Joint Prosodic and Segmental Unit Selection Speech Synthesis

Robert A. J. Clark, Simon King

CSTR, University of Edinburgh,
Edinburgh, U.K.
robert@cstr.ed.ac.uk , Simon.King@ed.ac.uk

Abstract

We describe a unit selection technique for text-to-speech synthesis which jointly searches the space of possible diphone sequences and the space of possible prosodic unit sequences in order to produce synthetic speech with more natural prosody. We demonstrates that this search, although currently computationally expensive, can achieve improved intonation compared to a baseline in which only the space of possible diphone sequences is searched. We discuss ways in which the search could be made sufficiently efficient for use in a real-time system.

Index Terms: speech synthesis, unit selection, prosody.

1. Introduction

At its best, unit selection speech synthesis can produce synthetic speech with a segmental quality almost indistinguishable from natural speech. As a consequence, the inadequacy of current models of prosody become much more apparent, if the F_0 and segment durations predicted by such models are imposed on the synthetic speech. In some circumstances, the prosody of synthetic speech can be considered of limited importance and one can use simple constraints to ensure the system produces conservative neutral prosody.

There are however many application of speech synthesis where correct, natural sounding prosody is important. Specifically, situations where a synthesiser is expected to convey meaning through the prosody. Language generation systems and any form of dialogue are likely to require the ability to produce emphasis and contrasts, and even pitch contours which express doubt or confirmation. In these situations, neutral prosody is entirely inappropriate, and a system using it will sound bad.

The usual approach to modelling prosody is to predict F_0 and duration, usually in terms of symbols which are subsequently realised in the F_0 contour and segment durations. In diphone synthesis, these F_0 and duration specifications are imposed using signal processing. In unit selection, it is more usual to incorporate them into the specification of target utterance and then to search for units (e.g. diphones or half-phones) that match the target (the closeness of the match being measured by the target cost function).

Festival has, to date, used CARTs [1] and linear regression models [2] for the prediction of prosody. These models are appropriate for diphone synthesis, or perhaps for HMM-based synthesis; in both cases, signal processing leads to a noticeably unnatural quality to the synthetic speech signal. However, when these models are used in a unit selection system, they are clearly the weakest link in the overall quality of the resulting speech; we have found that a system without any prosodic model at all often sounds better.

Recent work [3, 4] has shown that unit selection techniques can be used to search for sequences of prosodic units rather than segmental ones. However, in this previous work, these methods have only been used to find a single target prosodic specification which is then used, via the target cost, as a constraint for the segmental search. This method has the major drawback that the prosodic sequence is chosen independently of the segmental units; there is therefore no guarantee that a suitable sequence of segmental units exits in the database. Additionally, since there are likely to be many acceptable prosodic realisations for any given utterance, the early decision to choose a single target prosodic sequence is far from optimal (and reminiscent of early “phonetic typewriter” approaches to automatic speech recognition (ASR), in which the phone sequence was first decided, and then decoded into words).

2. Approach

We now introduce a unit selection method that adopts a key property from ASR: the principle of delayed decisions, or the propagation of uncertainty. This method *jointly* searches for a sequence of segments and a sequence of prosodic units that together minimise some cost function.

In the following explanation, we will first consider a simpler system in which a single F_0 contour is first predicted by a search for a sequence of prosodic units, and then segmental units are chosen that have similar F_0 values to this predicted contour. After this explanation, we describe how our system performs the two searches jointly.

2.1. Comparison to previous work

The closest previous work is that of [5] which composes a predictive prosodic model with the segmental search by the use of weighted finite state transducers, to allow for a search of more than one fixed prosodic target. Our approach differs in that instead of having a structured model trained on data incorporated dynamically into the search procedure, we rely only on the inherent structure of the unit selection database, plus the prosodic target and join costs, to predict prosody.

An advantage of our technique is that it requires minimal extra preparation of data when building a new voice, and the prosodic model for a given voice is always specific to that voice and does not use data from other speakers. A consequence of this is that the database must be designed to take prosodic coverage into account as well as segmental coverage.

2.2. Predicting an F_0 contour

We do not employ an explicit predictive model of F_0 to produce this contour. Instead, the F_0 contour is found using unit selection techniques by selecting a prosodic unit sequence that minimises a cost function, similarly to [4]. The prosodic units are syllables, from the same speech database used for the segmental units. So, the prosodic "model" is composed of the speech database plus the cost function.

