8,7% absolute (Table 6) when comparing the results of the smaller vocabulary, where the number of missing words decreased by 10 % absolute (Figure 8). With the basic search algorithm and sub-word model, recognition results improved by just 0.7%, while the number of OOV words decreased by 1,5%. This was caused by the acoustic and language interchangeability of units – shorter acoustic models, which represent vocabulary entries, achieve tighter acoustic discrimination compared to longer acoustic models. It is also true that probabilities, received from language models in the case of shorter vocabulary units, make a smaller contribution to the search algorithm than longer vocabulary units. The reason is in the compression of linguistic information (with a certain set of sub-word units we can describe a much larger set of words), which smoothes out probabilities between individual units. Lower probabilities between individual basic models cause less accurate restriction in the search space, and increase the probability of incorrect hypotheses, which also influences recognition error. By using the search algorithm with extended context, we decreased recognition error by 3% absolute, in comparison to the smaller vocabulary, while the greatest difference in comparison to the basic search algorithm with sub-word models is in the inclusion of longer context and restricting the search space to the correct order of hypotheses. This algorithm achieved the lowest recognition error (44,1 %) among all bigram language models. The relationship between the performance speeds of search algorithms remained the same as with a 20,000-unit vocabulary, because the new search algorithm (New_SB+ExtContext+LimEnding+Group) helped to achieve a practically identical speed to that of

ACCEPTED MANUSCRIPT

the word-based-model algorithm (only 0,5 % difference) and a similar speed was also achieved using the search algorithm with order limitation (New_SB+Order). The average number of active models also remained in a similar relationship to the recognition results, as in the case of the smaller-sized vocabulary.

With trigram language models similar improvements were achieved as in experiments with a vocabulary size of 20,000 units. Standard search algorithm with word-based models (Standard_WB) achieved the best result among all word-based recognition experiments (absolutely 42.3%). The recognition time increased for 113% compared to standard search algorithm with bigram language models. Standard search algorithm with sub-word-based models did not improve the results of word-based models (increase WER for 5.4% absolute), but we achieved almost the same recognition error (absolutely 42.0%) with the new search algorithm.

5. Final discussion of experimental results

The usage of standard recognition systems for successful recognition of Slavic languages is not always suitable because of their rich morphology. The biggest problem is in words with common word forms, which increase vocabulary size and decrease the acoustic separability of units, and, therefore, have a negative influence on word error rate. Due to the reduced efficiency of pruning techniques (beam pruning), search space increases, which results in longer recognition times. One solution for reducing vocabulary size is using sub-word units, which, however, does not solve the similarity problem. Instead, it increases the similarity, because of unit shortness.

The recognition problem for inflectional languages was addressed by replacing words with sub-word units: stems and endings. We did not limit ourselves to using the basic search algorithm. Instead, we included features of inflectional languages into the design of a new search algorithm. We added the possibility of restricting the correct order of sub-word units. By differentiating between sub-word units, we also incorporated separate pruning techniques for stems, endings and stems with an empty ending. The effect was positive, showing a smaller search space when compared to the basic search algorithm (10%), and similar search space size when compared to the standard word-based search algorithms. Recognition accuracy remained the same. Next we extended the context by using separate sub-word bigram (trigram) language models. Such a design increased the context of sub-word models to the context of word-based language models. The introduction of longer context had a positive effect on recognition efficiency, because error rate decreased by at least 3% absolute, when compared to the basic search algorithm with sub-word models. We limited the increase in search space by limiting the number of endings for individual stems, which we used to restrict the growth of stem trees. This resulted in the same recognition accuracy and higher recognition speeds by at least 3% relative compared to the search algorithm with the extended-context. The next improvement in speed and search space was made by combining the trees of stems, which were derived from the same tree of endings and combined into one common tree. With the new search algorithm we achieved the smallest search space amongst all search algorithms, using sub-word models and identical search space compared with standard word-based search algorithm. With a vocabulary size of 20,000 units and bigram language models the new search algorithm with sub-word models decreased error rate by 3.2% absolute compared with basic search algorithm with sub-word models and 6.3% absolute compared with the standard word-based search algorithm.

ACCEPTED MANUSCRIPT

Comparing the new search algorithm with basic search algorithm with sub-word models, bigram or trigram language models and a vocabulary size of 60,000 units it retained a performance gain (error rate decreased by at least 5.5% absolute) but improvement over standard word-based search algorithm was not achieved. One reason could be the decomposition algorithm. It is based on a data driven approach, with no emphasis on language-specific characteristics. The algorithm over-stem (or under-stem) some word forms in order to produce the minimal number of modelling units. Consequentially, words having the same lemma received different stems. Using morphological lexicon, decomposition could be derived from the information about lemmas. Using this information, words having the same lemma could obtain the same stem. Another problem is acoustic separability. We could control the length of the sub-word units and, consequently, acoustic separability, but at the same time we would violate the morphological rules and weaken the power of the language model. By enlarging the vocabulary the problem of acoustic confusability is even more evident. Larger vocabulary contains more candidates for acoustic confusability. Increasing the size of word-based vocabulary would also increase the acoustic confusability, because of more inflected word-forms included.

