VOICE TO VOICE: RECOGNITION AND RESPONSE SYNTHESISER

A PROJECT REPORT

Submitted by

NAEEM HADIQ (MES15CS064)

SUMEENA SALAM (MES15CS110)

NIBRAS NAZAR (MES15CS067)

NEHA PARVEEN (MES15CS066)

to

the APJ Abdul Kalam Technological University

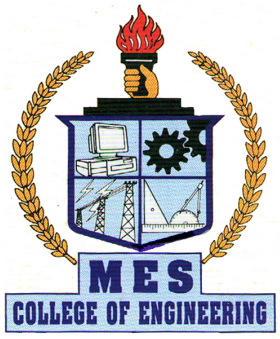
in partial fulfillment of the requirements for the award of the Degree

of

Bachelor of Technology

in

Computer Science and Engineering



(NBA accredited)

Department of Computer Science and Engineering

MES College of Engineering Kuttippuram

Thrikkanapuram P.O., Malappuram Dt., Kerala, India 679582

MAY 2019

is

DECLARATION

We, the undersigned, hereby declare that the project report "Voice to Voice: Recognition and Response Synthesiser", submitted for partial fulfillment of the requirements for the award of degree of Bachelor of Technology of the APJ Abdul Kalam Technological University, Kerala is a bonafide work done by our team under the supervision of Dr. Sobin C. C., Associate Professor, Department of Computer Science Engineering. This submission represents our ideas in our own words and where ideas or words of others have been included, We have adequately and accurately cited and referenced the original sources. We also declare that we have adhered to ethics of academic honesty and integrity and have not misrepresented or fabricated any data or idea or fact or source in our submission. We understand that any violation of the above will be a cause for disciplinary action by the institute and/or the University and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been obtained. This report has not been previously formed the basis for the award of any degree, diploma or similar title of any other University.

Place: Kuttippuram
Date:

Signature :
Name : Naeem hadiq
Signature :
Name : Sumeena Salam
Signature :
Name : Neha Parveen
Signature :
Name : Nibras Nazar

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

MES COLLEGE OF ENGINEERING, KUTTIPPURAM

(NBA accredited)

CERTIFICATE

This is to certify that the report entitled **"Voice to Voice: Recognition and Response Synthesiser"** submitted by **Naeem hadiq**, **Sumeena Salam**, **Neha Parveen** and **Nibras Nazar** to the APJ Abdul Kalam Technological University in partial fulfillment of the requirements for the award of the Degree of Bachelor of Technology in Computer Science and Eng is a bonafide record of the project work carried out by the team under my guidance and supervision. This report in any form has not been submitted to any other University or Institute for any purpose.

Internal Supervisor:

Dr. Sobin C. C.
Associate Professor
Dept. of Computer Science and Eng.
MES College of Engineering

HEAD OF THE DEPT:

Dr. Sasidaran Sreedharan
Dept.of Computer Science and Engg.
MES College of Engineering

Project Coordinators:
Mr. Sreekanth E. S.
Mrs. Asha G.

ACKNOWLEDGEMENT

At the outset, we would like to thank the Almighty for all his blessings that led us here.

We are grateful to *Dr. A. S. Varadarajan, Principal, MES College of Engineering, Kuttippuram*, for providing the right ambiance to complete this project. We would also like to extend our sincere gratitude to *Dr. Sasidharan Sreedharan, Head of the Department, Computer Science and Engineering, MES College of Engineering, Kuttippuram.*

We are deeply indebted to the project coordinators, *Mr. Sreekanth E. S. Assistant Professor, Department of Computer Science and Engineering* and *Mrs. Asha G, Assistant Professor, Department of Computer Science and Engineering* for their continued support.

It is with great pleasure that we express our deep sense of gratitude to our project guide, *Dr. Sobin C. C., Associate Professor, Department of Computer Science and Engineering*, for his guidance, supervision, encouragement and valuable advice in each and every phase.

We would also like to express our sincere thanks and gratitude to all staff members of the department, our friends and family members for their cooperation, positive criticism, consistent support and consideration during the preparation of this work.

Naeem Hadiq

Sumeena Salam

Neha Parveen

Nibras Nazar

ABSTRACT

Speech recognition is the ability of a machine or program to identify words and phrases in spoken language and convert them to a machine readable format. Voice recognition has gained prominence and use with the rise of AI and intelligent assistants, such as Amazon's Alexa, Apple's Siri and Microsoft's Cortana. There are plenty of applications and areas where speech recognition is used, including the military, as an aid for impaired persons, in the medical field, in robotics, etc.

One of the main difficulties is the immense variations among people in pronouncing words. Background noise can cause a whole system to fail. As a result, speech recognition fails in many cases due to noises that are out of the user's control.

The proposed model to synthesize natural synthetic speech as processed output from input voice. The input speech is converted to Text which is then syntactically and semantically processed to analyse the requirement and then the output is converted back to synthetic speech. Our system is set to outperforms previously published methods on the Switchboard Hub 500 corpus, achieving less than 16 percent error, and performs better than commercial systems in noisy speech recognition tests while also being able to generate Speaker Embedded high quality modulate synthetic speech outputs.

CONTENTS

LIST OF FIGURES

CHAPTER 1
INTRODUCTION

Speech recognition technology, which is able to recognize human speech and change to text, or to perform a command, has emerged as the next big thing of the IT industry. Speech recognition is technology that uses desired equipment and a service which can be controlled through voice without using items such as a mouse or keyboard. It also appeared as part of ongoing research in progress in 1950s, but was not popularized until the mid-2000s, with low voice recognition. Presently, related speech recognition technologies, which have been previously used limitedly for special-purposes, have been rapidly evolving because of the proliferation of portable computing terminals such as smartphones interconnected with the expansion of the cloud infrastructure.

One of the most prominent examples of a mobile voice interface is *Siri*, the voice-activated personal assistant that comes built into the latest iPhone. But voice functionality is also built into Android, the Windows Phone platform, and most other mobile systems, as well as many applications. While these interfaces still have considerable limitations,we are inching closer to machine interfaces we can actually talk.

Top speech recognition systems rely on sophisticated pipelines composed of multiple algorithms and hand-engineered processing stages. In this project, we describe an end-to-end speech system, where deep learning supersedes these processing stages. Combined with a language model, and a response generator this approach achieves higher performance than traditional methods on hard speech recognition and processing tasks while also being much simpler. These results are made possible by training a large recurrent neural network (RNN) using multiple GPUs and thousands

of hours of data for recognition and processing alongside a DNN trained for speech synthesis. As this system learns directly from data, we do not require specialized components for speaker adaptation or noise filtering. In fact, in settings where robustness to speaker variation and noise are critical. Our system is set to outperforms previously published methods, achieving less error, and performs better than commercial systems in noisy speech recognition tests while also being able to generate Speaker Embedded high quality modulate synthetic speech outputs.

