TAMIL TEXT TO ENGLISH EMOTIONAL SPEECH CONVERSION WITH UNL FOR TRANSLATION

by

ARCHANA B 2011103505
SIVA M R 2011103536
JAYAVASANTH R 2011103562

A project report submitted to the

FACULTY OF INFORMATION AND COMMUNICATION ENGINEERING

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

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE AND ENGINEERING

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

ANNA UNIVERSITY, CHENNAI – 25

APRIL 2015

BONAFIDE CERTIFICATE

Certified that this project report titled **TAMIL TEXT TO ENGLISH EMOTIONAL SPEECH CONVERSION WITH UNL FOR TRANSLATION** is the *bonafide* work of **ARCHANA B (2011103505)**, **SIVA M R (2011103536)** and **JAYAVASANTH R (2011103562)** who carried out the project work under my supervision, for the fulfillment of the requirements for the award of the degree of Bachelor of Engineering in Computer Science and Engineering. Certified further that to the best of my knowledge, the work reported herein does not form part of any other thesis or dissertation on the basis of which a degree or an award was conferred on an earlier occasion on these or any other candidates.

Place: Chennai
Date:

Dr.RAJESWARI SRIDHAR
Assistant Professor(Senior Grade)
Department of Computer Science and Engineering
Anna University, Chennai – 25

COUNTERSIGNED

Head of the Department,
Department of Computer Science and Engineering,
Anna University Chennai,
Chennai – 600025

ACKNOWLEDGEMENT

We express our deep gratitude to our guide, **Dr. Rajeswari Sridhar** for guiding us through every phase of the project. We appreciate her thoroughness,tolerance and ability to share her knowledge with us. We thank her for being easily approachable and quite thoughtful. Apart from adding her own input, she has encouraged us to think on our own and give form to our thoughts. We owe her for harnessing our potential and bringing the best in us. Without her immense support through every step of the project, we could never have done it to this extent.

We are extremely grateful to **Dr.D.Manjula**, Head of the Department of Computer Science and Engineering, Anna University, for extending the facilities of the Department towards our project and for her unstinting support.

We express our thanks to the panel of reviewers **Dr.V.Mary Anita Rajam**, **Dr. V. Vetriselvi** and **Dr. S. Sudha** for their valuable suggestions and critical reviews throughout the course of our project. We thank our parents, family, and friends for bearing with us throughout the course of our project and for the opportunity they provided us in undergoing this course in such a prestigious institution.

ARCHANA B **SIVA M R** **JAYAVASANTH R**

ABSTRACT

This project in the field of Natural Language Processing aims at translating an input Tamil sentence into an equivalent spoken English translation of the sentence. This brings together two major domains in NLP,i.e, machine translation and TTS(Text to Speech). Natural Language Processing is the field of computer science that deals with interactions between computer and human languages. In this project, a hybrid approach of translation called Universal Networking Language(UNL) is employed following which a Text-to-Speech system is used that gives a neutral emotionless voice as the output. Emotion based prosody is added to this neutral voice to give a rich, humanised voice output.

Universal Networking Language (UNL) is a declarative formal language that is used to represent semantic data extracted from natural language texts. During the process of translation between two languages, it is used as a pivot language. As UNL is language-independent, it is capable of storing information in a way that is independent of the original language from which the text was derived. This generic way of representation makes UNL very efficient for translation and so it has been employed in our project.

Once the text has been translated with the help of the UNL formalism, the emotional voice is synthesised from the neutral robotic voice by modifying certain key parameters that include fundamental frequency, intensity, and duration.

திட்டப்பணி சுருக்கம்

பரந்த இக்கண்டத்தில் பல்வேறு மொழிகள் வழங்கி வரப்படுகின்றன. ஆகவே, ஒரு மொழி தனில் இருந்து வேறொரு மொழிக்கு மொழி மாற்றம் செய்தல் இன்றியமையாதது ஆகின்றது. ஆதலால், இயற்கை மொழிச் செய-லாக்கத்தில் மொழி மாற்றம் ஒரு தன்னிச்சையான இடத்தைப் பெற்றுள்ளது.

இத்திட்டபணியானது ஓர் தமிழ் வாக்கியத்தை ஆங்கில மொழியில் பொருள் மாறாது மொழி மாற்றம் செய்து, பின் அவ்வாக்கியத்தை உணர்ச்சியுடன் கூடிய ஒலியாய் உருவகிக்கும் நோக்கோடு செய்யப்பட்ட-து ஆகும்.

மொழி மாற்றம் செய்வதற்கு உலகளாவிய தொடர்பு மொழி என்னும் இடைநிலை மொழியானது பயன் படுத்தப்பட்டுள்ளது. இம்மொழியானது, சொற்பொருள் அறிவிக்கும் தீர்மான மொழியாகும். இவ்வாறான இடை மொழியொன்றைப் பயன் படுத்துவதால் எம்மொழி தனில் இருந்தும் ஏனைய மொழிகள் அனைத்திற்கும் மொழி மாற்றம் செய்ய வழி வகுக்-கின்றது. ஒலியின் வீச்சு, தீவிரம் மற்றும் காலம் ஆகிய தன்மைகளில் மாற்றப்படின் ஒலியில் வெளிப்படும் உணர்ச்சியும் மாறுபடும். இவ்வழி-யில், ஒலியில் உணர்ச்சிகள் ஏற்றப்படுவதால் அவ்வொலி மனிதக் குரல் போல் மெருகேறுகின்றது.

CONTENTS

ABSTRACT – ENGLISH iii

ABSTRACT – TAMIL iv

List of Figures ix

List of Tables x

LIST OF ABBREVIATIONS xi

1 INTRODUCTION 1

1.1 Overview 1

1.2 Problem Description 2

1.2.1 Scope 2

1.3 Contribution 3

1.4 Organisation of thesis 3

2 RELATED WORK 5

2.1 Translation System 5

2.1.1 Rule-Based Machine Translation (RBMT) 5

2.1.2 Phrase Based Translation 6

2.1.3 Example Based Translation System 7

2.1.4 Context Based Machine Translation 9

2.1.5 UNL Based Approach 9

2.2 Prosody Generation in Text to Speech System 10

2.2.1 Text to Speech System 10

2.3 Observations from Survey 12

3 REQUIREMENTS ANALYSIS 14

3.1 Functional Requirements 14
3.2 Non - Functional Requirements 14
3.2.1 User Interface 14
3.2.2 Hardware Requirements 15
3.2.3 Software Requirements 15
3.3 System Models 15
3.3.1 Use case Diagram 15
3.3.2 Overall activity diagram 16
3.4 Constraints and Assumptions 18
3.4.1 Constraints 18
3.4.2 Assumptions 18
4 SYSTEM DESIGN 19
4.1 System Architecture 19
4.2 UI Design 21
4.3 Module Design 22
4.3.1 POS Tagger 22
4.3.2 Morphological Analyser 22
4.3.3 Tamil WORDNET 22
4.3.4 UNL Enconverter 23
4.3.5 UNL DeConverter 23
4.3.6 Emotion-Features Database Construction 24
4.3.7 Emotion Identification 26
4.3.8 Voice Modification 26
4.4 Complexity Analysis 26
4.4.1 Time Complexity 26