6. Conclusion

In this article we presented a new search algorithm with sub-word models, which restricts search space by using sub-word units in the correct order, limiting the number of endings for an individual stem, using separate sub-word language models (extended context), and combining the trees of stems, which were derived from the same tree of endings, into one common tree. The result was the smallest search space amongst all search algorithms, using sub-word models and identical search space, compared with standard word-based search algorithm. Using higher order sub-word-based language models (trigram), did not contribute so much to the performance of new search algorithm, because the problem of free word order arise. The essential feature of sub-word-based language model is the capability of modeling dependencies within a word. In general sub-word-based language models are less constrained and lead to increases in word-based perplexity. Therefore, such models would still have to be combined with ones that could produce probabilities for larger units (i.e. words, classes of words). One promising way for the future would be to combine those models, which capture different dependencies in language.

This recognition system is designed to be extendable to other inflectional languages and, with minor modifications, also used for other languages which include inflectional morphology. However, in recognition using sub-word models, some problems remain: How to preserve acoustic separability with shorter units. One of the possible solutions might be the integration of new knowledge sources from the field of speech understanding, or incorporating higher-level linguistic information (semantic and grammatical analysis) into the decoding process.

References:

Bellman R., 1957. Dynamic Programming, Princeton University Press.

Beyerlein P., Aubert X. L., Haeb-Umbach R., Harris M., Klakow D., Wendemuth A., Molau S., Pitz M., Sixtus A., 2002. Large Vocabulary Continuous Speech Recognition of Broadcast News - The Philips/RWTH approach, Speech Communication, vol. 37, No. 1-2, pp. 109-131.

ACCEPTED MANUSCRIPT

Byrne W., Hajič J., Ircing P., Krbec P. and Psutka J, 2000. Morpheme Based Language Models for Speech Recognition of Czech, pp. 211-216, in: Proc. Int. Conf. on Text Speech and Dialogue, Brno, Czech Republic, 2000.

Byrne W., Hajič J., Ircing P., Jelinek F., Khudanpur S., Krbec P. and Psutka J, 2001. On Large Vocabulary Continuous Speech Recognition of Highly Inflectional Language – Czech, in: Proc. European Conf. on Speech Communication and Technology, pp. 487-489, Allborg, Denmark.

Carki K., Geuntner P. and Schultz T., 2000. Turkish LVCSR: Towards Better Speech Recognition for Agglutinative Languages, in: Proc. Int. Conf. on Acoustics, Speech, and Signal Processing, pp. 1563-1566, Istanbul, Turkey.

Choi I., Yoon S. Y., Kim N., 2004. Large Vocabulary Continuous Speech Recognition Based on Cross-Morpheme Phonetic Information, in: Proc. Int. Conf. on Spoken Language Processing, Jeju Island, Korea.

Cilingir O. and Demirekler M., 2003. A New Decoder Design For Large Vocabulary Turkish Speech Recognition, in: Proc. European Conf. on Speech Communication and Technology, pp. 1185-1188, Geneva, Switzerland.

Comrie B. and Corbett G. G., 2001. The Slavonic Languages, Taylor & Francis Group.

Deshmukh N., Ganapathiraju A. and Picone J., 1999. Hierarchical Search for Large Vocabulary Conversational Speech Recognition, IEEE Signal Processing Magazine, Vol. 16, No. 5, pp. 84-107.

Dimec J., Džeroski S., Todorovski L. and Hristovski D., 1999. "WWW Search Engine for Slovenian and English Medical Documents", Medical Informatics Europe, Amsterdam: IOS Press.

Erdogan H., Buyuk O., Oflazer K., 2005. Incorporating Language Constraints in Sub-Word Based Speech Recognition, in: Proc. Automatic Speech Recognition and Understanding Workshop, San Juan, Puerto Rico.

Evermann G., Woodland P.C., 2003. Design of Fast LVCSR Systems, in: Proc. Automatic Speech Recognition and Understanding Workshop, pp. 7-12, U.S. Virgin Islands.

Geuntner P., 1995. Using Morphology Towards Better Large-Vocabulary Speech Recognition Systems, in: Proc. Int. Conf. on Acoustics, Speech, and Signal Processing, pp. 445-448, Detroit.