CHAPTER 2
LITERATURE SURVEY

Related approaches to speech recognition system introduced in the existing systems is briefly discussed in this chapter. Three approaches are listed below

1. Automatic Speech Recognition(ASR)

2. Support Vector Machine(SVM)

3. Acoustic Model

2.1 Automatic Speech Recognition (ASR)

The main focus is to analyze the performance of Automatic Speech Recognition (ASR) in different emotional environments using prosody modification. The majority of ASR systems are trained using neutral speech and the performance of such systems degrade when tested with the emotional speech. The various components of speech that contribute to the emotion characteristics are studied. The prosody features of the source emotional utterances are modified according to the target neutral utterances using Flexible Analysis Synthesis Tool (FAST). In the FAST, Dynamic Time Warping (DTW) is used to align the source emotional and target neutral utterances. Components of the prosody such as intonation, duration and excitation source are manipulated to incorporate the desired features into the source utterance. The modified (source emotional) utterances are then used for testing the ASR system which is trained using neutral speech. Three emotions (compassion, happiness and anger) are considered for the analysis. Experimental results indicate an average improvement in the speech recognition system performance by considering prosody modified speech.

Automatic Speech Recognition (ASR) involves the process of considering the machine driven transcription of the language spoken by the speaker and converting them to a readable text. The goal of an ASR system is to accurately convert a speech signal into sequence of symbols. ASR in the presence of different emotion conditions is one of the research problems in natural human-machine interaction. Emotional environments may be viewed as a state where speakers produce speech in different emotions such as compassion, anger and happiness.

The majority of ASR systems are trained on the neutral speech. The performance of the ASR system is degraded in different emotions. The naturalness of the human-machine interaction mainly depends on the ASR systems ability to recognize speech under emotional conditions. In the literature, it is stated that the ASR system performance can be improved in three different levels namely preprocessing level, robust feature representation level and model-based adaptation level. The prosody modification is done at the preprocessing level to convert the emotional utterance into neutral utterance and the MFCC features are extracted later for ASR system [2].

2.2 Support Vector Machine (SVM)

One of the powerful tools for pattern recognition that uses a discriminative approach is a SVM. SVMs use linear and nonlinear separating hyper-planes for data classification. However, since SVMs can only classify fixed length data vectors, this method cannot be readily applied to task involving variable length data classification. The variable length data has to be transformed to fixed length vectors before SVMs can be used. It is a generalized linear classifier with maximum-margin fitting functions.

This fitting function provides regularization which helps the classifier generalized better. The classifier tends to ignore many of the features. Conventional statistical and neural network methods control model complexity by using a small number of features. SVM controls the model complexity by controlling the VC dimensions of its model. This method is independent of dimensionality and can utilize spaces of very large dimensions spaces, which permits a construction of very large number of on-linear features and then performing adaptive feature selection during training. By shifting all non-linearity to the features, SVM can use linear model for which VC dimensions is known. For example, a support vector machine can be used as a regularized radial basis function classifier [1].

2.3 Acoustic model

Research in speech processing and communication for the most part, was motivated by people desire to build mechanical models to emulate human verbal communication capabilities. Speech is the most natural form of human communication and speech processing has been one of the most exciting areas of the signal processing. Speech recognition technology has made it possible for computer to follow human voice commands and understand human languages. The main goal of speech recognition area is to develop techniques and systems for speech input to machine. Speech is the primary means of communication between humans. For reasons ranging from technological curiosity about the mechanisms for mechanical realization of human speech capabilities to desire to automate simple tasks which necessitates human machine interactions and research in automatic speech recognition by machines has at-

tracted a great deal of attention for sixty years. Based on major advances in statistical

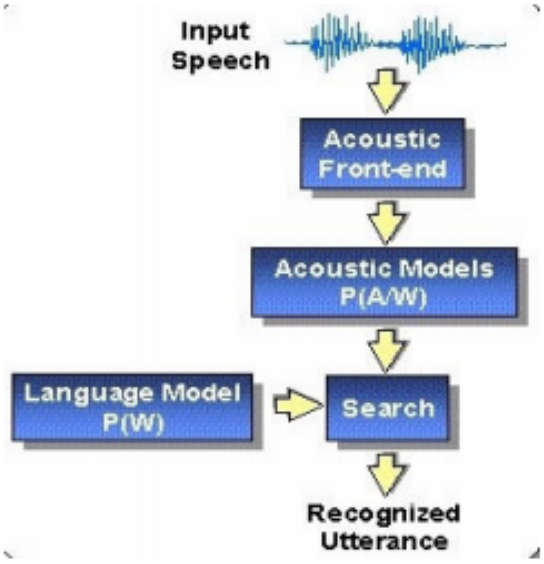


Figure 2.1: Basic model of speech recognition

modeling of speech, automatic speech recognition systems today find widespread application in tasks that require human machine interface, such as automatic call processing in telephone networks, and query based information systems that provide updated travel information, stock price quotations, weather reports, data entry, voice dictation, access to information: travel, banking, commands, avoinics, automobile portal, speech transcription, handicapped people (blind people) supermarket, railway reservations etc. Speech recognition technology was increasingly used within telephone networks to automate as well as to enhance the operator services. This report reviews major highlights during the last six decades in the research and development of automatic speech recognition, so as to provide a technological perspective. Although many technological progresses have been made, still there remains many research issues that need to be tackled. Fig 2.1 shows a mathematical representation

of speech recognition system in simple equations which contain front end unit, model unit, language model unit, and search unit. The recognition process is shown below (Fig 2.1).

The standard approach to large vocabulary continuous speech recognition is to assume a simple probabilistic model of speech production whereby a specified word sequence, W, produces an acoustic observation sequence Y, with probability P(W,Y). The goal is then to decode the word string, based on the acoustic observation sequence, so that the decoded string has the maximum a posteriori (MAP) probability [1].

CHAPTER 3
SYSTEM ANALYSIS

3.1 Existing System

The existing systems requires high computation requirements and mostly relies on nternet API calls to run which have requirement of internet connection which is a major fault of losing connectivity at critical situations.

Disadvantages of the current system are enlisted below:

- **Background Noise Interference**: To get the best out of voice recognition software, we need a quiet environment. Systems don't work so well if there is a lot of background noise. They may not be able to differentiate between your speech, other people talking and other ambient noise, leading to transcription mix-ups and errors. This can cause problems if you work in a busy office or noisy environment. Wearing close-talking microphones or noise-canceling headsets can help the system focus on your speech.

- **Costs and Productivity**: Speech recognition primarily increases productivity through its capacity to do simple, yet time-consuming tasks, such as meeting minute taking or transcribing audio files. Removing these menial duties from an employees list of things-to-do automatically frees up a significant amount of time and resources that can be allocated to other work that would not be possible by a machine.

- **Accents and Speech Recognition**: Voice recognition systems can have problems with accents. Even though some may learn to decode your speech over

time, you have to learn to talk consistently and clearly at all times to minimize errors. If you mumble, talk too fast or run words into each other, the software will not always be able to cope. Programs may also have problems recognizing speech as normal if your voice changes, say when you have a cold, cough, sinus or throat problem.