4.4.2 Complexity of the Project 26

5 SYSTEM DEVELOPMENT 28

5.1 Prototype Description 28

5.1.1 POS Tagger 28

5.1.2 Morphological Analyser 28

5.1.3 UNL Enconverter 30

5.1.4 Emotion Identifier 30

5.1.5 Voice Modifier 30

5.2 Algorithm 31

5.2.1 Overall Algorithm 31

5.2.2 POS Tagger Algorithm 31

5.2.3 Enconversion Algorithm 31

5.2.4 Deconversion Algorithm 33

5.2.5 Algorithm for Estimation of Fundamental Frequency 34

5.2.6 Voice Modification Algorithm 35

6 RESULTS AND DISCUSSION 36

6.1 Dataset for Testing 36

6.2 Sample Output 36

6.3 Performance Evaluation 39

6.3.1 BLEU Score 40

6.3.2 NIST 41

6.3.3 METEOR 43

6.3.4 Fluency and Adequacy 43

6.3.5 Evaluation of emotional speech 45

7 CONCLUSION 47
7.1 Summary 47
7.2 Criticism 47
7.3 Future Work 48
A Rules used in the system 50
A.1 Deconversion 50
Bibliography 51

LIST OF FIGURES

3.1 Use Case Diagram 16

3.2 Activity Diagram 17

4.1 Block Diagram 20

4.2 User Interface 21

5.1 Organisation of source code 29

5.2 Representation of a sample output of the enconverter 30

6.1 BLEU scores: Our system vs Google translator 41

6.2 NIST scores: Our system vs Google translator 42

6.3 METEOR scores: Our system vs Google translator 44

6.4 Fluency scores: Our system vs Google translator 45

6.5 Adequacy scores: Our system vs Google translator 45

LIST OF TABLES

4.1 Time complexity of each module 27
6.1 Fluency score description 44
6.2 Adequacy score description 44
6.3 Survey Results on Identifying the Emotion present in Modified Voice 46
6.4 Scores for Emotion present in the Modified Voice 46

LIST OF ABBREVIATIONS

BLEU	Bilingual Evaluation Understudy
CBMT	Context Based Machine Translation
GUI	Graphical User Interface
MATLAB	MATrix LABoratory
METEOR	Metric for Evaluation of Translation with Explicit ORdering
MIT	Madras Institute of Technology
MT	Machine Translation
NLP	Natural Language Processing
POS	Part Of Speech
RBMT	Rule-Based Machine Translation
TTS	Text To Speech
UNDL	Universal Networking Digital Language
UNL	Universal Networking Language
UW	Universal Word

CHAPTER 1

INTRODUCTION

1.1 OVERVIEW

Natural language processing is a field that lies at the intersection between artificial intelligence and linguistics[1]. It is primarily concerned with the interactions that occur between computers and human languages. NLP is within the domain of human–computer interaction. The predominant challenge in NLP involves the creation of a medium through which computers assume the ability to comprehend natural language and hence derive meaning from natural language input. Computational linguistics is a sub domain under NLP, which focuses specifically on language-related work such as language modelling and representation[2]. Machine Translation is one such field of computational linguistics which involves the development of a system for translation of given text from a source language to the target language.

Translation has been in existence ever since written literature started gaining traction. This was primarily due to the fact that knowledge in one language could easily be propagated to other languages, and hence to other cultures with the help of translation. In modern times, the preponderance of literature has made it mandatory to automate the process of translation which would otherwise be tedious if done manually . Thus various machine translation approaches were developed over the past decades[3]. Communication in human languages is embedded in context and so the major challenge faced by machine translation involves the identification of the context in which words are uttered. If the context has been properly identified, then the translation can be accomplished accurately. Some of the major approaches proposed for machine translation include Rule Based, Phrase Based , Context Based, Example Based and Hybrid approach.

In this project, we have developed a system that is capable of translating a Tamil sentence to English using Universal Networking Language (UNL) as an intermediate. The translated sentence is given to a Text-to-Speech system and then finally prosody is added to give an output voice that is humanised. Since, the final output is voice, it can be of great help to visually challenged people and also to illiterate. Many of the rich Tamil texts can be converted into their English equivalent using this system for the people in the west to understand and appreciate Tamil literature.

1.2 PROBLEM DESCRIPTION

With a Tamil sentence as an input, the MT system should translate it to a correct and equivalent English sentence with no loss in meaning.The sentence should also be voiced with the emotion conveyed in it. When a paragraph is given as input, it should translate the paragraph sentence by sentence.

1.2.1 Scope

English has become a universal language of trade and communication. Hence, it becomes necessary to translate regional languages to English. Tamil is one of the most widely spoken regional languages in India. In addition, Tamil is also spoken in several parts of Srilanka, Singapore, Malaysia, and Arab countries. Tamil has a rich history of literature and in order for non-Tamil speakers to understand and appreciate this great legacy, translation is very essential. Also, official documents, school samacheer text books and even dialects in Tamil could be hereby voiced providing a great value.

1.3 CONTRIBUTION

As UNL is a language independent framework, the translation system that we have created represents a generic approach to translation. This is the first system in which UNL rules have been applied to the Tamil language in order to carry out translation. The basis of the translation system also relies heavily on the morphological analyser and the POS-tagger and these two systems were also created by us during the project. Once the sentence has been given to the Text-to-Speech system, a neutral voice is obtained. This neutral voice is modified by adding prosody so that emotion can be conveyed. The emotional voice synthesis is carried out by manipulating certain key features such as fundamental frequency, intensity and duration. This system can thus be developed into a very efficient system for translation from Tamil text to English speech.

1.4 ORGANISATION OF THESIS

Chapter 2 elaborates on the major approaches to machine translation. Chapter 3 explains the requirements analysis of the system. It explains the functional and non-functional requirements, constraints and assumptions made in the implementation of the system and the various UML diagrams. Chapter 4 explains the overall system architecture and the design of various modules along with their complexity. Chapter 5 gives the implementation details of each module, describing the algorithms used. Chapter 6 elaborates on the results of the implemented system and gives 4 an idea of its efficiency. It also contains information about the dataset used for testing and other the observations made during testing. Chapter 7 concludes the thesis and discusses some issues.

CHAPTER 2

RELATED WORK

This project consists of two components:

1. Translation
2. Adding Emotion to Speech

This Chapter discusses the various approaches that have been developed in the area of Machine Translation. The algorithms that have been developed by researchers to convert read speech to humanised speech has also been discussed.

2.1 TRANSLATION SYSTEM

Translation is one domain of computational linguistics where many different approaches have been formulated. Some of these approaches are :

- Rule based translation
- Phrase based translation
- Example based translation
- Context based Translation
- UNL approach

2.1.1 Rule-Based Machine Translation (RBMT)

RBMT is based on rules from source and target languages and it depends on the morphology and grammar of those languages for translation. The main steps in rule based translation are

1. Analyse the input sentence grammatically (Tokenization and parsing)
2. Generating equivalent sentence in target language in accordance with target grammar. (Word mapping and Semantics)

Components of a rule based translator are:

1. Source language analyser
2. Source language parser
3. Translator (lookup between source and target language)
4. Target language generator
5. Target language parser

SYSTRAN, one of the very old and successful machine translation systems works on rule based translation to translate[4].

In general RBMT is not complete by itself as its a very rigid system which relies completely on the knowledge base(semantics of source and target language). The system has to be trained to increase its efficiency.

2.1.2 Phrase Based Translation

In phrase based translation, the input is translated by grouping together words into phrases. So phrases are mapped from the source language to the target language, which is an improvement over the traditional word to word mapping.

A statistical approach to phrase based translation has been discussed by Philip Koehn,et al[5]. This paper shows that extremely high performance can be achieved by heuristic learning and lexical weighting of phrase translations.