Geuntner P., Finke M., Scheytt P., Waibel A. and Wactlar H., 1998a. Transcribing Multilingual Broadcast News Using Hypothesis Driven Lexical Adaptation, DARPA Broadcast News Transcription and Understanding Workshop, Lansdowne.

Geuntner P., Finke M. and Scheytt P., 1998b. Adaptive Vocabularies for Transcribing Multilingual Broadcast News, in: Proc. Int. Conf. Acoustics, Speech, and Signal Processing, pp. 925-928, Seattle.

ACCEPTED MANUSCRIPT

Ircing P. and Psutka J., 2002. Lattice Rescoring in Czech LVCSR System Using Linguistic Knowledge, in: Proc. Int Conf. on Speech and Computer, pp. 23-26, St. Petersburgh, Russia.

Kačič Z., Horvat B., Zogling A., 2000. Issues in Design and Collection of Large Telephone Speech Corpus for Slovenian Language, in: Proc. Int. Conf. on Language Resources and Evaluation.

Kanthak S., Ney H., Riley M., Mohri M., 2002. A Comparison of Two LVR Search Optimization Techniques, in: Proc. Int. Conf. on Spoken Language Processing, pp. 1309-1312, Denver, Colorado.

Kwon O. and Park J., 2003. Korean Large Vocabulary Continuous Speech Recognition With Morpheme-based Recognition Units, Speech Communication, Vol. 39, No. 3-4, pp. 287-300.

Mohri M., Pereira F., Riley M., 2002. Weighted Finite-State Transducers in Speech Recognition, Computer Speech and Language, 16(1) pp. 69-88.

Ohtsuki K., Matsuoka T., Mori T., Yoshida T., Taguchi Y., Furui S. and Shirai K., 1999. Japanese Large-Vocabulary Continuous-Speech Recognition Using a Newspaper Corpus and Broadcast News, Speech Communication, Vol. 28, No. 2, pp. 83-166.

Popovič M. and Willett P., 1992. "The Effectiveness of Stemming for Natural-Language Access to Slovene Textual Data", Journal of the American Society for Information Science, No. 5, Vol. 43, pp. 384-390.

Rabiner L. R., 1989. A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition, in: Proc. of the IEEE, pp. 257–286.

Rotovnik T., Sepesy M. M., Horvat B., Kačič Z., 2002. Large vocabulary speech recognition of Slovenian language using data-driven morphological models, in: Proc. Int. Conf. on Text Speech and Dialogue, pp. 329-332.

Rotovnik T., Sepesy M. M., Horvat B., Kačič Z., 2003. Slovenian large vocabulary speech recognition with data-driven models of inflectional morphology, IEEE Automatic Speech Recognition and Understanding Workshop, U.S. Virgin Islands, 2003. pp. 83-88.

Sepesy M. M., 2002. The Topic Adaptation of Statistical Language Models (in: Slovenian), Ph.D. thesis, University of Maribor.

Sepesy M. M., Rotovnik T., Zemljak M., 2003. Modelling highly inflected Slovenian language, Int. journal of speech technology, pp. 245-257.

Sixtus A., Ney H., 2002. From Within-Word Model Search to Across-Word Model Search in Large Vocabulary Continuous Speech Recognition, Computer Speech & Language, Vol. 16, No. 2, pp. 245-271.

ACCEPTED MANUSCRIPT

Stolcke A., 2002. SRILM - An Extensible Language Modeling Toolkit, in: Proc. Intl. Conf. Spoken Language Processing, pp. 901-904, Denver, Colorado.

Szarvas M., Furui S., 2003. Finite-state transducer based modeling of morphosyntax with applications to Hungarian LVCSR, in: Proc. Intl. Conf. on Acoustics, Speech, and Signal Processing, pp. 368-371, Hong Kong, China.

Woodland P., Odell J., Valtchev V. and Young S., 1994. Large Vocabulary Continuous Speech Recognition Using HTK, in: Proc. Intl. Conf. on Acoustics, Speech, and Signal Processing, pp. 125-128, Adelaide.

Figure legends:

Figure 1: Structure of automatic speech recognition system.
Figure 2: An example of HMM model for word sequence "jaz sem" (*I am*).
Figure 3: An example of HMM model for sub-word sequence "0jaz se -m".
Figure 4: An illustration of redundant sub-word units in two consecutive trees.
Figure 5: Search network with a limited set of endings for each stem.
Figure 6: Structure of trees at limited search space.
Figure 7: Structure of search space in the search algorithm with extended context which uses grouping of stems, originating from the same previous stem trees, into a following common tree.
Figure 8: Diagram of OOV rate in test set.

Tables:

Table 1: An example of different word forms for the word "nesti" (to carry).