- **Requires large amounts of memory**: Voice recognition uses a lot of memory. The software has specific hardware requirements. The speech rate supported by even the most high end desktop microprocessors is extremely limited due the processing requirements of the software. The performance of processors that are more suitable in a portable environment are even worse.

- **Requires each user to train software to recognize voice** : These systems should be able to recognize 95% of the sound correctly. While using this software one should talk clearly. Each and every person has a different voice, hence speech recognition system should ask for enrollment of the voice before it gets used.

3.2 Proposed System

Speech recognition is the ability of a machine or program to identify words and phrases in spoken language and convert them to a machine-readable format.

The proposed system create a completely offline (Low Inference Compute) vocal recognition and response system capable of understanding the context of speech and differentiate between tones of speech. The system can be used as critical responses for computer controlled devices including robots and war equipments.

CHAPTER 4

System Requirements

4.1 Hardware Requirements

Training	**End product**
Core i7 Processor	2ghz/higher processor
nvidia 1000 series	Microphone
Microphone	speaker
speaker system	1GB Ram
16GB RAm	3GB Strorage
1TB Storage	

4.1.1 Core i7 Processor

Core i7 is a family of high-end performance 64-bit x86-64 processors designed by Intel for high-end desktops and laptops. Core i7 was introduced in 2008 following the retirement of the Core 2 Quad family. Core i7 microprocessors are the high-end brand from the Core family, positioned above both the Core i5 and the Core i3.

4.1.2 nvidia 1000 series

NVIDIA is the world leader in visual computing technologies and the inventor of the GPU, a high-performance processor which generates breathtaking, interactive graphics on workstations, personal computers, game consoles, and mobile devices.

4.2 Software Requirements

4.2.1 Python

Python is a widely used general-purpose, high level programming language. It was initially designed by Guido van Rossum in 1991 and developed by Python Software Foundation. It was mainly developed for emphasis on code readability, and its syntax allows programmers to express concepts in fewer lines of code.

4.2.2 Pytorch

PyTorch is an open-source machine learning library for Python, based on Torch, used for applications such as natural language processing. It is primarily developed by Facebook's artificial-intelligence research group, and Uber's "Pyro" Probabilistic programming language software is built on it.

PyTorch provides two high-level features:

1. Tensor computation (like NumPy) with strong GPU acceleration
2. Deep neural networks built on a tape-based autodiff system

4.2.3 Pandas

Pandas is an open source, BSD-licensed library providing high-performance, easy-to-use data structures and data analysis tools for the Python programming language. pandas is a NumFOCUS sponsored project. This will help ensure the success of development of pandas as a world-class open-source project, and makes it possible to donate to the project.

4.2.4 TensorFlow

TensorFlow is an end-to-end open source platform for machine learning. It has a comprehensive, flexible ecosystem of tools, libraries and community resources that lets researchers push the state-of-the-art in ML and developers easily build and deploy ML powered applications.

CHAPTER 5
DESIGN

5.1 Speech to Text Convertion

The complete RNN model is illustrated in figure 4.2. Its structure is considerably simpler than related models used in this domainwe have limited ourselves to a single recurrent layer (which is the hardest to parallelize) and we do not use Long-Short-Term-Memory (LSTM) circuits. One disadvantage of LSTM cells is that they require computing and storing multiple gating neuron responses at each step.

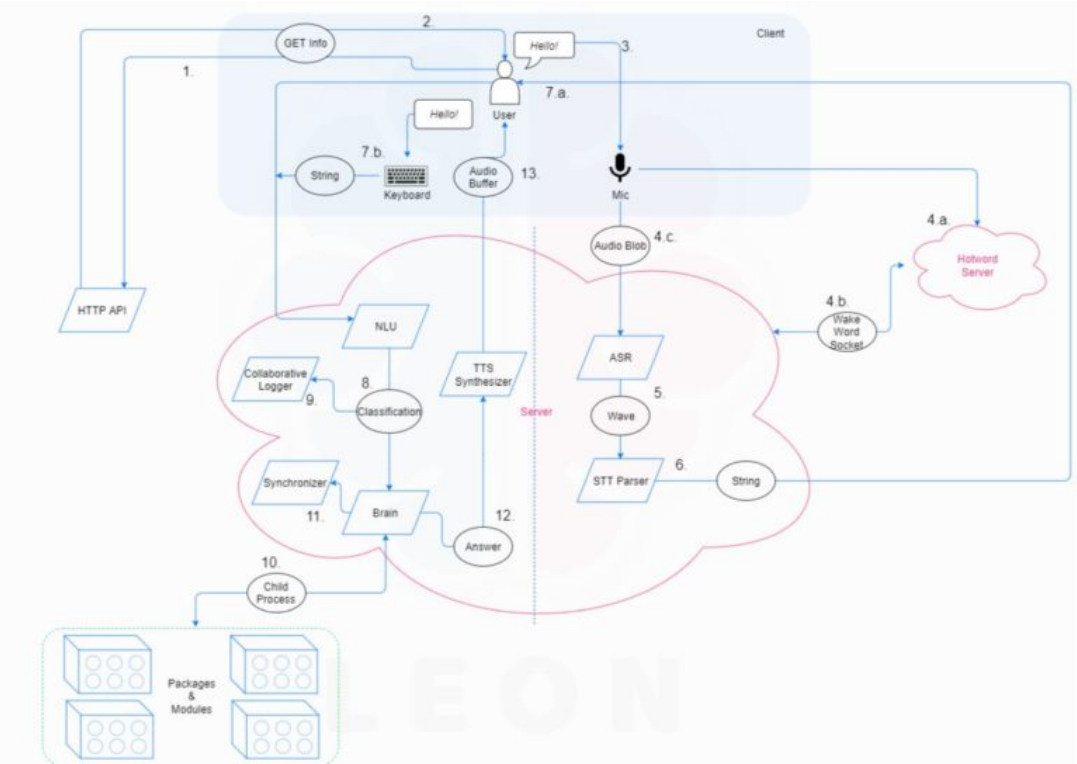


Figure 5.1: Detailed System Layout

Since the forward and backward recurrences are sequential, this small additional cost can become a computational bottleneck. By using a homogeneous model we have made the computation of the recurrent activations as efficient as possible: com-

puting the ReLu outputs involves only a few highly optimized BLAS operations on the GPU and a single point-wise nonlinearity.

