DESIGN AND IMPLEMENTATION OF PDF TO AUDIO SYSTEM

DESIGN AND IMPLEMENTATION OF PDF TO AUDIO SYSTEM

BY

OKWUIWE ALPHONSUS JONAS ICT/6252040312

BEING A PROJECT PRESENTED TO THE DEPARTMENT OF COMPUTER SCIENCE, SCHOOL OF INFORMATION AND COMMUNICATION TECHNOLOGY , AUCHI POLYTECHNIC, AUCHI.

IN PARTIAL FULFILLMENT OF THE REQUIREMENT FOR THE AWARD OF HIGHER NATIONAL DIPLOMA (HND) IN COMPUTER SCIENCE

JUNE, 2023

DECLARATION

I hereby declare that this report entitled "Design and Implementation of PDF to Audio System" is the product of my personal research work carried out under the supervision of DR. (MRS.) ODUNTAN E.B.

OKWUIWE ALPHONSUS JONAS Date

Project Student

CERTIFICATION

This project work titled "Design and Implementation of PDF to Audio System" has been assessed and approved to meet the requirement for the award of Higher National Diploma (HND) in the Department of Computer Science, Auchi Polytechnic, Auchi.

DR. (MRS.) ODUNTAN E.B. Date

Project Supervisor

Date

Head of Department

Date Dean, School Information and Communication Tech.

DEDICATION

This project is dedicated to the KING OF KINGS, my Mom MRS. H. OKWUIWE for her continuous support and prayers leading to this moment.

.

.

ACKNOWLEDGEMENT

First and foremost, my profound gratitude goes to the Almighty God for His infinite mercy and love over me, who in his loving kindness brings this programme to a successful completion.

My sincere gratitude goes to my mother MRS. H. OKWUIWE for her love, endless support, prayers and encouragement throughout the course of this programme.

My humble appreciation goes to my project supervisor DR. (MRS.) ODUNTAN E.B. for her patience, time and guidance in carrying out this research.

Special appreciation goes to my ELDER SIBLINGS and to my amiable Stan M.s.c AKINTAN E. SHINA

Appreciation to my friends and family since day one in Auchi polytechnic, members of

C.C.C. AUCHI PARISH 1, the HART T. FAMILY, PROPHETESS DR. F. AKINTAN (JP),

Chief G.I. Irorere (The Aigbedion of Oba of Benin Kingdom).

The list of all those who contributed to the successful completion of this programme is endless. Everyone who has contributed in one way or the other is greatly appreciated and my prayer is that God in his infinite mercy meet each and every one at the point of their needs in Jesus name. Amen.

ABSTARCT

Text-to-speech and related read audio tools are being widely implemented in an attempt to assist students' reading comprehension skills. PDF to audio system is a screen reader application designed and constructed for effective audio communication system. PDF's were designed to present and exchange documents reliably, PDF are an open standard document format used globally, maintained by the International Organization for Standardization (ISO). The document format is one of the most convenient method for electronic communication, and also for exchange of information. Hence, there is a need to make it more accessible to readers on screen through audio. PDF documents are designed and structured to contain links and buttons, form fields, audio or sounds, video, and business logic. The PDF to audio system will power text on screens to read aloud (speak) with support for many languages. In this research, the researcher has proposed to design and implement a PDF to audio system developed using HTML, CSS and PHP.

CHAPTER ONE INTRODUCTION

BACKGROUND OF THE STUDY

In 1991, Adobe co-founder Dr. John Warnock propelled the paper- to-advanced digital revolution with an idea he called, The Camelot Project. The objective was to empower the growing digital users the ability to capture documents from agony application, send electronic renditions of these documents anywhere, and view and print them on any machine. By 1992, Camelot had developed into PDF

These abstract classes are related to surface correlates by complicated integrative processes, the nature of which has only recently been studied. Finally, the influence of computer science, which can be regarded as the study of complex systems, is described and the requirements for aggressive research facilities needed for further progress are developed. Allen (1985).

Text-to-speech (TTS) synthesis is one of the rapidly emerging areas of computer-to-human interaction technology. Human-like speech is replicated by the computer with the introduction of input text which is usually very natural. Real-life applications of TTS synthesis technique make users task hassle-free. For example, reading book for the visually impaired people, paying electricity bill through automated call-centre, announcing train information at the railway station, etc. Jaiswal et al (2021).

Today, it is a document format trusted by businesses/organizations around the globe. PDF, or Portable Document Format, was the first file format of its kind to have the ability to store and offer content and images in a way that would protect the formatting of the original document. Regardless of which software, hardware or platform it is being viewed on. Inspired by the idea of digitizing the contents of the Library of Congress, Warnock's Team Camelot expanded to include developers with a diverse range of coding skills to build the platform- agnostic file format. Not only would the file format need to be compatible with the most popular platforms, they also needed to have developers that had experience working with printer drivers. It was this additional element that eventually helped to boost PDF's popularity around the globe - users everywhere could choose to 'Save

as PDF' instead of printing out the document. The team created the initial version of the format in roughly a year, but the public launch would wait until June 1993 - in tandem with Adobe Acrobat Version

1.0. Acrobat was released to huge fanfare, and as the first program with the ability to read the file format, it's adoption was crucial to the initial success of PDF. Inspired by the idea of digitizing the contents of the Library of Congress, Warnock's Team Camelot expanded to include developers with a diverse range of coding skills to build the platform-agnostic file format. Not only would the file format need to be compatible with the most popular platforms, they also needed to have developers that had experience working with printer drivers. It was this additional element that eventually helped to boost PDF's popularity around the globe - users everywhere could choose to 'Save as PDF' instead of printing out the document.

STATEMENT OF THE PROBLEM

With PDF being the most used document format globally, there is a need to converts text in PDF formats into Audio signal. These can be utilized for various purposes, e.g. in the educational system, car navigation, announcements in railway stations, response services in telecommunications, and e-mail reading. Furthermore, people with vision disabilities can't view or read PDF files and this is a major setback. This research addresses the problems in converting PDF text into speech. One is how to improve the naturalness of synthetic speech in PDF-based text into an Audio system.

AIMS AND OBJECTIVE OF THE STUDY

This research aims at the Design and Implementation of a PDF to Audio System to aid accessibility and easy text to voice assimilation of documents in PDF format.

The following are the objectives of the study:

Develop a system that will convert PDF text to audio for easy assimilation of document.

A system to easily detect a PDF file and convert to audio.

To design a system that will assist people with reading disabilities to easily convert PDF text to audio files.

To design and implement a system that will assist students'

reading comprehension skills.