Infinitive/supine	nesti nest
Present	nesem neseš nese (singular) neseva neseta neseta (dual) nesemo nesete nesejo/neso (plural)
Passive participle -n	nesen nesena neseno 54 possible different word forms (3 genders * 6 cases * 3 categories of person)
Passive participle -č	nesoč nesoča nesoče
Active participle -l	nesel nesla nesli nesle neslo
Imperative	nesi nesiva nesita nesimo nesite
Nominal	nesenje 18 possible different word forms (6 cases * 3 categories of person)

ACCEPTED MANUSCRIPT

Table 2: Morpheme alternations.

Verb	Participle	Ending / Rule
prevoziti (to drive)	prevožen	(-iti) / z->ž
pustiti (to leave)	puščen	(-iti) / s->šč
roditi (to bear)	rojen	(-iti) / d->j
zahvaliti (to thank)	zahvaljen	(-iti) / l->lj
pisati (to write)	pišem	(-ati) / s->š
prenesem (to transport)	prenašam	(-em) / es->aš
vtaknem (to put into)	vtikam	(-em) / ak->ik
začnem (to begin)	začenjam	(-em) / ne->enj

Table 3: Statistics of acoustic models.

Models	Word-based (WB)		Sub-word (SB)	
Vocabulary size	20,000	60,000	20,000	60,000
Trained models	4462	5247	4768	7059
Total models	5103	6484	7290	10792
States	5389		4905	

Table 4: Language models statistics.

Models	Word-based (WB)		Sub-word (Basic_SB)		Sub-word (New_SB)	
Vocabulary size	20,000	60,000	20,000	60,000	20,000	60,000
Perplexity (bigram)	366	686	1872	2485	1365	1821
No. of 2-grams	5.22M	7.72M	3.68M	4.84M	6.93M	8.85M
Perplexity (trigram)	315	602	995	1351	843	1146
No. of 3-grams	17.55M	23.08M	21.21M	24.42M	28.28M	31.92M
OOV [%]	17.5	8.7	2.7	1.2	2.7	1.2

ACCEPTED MANUSCRIPT

ACCEPTED MANUSCRIPT

Table 5: Recognition results for different search algorithms at the size of 20,000 units. The speed is expressed relative to the speed of standard algorithm with word-based models and bigram language model which achieved 24.9-times the real time.

Experiments	WER[%]	Speed	No. of active models
Bigram LM			
Standard_WB	53.3	1.000	25254
Basic_SB	50.3	1.141	28198
New_SB+Order	50.4	1.028	25546
New_SB+ExtContext	47.1	1.474	35733
New_SB+ExtContext+LimEnding	47.1	1.361	34754
New_SB+ExtContext+LimEnding+Group	47.0	1.004	25928
Trigram LM			
Standard_WB	50.2	2.241	50673
Basic_SB	47.9	2.624	59934
New_SB+ExtContext+LimEnding+Group	44.7	2.254	51514

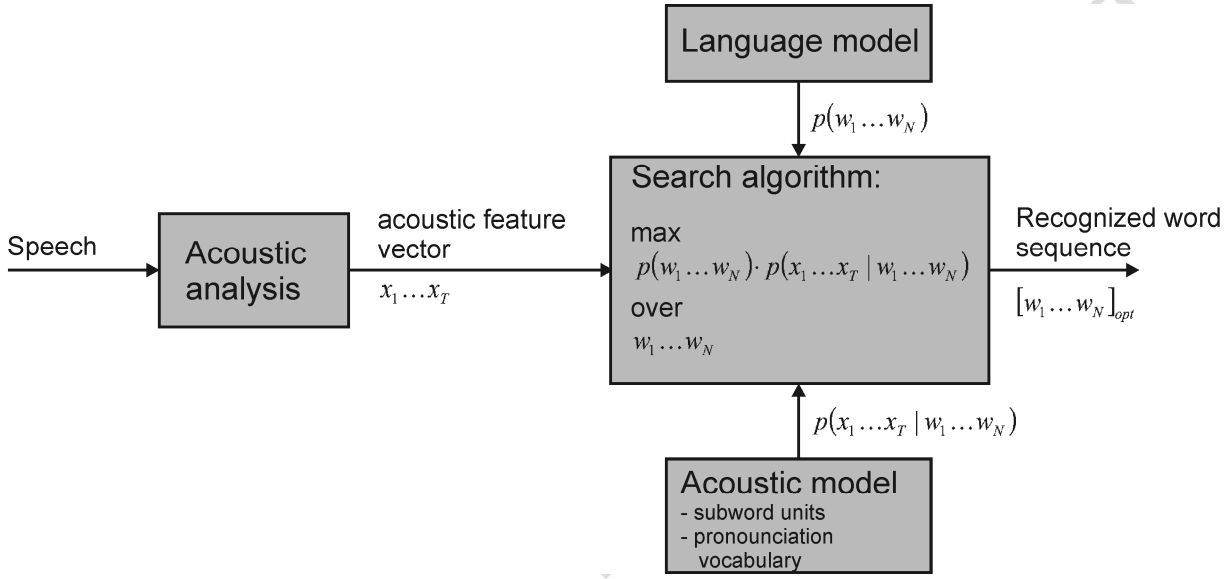
Table 6: Recognition results in using different search algorithms and vocabulary size of 60,000 units. The speed is expressed relative to the speed of standard algorithm with word-based models and bigram language model which achieved 45.7-times the real time).