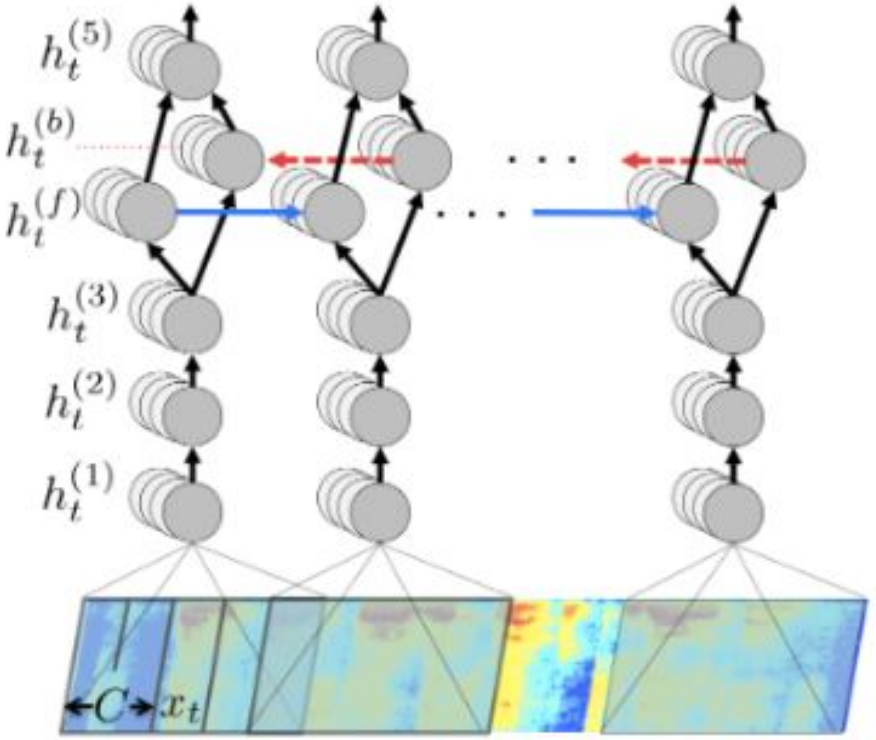


Figure 5.2: Structure of RNN model and notation

5.2 Query Analyzer

The system performs Natural Language processing on the input string. Figure 4.3 shows the steps involved in the process. This model consist of combination of analysis like lexical analysis, Query Analysis, Morphological Analysis, Syntactic Analysis, Semantic Analysis. The output obtained is a JSON data which is then Fed into a Response generator.

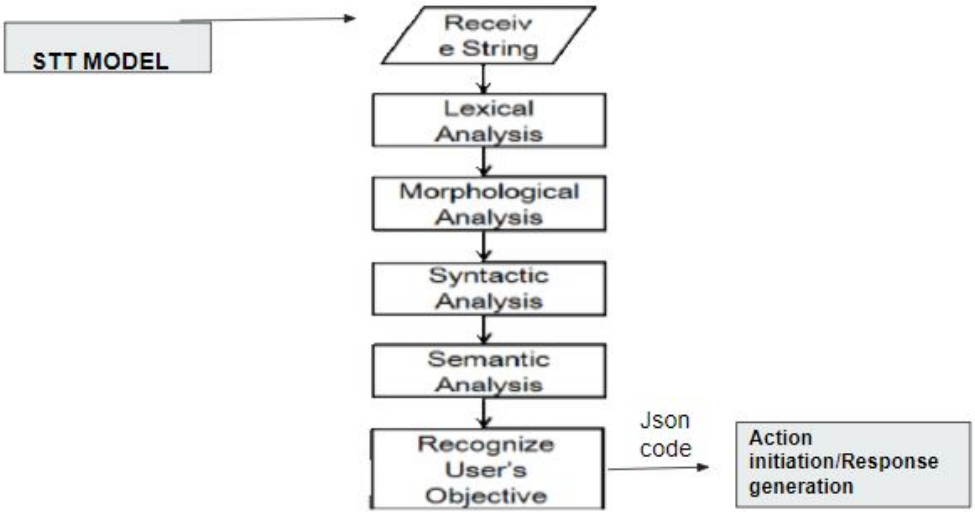


Figure 5.3: Query Analyzer

5.2.1 Lexical Analysis

Lexical Analyzer is the first state in translation. It is called by parser to fulfill the demand of words type. It generates the tokens for the requested words as their types and then handover to the parser for further processing converts sequence of characters into a sequence of tokens. A lex file contains, mainly, set of regular expressions or word patterns that will be applied on each word to recognize the words as a valid token. These regular expressions are written according to the language we are using. So Lexical Analyzer performs two things, mapping of source language words to target language words and returns the appropriate token of target language word to the parser.

5.2.2 Morphological Analysis

Identifies, analyzes, and describes the structure of a given languages linguistic units. It analysis shape of the text which defers by the positioning of different words in it. It provides a primary source of evidence of facilitation between words formed from

the same morpheme (i.e.,morphological relatives). Generally, target (second presentation) decision latency's and error rates are reduced in the context of morphological related primes (first presentation).

5.2.3 Syntactic Analysis

Syntactic parser analyze sentences in terms of a grammar and parts of speech. This analysis does not attempt to identify constituents that represent similar or related meanings.It analyzes texts, which are made up of a sequence of tokens, to determine their grammatical structure. Syntax analyzer validates the English Text query whether the input query is syntactically correct.

5.2.4 Semantic Analysis

Relates syntactic structures from the levels of phrases and sentences to their language-independent meanings. semantic analysis, attempts to analyze sentences based on constituents that represent concepts or meaning.

5.3 Response Generator

The system accepts a JSON code from the previous module and then processes it to perform the required action. Figure 4.4 shows the various steps involved in the process.

5.3.1 Action or Response Analyzer

Analyses if the given JSON is an action command or a question and diverts the control flow to action block user expects an action or response block if it is a response

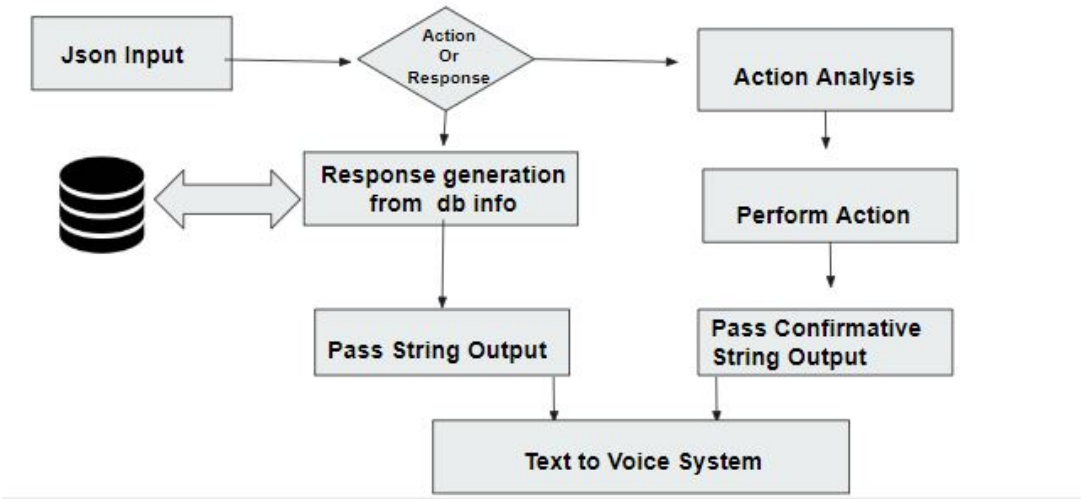


Figure 5.4: Response Generator

accordingly.Implemented using a desicive tree logic that analyse the Noun,Verb combinations.

5.3.2 Action Block

Analyses the action command from JSON and initiates the action performance that is most appropriate and passes a confirmation string when action is complete.