MOTIVATION OF THE STUDY

Presenting reading material orally in addition to a traditional paper presentation format increases the inability of users to be able to decode reading material, and therefore, has the potential to prevent students with reading disabilities better comprehend written texts. There are several different technologies for presenting oral materials (e.g., text-to-speech, reading pens, audiobooks). Already text were accessible orally through books-on-tape and through human readers. There is a need to develop and implement a text-to-speech system that will be used widely in the educational settings from elementary school through universities. With the implementation of the PDF to Audio system, they will be an improved effects of text-to- speech and related tools for oral presentation of material on reading comprehension for students with reading disabilities.

SCOPE OF THE STUDY

The scope of the research is focused on implementing a PDF to Audio system to improve the usage of PDF documents and in order to achieve a more flexible audio speech system.

LIMITATION OF THE STUDY

During the development of the research study, they following limitations were encountered;

Limited research material available at the school library and on the internet.

High cost of setting up the system as it requires a high programming language.

Combining school work and carrying out the research.

DEFINITION OF TERMS

PDF – Portable Document Format.

Audio - Audio most commonly refers to sound, as it is transmitted in signal form.

Speech- the expression of or the ability to express thoughts and feelings by articulate sounds.

Oral - relating to the transmission of information or literature by word of mouth rather than in writing.

Reading Disabilities - a condition in which a sufferer displays difficulty reading.

File - a collection of data treated as a unit by a computer.

Document- A document is a written, drawn, presented, or memorialized representation of thought.

CHAPTER TWO

LITERATURE REVIEW AND CONCEPTUAL FRAMEWORK

2.0 OVERVIEW OF PDf

The Portable Document Format (PDF) is a developed by in the 1990s to present , including text formatting and images, in a manner independent of ,

, and . Based on the language, each PDF file encapsulates a complete description of a fixed-layout flat document, including the text, , , and other information needed to display it. PDF was standardized as ISO 32000 in 2008, and no longer requires any royalties for its implementation.

The use of the Time-Domain Pitch Synchronous Overlap-Add (TD-PSOLA) algorithm in a Text-To-Speech synthesizer is reviewed. Its drawbacks are underlined and three conditions on the speech database are examined. In order to satisfy them, a previously described high quality resynthesis process is developed and enhanced, which makes use of the well-known Multi-Band Excited (MBE) model. An important by-product of this operation is that optimal Pitch Marking turns out to be automatic. A temporal interpolation block is finally added. The resulting Multi-Band Resynthesis Pitch Synchronous Overlap Add (MBR-PSOLA) synthesis algorithm supports spectral interpolation between voiced parts of segments, with virtually no increase in complexity. It provides the basis of a high-quality Text-To-Speech (TTS) synthesizer. Dutoit. (1993).

PDF files may contain a variety of content besides flat text and graphics including logical structuring elements, interactive elements such as annotations and form-fields, layers, (including video content) and three dimensional objects using or , and various other data formats. The PDF specification also provides for encryption and , file attachments and metadata to enable workflows requiring these features.

2.1 PDF TEXT TO SPEECH

A Text-To-Speech (TTS) synthesizer is a computer-based system that should be able to read any text aloud, whether it was directly introduced in the computer by an operator or scanned and submitted to an Optical Character Recognition (OCR) system. Let us try to be clear. There is a fundamental difference between the system we are about to discuss here and any other talking machine (as a cassette- player for example) in the sense that we are interested in the automatic production of new sentences. This definition still needs some refinements. Systems that simply concatenate isolated words or parts of sentences, denoted as Voice Response Systems, are only applicable when a limited vocabulary is required (typically a few one hundreds of words), and when the sentences to be pronounced respect a very restricted structure, as is the case for the announcement of arrivals in train stations for instance. In the context of TTS synthesis, it is impossible (and luckily useless) to record and store all the words of the language. It is thus more suitable to define Text-To-Speech as the automatic production of speech, through a grapheme-to-phoneme transcription of the sentences to utter.

At first sight, this task does not look too hard to perform. After all, is not the human being potentially able to correctly pronounce an unknown sentence, even from his childhood ? We all have, mainly unconsciously, a deep knowledge of the reading rules of our mother tongue. They were transmitted to us, in a simplified form, at primary school, and we improved them year after year. However, it would be a bold claim indeed to say that it is only a short step before the computer is likely to equal the human being in that respect. Despite

the present state of our knowledge and techniques and the progress recently accomplished in the fields of Signal Processing and Artificial Intelligence, we would have to express some reservations. As a matter of fact, the reading process draws from the furthest depths, often unthought of, of the human intelligence.

Automatic Reading : what for ?

Each and every synthesizer is the result of a particular and original imitation of the human reading capability, submitted to technological and imaginative constraints that are characteristic of the time of its creation. The concept of high quality TTS synthesis appeared in the mid eighties, as a result of important developments in speech synthesis and natural language processing techniques, mostly due to the emergence of new technologies (Digital Signal and Logical Inference Processors). It is now a must for the speech products family expansion.

Potential applications of High Quality TTS Systems are indeed numerous. Here are some examples :

Telecommunications services. TTS systems make it possible to access textual information over the telephone. Knowing that about 70 % of the telephone calls actually require very little interactivity, such a prospect is worth being considered. Texts might range from simple messages, such as local cultural events not to miss (cinemas, theatres,...), to huge databases which can hardly be read and stored as digitized speech. Queries to such information retrieval systems could be put

through the user's voice (with the help of a speech recognizer), or through the telephone keyboard (with DTMF systems). One could even imagine that our (artificially) intelligent machines could speed up the query when needed, by providing lists of keywords, or even summaries. In this connection, AT&T has recently organized a series of consumer tests for some promising telephone services [Levinson et al. 1993].

They include : Who's Calling (get the spoken name of your caller before being connected and hang up to avoid the call), Integrated Messaging (have your electronic mail or facsimiles being automatically read over the telephone), Telephone Relay Service (have a telephone conversation with speech or hearing impaired persons thanks to ad hoc text-to-voice and voice-to- text conversion), and Automated Caller Name and Address (a computerized version of the "reverse directory"). These applications have proved acceptable, and even popular, provided the intelligibility of the synthetic utterances was high enough. Naturalness was not a major issue in most cases.

Language education. High Quality TTS synthesis can be coupled with a Computer Aided Learning system, and provide a helpful tool to learn a new language. To our knowledge, this has not been done yet, given the relatively poor quality available with commercial systems, as opposed to the critical requirements of such tasks.