Experiments	WER[%]	Speed	No. of active models
Bigram LM			
Standard_WB	44.6	1.000	31174
Basic_SB	49.7	1.130	35082
New_SB+Order	49.7	1.028	31714
New_SB+ExtContext	44.1	1.529	44744
New_SB+ExtContext+LimEnding	44.1	1.483	43620
New_SB+ExtContext+LimEnding+Group	44.1	1.005	32473
Trigram LM			
Standard_WB	42.3	2.137	59949
Basic_SB	47.7	2.602	71031
New_SB+ExtContext+LimEnding+Group	42.0	2.148	60762

ACCEPTED MANUSCRIPT

ACCEPTED MANUSCRIPT

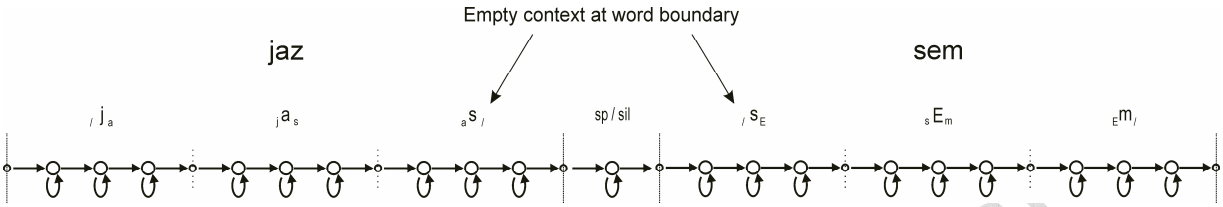
Figure 1



ACCEPTED

ACCEPTED MANUSCRIPT