The performance of a phrase based translation system can be further enhanced by incorporating dialog acts[6].The dialog act is primarily used to characterize the nature of the sentence. It is capable of giving the additional information of how some data has been presented, and not just the contents of the data. In this case, every utterance is tagged with a DA label (made possible by a DA Tagger). After having completed the phrase based translation, these Dialog Acts are used to further enhance the performance. Farsi English, Japanese English and Chinese English have been translated using this approach. BLEU scores were improved from +1 to +4 upon addition of Dialog Acts.

David Chiang proposed a model that makes use of hierarchical phrases(phrases that contain sub phrases)[7]. In this model,logically re ordered phrases are translated by creating rules within the framework of a Context Free grammar.

The phrase based translation system can be extended by integrating certain morphological and syntactic features[8]. In the English to Hindi translation, additional processing is done as follows:

- The English input sentence is reordered according to the Hindi syntax by applying simple transformation rules on the English parse tree.(syntactic processing)
- A suffix separation program is used for making use of Hindi suffixes.(morphological processing)

For a morphologically rich language like Hindi, this approach is suitable which is evident in the fact that translation fluency for English to Hindi translation is observed to have increased from 10% to 46%.

2.1.3 Example Based Translation System

Humans learn to translate by way of examples. The example based translation approach is also modelled on this. It tries to draw similarity between parts of text at the word or stem level. Due to this mapping, an extensive bilingual or parallel corpus is required. As it is rather difficult to build such corpora, we make use of the world wide web that can be thought of a large corpus of data. The example based approach involves

3 steps:

- Matching
- Transfer
- Recombination

Matching

The corpus is searched for a sequence of words in the source text and if there is a match, its score is incremented. i.e, greater the number of matches of the source text, greater will be its score. Those word sequences that have a score lesser than some threshold value are discarded from the corpus.

Transfer

Once the word sequence has been identified, the corresponding sequence is extracted from the target language. The total score is to be calculated as:

$$Total = TranslationScore * MatchScore$$

Recombination

All translated word sequences are concatenated and the best N possibilities of recombinations which give the entire translated sentence are chosen.

Y. Choueka Bar, et al discuss the application of the example based approach for Arabic to English translation[9]. But this approach has few drawbacks as it gives way for many exceptions to arise, for which new rules need to be formulated.

Also, according to Yves Lepage, et al., pure example based approach is feasible only by adopting proportional analogy[10].

2.1.4 Context Based Machine Translation

This method is similar to example based approach but does not require a parellel corpus like the other one.It is efficient for languages, where translation is dependant on the context rather than rules. A bilingual dictionary, source language corpus and target language corpus are maintained.N-grams are identified from a given text and the words in each of these N-grams are converted to target language with the help of the bilingual dictionary.

More than one n-gram can be generated in the target language for a single source language text.These N-grams are then ranked to find the best possible translation.Finally a decoder produces the translated output. Spanish to English, Arabic to English and Chinese to English translations have been experimented using this CBMT and they have attained a BLEU score of 0.6950[11].

2.1.5 UNL Based Approach

Universal networking language [12] is a declarative formal language specifically designed to represent semantic data extracted from natural language texts.It can be used as a pivot language in interlingual machine translation systems or as a knowledge representation language in information retrieval applications.

UNL expresses information and knowledge in the form of semantic network. The semantic network of the UNL is a directed graph. Its nodes are called Universal Words (UW).

There are 46 relations in the UNL, such as ‘agt’ (agent), ‘gol’ (goal/-final state) , ‘obj’ (object), etc. They are used to connect every two UWs to construct the semantic networks of UNL Expressions.

Translation from a source language to target language involves converting source language to UNL (Enconversion) and then converting

UNL structure to the target language (Deconversion)

A UNL based MT system for conversion from Punjabi to Hindi has achieved a BLEU score of 0.72 [13].

BLEU (Bilingual Evaluation Understudy) is an algorithm for evaluating the quality of machine- translated text. It is used to determine how close the machine translated output is to that of a human translated one.

To summarise the above approaches, it can be said that each approach has some drawbacks. Rule based translation systems are very rigid and dependent on the semantics of the source and target languages. Example based translation requires a large corpus. Thus, as a part of our project we will be working with UNL, which is a generic framework and independent of any language. Also, this approach does not require the presence of a corpus. Owing to these reasons we have chosen to work with UNL which is one of the foremost applications of the **Interlingual** approach. It falls under the **hybrid translation** from the source language to the UNL framework(called enconversion).

2.2 PROSODY GENERATION IN TEXT TO SPEECH SYSTEM

2.2.1 Text to Speech System

A text-to-speech (TTS) system converts normal language text into speech. This synthesis of speech from text can be carried out by the following approaches:

- Concatenative synthesis [14]
- Formant synthesis [15]
- Articulatory synthesis [16]
- HMM based synthesis [17]
- Sinewave synthesis [18]

Concatenative synthesis

It is based on the concatenation (or stringing together) of segments of recorded speech.

Formant synthesis

The synthesized speech output is created using additive synthesis and an acoustic model.Parameters such as fundamental frequency, voicing, and noise levels are varied over time to create a waveform of artificial speech.

Articulatory synthesis

It refers to computational techniques for synthesizing speech based on models of the human vocal tract and the articulation processes occurring there.

HMM-based synthesis

HMM-based synthesis is a synthesis method based on hidden Markov models.

Sinewave synthesis

Synthesizes speech by replacing the formants (main bands of energy) with pure tone whistles.

MaryTTS

MaryTTS is an open-source, multilingual Text-to-Speech Synthesis platform written in Java. It was originally developed as a collaborative project of DFKI's Language Technology Lab and the Institute of Phonetics at Saarland University. It is now maintained by the Multimodal Speech Processing Group in the Cluster of Excellence MMCI and DFKI.

FLITE

Flite (festival-lite) is a small, fast run-time synthesis engine developed at CMU and primarily designed for small embedded machines and/or large servers.[19]

A TTS can be more natural if prosodic features are also incorporated in it. Prosodic components have to be extracted from text and methods which use Hidden Markov models are found to be more successful.

A Syllable based text to speech synthesis system was developed using a five layer auto associative neural network[20]. The prosodic features extracted from the text will be used for the prosody generation in speech.

Human speech varies depending on the variation of pressure in vocal chord. Different acoustic factors like pitch, amplitude affect the voice.

While trying to humanise a robotic voice, it was found that a set of parameters greatly influenced the prosodic content of the voice. These parameters include fundamental frequency, intensity and duration[21].

2.3 OBSERVATIONS FROM SURVEY

Based on the analysis of various translation approaches, it can be seen that some of these approaches won't suit for a morphologically rich, free word order language.

An example based translation approach requires a very large corpus where every word of the language needs to be present. In languages like Tamil where more and more words can be formed by adding suffixes, this approach would not be efficient.

The same analogy can be said for context based and phrase based translation.In case of a rule based translation, the translation scope is limited to only the two languages involved.New set of rules have to be devised if the text is to be translated to another language.

Hence, we have decided to use an Interlingua - UNL thereby widening the scope for translation.

The emotions in human voice are a result of amplitude,pitch and frequency.By tweaking those in a machine voice, an attempt can be made to humanise that voice.

CHAPTER 3

REQUIREMENTS ANALYSIS

This chapter analyses the requirements needed for building the system.It also models the system using a Usecase Diagram and an Activity Diagram.