Aid to handicapped persons. Voice handicaps originate in mental or motor/sensation disorders. Machines can be an invaluable support in the latter case : with the help of an especially designed keyboard and a fast sentence assembling program, synthetic speech can be produced in a few seconds to remedy these impediments. Astro-physician Stephen Hawking gives all his lectures in this way. The aforementioned Telephone Relay Service is another example. Blind people also widely benefit from TTS systems, when coupled with Optical Recognition Systems (OCR), which give them access to written information. The market for speech synthesis for blind users of personal computers will soon be invaded by mass-market synthesizers bundled with sound cards. DECtalk (TM) is already available with the latest SoundBlaster (TM) cards now, although not yet in a form useful for blind people.

Talking books and toys. The toy market has already been touched by speech synthesis. Many speaking toys have appeared, under the impulse of the innovative 'Magic Spell' from Texas Instruments. The poor quality available inevitably restrains the educational ambition of such products. High Quality synthesis at affordable prices might well change this.

Vocal Monitoring. In some cases, oral information is more efficient than written messages. The appeal is stronger, while the attention may still focus on other visual sources of information. Hence the idea of incorporating speech synthesizers in measurement or control systems.

Multimedia, man-machine communication. In the long run, the development of high quality TTS systems is a necessary step (as is the enhancement of speech recognizers) towards more complete means of communication between men and computers. Multimedia is a first but promising move in this direction.

Fundamental and applied research. TTS synthesizers possess a very peculiar feature which makes them wonderful laboratory tools for linguists : they are completely under control, so that repeated experiences provide identical results (as is hardly the case with human beings). Consequently, they allow to investigate the efficiency of intonative and rhythmic models. A particular type of TTS systems, which are based on a description of the vocal tract through its resonant frequencies (its formants) and denoted as formant synthesizers, has also been extensively used by phoneticians to study speech in terms of acoustical rules. In this manner, for instance, articulatory constraints have been enlightened and formally described.

How does a machine read ?

From now on, it should be clear that a reading machine would hardly adopt a processing scheme as the one naturally taken up by humans, whether it was for language analysis or for speech production itself. Vocal sounds are inherently governed by the partial differential equations of fluid mechanics, applied in a dynamic case since our lung pressure, glottis tension, and vocal and nasal tracts

configuration evolve with time. These are controlled by our cortex, which takes advantage of the power of its parallel structure to extract the essence of the text read : its meaning. Even though, in the current state of the engineering art, building a Text-To-Speech synthesizer on such intricate models is almost scientifically conceivable (intensive research on articulatory synthesis, neural networks, and semantic analysis give evidence of it), it would result anyway in a machine with a very high degree of (possibly avoidable) complexity, which is not always compatible with economical criteria. After all, flies do not flap their wings !

Figure 1 introduces the functional diagram of a very general TTS synthesizer. As for human reading, it comprises a Natural Language Processing module (NLP), capable of producing a phonetic transcription of the text read, together with the desired intonation and rhythm (often termed as prosody), and a Digital Signal Processing module (DSP), which transforms the symbolic information it receives into speech. But the formalisms and algorithms applied often manage, thanks to a judicious use of mathematical and linguistic knowledge of developers, to short-circuit certain processing steps. This is occasionally achieved at the expense of some restrictions on the text to pronounce, or results in some reduction of the "emotional dynamics" of the synthetic voice (at least in comparison with human performances), but it generally allows to solve the problem in real time with limited memory requirements.

Figure 2.1 A simple but general functional diagram of a TTS system.

The NLP component

Figure 2 introduces the skeleton of a general NLP module for TTS purposes. One immediately notices that, in addition with the expected letter-to-sound and prosody generation blocks, it comprises a morpho-syntactic analyzer, underlying the need for some syntactic processing in a high quality Text-To-Speech system. Indeed, being able to reduce a given sentence into something like the sequence of its parts-of-speech, and to further describe it in the form of a syntax tree, which unveils its internal structure, is required for at least two reasons :

Accurate phonetic transcription can only be achieved provided the part of speech category of some words is available, as well as if the dependency relationship between successive words is known.

Natural prosody heavily relies on syntax. It also obviously has a lot to do with semantics and pragmatics, but since very few data is currently available on the generative aspects of this dependence, TTS systems merely concentrate on syntax. Yet few of them are actually provided with full disambiguation and structuration capabilities.

Fig 2.2 The NLP module of a general Text-To-Speech conversion system.

Text analysis

The text analysis block is itself composed of :

A pre-processing module, which organizes the input sentences into manageable lists of words. It identifies numbers, abbreviations, acronyms and idiomatic and transforms them into full text when needed. An important problem is encountered as soon as the character level : that of punctuation ambiguity (including the critical case of sentence end detection). It can be solved, to some extent, with elementary regular grammars.

A morphological analysis module, the task of which is to propose all possible part of speech categories for each word taken individually, on the basis of their spelling. Inflected, derived, and compound words are decomposed into their elementary graphemic units (their morphs) by simple regular grammars exploiting lexicons of stems and affixes (see the CNET TTS conversion program for French [Larreur et al. 1989], or the MITTALK system [Allen et al. 1987]).

The contextual analysis module considers words in their context, which allows it to reduce the list of their possible part of speech categories to a very restricted number of highly probable hypotheses, given the corresponding possible parts of speech of neighboring words. This can be achieved either with n-grams [see Kupiec 92, Willemse & Gulikers 92, for instance], which describe local syntactic dependences in the form of probabilistic finite state automata (i.e. as a Markov model), to a lesser extent with multi-layer perceptron (i.e., neural networks) trained to uncover contextual rewrite rules, as in [Benello et al. 1989], or with local, non-stochastic grammars provided by expert

linguists or automatically inferred from a training data set with classification and regression tree (CART) techniques [Sproat et al. 1992, Yarowsky 1994].

Finally, a syntactic-prosodic parser, which examines the remaining search space and finds the text structure (i.e. its organization into clause and phrase-like constituents) which more closely relates to its expected prosodic realization (see below).

Automatic phonetization

A poem of the Dutch high school teacher and linguist G.N. Trenite surveys this problem in an amusing way. It desperately ends with :

Finally, which rimes with "enough", Though, through, plough,
cough, Hough, or tough ? Hiccough has the sound of "cup",
My advice is ... give it up !

The Letter-To-Sound (LTS) module is responsible for the automatic determination of the phonetic transcription of the incoming text. It thus seems, at first sight, that its task is as simple as performing the equivalent of a dictionary look-up ! From a deeper examination, however, one quickly realizes that most words appear in genuine speech with several phonetic transcriptions, many of which are not even mentioned in pronunciation dictionaries. Namely :

Pronunciation dictionaries refer to word roots only. They do not explicitly account for morphological variations (i.e. plural,

feminine, conjugations, especially for highly inflected languages, such as French), which therefore have to be dealt with by a specific component of phonology, called morphophonology.

