

Scottish Gaelic Text-to-Speech Synthesis

Jeff Berry

Student Showcase
March 7, 2008

Approaches to Synthesis

Rule-based Synthesis

- Formant Synthesis
- Articulatory Synthesis

Approaches to Synthesis

Rule-based Synthesis

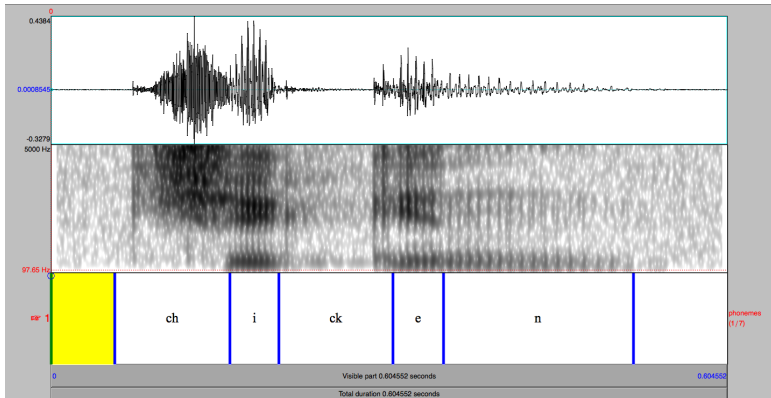
- Formant Synthesis
- Articulatory Synthesis

Concatenative Synthesis

- Diphone Synthesis
- Unit Selection Synthesis
- HMM-based Synthesis

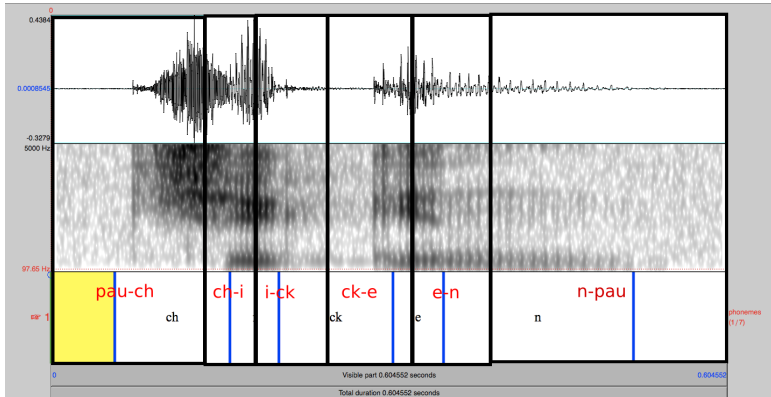
Diphone Synthesis

A cut-and-paste approach



Diphone Synthesis

A cut-and-paste approach



Diphone Synthesis – Pros & Cons

Pros

- More natural sounding than rule-based synthesis
- Small diphone database – easy to store
- Not difficult to make new voices

Diphone Synthesis – Pros & Cons

Pros

- More natural sounding than rule-based synthesis
- Small diphone database – easy to store
- Not difficult to make new voices

Cons

- Still not very natural sounding
- Not flexible

Unit Selection Synthesis

- Based on a large database of a single speaker
- Units can include words, syllables, demisyllables, phonemes
- Text input is converted into a set of targets
- Units are selected based on their similarity to the targets:

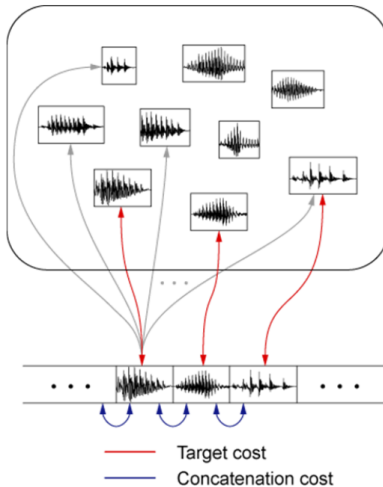
Target cost between candidate unit u and intended output t

$$C^t(t_i, u_i) = \sum_{j=1}^p w_j^t C_j^t(t_i, u_i)$$

Concatenation cost

$$C^c(u_{i-1}, u_i) = \sum_{k=1}^q w_k^c C_k^c(u_{i-1}, u_i)$$

Unit Selection Synthesis



Unit Selection Synthesis – Pros & Cons

Pros

- Most natural sounding because of less signal processing
- Nearly perfect performance on limited domains
- Widely used