5.3.3 Response block

Analyses the question from JSON and generates a response that is most appropriate from data in Database or earlier question buffer using n dimensional search RNN and passes a generated string response to the next module. The response generated is a syntactically correct language expression having meaningful intents in the given language and are valuable information in the context of the question.

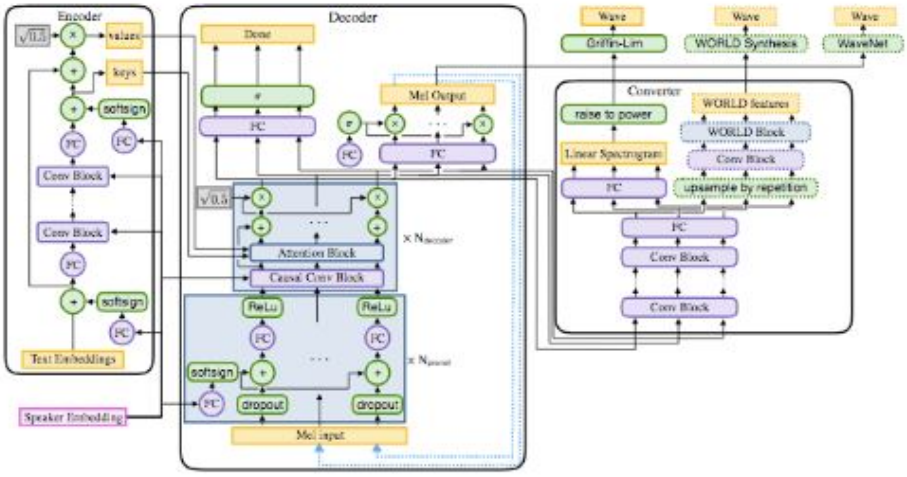


Figure 5.5: Text to Voice System

5.4 Text to Voice System

Encoder: A fully-convolutional encoder, which converts textual features to an internal learned representation.

Decoder: A fully-convolutional causal decoder, which decodes the learned representation with a multi-hop convolutional attention mechanism into a low-dimensional audio representation (mel-scale spectrograms) in an autoregressive manner.

Converter: A fully-convolutional post-processing network, which predicts final vocoder parameters (depending on the vocoder choice) from the decoder hidden states. Unlike the decoder, the converter is non-causal and can thus depend on future context information

5.4.1 Text Processing

Text preprocessing is crucial for good performance. Feeding raw text (characters with spacing and punctuation) yields acceptable performance on many utterances. However, some utterances may have mispronunciations of rare words, or may yield

skipped words and repeated words. We alleviate these issues by normalizing the input text as follows:

1. We uppercase all characters in the input text.

2. We remove all intermediate punctuation marks.

3. We end every utterance with a period or question mark.

4. We replace spaces between words with special separator characters

The duration of pauses inserted by the speaker between words. We use four different word separators, indicating (i) slurred-together words, (ii) standard pronunciation and space characters, (iii) a short pause between words, and (iv) a long pause between words.

5.4.2 Encoder

The encoder network begins with an embedding layer, which converts characters or phonemes into trainable vector representations, *he*. These embeddings *he* are first projected via fully-connected layer from the embedding dimension to a target dimensionality. Then, they are processed through a series of convolution blocks described to extract time-dependent text information. Lastly, they are projected back to the embedding dimension to create the attention key vectors *hk*. The attention value vectors are computed from attention key vectors and text em-beddings, $hv = 0.5 (hk + he)$, to jointly consider the local information in he and the long-term context information in *hk*. The key vectors are used by each attention block to compute attention weights, whereas the final context vector is computed as a weighted average over the value vectors *hv*.

5.4.3 Decoder

The decoder generates audio in an auto regressive manner by predicting a group of future audio frames conditioned on the past audio frames. Since the decoder is auto regressive,it must use causal convolution blocks. We choose mel-band log-magnitude spectrogram as the compact low-dimensional audio frame representation. We empirically observed that decoding multiple frames together (i.e. having r greater than 1) yields better audio quality. The decoder network starts with multiple fully-connected layers with rectified linear unit (ReLU) nonlinearities to preprocess input mel-spectrograms (denoted as PreNet). Then, it is followed by a series of causal convolution and attention blocks. These convolution blocks generate the queries used to attend over the encoders hidden states. Lastly, a fully-connected layer output the next group of r audio frames and also a binary final frame prediction (indicating whether the last frame of the utterance has been synthesized). Dropout is applied before each fully-connected layer prior to the attention blocks, except for the first one. L1 loss 6 is computed using the output mel-spectrograms and a binary cross-entropy loss is computed using the final-frame prediction.

5.4.4 Attention Block

A dot-product attention mechanism is used. The attention mechanism uses a query vector (the hidden states of the decoder) and the per-time step key vectors from the encoder to compute attention weights, and then outputs a text vector computed as the weighted average of the value vectors. Empirical benefits from introducing a inductive bias where the attention follows a mono-tonic progression in time is observed. Thus, we add a positional encoding to both the key and the query vectors.

These positional encodings hp are chosen as hp(i) = sin (si/10000k/d)(for even i) or cos (si/10000k/d)(for odd i), where i is the time step index,k is the channel index in the positional encoding,d is the total number of channels in the positional encoding, andsis the position rate of the encoding. The position rate dictates the average slope of the line in the attention distribution,roughly corresponding to speed of speech. For a single speaker, Omega s is set to one for the query, and be fixed for the key to the ratio of output timesteps to input timesteps (computed across the entire data set). For multi-speaker datasets,sis computed for both the key and query from the speaker embedding for each speaker. As sine and cosine functions form an orthonormal basis, this initialization yields an attention distribution in the form of a diagonal line. We initialize the fully-connected layer weights used to compute hidden attention vectors to the same values for the query projection and the key projection. Positional encodings are used in all attention blocks. We use context normalization as a fully-connected layer is applied to the context vector to generate the output of the attention block. Overall, positional encodings improve the convolutional attention mechanism.

5.4.5 Converter

The converter network takes as inputs the activations from the last hidden layer of the decoder, applies several non-causal convolution blocks, and then predicts parameters for downstream vocoders.Unlike the decoder, the converter is non-causal and non-autoregressive, so it can use future context from the decoder to predict its outputs.The loss function of the converter network depends on the type of the vocoder used

- Griffin-Lim Vocoder: Griffin-Lim algorithm converts spectrograms to time-domain audio waveforms by iteratively estimating the unknown phases. We

find raising the spectrogram to a power parameterized by a sharpening factor before waveform synthesis is helpful for improved audio quality, as suggested. L_1 loss is used for prediction of linear-scale log-magnitude spectrograms

- WORLD Vocoder: The WORLD vocoder is based on [6]. As vocoder parameters, we predict a boolean value (whether the current frame is voiced or unvoiced),an F0 value (if the frame is voiced), the spectral envelope, and the a periodicity parameters. We use a cross-entropy loss for the voiced-unvoiced prediction, and L_1 losses for all other predictions as shown in Figure 4.6.

