Databases of Heterogeneous Segments for Concatenative Speech Synthesis

Ivan Kopeček, Karel Pala

Faculty of Informatics
Masaryk University Brno
Botanicka 68a
602 00 Brno
Czech Republic
kopecek@fi.muni.cz, pala@fi.muni.cz

Abstract

Heterogeneous segments can enhance the quality of concatenative speech synthesis especially for highly inflected languages. In this paper we present a brief analysis of the segment types on a general level and discuss the problems related to optimising databases of heterogeneous segments. We present a brief discussion of the algorithmical complexity for the proposed approach and offer some heuristics for optimizing databases of heterogeneous segments. We also mention the syllable and morphemic segments in relation to the development of the Czech speech synthesis system Demosthenes.

1. Introduction

Present concatenative speech synthesis systems may use a wide variety of segments: allophones, diphones, triphones, half-syllables, demi-syllables, syllabic segments and some other types (e.g. Doddington, 1997; Dutoit, 1997; Deligne, 1997).

With the increasing technical parameters of present computers, the number of segments is nowadays not so critical, and we can see attempts to taking advantage of this to use larger segments involving more coarticulation (e.g. Deligne, 1997; Dodington, 1997; Greenberg, 1998; Kopeček, 1998).

Hence, the natural question emerges when we look for an appropriate set of segments for a given model: is the chosen set, in any sense, optimal? This question sounds especially natural when using heterogeneous segments, i.e. a combination of segments of a different type (e.g. syllables combined with morphological segments).

2. Goal of the paper

In this paper we will deal with the problem of creating and optimizing a large database of heterogeneous segments for use in speech synthesis systems.

The relatively large number of heterogeneous segments implies the necessity to take into account the algorithmical complexity and to avoid approaches that are not effectively solvable.

Because we do not know a polynomial algorithm to solve the optimality problem, we present some heuristics that enables to optimize the database of heterogeneous segments in polynomial time (but only as an approximative solution to the global optimization problem).

3. Homogeneous Databases of Speech Segments

We shall say that a database of segments S is homogeneous if each element of a considered corpus can be obtained uniquely as concatenation of the segments belonging to the set S.

Practically, this condition is fulfilled for many instances of segments databases, like allophones, diphones, triphones, syllable segments (in the sense as they are described in the following section).

In what follows, we introduce a stronger type of homogeneity (strong homogeneity, see Section 6) as well.

4. Syllable Segments

Syllable segments is a special instance of the segment databases, which is interesting from our point of view, because it can demonstrate the tendency to apply larger segments, which leads to the attempts to use heterogeneous segments.

The syllable based approach to speech synthesis (e.g. Greenberg, 1998; Josifovski, 1997; Kopeček, 1997,1998) uses the fact that syllables create perceptually and acoustically coherent units and that they also represent the basic prosodic units.

However, there is no exact general specification of those segments. The feeling of syllable boundaries is subjective and in many cases not unique. This leads us to define an independent subset of syllables that is uniquely defined.

Another problem that appears when we try to determine an appropriate set of syllable segments is the necessity to respect the coarticulation effect between the adjacent syllables and to keep the number of segments within reasonable limits.

Although this problem can be effectively solved, there is a natural temptation to involve some frequently occurring bi-syllables in the database, enhancing the

coarticulation quality of the system. Then, however, we obtain a database that is not more homogenous.

Another possibility is to add some frequently appearing words or to enrich the database with morphological segments, which also leads to databases being not homogenous.

5. Databases of Heterogeneous Speech Segments

In order to describe the possible structures of segment databases, let us recall briefly some basic related notions (Kopeček, 2001).

First, we will say that a segment database is compatible with a given corpus, if each element of the corpus can be obtained as a concatenation of the segments belonging to the segment database.

This condition should (of course) be fulfilled for any segment database that is used for concatenative speech synthesis, otherwise there would be some cases for which the relevant segments would not exist.

A compatible set of segments S is said to be heterogeneous if it is not homogeneous. Heterogeneous segments can be used in order to achieve natural sounding speech by means of involving more coarticulation inside the segments.

An important case of particular interest to us is databases of "large" segments, e.g. syllabic segments combined with morphemes. This can be very interesting particularly for highly inflected languages, like Czech, for which the rich morphology dramatically increases the number of word forms.

6. Some Basic Types of the Segment Databases

We will say that a compatible database S is strongly homogeneous, if no segment of S is a part of a different segment of S.

It can be easily seen that this condition is stronger than homogeneity, i.e. if S is strongly homogenous, then it is homogenous.

Another useful term is consistency. We say that a compatible set of segments is consistent with a given corpus, if each segment is a part of an element belonging to the corpus.

If our corpus is large enough to be really representative, then the segments that violate this condition are suspected to be superfluous (and might possibly be eliminated).