Unit Selection Synthesis – Pros & Cons

Pros

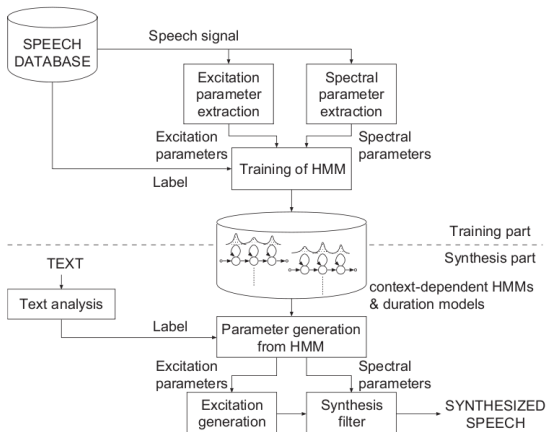
- Most natural sounding because of less signal processing
- Nearly perfect performance on limited domains
- Widely used

Cons

- Requires large databases – 10+ hours of recorded, labelled speech
- Expensive

HMM-based Synthesis

Overview of a HMM-based system (Black et al. 2007)



HMM-based Synthesis – Pros & Cons

Pros

- Once trained, has a very small footprint
- Training data does not have to be as large as Unit Selection
- Versatility – easy to change the quality of the voice
- Should be easily adaptable to other languages

HMM-based Synthesis – Pros & Cons

Pros

- Once trained, has a very small footprint
- Training data does not have to be as large as Unit Selection
- Versatility – easy to change the quality of the voice
- Should be easily adaptable to other languages

Cons

- Does not have the same quality as Unit selection
- Still being developed

Demos

Online demos for all three concatenative approaches using the same voices:

http:

`//www.cstr.ed.ac.uk/projects/festival/morevoices.html`

Background



- SG is the focus of an ongoing project here at UA to document the language
- SG is severely endangered, with no monolingual speakers
- SG has many interesting grammatical properties

Phonology

- SG has 9 vowels, 10 diphthongs, 32 consonants (Gillies 1992)
- SG has phonetics settings at the syllable level or higher:
 - neutral
 - palatalized
 - velarized
 - nasalized
- Variation among dialects

Question: How many diphones do we need?

Previous Attempts

- Murray & Black 1993 for SG – built out of English sounds
- Williams 1994 for Welsh – 2800 diphones
- Wolters 1997 for Bayble dialect of SG – about 900 diphones

Phase 1: uses Wolters diphone set

Building the Voice

Recording

- Native speaker read a list of 500 Gaelic words
- We couldn't use nonwords because of possible influence of English
- Use EGG for pitchmarking

Building the Voice

Recording

- Native speaker read a list of 500 Gaelic words
- We couldn't use nonwords because of possible influence of English
- Use EGG for pitchmarking

Labelling

- Each word must be labelled
- Hand labelling is labor intensive
- Use hand labelled samples to train HMMs for automatic labelling

Building the Voice

Festvox

- Festvox uses labels and EGG pitchmarks to create diphone database
- Dictionary must be created to perform grapheme to phoneme conversion along with a machine readable phonetic alphabet:

```
(lex.add.entry '(" glaschu" nn (((g l a s) 1) ((x ub) 0))))
```

```
(lex.add.entry '(" gainne" nn (((g a) 1) ((nj E)0))))
```

```
(lex.add.entry '(" thraigh" nn (((thr a G) 1))))
```

```
(lex.add.entry '(" thu" nn (((h uub) 1))))
```

...

- A set of LTS rules must be constructed to handle words not in the dictionary

Demo

- Prototype of Phase 1 can be found at:
<http://yllab.dyndns.org/~group2>
- I'm still in the process of finishing the labelling task, so this only says a few words:

gainne	thraigh
thu	thubhairt
thuir	tim
tighinn	tilg
tilleadh	timcheall
tinn	

- These words illustrate some issues with the orthography

Current work

Phase 2 Diphone

- Make new recordings of a speaker from Skye
- The Phase 1 synthesizer will be used to generate prompts for the new recordings – these can be nonsense words
- Revised diphone set
- The new system will have better sound quality, faster to build

HMM-based

- Record a set of sentences to train an HMM-based system using the HTS extension for the HTK software package

The End

Thank You!

References available upon request jjberry@email.arizona.edu