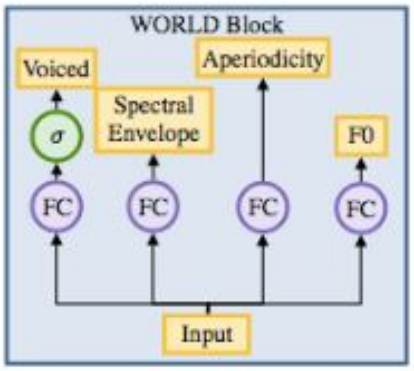


Figure 5.6: Generated WORLD vocoder parameters with fully connected (FC) layers

- WaveNet Vocoder: We separately train a WaveNet to be used as a vocoder treating mel-scale log-magnitude spectrograms as vocoder parameters. These vocoder parameters are input as external conditioners to the network. The WaveNet is trained using ground-truthmel-spectragrams and audio waveforms. The architecture besides the conditioner is similar to the WaveNet. While the WaveNet is conditioned with linear-scale log-magnitude spectrograms, we observed better performance with mel-scale spectrograms, which corresponds to a more compact representation of audio. In addition to L1 loss on mel-scale

spectrograms at decode, L1 loss on linear-scale spectrogram is also applied as Griffin-Lim vocoder.

CHAPTER 6

PERFORMANCE EVALUATION

Here the system acts as a personal assistant answering to your questions, exchange text messages and even communicate offline. A demo of how the system communicates is shown below with suitable screenshots.

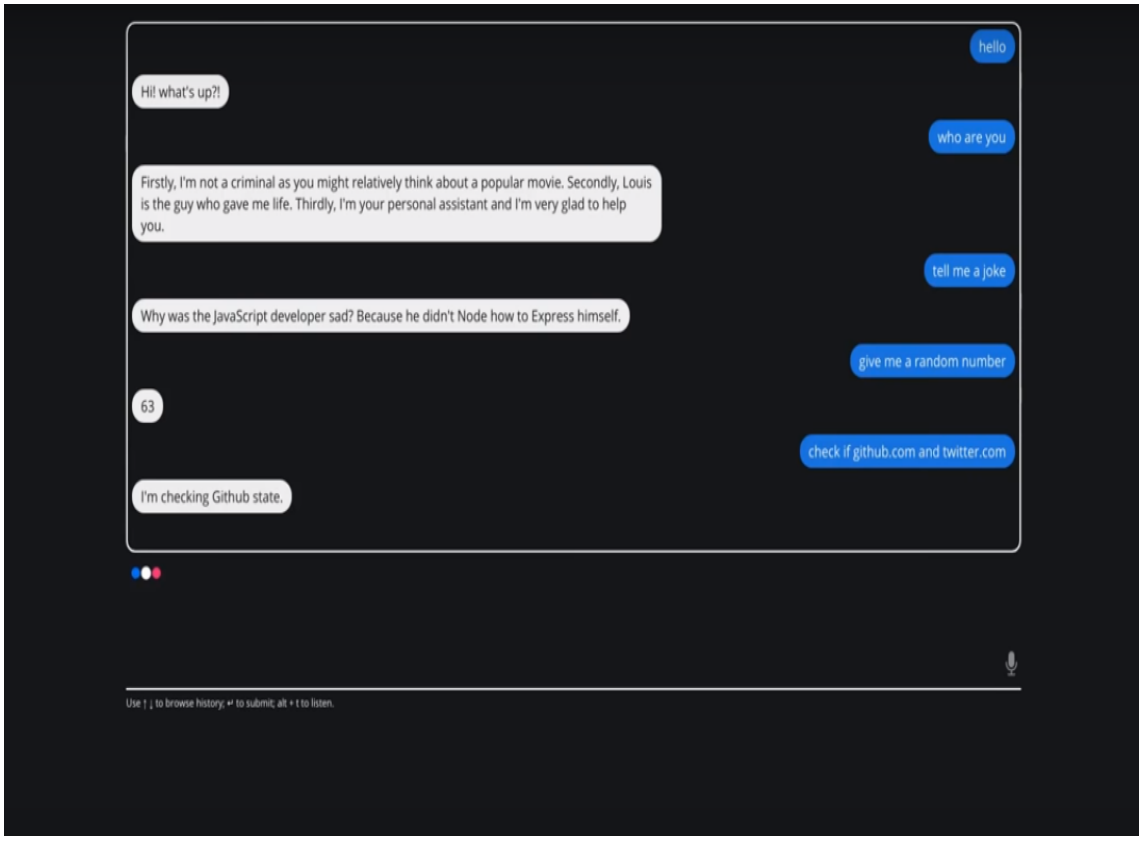


Figure 6.1: Sample output 1

When trained on the combined 2300 hours of data the Deep Speech system improves upon this baseline by 2.4 percent absolute WER (Word Error Rate) and