7. Segment Bases and Minimality of the Segment Databases

We say that a compatible set is a base, if removing any segment causes the resulting database to be no longer compatible. A compatible database is said to be minimal, if there is no compatible database that has lower number of segments.

We can see that any base is a consistent set of segments. Because we are trying to keep the number of the segments as low as possible, bases are usually the candidates for suitable segment databases.

It can be easily seen that any minimal set is a base. (On the contrary, a base need not be a minimal set of segments.)

8. Optimality Problem for Heterogeneous Databases

When choosing the set of segments, we try to keep the number of segments as low as possible while simultaneously trying to involve maximum coarticulation in the segments.

A general formulation of the corresponding objective function g(S, C) (S denotes the considered segment database and C the considered corpus) can be expressed as a linear combination of the function b(S, C) (the minimal number of the segment boundaries when concatenating all the elements of the considered corpus C) and the function n(S, C) (the number of the segments belonging to S). To put it more formally,

$$g(S, C) = \alpha b(S, C) + \beta n(S, C)$$

where α and β are non-negative weights assigned to the function according to how much important the criterion expressed by the function g(S, C) or n(S, C) is in relation to the speech synthesis system architecture.

The objective function g(S, C) expresses that we try to choose the segments of the database S in such a way that their number is as low as possible and simultaneously we try to involve maximum coarticulation in the segments (i.e. we try to minimize the number of the segment boundaries).

9. Optimizing Databases of Heterogeneous Segments

With respect to the definition of the objective function that was given above we can postulate our optimization task as follows:

Problem: Find a segment database S that minimizes the function g(S, C).

First, corpus C is supposed to be fixed, i.e. we minimize the function g(S, C) with respect to the choice of the database S only.

Further, we can see that if we put $\alpha = 0$, the problem of minimizing g(S, C) is trivial, because we can simply take S = C obtaining g(S, C) = 0. Of course, for real applications this solution has practically no meaning. Putting $\beta = 0$ we obtain a polynomially solvable problem, which is however also not very interesting for applications.

Real applications force us to consider that the weights are non-zero. Of course, this problem can be solved by trying to evaluate the objective function for all possible set of segments, but this algorithm is exponential and therefore not usable.

Unfortunately, the authors do not know a polynomial algorithm that solves the problem. Even the theoretical question, if there exists a polynomial algorithm that solves the problem (or, whether the problem is NP – complete) is open.

Instead of finding a global minimum of the function g(S, C), we can try to find such a database S, that has the following property: by both adding or removing an

arbitrary segment, the value g(S, C) increases. Clearly, the database that minimizes g(S, C) globally has such a property.

The proposed algorithms are based on the following scheme:

1. Take a compatible database of segments S (first approximation of the database S).
2. Try to add a segment that decreases the objective function.
3. If such a segment does not exist, try to remove a segment that decreases the objective function.
4. If such a segment does not exist, S locally minimizes the objective function and the computation stops, else go to 2.

Let us remark, that the algorithm can be also modified by specifying the order of adding and removing the segments.

10. Some Heuristics for Optimizing Databases of Heterogeneous Segments

In the scheme of the optimization algorithm for optimizing databases of heterogeneous segments that is described in the previous section, some points are not fully determined.

- how to choose the first approximation of the segment database S;
- what segments should be add or removed in the steps of the optimizing process;
- possible modification of the order of adding and removing the segments.

We will briefly discuss this issue and present some heuristics that can be used in this connection.

First, there is the problem of determining the first approximation of the database S. Basically, we have the following natural ways of doing it:

1. Take a segment database that is minimal (or nearly minimal). It can be a compatible homogenous segment database (for instance syllable segments) that will be mostly enlarged in the optimizing process (for instance, by morphemic segments).
2. Take a database that is maximal (for instance, all syllable segments and all morphemic segments). This database will be mostly reduced in the optimizing process.
3. Take a segment database that is especially chosen with respect to the frequencies of appearances of the segments in the given corpus. Another possibility is to take into account the grammar and/or semantic nets – ontologies (Wordnet, EuroWordnet, see (Miller, 1990; Vossen at all, 1998, 1999) to estimate the segments that have a high chance of appearing frequently in the language.

In the first case, where we can assume that the database will be mostly enlarged in the optimizing process, we can modify the optimizing algorithm in this direction, which means that we would prefer the step of adding a new segment.

The second case is the opposite of the first one, and analogously our modification give preference to the step of removing a segment.

These strategies are not directly applicable to the third case, where the strategy should follow from an analysis of the specific situation.

For all the previously discussed cases, a natural strategy of determining the segment that will be added/removed is to take a segment that will cause a maximal decrease of the function b(S, C) (when adding a segment) or minimal increase of the function b(S, C) (when removing a segment). These values can be obtained in polynomial time.