Some words actually correspond to several entries in the dictionary, or more generally to several morphological analyses, generally with different pronunciations. This is typically the case of heterophonic homographs, i.e. words that are pronounced differently even though they have the same spelling, as for 'record' (/rekùd/ or /rIkùd/), constitute by far the most tedious class of pronunciation ambiguities. Their correct pronunciation generally depends on their part-of-speech and most frequently contrasts verbs and non-verbs , as for 'contrast' (verb/noun) or 'intimate' (verb/adjective), although it may also be based on syntactic features, as for 'read' (present/past)

Pronunciation dictionaries merely provide something that is closer to a phonemic transcription than from a phonetic one (i.e. they refer to phonemes rather than to phones). As denoted by Withgott and Chen [1993] : "while it is relatively straightforward to build computational models for morphophonological phenomena, such as producing the dictionary pronunciation of 'electricity' given a base form 'electric', it is another matter to model how that pronunciation actually sounds". Consonants, for example, may reduce or delete in clusters, a phenomenon termed as consonant cluster simplification, as in 'softness'

[snIs] in which [t] fuses in a single gesture with the following [n].

Words embedded into sentences are not pronounced as if they

were isolated. Surprisingly enough, the difference does not only originate in variations at word boundaries (as with phonetic liaisons), but also on alternations based on the organization of the sentence into non-lexical units, that is whether into groups of words (as for phonetic lengthening) or into non-lexical parts thereof (many phonological processes, for instance, are sensitive to syllable structure).

Finally, not all words can be found in a phonetic dictionary : the pronunciation of new words and of many proper names has to be deduced from the one of already known words.

Clearly, points 1 and 2 heavily rely on a preliminary morphosyntactic (and possibly semantic) analysis of the sentences to read. To a lesser extent, it also happens to be the case for point 3 as well, since reduction processes are not only a matter of context-sensitive phonation, but they also rely on morphological structure and on word grouping, that is on morphosyntax. Point 4 puts a strong demand on sentence analysis, whether syntactic or metrical, and point 5 can be partially solved by addressing morphology and/or by finding graphemic analogies between words.

It is then possible to organize the task of the LTS module in many ways (Fig 2.3), often roughly classified into dictionary-based and rule- based strategies, although many intermediate solutions exist.

Dictionary-based solutions consist of storing a maximum of phonological knowledge into a lexicon. In order to keep its size reasonably small, entries are generally restricted to morphemes, and the pronunciation of surface forms is accounted for by inflectional, derivational, and compounding morphophonemic rules which describe how the phonetic transcriptions of their morphemic constituents are modified when they are combined into words. Morphemes that cannot be found in the lexicon are transcribed by rule. After a first phonemic transcription of each word has been obtained, some phonetic post-processing is generally applied, so as to account for coarticulatory smoothing phenomena.

This approach has been followed by the MITTALK system from its very first day. [Allen et al. 1987].

A dictionary of up to 12,000 morphemes covered about 95% of the input words. The AT&T Bell Laboratories TTS system follows the same guideline with an augmented morpheme lexicon of 43,000 morphemes. [Levinson et al. 1993].

A rather different strategy is adopted in rule-based transcription systems, which transfer most of the phonological competence of dictionaries into a set of letter-to-sound (or grapheme-to-phoneme) rules. This time, only those words that are pronounced in such a particular way that they constitute a rule on their own are stored in an exceptions dictionary. Notice that, since many exceptions are found in the most frequent words, a reasonably small exceptions

dictionary can account for a large fraction of the words in a running text. In English, for instance, 2000 words typically suffice to cover 70% of the words in text [Hunnicut 1980].

It has been argued in the early days of powerful dictionary-based methods that they were inherently capable of achieving higher accuracy than letter-to-sound rules given the availability of very large phonetic dictionaries on computers. [Coker et al 1990].

On the other hand, considerable efforts have recently been made towards designing sets of rules with a very wide coverage (starting from computerized dictionaries and adding rules and exceptions until all words are covered, as in the work of Daelemans & van den Bosch [1993] or that of Belrhali et al [1992]).

Clearly, some trade-off is inescapable. Besides, the compromise is language-dependent, given the obvious differences in the reliability of letter-to-sound correspondences for different languages.

Fig. 2.3. Dictionary-based (left) versus rule-based (right) phonetization.

Prosody generation

The term prosody refers to certain properties of the speech signal which are related to audible changes in pitch, loudness, syllable length. Prosodic features have specific functions in speech communication (see Fig. 2.4). The most apparent effect of prosody is that of focus. For instance, there are certain pitch events which make a syllable stand out within the utterance, and indirectly the word or syntactic group it belongs to will be highlighted as an important or new component in the meaning of that utterance. The presence of a focus marking may have various effects, such as contrast, depending on the place where it occurs, or the semantic context of the utterance.

Fig. 2.4. Different kinds of information provided by intonation (lines indicate pitch movements; solid lines indicate stress).

Focus or given/new information;

Relationships between words (saw-yesterday; I-yesterday; I-him)

Finality (top) or continuation (bottom), as it appears on the last syllable;

Segmentation of the sentence into groups of syllables.

Although maybe less obvious, there are other, more systematic or general functions.

Prosodic features create a segmentation of the speech chain into groups of syllables, or, put the other way round, they give rise to the grouping of syllables and words into larger chunks. Moreover, there are prosodic features which indicate relationships between such groups, indicating that two or more groups of syllables are linked in

some way. This grouping effect is hierarchical, although not necessarily identical to the syntactic structuring of the utterance.

So what ? Does this mean that TTS systems are doomed to a mere robot-like intonation until a brilliant computational linguist announces a working semantic-pragmatic analyzer for unrestricted text (i.e. not before long) ? There are various reasons to think not, provided one accepts an important restriction on the naturalness of the synthetic voice, i.e. that its intonation is kept 'acceptable neutral'

:

"Acceptable intonation must be plausible, but need not be the most appropriate intonation for a particular utterance : no assumption of understanding or generation by the machine need be made. Neutral intonation does not express unusual emphasis, contrastive stress or stylistic effects : it is the default intonation which might be used for an utterance out of context. (...) This approach removes the necessity for reference to context or world knowledge while retaining ambitious linguistic goals." (Monaghan 1989)

The DSP component

Intuitively, the operations involved in the DSP module are the computer analogue of dynamically controlling the articulatory muscles and the vibratory frequency of the vocal folds so that the output signal matches the input requirements. In order to do it

properly, the DSP module should obviously, in some way, take articulatory constraints into account, since it has been known for a long time that phonetic transitions are more important than stable states for the understanding of speech [Libermann 59]. This, in turn, can be basically achieved in two ways :

Explicitly, in the form of a series of rules which formally describe the influence of phonemes on one another;

Implicitly, by storing examples of phonetic transitions and co- articulations into a speech segment database, and using them just as they are, as ultimate acoustic units (i.e. in place of phonemes).

