**ABSTRACT**

Concatenative speech synthesis requires a speech database comprising of speech units. For a particular language a unique speech database will comprise all the various speech units of the language and will vary for different languages. Two crucial tasks for the formation of speech databases will be to determine the phonetic alphabet for the target language and the ability to phonetically transcribe speech from the target language into a formal orthographic system.

This project examines a statistical DSP method for preparing concatenative speech synthesis speech databases and proposes automatic correlational means for determining the phonetic alphabet speech units as a reverse engineering method of automatically transcribing speech from a target language.

**Introduction**

Text to Speech (TTS) by concatenative model requires speech segments to be concatenated to form audible speech in a text-to-speech implementation. In order to achieve this, speech segments are created from pre-existing recordings. The creation of this segment database is usually strenuous, time consuming and manual.

**Aim**

The aim of this research will be to discover through statistical autocorrelation a method to automatically segment speech and discover the necessary segments or phonemes using DSP, NLP and machine learning methods.  For this initial implementation only DSP will be applied.

**Objectives**

* Generation of diphone/triphone-based segments
* determination of unique di-phones and tri-phones
* determination of phonemes
* measure of correctness of phoneme results
* measure of closeness of diphones and phonemes to one another
* comparison of results with alternative means of phonetic alignment.

### IMPLEMENTATION PREREQUISITE KNOLWEDGE

* cross-correlation
* speech acoustic analysis

### ADVANCED METHODS

* test driven and integration approach
* analysis of speech segmentation tools
* comparison of speech dsp methods
* analysis of machine learning techniques

**Software Specification**

The software will be implemented at two levels.  A lower level API and a higher level web interface. The web interface will be implemented typically as a web application and optionally as a web service.

### API target platforms

* Matlab
* Java
* C++
* python

### Web target platforms

* PHP
* Django
* Asp MVC
* Rails???

The input data upload will be within  restricted size range of 4-40hrs of recording i.e. 500mb - 5gb in order to get a large enough corpus.

### API SPECIFICATION

#### Inputs:

* Digital Continuous speech. Format: wav, mp3

#### Output:

* Speech database
  + Tables:
    1. corpora file
    2. auto-sequenced speech unit
    3. auto correlation results
  + Fields
    1. corpora file table
       1. file\_id
       2. file\_name
       3. file\_path\_relative
       4. file\_path\_absolute
    2. speech unit table
       1. unique\_id
       2. sequence\_id
       3. sequence\_position
       4. sequence\_data
       5. file\_id
    3. correlation results table
       1. unit\_id
       2. relation\_id
       3. correlation\_result

### WEB SPECIFICATION

* Input -> ability to upload mp3/wav based on specification restrictions
* Output
  + Ability to play initial input
  + Ability to play initial segment and correlates
  + visualization of a segment and correlational data

### ADVANCED OUTPUTS

* Sentence Segmentation
* Phrase/clause segmentation
* word segmentation
* Phoneme to sentence/phrase/word matching.
* Play word/phrase/clause/sentence within which phoneme occurred

**SOFTWARE PROJECT MANAGEMENT NOTES**

The project is to be completed in three phases

1. System Pragmatics
   1. User Interface
   2. I/O management
2. Ability to segment corpora
3. Auto correlation implementation

### SOFTWARE ROLE MAPPING

1. UML modeling - group leader
2. Test document preparation - group leader
3. Documentation – ISM
4. Database Design – ISM
5. Data Generation – nw tech/internet security
6. Development - web tech/adv msc
7. Unit Testing - web tech/adv msc
8. Integration testing - nw technology
9. Authentication – web tech/int security
10. Load testing - internet security/nw technology

**REFERENCES**

McLoughlin, I. (2009). *Applied speech and audio processing: with Matlab examples*. Cambridge University Press.

Gaydecki, P. (2004). *Foundations of Digital Signal Processing: theory, algorithms and hardware design* (Vol. 15). Iet.