The cost function used to select the optimal sequence of prosodic syllable units ignores the segmental constituents of the units and is composed of target and join sub-costs, as in a conventional segmental search. The join cost is very simple and only measures the F_0 mismatch at prosodic unit concatenation points.

The target cost uses the following component features: the *phrase type* that a syllable is found in, the *position in the phrase* of the syllable, the *position in the word* of the syllable, the presence of *lexical stress* on the syllable, and the van Santen and Hirschberg [6] classifications of the structure of the onset and coda of the syllable.

The *phrase type* takes values such as statement, YN-question, Wh-question, and is intended to allow preselection of units of a particular intonational tune type in an attempt to provide consistency at the intonational tune level. The work in this paper only uses units of the phrase type statement; design and annotation of datasets containing a wider rage of phrase types is work in progress.

Both the *position in the phrase* and *position in the word* features take one of six values each to represents the position of the syllable. The values (illustrated in figure 1 for *position in the word*) are designed to determine whether a syllable is initial, medial or final in the larger local utterance structure. The six possible values for these two features are:

IF (Initial and Final) A syllable is the only syllable in the word/phrase

IP (Initial and Penultimate) The syllable is the first of only two syllables in the word/phrase.

I (Initial) The syllable is the first syllable of three or more syllables in the word/phrase

FS (Final and Second) The syllable is the final syllable of a two syllables word/phrase.

F (Final) The syllable is the final syllable of a word/phrase of three or more syllables.

M The syllable is medial in a word/phrase of three or more syllables.

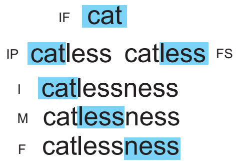

Figure 1: Examples of syllables for each of the values of the *position in the word* feature.

The *lexical stress* feature is binary, and the van Santen and Hirschberg features take the values: -V for unvoiced, +V-S for voiced but no sonorants, and +S for sonorants.

For each target syllable we currently preselect as suitable candidates only syllables which have the correct *phrase type*, *position in phrase* and *position in word* features. This is done to ensure a level of consistency in the intonation tune at the utterance level. These features are then left out of the target cost.

For example the first syllable unit in the utterance "Take the black one." would be have the features statement;I;IF (the syllable is from an utterance with a statement contour, is the first syllable in a phrase containing more than three syllables and is in a word consisting of a single syllable) and only syllables in the database matching this description would be considered as suitable prosodic candidates for the syllable 'take'

2.3. Using the resulting F_0 contour to guide the segmental search

For simplicity of explanation, let us continue to assume that the prosodic search has predicted a single target F_0 contour. The segmental search now simply proceeds in the usual way [7], with the target cost incorporating a component that measures the difference between each candidate unit's F_0 and the target F_0. Note that the F_0 contour found by the prosodic search is *not* imposed on the segmental units, it is merely a constraint guiding the segmental unit search.

2.4. Joint search

In our system, the prosodic search is not carried out first, but is done jointly with the segmental search. This is equivalent to first finding the N best prosodic unit sequences (for very large N) then, for each of them, finding the best segmental unit sequence, and combining the prosodic and segmental costs to make the final decision as to the best segmental sequence.

Rather than constructing a very large finite automaton (FSA) from the product of the two FSAs for the segmental and prosodic candidate units, we implement an equivalent algorithm which we describe as a tied search of the two FSAs.

In addition to target and join costs for each search space, a *tie cost* is introduced. The tie cost replaces the F_0 component of the segmental target cost and compares a segmental candidate and a prosodic candidate in terms of F_0.

An updated version of the Festival's [8] Multisyn [7] engine was used to implement the proposed method. From a single voice database, two *inventories* are indexed. This indexing specifies how the data is to be used in each part of search. The database is first indexed as a set of diphones, to produce and inventory for use in the segmental search; it is then indexed as a set of syllable sized units to be used as the inventory for the prosodic search.

Before the search is performed for a target utterance, the language processing stage of text-to-speech synthesis is carried using the default Festival front end, resulting in a heterogeneous relation graph structure [9] representing the utterance. From this structure, *two* target sequences are created. A segmental target sequence of diphones is created from the phone sequence of the target utterance, and a prosodic target sequences of syllables is created from the syllable sequence of the target utterance. Two time-aligned finite state networks are then constructed from the two sets (prosodic and segmental) of candidate units retrieved from the database.