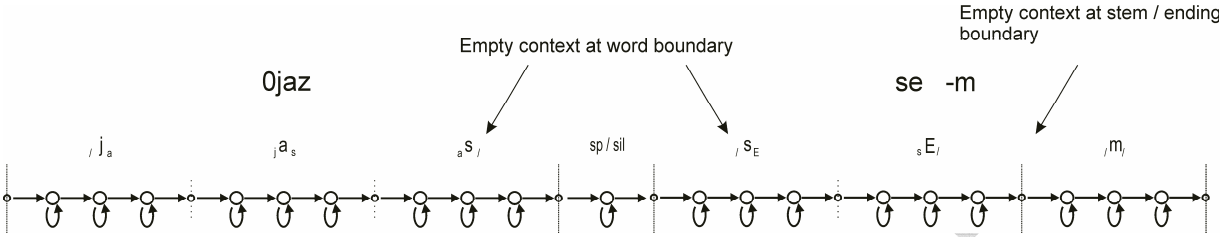
Figure 2



ACCEPTED MANUSCRIPT

ACCEPTED MANUSCRIPT

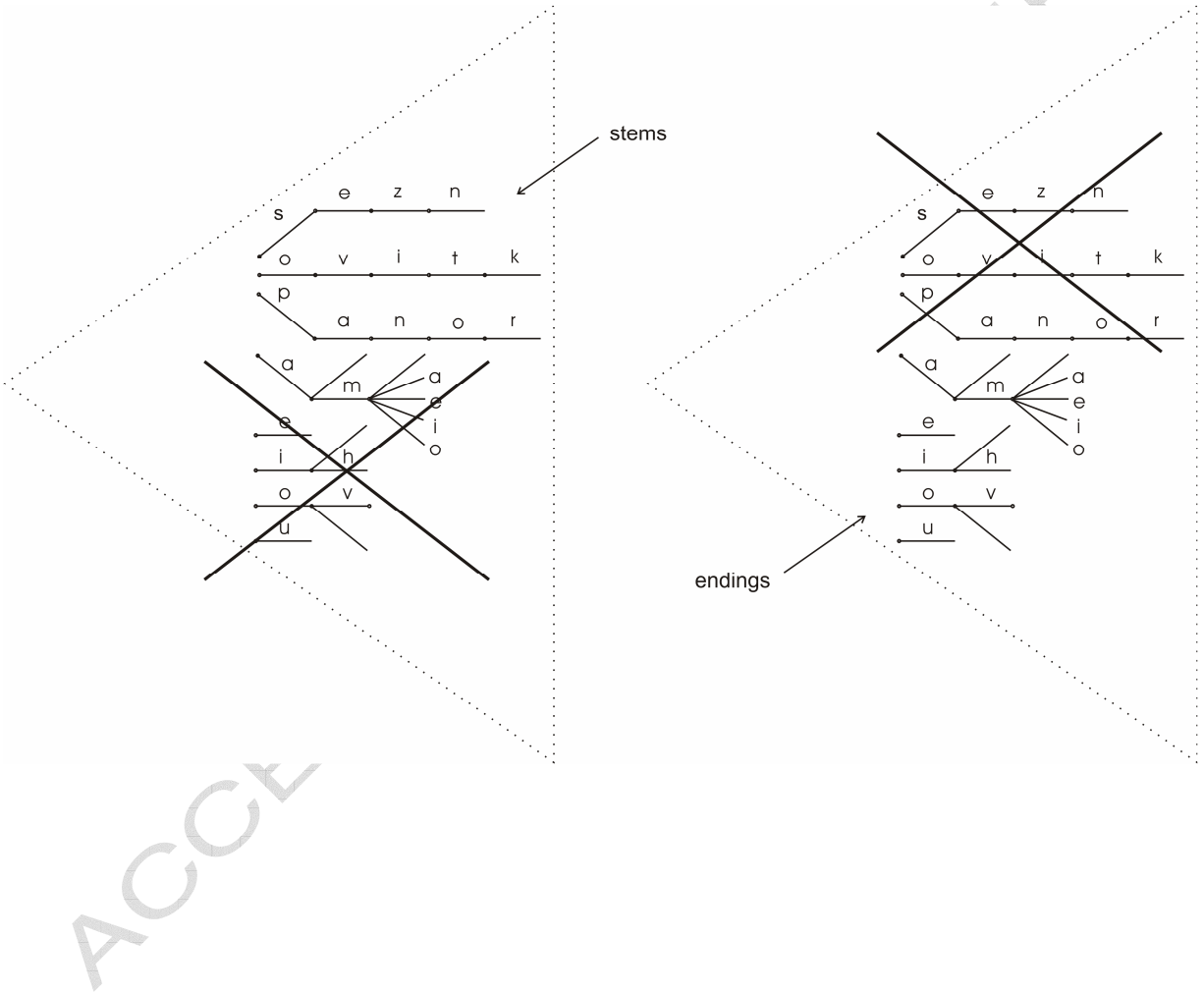
Figure 3



ACCEPTED MANUSCRIPT

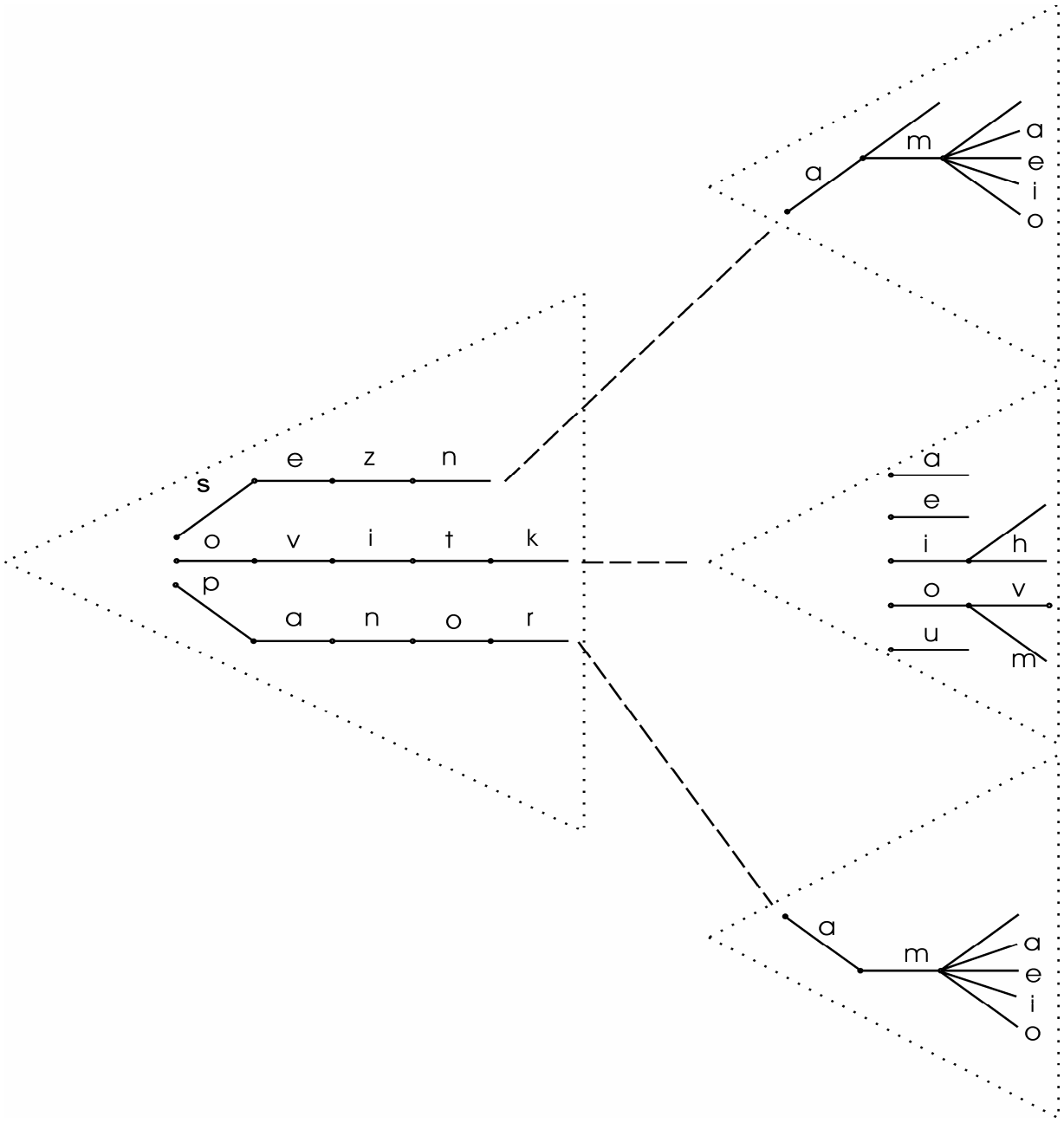
ACCEPTED MANUSCRIPT

Figure 4



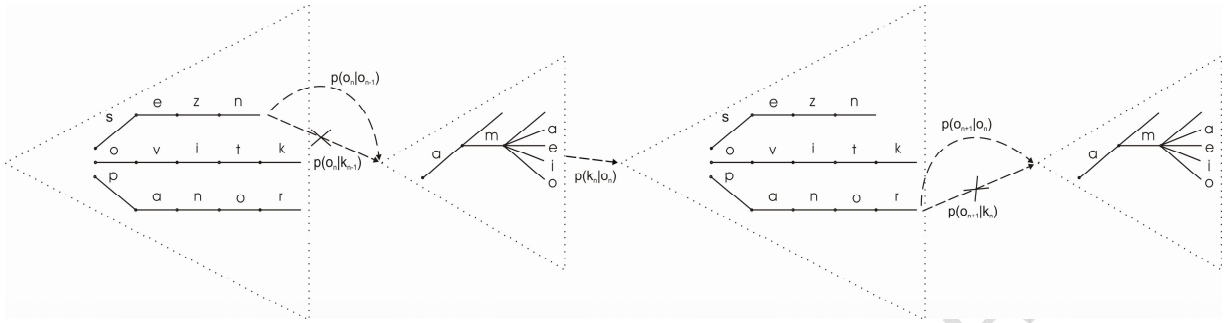