Two main classes of TTS systems have emerged from this alternative, which quickly turned into synthesis philosophies given the divergences they present in their means and objectives : synthesis- by-rule and synthesis-by-concatenation.

Rule-based synthesizers

Rule-based synthesizers are mostly in favour with phoneticians and phonologists, as they constitute a cognitive, generative approach of the phonation mechanism. The broad spreading of the Klatt synthesizer [Klatt 80], for instance, is principally due to its invaluable assistance in the study of the characteristics of natural speech, by analytic listening of rule-synthesized speech. What is more, the existence of relationships between articulatory parameters and the inputs of the Klatt model make it a practical tool for investigating physiological constraints [Stevens 90].

For historical and practical reasons (mainly the need for a physical interpretability of the model), rule synthesizers always appear in the form of formant synthesizers. These describe speech as the dynamic evolution of up to 60 parameters [Stevens 90], mostly related to formant and anti-formant frequencies and bandwidths together with glottal waveforms. Clearly, the large number of (coupled) parameters complicates the analysis stage and tends to produce analysis errors. What is more, formant frequencies and bandwidths are inherently difficult to estimate from speech data. The need for intensive trials and errors in order to cope with analysis errors, makes them time- consuming systems to develop (several years are commonplace). Yet, the synthesis quality achieved up to now reveals typical busyness problems, which originate from the rules themselves : introducing a high degree of naturalness is theoretically possible, but the rules to do so are still to be discovered.

Rule-based synthesizers remain, however, a potentially powerful approach to speech synthesis. They allow, for instance, to study speaker-dependent voice features so that switching from one synthetic voice into another can be achieved with the help of specialized rules in the rule database. Following the same idea, synthesis-by-rule seems to be a natural way of handling the articulatory aspects of changes in speaking styles (as opposed to their prosodic counterpart, which can be accounted for by concatenation- based synthesizers as well). No wonder then that it has been widely integrated into TTS systems (MITTALK (Allen et al. 1987) and the JSRU synthesizer (Holmes et al. 1964) for English, the multilingual

INFOVOX system (Carlson et al. 1982), and the I.N.R.S system (O'Shaughnessy 1984) for French).

Concatenative synthesizers

As opposed to rule-based ones, concatenative synthesizers possess a very limited knowledge of the data they handle : most of it is embedded in the segments to be chained up. This clearly appears in figure 6, where all the operations that could indifferently be used in the context of a music synthesizer (i.e. without any explicit reference to the inner nature of the sounds to be processed) have been grouped into a sound processing block, as opposed to the upper speech processing block whose design requires at least some understanding of phonetics.

Database preparation

A series of preliminary stages have to be fulfilled before the synthesizer can produce its first utterance. At first, segments are chosen so as to minimize future concatenation problems. A combination of diphones (i.e. units that begin in the middle of the stable state of a phone and end in the middle of the following one), half-syllables, and triphones (which differ from diphones in that they include a complete central phone) are often chosen as speech units, since they involve most of the transitions and co-articulations while requiring an affordable amount of memory. When a complete list of segments has emerged, a corresponding list of words is carefully completed, in such a way that each segment appears at least once (twice is better, for security). Unfavorable positions, like inside

stressed syllables or in strongly reduced (i.e. over-co-articulated) contexts, are excluded. A corpus is then digitally recorded and stored, and the elected segments are spotted, either manually with the help of signal visualization tools, or automatically thanks to segmentation algorithms, the decisions of which are checked and corrected interactively. A segment database finally centralizes the results, in the form of the segment names, waveforms, durations, and internal sub-splitting. In the case of diphones, for example, the position of the border between phones should be stored, so as to be able to modify the duration of one half-phone without affecting the length of the other one.

Figure 2.5. A general concatenation-based synthesizer. The upper left hatched block corresponds to the development of the synthesizer (i.e. it is processed once for all). Other blocks correspond to run-time operations. Language-dependent operations and data are indicated by a flag.

Segments are then often given a parametric form, in the form of a temporal sequence of vectors of parameters collected at the output of a speech analyzer and stored in a parametric segment database. The advantage of using a speech model originates in the fact that :

Well chosen speech models allow data size reduction, an advantage which is hardly negligible in the context of concatenation-based synthesis given the amount of data to be stored. Consequently, the analyzer is often followed by a parametric speech coder.

A number of models explicitly separate the contributions of respectively the source and the vocal tract, an operation which remains helpful for the pre-synthesis operations : prosody matching and segments concatenation.

Indeed, the actual task of the synthesizer is to produce, in real-time, an adequate sequence of concatenated segments, extracted from its parametric segment database and the prosody of which has been adjusted from their stored value, i.e. the intonation and the duration they appeared with in the original speech corpus, to the one imposed by the language processing module. Consequently, the respective

parts played by the prosody matching and segments concatenation modules are considerably alleviated when input segments are presented in a form that allows easy modification of their pitch, duration, and spectral envelope, as is hardly the case with crude waveform samples.

Since segments to be chained up have generally been extracted from different words, that is in different phonetic contexts, they often present amplitude and timbre mismatches. Even in the case of stationary vocalic sounds, for instance, a rough sequencing of parameters typically leads to audible discontinuities. These can be coped with during the constitution of the synthesis segments database, thanks to an equalization in which related endings of segments are imposed similar amplitude spectra, the difference being distributed on their neighbourhood. In practice, however, this operation, is restricted to amplitude parameters : the equalization stage smoothly modifies the energy levels at the beginning and at the end of segments, in such a way as to eliminate amplitude mismatches (by setting the energy of all the phones of a given phoneme to their average value). In contrast, timbre conflicts are better tackled at run- time, by smoothing individual couples of segments when necessary rather than equalizing them once for all, so that some of the phonetic variability naturally introduced by co-articulation is still maintained. In practice, amplitude equalization can be performed either before or after speech analysis (i.e. on crude samples or on speech parameters).

Once the parametric segment database has been completed, synthesis itself can begin.