3.1 FUNCTIONAL REQUIREMENTS

- The system should be able to translate a given Tamil text to grammatically correct English Sentence with no loss in meaning.
- The system should be able to identify the emotions from the Universal Words in UNL.
- The system should be able to add emotions to the voice output of the TTS engine.
- The system should be able handle various formats of inputs including simple sentences, complex sentences, compound sentences and paragraphs.
- The system should be designed to work optimally and efficiently.
- The system should handle sentences from any domain.

3.2 NON - FUNCTIONAL REQUIREMENTS

3.2.1 User Interface

The UI should be simple and easy to use.User should be able to copy paste Tamil into the input box. The UI should also show the UNL output to the user.The unmodified voice output from TTS should be played after the translation is complete.The modified emotional voice output should follow that, so as to comprehend the difference in tone between the two.

3.2.2 Hardware Requirements

The system should have at least 1 GB RAM to run all the needed tools concurrently.

3.2.3 Software Requirements

- Operating System: Any
- Programming Languages
  - Java
  - PHP
- Tools
  - Eclipse Luna
  - XAMPP
  - Neo4j
  - MATLAB
  - Simple NLG
  - MaryTTS
  - PRAAT
  - Audacity

3.3 SYSTEM MODELS

3.3.1 Use case Diagram

The use case diagram is depicted in Figure 3.1

Description

The user gives a Tamil sentence as input. This sentence is converted to UNL by an enconverter.The deconverter then converts this UNL notation to English text. MaryTTS is used to generate a machine voice for

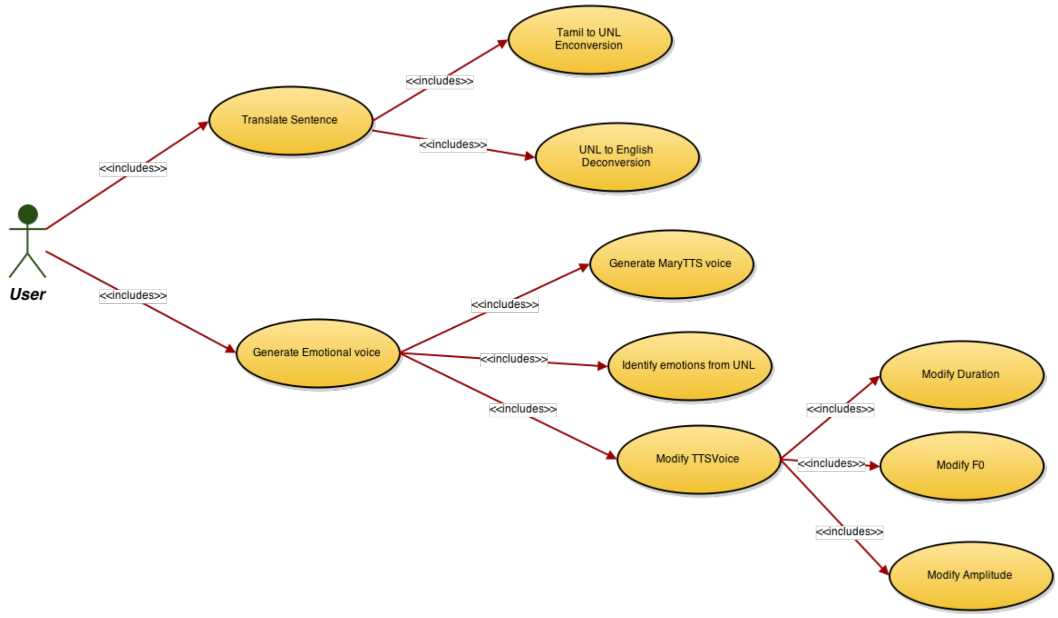

Figure 3.1 Use Case Diagram

the deconverted English text.The UNL is analysed to identify the emotion in the sentence. Based on the identified emotion, the TTS Voice is modified to add prosody to it.

Pre condition

A grammatically correct input sentence with an identifiable emotion is given as input.

Postcondition

The emotion should be correctly identified and an emotional voice is generated

3.3.2 Overall activity diagram

The overall activity diagram of the system is depicted in Figure 3.2

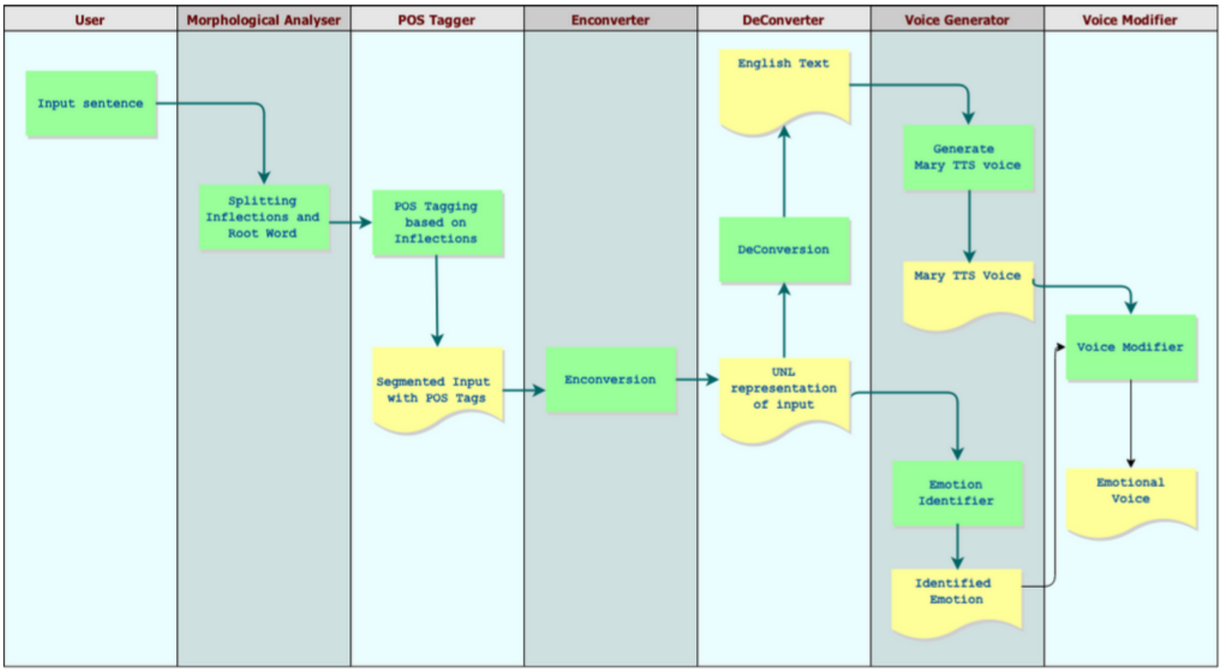

Figure 3.2 Activity Diagram

Description

The user gives a Tamil sentence as input. Each word in the sentence is analysed by an Morphological analyser to find the root word and inflection merged in it.These inflections are then used in POS Tagger to identify the POS tag of the word.This segmented output is sent to the Enconverter,which converts this text to an UNL representation with the help of a Wordnet and custom made rules.The Deconverter then converts this UNL notation to English text with the help of Simple NLG.

MaryTTS is used to generate a machine voice for the deconverted English text.Then, the UNL is analysed to identify the emotion conveyed. Based on the identified emotion, the TTS Voice parameters are modified to generate an emotional voice.