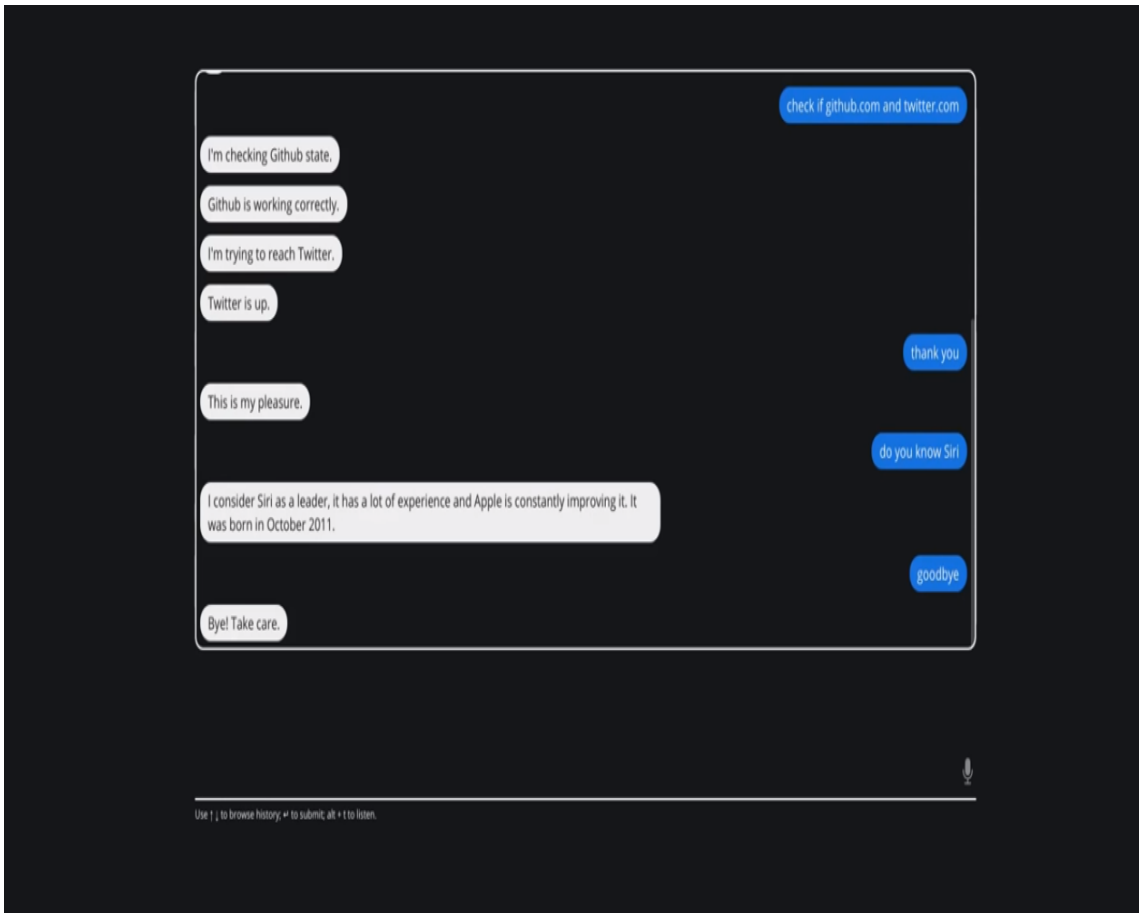


Figure 6.2: Sample output 2

13.0 percent relative. The model from Maas et al. (DNN-HMM FSH) achieves 19.9 percent WER(Word Error Rate) when trained on the Fisher 2000 hour corpus. Graph of this comparison is given in figure 5.3.

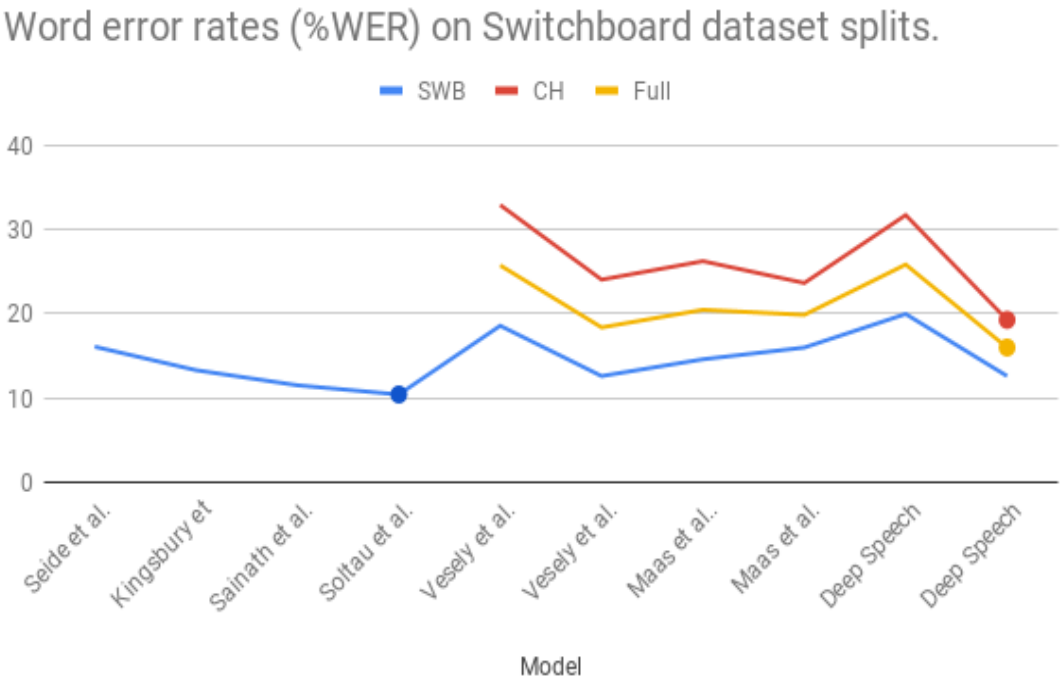


Figure 6.3: Compared WER(Word Error Rate) on Switchboard dataset splits.

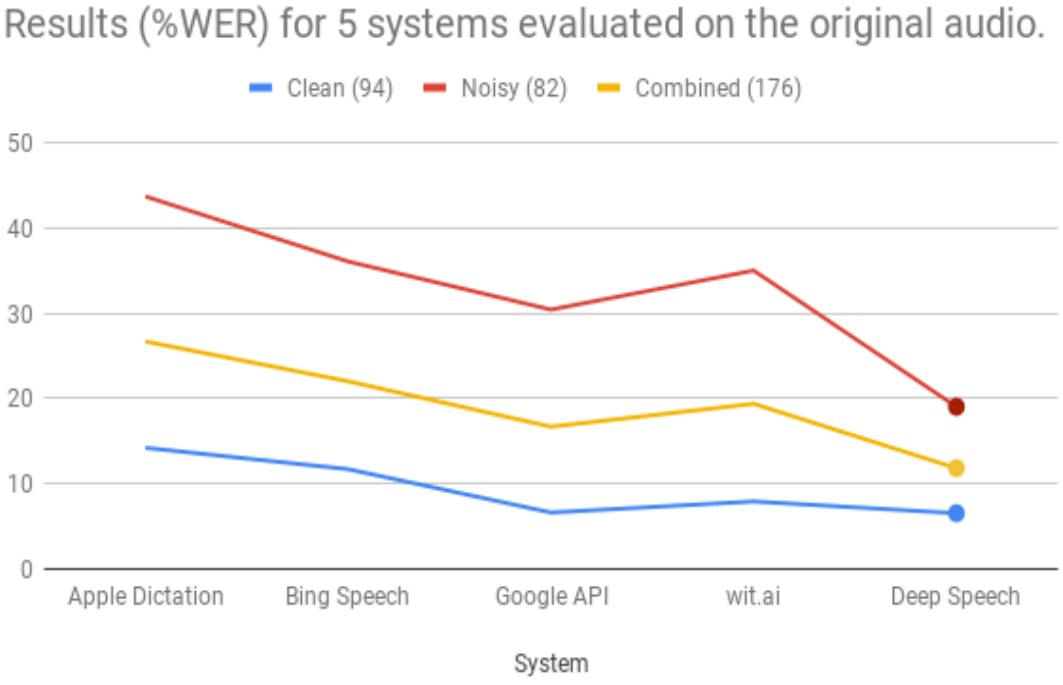


Figure 6.4: Results WER(Word Error Rate) for 5 systems evaluated on the original audio. Scores are reported only for utterances with predictions given by all systems.