Speech synthesis

A sequence of segments is first deduced from the phonemic input of the synthesizer, in a block termed as segment list generation in Fig. 5, which interfaces the NLP and DSP modules. Once prosodic events have been correctly assigned to individual segments, the prosody matching module queries the synthesis segment database for the actual parameters, adequately uncoded, of the elementary sounds to be used, and adapts them one by one to the required prosody. The segment concatenation block is then in charge of dynamically matching segments to one another, by smoothing discontinuities. Here again, an adequate modelization of speech is highly profitable, provided simple interpolation schemes performed on its parameters approximately correspond to smooth acoustical transitions between sounds. The resulting stream of parameters is finally presented at the input of a synthesis block, the exact counterpart of the analysis one. Its task is to produce speech.

Segmental quality

The efficiency of concatenative synthesizers to produce high quality speech is mainly subordinated to :

The type of segments chosen.

Segments should obviously exhibit some basic properties :

They should allow to account for as many co-articulatory effects as possible.

Given the restricted smoothing capabilities of the concatenation block, they should be easily connectable.

Their number and length should be kept as small as possible.

On the other hand, longer units decrease the density of concatenation points, therefore providing better speech quality. Similarly, an obvious way of accounting for articulatory phenomena is to provide many variants for each phoneme. This is clearly in contradiction with the limited memory constraint. Some trade-off is necessary. Diphones are often chosen. They are not too numerous (about 1200 for French, including lots of phoneme sequences that are only encountered at word boundaries, for 3 minutes of speech, i.e. approximately 5 Mbytes of 16 bits samples at 16 kHz) and they do incorporate most phonetic transitions. No wonder then that they have been extensively used. They imply, however, a high density of concatenation points (one per phoneme), which reinforces the importance of an efficient concatenation algorithm. Besides, they can only partially account for the many co- articulatory effects of a spoken language, since these often affect a whole phone rather than just its right or left halves independently. Such effects are especially patent when somewhat transient phones, such as liquids and (worst of

all) semi-vowels, are to be connected to each other. Hence the use of some larger units as well, such as triphones.

The model of speech signal, to which the analysis and synthesis algorithms refer.

The models used in the context of concatenative synthesis can be roughly classified into two groups, depending on their relationship with the actual phonation process. Production models provide mathematical substitutes for the part respectively played by vocal folds, nasal and vocal tracts, and by the lips radiation. Their most representative members are Linear Prediction Coding (LPC) synthesizers (Markel & Gray 1976), and the formant synthesizers we mentioned in section 2.2.1. On the contrary, phenomenological models intentionally discard any reference to the human production mechanism. Among these pure digital signal processing tools, spectral and time-domain approaches are increasingly encountered in TTS systems. Two leading such models exist : the hybrid Harmonic/Stochastic (H/S) model of (Abrantes et al. 1991) and the Time-Domain Pitch-Synchronous-OveraLap-Add (TD-PSOLA) one ([oulines & Charpentier 1990). The latter is a time-domain algorithm

: it virtually uses no speech explicit speech model. It exhibits very interesting practical features : a very high speech quality (the best currently available) combined with a very low computational cost (7 operations per sample on the average). The hybrid Harmonic/stochastic model is intrinsically more powerful than the

TD-PSOLA one, but it is also about ten times more computationally intensive. PSOLA synthesizers are now widely used in the speech synthesis community. The recently developed MBROLA algorithm [Dutoit 93,96] even provides a time-domain algorithm which exhibits the very efficient smoothing capabilities of the H/S model (for the spectral envelope mismatches that cannot be avoided at concatenation points) as well as its very high data compression ratios (up to 10 with almost no additional computational cost) while keeping the computational complexity of PSOLA.

CONCEPTUAL FRAMEWORK

SAMPLE INPUT

VOICE SELECTION

SAMPLE OUTPUT

CHAPTER THREE

SYSTEM ANALYSIS AND DESIGN

GENERAL DESCRIPTION OF THE EXISTING SYSTEM

In today's world where most information is shared digitally, visually impaired persons always require their reading glasses to have access to this information, in a situation where they somehow forgot their reading glasses, they won't be to have access this information. But with text to speech system digital information can be read out to a visually impaired person.

FACT FINDING METHODS USED

There are two main sources of data collection in carrying out this study, information was basically obtained from the two sources which are:

Primary source and

Secondary source

Primary Source

Primary source refers to the sources of collecting original data in which the researcher makes use of empirical approach such as personal interview, questionnaires or observation.

In my research, I used a method of observation were I was attentive to how contact are being operated and saved using a manual method.

Secondary Source

The need of the secondary sources of data for this kind of project cannot be over emphasized. The secondary data were obtained by me from the library source and most of the information from the library research has been covered in my literature review in the previous chapter of this project.

OBJECTIVE OF THE NEW SYSTEM

The main objective the new system to convert text to speech, i.e. a system capable reading out type words

INPUT FORM DESIGN

The system has only one input structure which the text input form.

PROCESS ANALYSIS

Text-to-speech synthesis takes place in several steps. The TTS systems get a text as input, which it first must analyze and then transform into a phonetic description. Then in a further step it generates the prosody. From the information now available, it can produce a speech signal.

The structure of the text-to-speech synthesizer can be broken down into major modules:

Natural Language Processing (NLP) module: It produces a phonetic transcription of the text read, together with prosody.

Digital Signal Processing (DSP) module: It transforms the symbolic information it receives from NLP into audible and intelligible speech.

The major operations of the NLP module are as follows:

Text Analysis: First the text is segmented into tokens. The token-to-word conversion creates the orthographic form of the token. For the token “Mr” the orthographic form “Mister” is formed

by expansion, the token “12” gets the orthographic form “twelve” and “1997” is transformed to “nineteen ninety seven”.

Application of Pronunciation Rules: After the text analysis has been completed, pronunciation rules can be applied. Letters cannot be transformed 1:1 into phonemes because correspondence is not always parallel. In certain environments, a single letter can correspond to either no phoneme (for example, “h” in “caught”) or several phoneme (“m” in “Maximum”). In addition, several letters can correspond to a single phoneme (“ch” in “rich”).

OUTPUT FORM DESIGN

The output from the system designed is generated from the system inputs. The system format for this system audio format

CHAPTER 3.8 SYSTEM FLOW CHART

text

CHAPTER FOUR

DESIGN AND IMPLEMENTATION

DESIGN STANDARD

Choice of Programming language used :

HTML ADVANTAGES OF HTML

It is easy to learn and use

Useful for beginners in web design

The software is available for free DISADVANTAGES OF HTML

We need to create a lot of code for a simple webpage

It is not completely secured

Long codes becomes complex

JAVASCRIPT ADVANTAGES OF JAVASCRIPT

JavaScript is executed on the client side. So, it is very fast.