3.4 CONSTRAINTS AND ASSUMPTIONS

3.4.1 Constraints

- The translation is limited by the 20000 Tamil Words in our Wordnet.Words which are not included in those will be transliterated.More words can be added to improve accuracy.
- The accuracy of the UNL relations depend on the available domain information of the input Tamil words in the Wordnet.In cases, where sufficient domain information is not available, relations are assigned based on the inflections arbitrarily.
- Words without inflections depend entirely on the Wordnet to identify the correct POS Tag for them. If those words are not found in the Wordnet, translation would not be perfect.

3.4.2 Assumptions

- The input sentence is grammatically correct and has an identifiable emotion.
- It contains a proper sentence structure and with no spelling mistakes.

CHAPTER 4

SYSTEM DESIGN

This chapter provides a detailed explanation of all the inbuilt modules in the system.

4.1 SYSTEM ARCHITECTURE

The system is depicted in a block diagram as in Figure 4.1. Colour codes are used to indicate modules that have been developed from scratch or modified/unmodified version of an existing standard module.

This system aims at translating a Tamil sentence to English and then voicing it out with emotions ensuring that there is no loss in the meaning conveyed and the emotions are elegantly heard in the output.

The rules for Morphological analyser and POS Tagger are taken from Tamil grammar rules[22].They are developed from scratch and are designed to split each input word in to root word, inflection, tense suffix, count suffix and corresponding POS.

The Wordnet used in this system is actually a NEO4j GraphDB.This DB is extended over an existing graph wordnet[23] and MIT Wordnet. The missing English gloss for the English words are crawled from taWiktionary.

The rules regarding UNL Enconversion are devised based on UNL specifications[12] and Tamil Grammar.The Enconversion module is designed to generate UNL graph from the input.

For Deconversion process a new algorithm has been devised.This algorithm reorders the UNL Graph and generates the English sentence

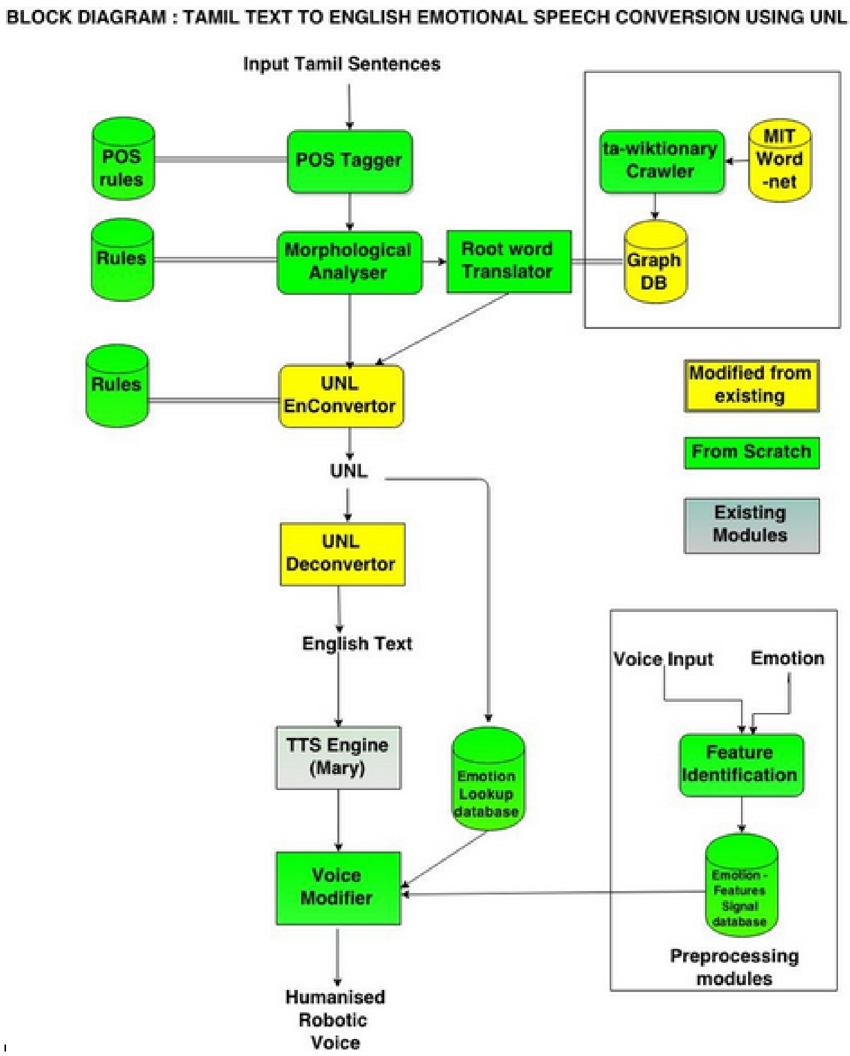

Figure 4.1 Block Diagram

with the help of Simple NLG.

The emotion feature database contains a set of features that vary for each emotion in a voice.It was constructed by analysing various parameters of different sentences with emotion.

For the machine voice, a TTS system known as Mary TTS is used. This produces a robotic voice which is then modified by adding prosody. The prosody is added with the help of MATLAB.

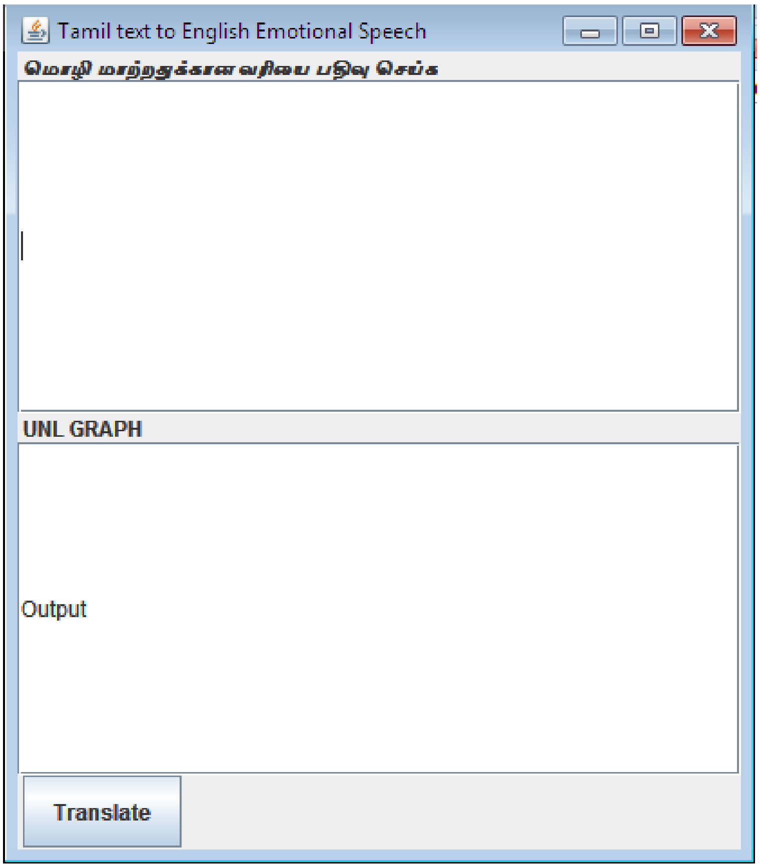

Figure 4.2 User Interface

4.2 UI DESIGN

The UI contains three segments one each for the Tamil input,intermediate UNL representation and the English Output. The input sentences/paragraph are/is inserted in the first one. On clicking the Translate button at the bottom, the intermediate UNL will be displayed in the second text area and the English text will be displayed in the third. Soon after that the TTS voice followed by the modified voice will be played through a MATLAB call.

The user interface is shown in Figure 4.2.