11. Morphemic and Syllabic Segments in Czech

If we consider a highly inflected language like Czech, an interesting task arises: to examine the relations between the syllables and morphemic segments.

In the Laboratory of Natural Language Processing at Faculty of Informatics, Masaryk University, a Czech morphological analyzer Ajka has been developed (Osolsobě, 1990; Sedláček, 1999).

The analyzer is able to recognize almost any Czech input word form and/or to generate all the possible word forms that can be derived from a given input form. The implemented algorithm is, roughly speaking, based on the idea of inflectional paradigms and the respective sets of endings.

First, we are interested in what relations can be observed between morphemic and syllabic segments. First estimation show that about 80% of prefixes in Czech corresponds to the respective syllables.

The estimation for roots is not easy to give since at the present moment we possess only a partial list of roots in Czech (approx. 3300 items), however, at least 70% of the existing roots can be treated as syllables.

On the other hand, the correspondence between syllables and intersegments (infixes and suffixes) is much lower. As far as we know the question of the correspondence between the morpheme segments and syllables has not been studied in a detailed way though there are some obvious links.

In our opinion the morphological segments display a optimization power which follows from their frequencies and regular appearance.

From the analysis, it follows that the use of the morphological segments in combination with syllabic ones is reasonable; we must however use some optimization criteria that will eliminate the non-effective segments. We are currently preparing a corpus suitable for the application of the proposed approach.

12. Applications - Heterogeneous Database of Segments Based on Morphological and Phonological Segmentation for Czech Speech Synthesis

Most of the problems we have mentioned in this paper appeared when creating and optimizing the heterogeneous segment database built for Czech speech synthesizer Demosthenes (Kopeček, 1997, 1998) based on the syllable segments.

To enhance the quality of the synthesized speech we have decided to enrich the segment database by bi-syllables (chosen by statistics) and morphological segments. The next stage is to add some very frequent words and phrases.

Acknowledgment

The authors are grateful to James Edward Thomas for reading a draft of the paper and for valuable comments. This research has been partially supported by Grant LI200027 of Czech Ministry of Education.

References

Deligne S., Bimbot F., 1997. Inference of Variable-Length Linguistic and Acoustic Units by Multigrams. *Speech Communication* 23 (1997): 223-241.

Doddington G., 1997. Syllable Based Speech Processing; *WS97 Project Report*, Research Notes No. 30, J. Hopkins University.

Dutoit T., 1997. An Introduction to Text-to-Speech Synthesis, *Kluwer Acadenic Publishers*, London.

Greenberg S., 1998. Speaking in Shorthand - A Syllable-Centric Perspective for Understanding Pronunciation Variation. *Proceedings of the international workshop Modeling Pronunciation Variation for ASR*: 47-56.

Hunt A., Black A., 1996. Unit Selection in A Concatenative Speech Synthesis System Using a Large Database. *Proceedings of Internationl Conference ICSLP*: 373-376,

Josifovski L., Mihajlov D., Gorgevik D., 1997. Speech Synthesizer Based on Time Domain Syllable Concatenation. *Proceedings SPECOM'97, Cluj-Napoca*: 165-170.

Kopeček I., 1997. Syllable Based Speech Synthesis; *Proceedings of the workshop SPECOM'97, Cluj-Napoca:* 161-165.

Kopeček I, 1998. Speech Synthesis Based on the Composed Syllable Segments. *Proceedings of the First Workshop on Text, Speech and Dialogue - TSD'98*: 259-262.

Kopeček I., 1998. Automatic Segmentation into Syllable Segments. *Proceedings of the Conference LREC'98*: 1275-1279.

Kopeček I., 1999. Speech Recognition and Syllable Segments. *Proceedings of the International Workshop on Text, Speech and Dialogue - TSD'99, Lectures Notes in Artificial Intelligence 1692, Springer-Verlag*: 203-208.

Kopeček I., 2001. Algebraic Models of Speech Segment Databases, *Proceedings of TSD 2001, Springer Verlag, LNAI 2166*: 208-213.

Miller G. at all, 1990. Five Papers on WordNet. *Technical Report, Princeton University. CSL Report 43, Cognitive Science Laboratory.*

Osolsobě K., Pala K., 1990. Stem Dictionary for IBM PC, *Proceedings of the Conference on the Computer Lexicography, Balatonfured*: 125-128.

Sedláček R., 1999. *A Morphological Analyser for Czech*, Diploma Thesis, Brno, Masaryk University.

Vossen P. at all, 1998. The EWN Base Concepts and Top Ontology. *Technical Report, University of Amsterdam, Amsterdam. Deliverables D017, D034, D036, EuroWordNet, LE2-4003, Final Version.*

Vossen P. at all, 1999. Final report on EuroWordNet 2. *Technical Report, University of Amsterdam, Amsterdam [CD ROM]*.