JavaScript is easy to learn. Any one which have basic knowledge of programming can easily lean JavaScript.

JavaScript supports all modern browsers. It can execute on any browser and produce same result

DISADVANTAGES OF JAVASCRIPT

JavaScript code is visible to every one and this is the biggest disadvantage of JavaScript.

JavaScript code is visible to every one and this is the biggest disadvantage of JavaScript.

JavaScript only support single inheritance.

CSS(Cascading style sheet) ADVANTAGES OF CSS

The main advantage of CSS is that style is applied consistently across variety of sites. One instruction can control several areas which is advantageous.

Web designers needs to use few lines of programming for every page improving site speed.

It is less complex therefore the effort are significantly reduced DISADVANTAGE OF CSS

SQL(Structural query language)

ADVANTAGES OF SQL

Large amount of data is retrieved quickly and efficiently.

For data retrieval, large number of lines of code is not required.

Easy to learn and understand, answers to complex queries can be received in seconds.

DISADVANTAGES OF SQL

SQL has a difficult interface that makes few users uncomfortable while dealing with the database.

Some versions are costly and hence, programmers cannot access it

Due to hidden business rules, complete control is not given to the database.

SYTEM REQUIREMENTS

The requirements needed to implement this system are as follows:

HARD REQUIREMENTS

The software design needed the following hardware for an effective operation of the newly designed system

A system running on AMD, Pentium 2 or higher processor

The Random Access Memory (RAM) should be at least 512mb.

Enhanced keyboard.

At least 20 GB hard disk.

V.G.A or a colored monitor.

Software Requirements

The software requirements includes:-

A Windows XP operating system or higher version for faster processing

MySQL database

Apache webserver

PHP 5.6+ runtime environment

Visual Studio Code

Chrome Browser

CHAPTER FIVE

SUMMARY, CONCLUSION AND RECOMMENDATIONS

SUMMARY

In summary, this Academic Work project has done a great deal of giving a broad knowledge of what Text-To-Speech system is all about and how it can be operated.

CONCLUSION

From this Academic Work, I have been able to show the application of Text-to-Speech and how text can be synthesized for the visually impaired as well as children to read easily.

5.3 RECOMMENDATION

Below are some recommendations:

I hereby recommend this Academic work to be used by staff and management of Auchi Polytechnic, Auchi.

Training of users to be done by Organization who needs it

for further research, text to more languages should be carried out.

REFERENCES

[Abrantes et al. 91] A.J. ABRANTES, J.S. MARQUES, I.M.

TRANSCOSO, "Hybrid Sinusoidal Modeling of Speech without Voicing Decision", EUROSPEECH 91, pp. 231-234.

[Jaiswal et al. 21] R.K. Jaiswal, R.K. Dubey, “Text-to-speech (TTS) synthesis is one of the rapidly emerging areas of computer-to-human interaction technology” 2021 5th International Conference on Electronics, Communication and Aerospace Technology (ICECA) IEEE, 2021, p.867-872.

[Allen 85] J. ALLEN, "A Perspective on Man-Machine Communication by Speech", Proceedings of the IEEE, vol. 73, 11, November 1985, pp. 1541-1550.

[Allen et al. 87] J. ALLEN, S. HUNNICUT, D. KLATT, From Text To

Speech, The MITTALK System, Cambridge University Press, 1987, 213 pp.

[Bachenko & Fitzpatrick 90] J. BACHENKO, E. Fitzpatrick, "A computational grammar of discourse-neutral prosodic phrasing in English", Computational Linguistics, n16, September 1990, pp. 155- 167.

[Belrhali et al. 94] R. BELRHALI, V. AUBERGE, L.J. BOE, "From

lexicon to rules : towards a descriptive method of French text-to- phonetics transcription", Proc. ICSLP 92, Alberta, pp. 1183-1186.

[Benello et al. 88] J. BENELLO, A.W. MACKIE, J.A. ANDERSON,

"Syntactic category disambiguation with neural networks", Computer Speech and Language, 1989, n3, pp. 203-217.

[Carlson et al. 82] R. CARLSON, B. GRANSTRÖM, S. HUNNICUT, "A

multi-language Text-To-Speech module", ICASSP 82, Paris, vol. 3, pp. 1604-1607.

[Coker 85] C.H. COKER, "A Dictionary-Intensive Letter-to-Sound Program", J. Ac. Soc. Am., suppl. 1, n8, 1985, S7.

[Coker et al. 90] C.H. COKER, K.W. CHURCH, M.Y. LIBERMAN,

"Morphology and rhyming : Two powerful alternatives to letter-to- sound rules for speech synthesis", Proc. of the ESCA Workshop on Speech Synthesis, Autrans (France), 1990, pp. 83-86.

[Daelemans & van den Bosch 93] W. DAELEMANS, A. VAN DEN BOSCH, "TabTalk : Reusability in data-oriented grapheme-to- phoneme conversion", Proc. Eurospeech 93, Berlin, pp. 1459-1462.

[Dutoit 93] T. DUTOIT, H. LEICH, "MBR-PSOLA : Text-To-Speech

Synthesis based on an MBE Re-Synthesis of the Segments Database", Speech Communication, Elsevier Publisher, November 1993, vol. 13, n3-4.

[Dutoit 96] T. DUTOIT,

Kluwer Academic Publishers, 1996, 326 pp.

[Flanagan 72] J.L. FLANAGAN, Speech Analysis, Synthesis, and Perception, Springer Verlag, 1972, pp. 204-210.

[Hirschberg 91] J. HIRSCHBERG, "Using text analysis to predict intonational boundaries", Proc. Eurospeech 91, Genova, pp. 1275- 1278.

[Holmes et al. 64] J. HOLMES, I. MATTINGLY, J. SHEARME,

'Speech synthesis by rule', Language and Speech, Vol 7, 1964, pp.127-143

[Hunnicut 80] S. HUNNICUT, "Grapheme-to-Phoneme rules : a Review", Speech Transmission Laboratory, Royal Institute of Technology, Stockholm, Sweden, QPSR 2-3, pp. 38-60.

[Klatt 80] D.H. KLATT, 'Software for a cascade /parallel formant synthesizer', J. Acoust. Soc. AM., Vol 67, 1980, pp. 971-995.

[Klatt 86] D.H. KLATT, "Text-To-Speech : present and future", Proc. Speech Tech '86, pp. 221-226.