4.3 MODULE DESIGN

4.3.1 POS Tagger

The input sentence is given to this rule-based POS tagger which returns the part of speech of every word in the sentence.In Tamil there are 'vallinammigumidam' where a vallinamei *(க், ச், த், ப்)* will be added to the inflected word, they are scanned from right to left. Once those vallinamei(if any) is removed, the inflections are used to determine the POS tags. A customised tagset tailored to fit the UNL relations and Enconversion module is created and used by this tagger. These tags are used in Morphological analysis and root word extraction.

4.3.2 Morphological Analyser

Tamil is a morphologically rich language. A number of words can be formed from a root word by inflections. But only the root word will be mapped for translation and features like tense, gender and plurality can be extracted from the inflections which will be used in Deconversion to target language. A rule based morphological analyser has been developed for this purpose which takes inflected word with POS tag as input and it returns the root word which is the 'Universal Word' (UW) in the UNL representation.

4.3.3 Tamil WORDNET

The Wordnet consists of words and relationships between them such as hypernymy , hyponymy , holonymy , synonyms and antonyms etc., For the purpose of Word sense disambiguation - the Wordnet becomes essential.For this purpose we would use the Madras Institute of Technology's Tamil Wordnet along with few online dictionaries to fill in the English gloss attribute of the MIT Wordnet. The Wordnet would be deromanized before usage. Also the Wordnet will be converted to a Graph database format with Neo4j.

4.3.4 UNL Enconverter

The process of converting Source text to UNL Graph is called Enconversion. UNL Graph are composed of nodes called as Universal Words (UW) and relations between them.Also each UW has one or more attributes attached to them.The output from a morphological analyser is processed using a manually defined set of rules to form UNL expressions.The root word of each important word in a sentence forms an UW and the suffixes attached to these words determine the relations between Uws. Some words including articles, prepositions and adjectives form attributes of these UWs.

Also the UWs that have to be processed together form a hyper node or a sub graph .The hyper nodes are identified using suffixes of the Tamil words and are represented by a sub tree in UNL Graph.

The equivalent English terms for these Tamil words are taken from the Wordnet based on the POS of the term. If no translation is found, these words are transliterated.

4.3.5 UNL DeConverter

The process of converting UNL Graph to target language text is called Deconversion.

A new algorithm is developed for deconversion.This algorithm traverses the UNL Graph in DFS manner and adds the UWs linearly along with few custom defined connectors in between to form the English sentence.

Every relation in UNL can be attributed to one POS in English language. For example,an agt relation denotes agent and can be attributed to the Subject.Considering an SVOA pattern the subject should come first and so the the node corresponding to the Subject should be traversed first.Consider tmi and tmf relations - if they both exist in the sentence they would contribute to the phrase “ from <tmi> till <tmf> ”. Hence, tmi should be traversed before tmf.Same way ordering between all the relations can be defined.Also the connectors for each relation like “from” for tmi, “till” for tmf , “during” for dur etc., are custom defined.This ordering and connectors are mentioned in the appendix.

The UNL graph is reordered on its relations based on this. Then, based on the UNL attributes prefixes are added to the UWs.These UWs are modified with SimpleNLG to form tense verbs, if they have attributes pertaining to the tenses.Then a DFS is performed on the graph to create the sentence from it by joining the words together connected by words representing the relations.

4.3.6 Emotion-Features Database Construction

This database consists of all features that have been extracted from the voice samples. The voice samples are obtained from 3 female speakers. Since the output of the “Mary” Text to Speech Synthesis system which has a female voice that need to be modified, the voice samples are taken from female speakers. 7 sentences are recorded in a variety of emotions such as angry, happy, sad and neutral. The voices are recorded in a fairly quiet room so as to avoid any external noise getting recorded. Then the sentences are read into Audacity, an audio analysis software in order to distinguish between the word boundaries clearly and to split them into audio files corresponding to each word. If excess noise has been recorded, then using the noise removal feature of Audacity, the audio sample can be made relatively noise free.

Fundamental Frequency Estimation

Once the word samples are obtained, they are given as input to the MATLAB program in order to measure the dominant fundamental frequency F0 present in each word. They are recorded systematically in the features database. In order to make sure that the correct F0 values have been measured, it is carried out using 2 different approaches.

In frequency domain using cepstral analysis:

The cepstrum is a Fourier analysis of the logarithmic amplitude spectrum of the signal. If the log amplitude spectrum contains many regularly spaced harmonics, then the Fourier analysis of the spectrum will contain a peak corresponding to the spacing between the harmonics: i.e. the fundamental frequency.

In time domain using autocorrelation:

The autocorrelation function for a section of signal shows how well the waveform shape correlates with itself at a range of different delays, thereby determining the periodicity in the waveform itself.

Once the F0 value has been determined, it can be used for pitch modification in the output of the text to speech synthesis system for prosody addition.

Duration of each word

This is measured using Audacity that has a millisecond calibration so that the exact duration of every word can be estimated.

Intensity Estimation

Generally, words said with different emotions differ in the intensity with which they are spoken. For eg, when a person speaks with anger, there is a greater intensity in the voice than when compared to a person who speaks with sadness. So, this measurement is done with the help of PRAAT tool, that can accurately measure with what intensity the word was spoken. It is measured in decibels.

4.3.7 Emotion Identification

The emotion present in the sentence is calculated by using the output of the UNL which is annotated with the tone of the sentence. Also, using a weighting system, the emotion present in the sentence can be determined. Various degrees of emotion can be identified based on the strength of the words that are used in the sentence. After this has been estimated, it is mapped onto one of the emotions that have been handled in the emotion-features database.

4.3.8 Voice Modification

The output of the text to speech system is modified to add the emotion and prosody. The emotion features database is consulted and the corresponding values of various parameters (like fundamental frequency, intensity, duration etc) are calculated. Then the robotic voice is modified based on the values of these parameters to produce the humanised voice.

4.4 COMPLEXITY ANALYSIS

4.4.1 Time Complexity

The time complexity of each module of the MT system has been shown in Table 4.1

4.4.2 Complexity of the Project

- The complexity of the project lies in the well know fact that Tamil is a free word-order language.Hence rules have to be written for every possible case of the ordering.

Module	Complexity
Morphological Analyser	O (N * k)
POS Tagger	O (N * k)
UNL Enconverter	O (N * N * k)
UNL Deconverter	O (N)
Emotion Identifier	O (N)

Table 4.1 Time complexity of each module

N - Number of words in the Sentence
k - Number of rules

- Splitting up of Tamil words in the morphological analyser often becomes erroneous with named entities.
- Most Tamil sentences, by nature tend to have a complex sentence structure.POS Tagging may fail in such cases and may propagate the error throughout the system.
- Tamil language encompasses more than 50 lakh words.And the largest available wordnet counts only to 50,000 words with no English gloss for any word.
- Proper indexing is needed on this 50,000 word DB so as to reduce the DB fetch latency.
- Creating emotions by modifying the signal, is an extremely new approach and is complex.

CHAPTER 5

SYSTEM DEVELOPMENT

The overall system code is depicted in the Figure 5.1. The system includes packages like analyser,UNL DeConverter,UNL EnConverter, UNL Word Format and Voice Generator ,each performing a function related to its name.

5.1 PROTOTYPE DESCRIPTION

The prototype description across each module is as follows :

5.1.1 POS Tagger

Input : The Tamil sentence is given as input.
Output : POS Tag for each word in the sentence is identified here.
eg.
Input - அவன் புத்தகத்தை படித்தான்
Output -
அவன் pronoun
புத்தகத்தை – Noun
படித்தான்- Verb

5.1.2 Morphological Analyser

Input : Each word of the POS tagged Tamil sentence is given as input.
Output :Each Tamil word is split into its root word and inflections.This output format is denoted as Segmented Word Format.
eg.