Each of the states in the prosodic network, which correspond to syllable-sized units, are assigned a time index to match the time index of the first phone in the rhyme of the syllable. As the segmental units represent diphones rather than phones, the match is actually made with the segmental units whose left half is the first phone in the syllable rhyme. An example of this alignment is shown in figure 2

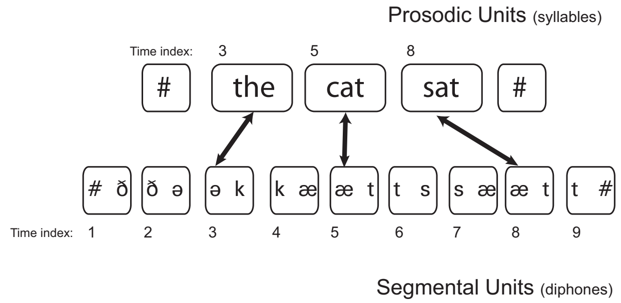

Figure 2: Alignment between prosodic syllable units and segmental diphone units. Prosodic unit candidates receive a time index corresponding to the first diphone whose left edge is part of the syllable rhyme.

The search for units is implemented as token passing [10]. Tokens are associated with a pair of states: one from each of the two FSAs. As a token is duplicated and passed on to the connecting states in one network, the copies of the token keep their association with the original token's state in the other network. As propagation occurs, tokens maintain a record of their path history through both networks and the cumulative cost of the path so far.

Propagation of tokens occurs in a time synchronous fashion. As the time variable is incremented, any tokens associated with a state in the segment network with a time index of less than this value are propagated forward one state in that network. Once there are no tokens in the segment network at times less than the current time, tokens are propagated through the prosodic network in a similar manner. Note that the units in the prosodic network are syllables, so are generally of longer duration than the segmental units.

As a token enters a new state in either network, the local (either prosodic or segmental) target and join costs are added to the cumulative cost of that token. In addition, if a token enters a state in one network with the same time index as the node that the token is associated with in the other network, then the tie cost is calculated and added to the cumulative cost of that token.

2.5. Dynamic programming and pruning

Since the search space is very large (the square of the usual segmental unit selection search space), an efficient search strategy is of paramount importance. The first technique considered is dynamic programming, in which tokens which are in the same pair of states can be directly compared and only the best (lowest cost) one retained. This is known as "Viterbi search" in ASR and leads to much faster search with no reduction in accuracy.

In practice, we have found that Viterbi search alone is not sufficient to make the search computationally feasible, so we currently allow the comparison of two or more tokens that are in the same state *in only one of the machines*. Unlike Viterbi search, this technique may lead to a reduction in accuracy.

To further reduce the computational cost of the search, beam pruning is employed performed at two different points in the search. An initial pruning occurs as the the networks are first constructed. Target costs are pre-calculated for each candidate unit and a beam is used to prune away high cost candidates. The second beam pruning occurs during token propagation, as in ASR.

3. Performance issues and system analysis

The current implementation is designed to demonstrate that the proposed method can produce improved prosody. For computational reasons, we currently only use a voice database of limited size (from the ARCTIC [11] datasets).

3.1. Setting Parameters

The search depends on a number of parameters which require tuning. There are now five sub-components making up the cost of a chosen segmental unit sequence: prosodic and segmental target costs, prosodic and segmental join costs and the tie cost.

Each of these costs ranges from 0 to 1 and they are combined in a weighted sum to form the overall cost. Initial results suggest the tie cost needs to be weighted quite heavily to ensure that synchronisation between the two candidate paths outweighs other selection criteria. With the target and join costs all weighted at unity and a tie cost with a weight of five, the system produces reasonable results.

To compare the proposed approach to a baseline system without the prosodic search, a series of sentences generated with both systems were compared. It was found that, in general, the segments chosen by the proposed method had exactly the same segmental target costs as the corresponding units chosen by the baseline system[1] but that different segmental units were being chosen for a given target the majority of the time. In other words, including the prosodic search does not lead to worse selections of segmental units (as measured by the segmental target cost).In contrast, the segmental join costs were generally different and, on average, slightly higher for the full prosodic search than for the segmental-only search. This result is important because it shows that candidate unit sequences are available from the inventory that have better prosody without having worse segmental quality.

4. Discussion

As the ARCTIC datasets only provide a basic level of diphone coverage, with no specific account of prosodic coverage, we expected the resulting synthetic speech to have reduced segmental quality in return for improved prosodic quality. However, it appears that the alternative segmental unit sequence chosen when the prosodic search is included is generally as good as the sequence chosen when only the segmental search is performed.