[Kupiec 92] J. KUPIEC, "Robust part-of-speech tagging using a Hidden Markov Model", Computer Speech and Language, 1992, n6,

pp. 225-242.

[Larreur et al. 89] D. LARREUR, F. EMERARD, F. MARTY,

"Linguistic and prosodic processing for a text-to-speech synthesis system", Proc. Eurospeech 89, Paris, pp. 510-513.

[Levinson et al. 93] S.E. LEVINSON, J.P. OLIVE, J.S. TSCHIRGI,

"Speech Synthesis in Telecommunications", IEEE Communications Magazine, November 1993, pp. 46-53.

[Liberman & Church 92] M.J. LIBERMAN, K.W. CHURCH, "Text analysis and word pronunciation in text-to-speech synthesis", in Advances in Speech Signal Processing, S. Furuy, M.M. Sondhi eds., Dekker, New York, 1992, pp.791-831.

[Lingaard 85] R. LINGAARD, Electronic synthesis of speech, Cambridge University Press, 1985, pp 1-17.

[Markel & Gray 76] J.D. MARKEL, A.H. GRAY Jr, Linear Prediction of Speech, Springer Verlag, New York, pp. 10-42, 1976.

[Monaghan 90a] A.I.C. MONAGHAN, "A multi-phrase parsing strategy for unrestricted text", Proc. ESCA Workshop on speech synthesis, Autrans, 1990, pp. 109-112.

[Moulines & Charpentier 90] E.MOULINES, F. CHARPENTIER, "Pitch Synchronous waveform Processing techniques for Text-To- Speech Synthesis using diphones", Speech Communication, Vol. 9, no 5-6.

[O' Shaughnessy 84] D. O' SHAUGHNESSY, 'Design of a real-time French text-to-speech system', Speech Communication, Vol 3, pp. 233-243.

[O'Shaughnessy 90] D. O'SHAUGHNESSY, "Relationships between syntax and prosody for speech synthesis", Proceedings of the ESCA tutorial day on speech synthesis, Autrans, 1990, pp. 39-42.

[Sproat et al. 92] R. SPROAT, J. HIRSHBERG, D. YAROWSKY, "A Corpus-based Synthesizer", Proc. ICSLP 92 Alberta, pp. 563-566.

[Stevens 90] K.N. STEVENS, 'Control parameters for synthesis by rule', Proceedings of the ESCA tutorial day on speech synthesis, Autrans, 25 sept 90, pp. 27-37.

[Traber 93] C. TRABER, "Syntactic Processing and Prosody Control in the SVOX TTS System for German", Proc. Eurospeech 93, Berlin, vol. 3, pp. 2099-2102.

[Willemse & Gulikers 92] R. WILLEMSE, L. GULIKERS, "Word class assignment in a Text-To-Speech system", Proc. Int. Conf. on Spoken Language Processing, Alberta, 1992, pp. 105-108.

[Withgott & Chen 93] M. M. WITHGOTT, F.R. CHEN, Computational models of American English, CSLI Lecture Notes, no 32, 143pp.

[Witten 82] I.H. WITTEN, Principles of Computer Speech, Academic Press, 1992, 286 pp.

[Yarowsky 94] D. YAROWSKY, "Homograph Disambiguation in Speech Synthesis'', Proceedings, 2nd ESCA/IEEE Workshop on Speech Synthesis, New Paltz, NY, 1994.

APPENDIX

Source Codes

Home Page

```
<!DOCTYPE html>

<html>

<head>

<meta charset="utf-8">

<meta  http-equiv="X-UA-Compatible" content="IE=edge,chrome=1">

<meta name="viewport" content="width=device-width">

<title>Speech synthesiser</title>

<link rel="stylesheet" href="style.css">

<!--[if lt IE 9]>

<script src="//html5shiv.googlecode.com/svn/trunk/html5.js"></script>

<![endif]-->

</head>
```

```
<body>
<h1>Design and Implementation of Text to Speech/Audio System
</h1>
<p>Enter some text in the input below and press ENTER to hear it. change voices using the dropdown menu.</p>
<form>
<input type="text" class="txt">
<div>
<label for="rate">Rate</label><input type="range" min="0.5" max="2" value="1" step="0.1" id="rate">
<div class="rate-value">1</div>
<div class="clearfix"></div>
</div>
<div>
<label for="pitch">Pitch</label><input type="range" min="0" max="2" value="1" step="0.1" id="pitch">
<div class="pitch-value">1</div>
<div class="clearfix"></div>
</div>
<select>
</select>
</form>
<script src="script.js"></script>
</body>
</html>
```

Tex-to-Speech Engine

```
var synth = window.speechSynthesis;
var inputForm = document.querySelector('form'); var inputTxt =
document.querySelector('.txt');

var voiceSelect = document.querySelector('select');

var pitch = document.querySelector('#pitch');

var pitchValue = document.querySelector('.pitch-value'); var rate =
document.querySelector('#rate');

var rateValue = document.querySelector('.rate-value');

var voices = [];

function populateVoiceList() { voices = synth.getVoices();

var selectedIndex = voiceSelect.selectedIndex < 0 ? 0
        : voiceSelect.selectedIndex;

voiceSelect.innerHTML = '';

for(i = 0; i < voices.length ; i++) {

var option = document.createElement('option'); option.textContent = voices[i].name + ' (' +
voices[i].lang + ')';

if(voices[i].default) { option.textContent += ' -- DEFAULT';

}

option.setAttribute('data-lang', voices[i].lang); option.setAttribute('data-name',
voices[i].name); voiceSelect.appendChild(option);

}

voiceSelect.selectedIndex = selectedIndex;

}

populateVoiceList();

if (speechSynthesis.onvoiceschanged !== undefined) {

speechSynthesis.onvoiceschanged = populateVoiceList;

}

inputForm.onsubmit = function(event) { event.preventDefault();

var utterThis = new SpeechSynthesisUtterance(inputTxt.value); var selectedOption =
voiceSelect.selectedOptions[0].getAttribute('data-name'); for(i = 0; i < voices.length ; i++) {

if(voices[i].name === selectedOption) { utterThis.voice = voices[i];

}

}

utterThis.pitch = pitch.value; utterThis.rate = rate.value; synth.speak(utterThis);

utterThis.onpause = function(event) {

var char = event.utterance.text.charAt(event.charIndex); console.log('Speech paused at character ' + event.charIndex + ' of

"' +

event.utterance.text + '", which is "' + char + '".');

}

inputTxt.blur();

}

pitch.onchange = function() {

pitchValue.textContent = pitch.value;

}

rate.onchange = function() {

rateValue.textContent = rate.value;

}
```