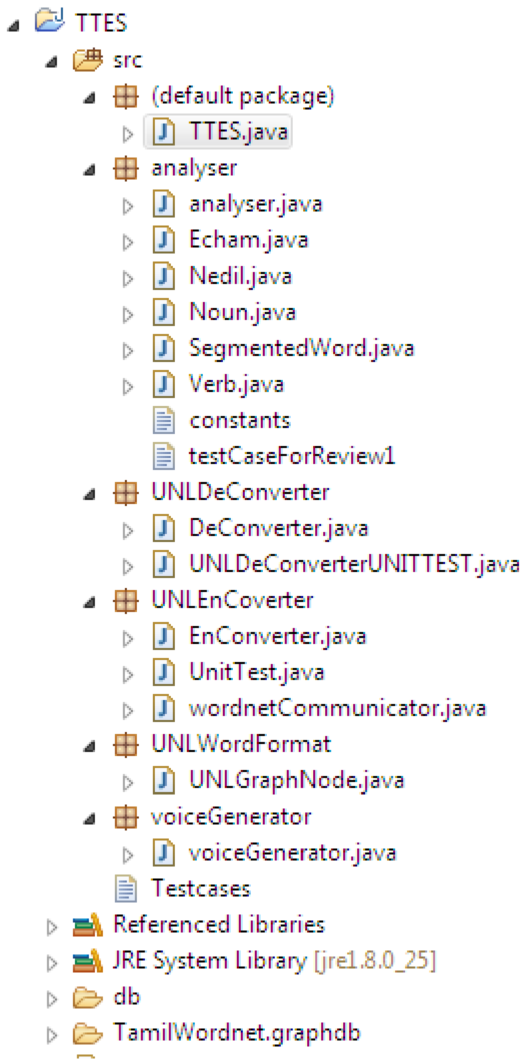

Figure 5.1 Organisation of source code

Input - படித்தான்

Output -

Root Word - படி

POS - Verb

POS ID - 3

Tense Suffix - த்

Doer Suffix - ஆன்

Doer Suffix Id - 6

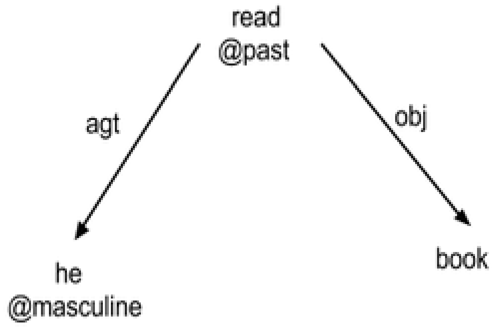

Figure 5.2 Representation of a sample output of the enconverter

5.1.3 UNL Enconverter

Input: A segmented word set for the input Tamil sentence
Output: An UNL Graph representing the UNL format of the sentence.
eg.
For the sentence in previous input the output in Graphical representation will be as in Figure 5.2.

5.1.4 Emotion Identifier

Input : The UNL Graph formed from the input sentence.
Output: This module should say whether sad or happy or anger emotion is present in the input sentence

5.1.5 Voice Modifier

Input : The deconverted English sentence and the identified emotion are given as input.
Output: Emotional voice corresponding to the input text and emotion.

5.2 ALGORITHM

5.2.1 Overall Algorithm

An overview of the overall algorithm of the system is as follows:

Overall Algorithm

function GenerateVoice($Sentence_{tamil}$)
  $SegmentedWords \leftarrow$ POS Tagging($Sentence_{tamil}$) + Morphological Analysis($Sentence_{tamil}$)
  $UNL \leftarrow$ EnConvert($SegmentedWords$)
  $S_{english} \leftarrow$ DeConvert(UNL)
  $emotion \leftarrow$ IdentifyEmotion(UNL)
  $Output \leftarrow$ ModifyVoice($S_{english}, emotion$)
end function

5.2.2 POS Tagger Algorithm

An overview of the algorithm used in our POS Tagger is as follows:

Algorithm for POS Tagger

Split the input sentence into words
Right to left scanning of words
Remove "க் || ச்||த் || ப்" if any at the end
Assign tags based on the inflection at the end the result from wordnet

5.2.3 Enconversion Algorithm

The Enconversion algorithm is defined as

Enconversion Algorithm

function Enconvert(SegmentedWord)
  $UNLGraphroot$
  $root.segments \leftarrow SegmentedWord$
  for $segment(i)$ in $root.segments$ **do**
    if segment.rootWord is a conjugation **then**

```
            root.addChildren(conjugation,Enconvert(new Segment(i,end))
            root.removeSegments(i,end)
        end if
        if Segment.rootWord matches Rule_k.Word then
            root.addChildren (Rule_k.relation , EnConvert(Rule_k.Corresponding_segments))
            root.remove(Rule_k.Corresponding_segments)
        end if
    end for
    for segment(i) in root.segments do

        if segment.POS is verb then
            root.addAttribute(tense(segment))
        end if
        if segment.POS is noun then
            root.addAttribute(gender(segment))
            root.addAttribute(singular/plural(segment))
        end if
        root.addAtrributes(domainAnalysis(segment.rootWord))
    end for
    return root
end function
```

5.2.4 Deconversion Algorithm

The Deconversion algorithm is defined as

Deconversion algorithm

```
function Deconvert(UNL)
    Re-orderWithDFS(UNL)
    Output = ModifiedDFS(UNL)
end function
function ModifiedDFS(UNL)
    root ← UNL
    if root has no children then
        if root has tense attribute then
            output ← tense_form(root.RootWord, root.tense)
        end if
        if root has preposition,article attributes then
            prefix ← preposition/article
            return prefix + output
        end if
    end if
    for c in children(root) do
        output+ ← wordMap(relation(c)) + ModifiedDFS(c)
    end for
    return output
end function
```

The priority order for the relations on which Re-Ordering happens is given in the appendix.

5.2.5 Algorithm for Estimation of Fundamental Frequency

The Algorithm for the estimation of fundamental frequency is as follows:

In frequency domain using cepstrum

Fundamental frequency estimation using cepstrum

$(samples, samplingrate) \leftarrow \text{wavread}(audiofile)$

$millisecond1 \leftarrow \frac{samplingrate}{1000}$

$millisecond20 \leftarrow \frac{samplingrate}{50}$

$Y \leftarrow \text{Fast Fourier Transform}(samples * hamming(length(samples)))$

$Cepstrum \leftarrow \text{Fast Fourier Transform}(log(absolute(samples) + eps))$

$[c, fundamentalfrequency] \leftarrow \max(absolute(Cepstrum(ms1 : ms20))$

$Fundamentalfrequency \leftarrow \frac{fs}{millisecond1+fx-1}$

In time domain using autocorrelation

Fundamental frequency estimation using autocorrelation

$(samples, samplingrate)] \leftarrow \text{wavread}(audiofile)$

$millisecond2 \leftarrow \frac{samplingrate}{500}$

$millisecond20 \leftarrow \frac{samplingrate}{50}$

$rvalue \leftarrow XCorrelation(samples, millisecond20)$

$[rmaxvalue, tx] \leftarrow \max(rvalue(millisecond2 : millisecond20))$

$Fundamentalfrequency \leftarrow \frac{fs}{millisecond2+tx-1}$

5.2.6 Voice Modification Algorithm

The Algorithm for voice modification is as follows:

Voice Modification Algorithm

function GetEmotionalVoice($Sentence_{english}, emotion$)
  $FundamentalFrequency \leftarrow EmotionFeaturesDatabase(emotion)$
  $Intensity \leftarrow EmotionFeaturesDatabase(emotion)$
  $Duration \leftarrow EmotionFeaturesDatabase(emotion)$
  $TTSOutput \leftarrow$ MaryTTSGenerator($Sentence_{english}$)
  ModifyVoice($FundamentalFrequency, Intensity, Duration, TTSOutput$)
end function

CHAPTER 6

RESULTS AND DISCUSSION

This chapter elaborates the efficiency of the system we have built.We will analyse the two parts of our system separately.