The main drawback of the current system is that the small database does not really provide sufficient prosodic coverage to generate anything other than statement intonation. However as the system is designed to partition the prosodic data based on tune type, there is no reason to think that system could not generate other tunes if the underlying database was sufficiently rich.

To further reduce the computational cost of the method, a variety of standard techniques are available from ASR, including multi-pass search in which an initial N-best search of the prosodic and segmental spaces is performed using simpler models (e.g. with no join cost).

One situation where the search space becomes very large is when, for two adjacent target units, a large number of candidates are found in the inventory. It may be possible to resolve this problem by applying pruning during token propagation, rather than

[1] Note that the target cost is essentially quantised (it has only a limited set of possible values within the interval 0 to 1) because it is composed of a small number of discrete features

only at the end of each time cycle.

4.1. Specification of Intonation

The method we have described does not yet allow direct control over intonation, either symbolically or acoustically. This is probably acceptable the statement intonation generated so far, but the main motivation behind our development of the method is to allow prosody to carry specific meanings. Once a larger database is used, which contains more variation in prosody, we predict that performance will degrade unless intonation is symbolically represented. The categorisation of the prosodic units into subsets of individual phrase types, so that only prosodic units from the phrase type of the target are chosen, should allow the system to perform reasonably well for different types of pitch contour, but this will not be enough to deal with contrastive stress, for example. In future, we plan to use the method with a voice database designed specifically for prosodic richness [12]. Precisely how intonation should be represented, in order to facilitate the control required to realise contrastive stress and other phenomena, is still open to question. The main requirements of such a representation include that it should be a simple representation with which the the database can be automatically labelled, given the speech and the text of each utterance in the database.

5. Conclusions

We have demonstrated that a parallel search of the segmental candidate unit space and a prosodic unit space is feasible, at least with a small database. The method produces improved synthetic speech with more natural pitch contours, without reduction in segmental quality. This technique is computationally expensive, but we believe we can use ASR-like techniques to provide a real-time solution.

6. Acknowledgements

This work was funded by Scottish Enterprise through the Edinburgh-Stanford Link project *Expressive Intonation for Unit Selection Speech Synthesis*. We would also like to thank Mark Steadman, Korin Richmond and Volker Strom who have contributed to aspects of this work.

7. References

[1] L. Breiman, J. H. Friedman, J. A. Olshen, and C. J. Stone, *Classification and Regression Trees*, London : Chapman & Hall, 1993.

[2] A. Black and A. Hunt, "Generating f0 contours from ToBI labels using linear regression," in *Proc. ICSLP 96*, Philadelphia, Penn., 1996.

[3] Joram Meron, "Prosodic unit selection using an imitation speech database," in *4th ISCA Workshop on Speech synthesis*, Perthshire, Scotland, 2001, pp. 53–57.

[4] A. Raux and A Black, "Unit selection approach to F0 modeling and its application to emphasis," in *ASRU 2003*, St Thomas, US Virgin Is, 2003.

[5] I. Bulyko and M. Ostendorf, "Joint prosody prediction and unit selection for concatenative speech synthesis," in *Proc. of ICASSP, 2001, Salt Lake City, USA*, 2001.

[6] J.P.H. van Santen and J. Hirschberg, "Segmental effects on timing and height of pitch contours," in *ICSLP*, Yokohama, 1994, vol. 2, pp. 719–722.

[7] Robert A.J. Clark, Korin Richmond, and Simon King, "Festival 2 – build your own general purpose unit selection speech synthesiser.," in *5th ISCA Speech Synthesis Workshop*, Pittsburgh, PA, 2004, pp. 173–178.

[8] Paul Taylor, Alan Black, and Richard Caley, "The architecture of the Festival speech synthesis system," in *Proc. The Third ESCA Workshop in Speech Synthesis*, 1998, pp. 147–151.

[9] P Taylor, R Caley, and A Black, "Heterogeneous relation graphs as a mechanism for representing linguistic information," *Speech Communication*, vol. 33, pp. 153–174, 2001.

[10] S. J. Young, N. H. Russell, and J. H. S. Thornton, "Token passing: A simple conceptual model for connected speech recognition systems," 1989.

[11] J. Kominek and A. Black, "The CMU ARCTIC speech databases," in *5th ISCA Speech Synthesis Workshop*, Pittsburgh, PA, 2004, pp. 223–224.

[12] Volker Strom, Robert A. J. Clark, and Simon King, "Expressive prosody for unit-selection speech synthesis," in *Proc Interspeech 2006, Pittsburgh, USA*, 2006.