ACCEPTED MANUSCRIPT

Figure 5



ACCEPTED MANUSCRIPT

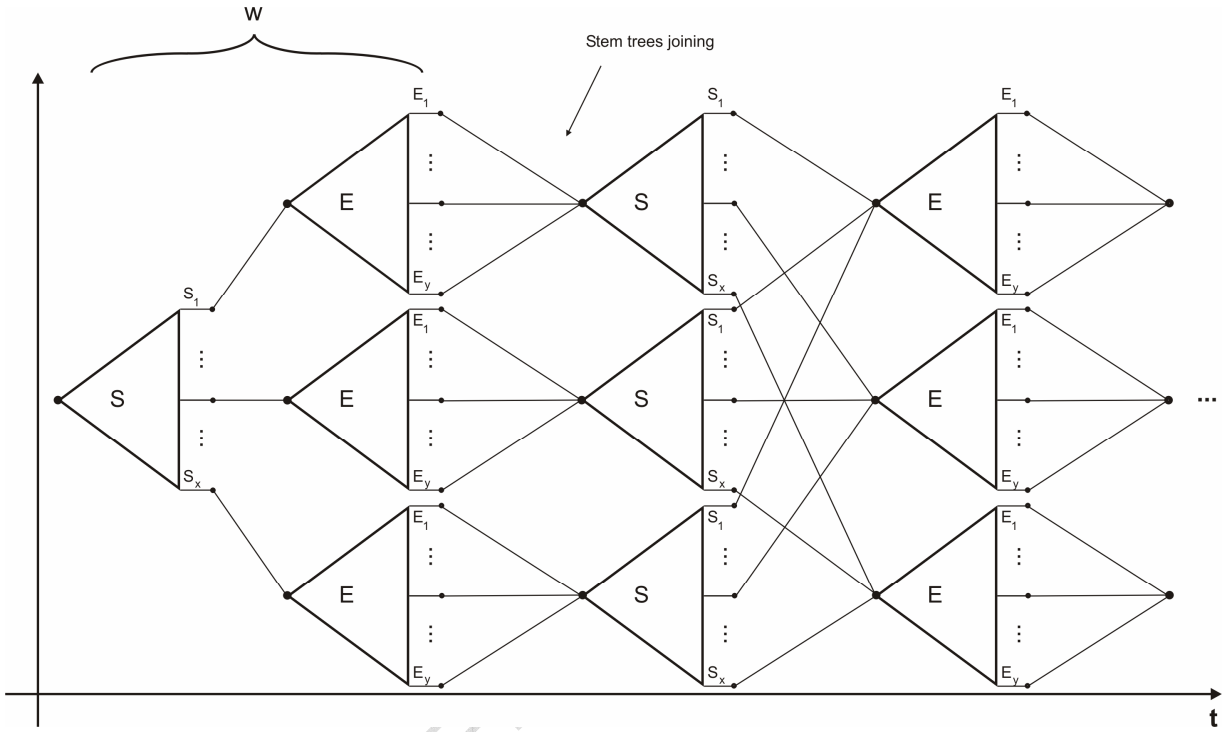
Figure 6



ACCEPTED MANUSCRIPT

ACCEPTED MANUSCRIPT

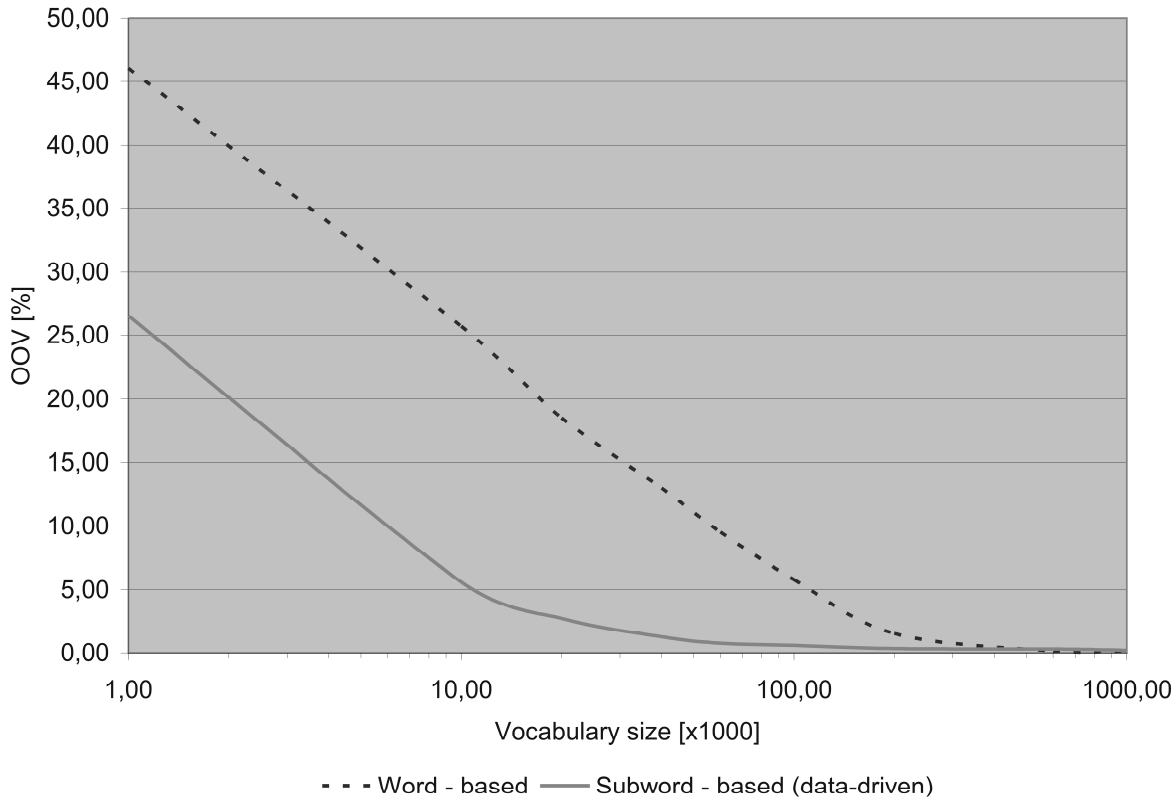
Figure 7



ACCEPTED

ACCEPTED MANUSCRIPT

Figure 8



ACCEPTED