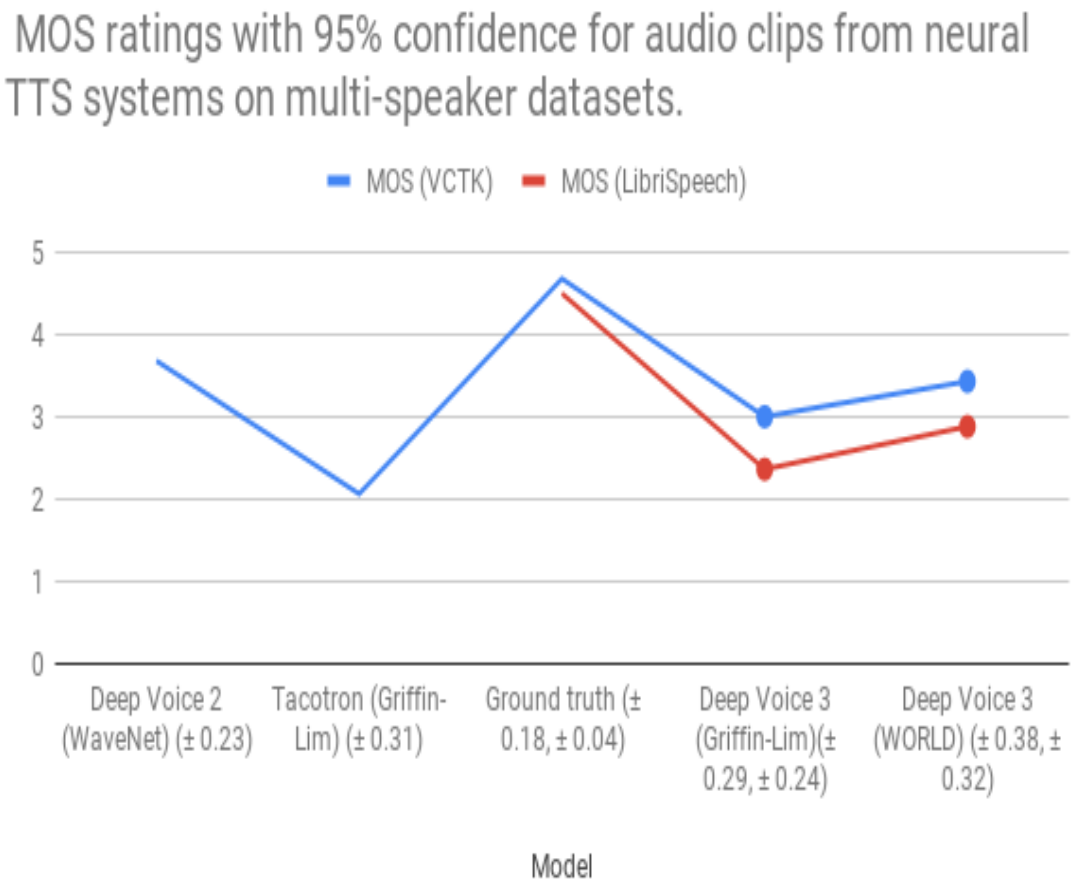


Figure 6.5: : MOS ratings with 95 perecentconfidence intervals for audio clips from neural TTS systems on multi-speaker datasets.

CHAPTER 7
CONCLUSION

The Proposed system takes the input and processed through a multi-microphone array and processed to generate an intermediate noise free audio signal which is then passed to an end-to-end speech system, where deep learning supersedes the processing stages. Combined with a language model, this approach achieves higher performance than traditional methods on hard speech recognition tasks while also being much simpler. These results are made possible by training a large recurrent neural network (RNN) using multiple GPUs and thousands of hours of data. Because this system learns directly from data, we do not require specialized components for speaker adaptation or noise filtering. In fact, in settings where robustness to speaker variation and noise are critical,it produces a higher accuracy on output.

The output is further processed by a Query Analyzer block to check for the structural and semantic intents of the speech input, The recieved text block (string) is analyzed by a lexical analyzer, Morphological Analyzer, Syntactic Analyzer Semantic Analyzer to generate a JSON code that contains information of the input split to language blocks(noun,verb,intent etc). The JSON response is further processed by the Response generator module to analyse the query's intent, to actions or questions. A suitable process section is then activated and a confirmatory statement of action or a response for the question is generated and passed to the next module.

A neural text-to-speech system based on a novel fully-convolutional sequence-to-sequence acoustic model with a position-augmented attention mechanism is used to convert the text to Speech output. The common error modes in sequence-to-sequence

speech synthesis models are fully avoided. The model is agnostic of the waveform synthesis method, and adapt it for Griffin-Lim spectrogram inversion, WaveNet, and WORLD vocoder synthesis. The architecture is capable of multispeaker speech synthesis by augmenting it with trainable speaker embeddings, a technique used in existing system. The text to speech system includes text normalization and performance characteristics, and demonstrate state-of-the-art quality through extensive MOS evaluations.

Future work will involve improving the implicitly learned grapheme-to-phoneme model, jointly training with a neural vocoder, and training on cleaner and larger datasets to scale to model the full variability of human voices and accents from hundreds of thousands of speakers to enable better synthetic speech production. Along with improving the RNN based response generator and analyzer by training it with real world question answer data sets.

REFERENCES

[1] Anusuya, M. A., and Shriniwas K. Katti. Khilari, Prachi, and V. P. Bhope. *"Speech recognition by machine, a review."*, arXiv preprintarXiv:1001.2267 (2010). "A review on speech to text conversion methods." *International Journal of Advanced Research in Computer Engineering Technology 4.7 (2015).*

[2] V. V. Vidyadhara Raju, P. Gangamohan, Suryakanth V Gangashetty, and Anil kumar Vuppala. Raju, V. V., Gangamohan, P., Gangashetty, S. V., kumar Vuppala, A. (2016, November). *Application of prosody modification for speech recognition in different emotion conditions. In 2016 IEEE Region 10 Conference (TENCON) (pp. 951-954). IEEE.*

[3] B. Pendharkar, P. Ambekar,P. Godbole, S. Joshi, and S. Abhyankar. Moore, T. R. (1994). *"Twenty things we still dont know about speech proc. CRIM"*. In FORWISS *Workshop on Progress and Prospects of speech Research and Technology.*

[4] Hannun, A., Case, C., Casper, J., Catanzaro, B., Diamos, G., Elsen, E., Ng, A. Y. (2014). *"Deep speech: Scaling up end-to-end speech recognition"*. arXiv preprint arXiv:1412.5567 (2014).

[5] Ping, W., Peng, K., Gibiansky, A., Arik, S. O., Kannan, A., Narang, S., Miller, J. (2017) *"Deep voice 3: Scaling text-to-speech with convolutional sequence learning."*.arXiv preprint arXiv:1710.07654. (2018)

[6] Morise, M., Yokomori, F.,Ozawa, K. (2016). *"WORLD: a vocoder-based high-quality speech synthesis system for real-time applications."* (2016) *I*EICE TRANSACTIONS on Information and Systems, 99(7), 1877-1884.

[7] Wang, Y., Skerry-Ryan, R. J., Stanton, D., Wu, Y., Weiss, R. J., Jaitly, N., Le, Q. (2017). *" Tacotron: Towards end-to-end speech synthesis"*.,arXiv preprint arXiv:1703.10135. (In Interspeech, 2017.)

[8] Hae-Duck J. Jeong, Sang-Kug Ye, Jiyoung Lim, Ilsun You, and WooSeok Hyun. *"A Computer Remote Control System Based on Speech Recognition Technologies of Mobile Devices and Wireless Communication Technologies"*. Computer Science and Information Systems 11(3):10011016 DOI: 10.2298/CSIS130915061J.