6.1 DATASET FOR TESTING

The input sentences are taken from fourth, fifth and sixth standard Samacheer book. The reference translations are also obtained from the English version of the same book.About hundred sentences in each of the types - simple, complex and compound are tested and the results are measured.

6.2 SAMPLE OUTPUT

A set of sample input and outputs of translation is given below:

- Test case ID : TC-01

  Input: அவள் சிரித்தாள்

  Output: She laughed.

- Test case ID : TC-02

  Input: அவன் புத்தகத்தை படிப்பான்

  Output: He will read book.

- Test case ID : TC-03

  Input: அவன் கோபத்தில் கத்தினான்

  Output: In anger he bawled.

- Test case ID : TC-04

  Input: அவள் சோகத்தில் அழுதாள்

  Output: She cried in grief.

- Test case ID : TC-05

  Input: அவள் பெயர் தமிழரசி

  Output: She is thamizharasi.

- Test case ID : TC-06

  Input: அவள் மகிழ்ச்சியில் குதித்தாள்

  Output: She jumped in joy.

- Test case ID : TC-07

  Input: மாசி மாதத்தில் அவன் இறந்தான்

  Output: During February he ceased to exist.

- Test case ID : TC-08

  Input: புத்தகம் படித்தான்

  Output: He learned book.

- Test case ID : TC-09

  Input: அவன் படியில் நின்றான்

  Output: He stood in step.

- Test case ID : TC-10

  Input: அவர்கள் கடைக்குச் செல்கிறார்கள்

  Output: They go to shop.

- Test case ID : TC-11

  Input: கன்னியாகுமரியில் இருந்து சென்னை வரை நான் நடந்-தேன்

  Output: I walked from kanniyaakumari to sennai.

- Test case ID : TC-12

  Input: அவள் என்ன செய்தாள்?

  Output: What did she do?

- Test case ID : TC-13

  Input: கண்ணகி மதுரையை எரித்தாள்

  Output: Kannaki parched mathurai.

- Test case ID : TC-14

  Input: வசந்த் வேகமாக கிளம்பினான்

  Output: Vasanhth emerged fast.

- Test case ID : TC-15

  Input: ஹனுமன் மோதிரத்தைக் கொடுத்தான்

  Output: Hanuman gave ring.

- Test case ID : TC-16

  Input: காவேரி பேயைப் பார்த்தாள்

  Output: Kaaveeri saw ghost.

- Test case ID : TC-17

  Input: அவள் அப்படி என்ன செய்தாள் ?

  Output: What did she do like that?

- Test case ID : TC-18

  Input: காலையில் நான் படிப்பேன்

  Output: I will learn in morning.

- Test case ID : TC-19

  Input: அவள் ஒரு காவியம்

  Output: Output: She is an epic.

- Test case ID : TC-20

  Input: அசோகவனத்தில் சீதையைக் கண்ட ராமன் அழுதான்

  Output: Raaman cried after seeing siithai in asokavanam.

6.3 PERFORMANCE EVALUATION

The translation via UNL and adding emotions to TTS voice are evaluated separately.For evaluating translation four measures namely, BLEU, NIST, METEOR and Fluency and Adequacy are employed.Since Tamil does not have accurate sentence types, the evaluation is categorized based on the type of sentence (simple,compound or complex) of the reference text.

The same input sentences are tested on Google Translator and evaluated against the same reference text. It is to be noted that Google Translate utilises an Example Based Translation System that operates on a very huge corpus.Having a huge corpus increases the time complexity of the system drastically.Google Translator is backed up by ennumerous servers which can process petabytes of data in a few seconds. On the other hand, UNL based translation neither requires such a huge corpus nor a heavy corpus processing and thus is both time and space efficient.

The emotional voice generation part of this project is evaluated manually through a survey.

6.3.1 BLEU Score

BLEU stands for Bilingual Evaluation Understudy.It is a n-gram co-occurrence based MT evaluation system that verified how close the MT translated text is to , to a human translated reference. N-gram precision in BLEU is computed as follows:

$$p_n = \frac{\sum\limits_{C\varepsilon\{Candidates\}} \sum\limits_{n-gram\varepsilon C} Count_{clip}(n-gram)}{\sum\limits_{C\varepsilon\{Candidates\}} \sum\limits_{n-gram\varepsilon C} Count(n-gram)} \tag{6.1}$$

Where $Count_{clip}(n-gram)$ is the maximum number of n-grams co-occurring in a candidate translation. To prevent very short translations that try to maximize their precision scores, BLEU adds a brevity penalty, BP, to the formula:

$$BP = \begin{cases} 1 & \text{if } |c| > |r| \\ \text{e^(1-|r|-|c|)} & \text{if } |c| \leq |r| \end{cases} \tag{6.2}$$

Where $|c|$ is the length of the candidate translation and |r| is the length of the reference translation. The BLEU formula is then written as follows.

$$BLEU = BP * exp\left\{\sum_{n=1}^{N} w_n \, log p_n\right\} \tag{6.3}$$

The weighting factor w_n, is set at $\frac{1}{N}$.

The BLUE score comparison of our System to that of the Google translator is shown in Figure 6.1.

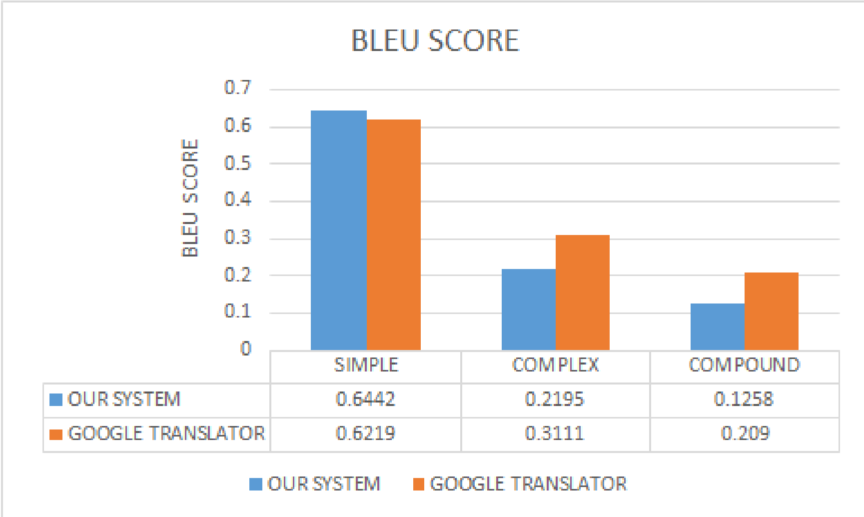

Figure 6.1 BLEU scores: Our system vs Google translator

6.3.2 NIST

The NIST 2005 Machine Translation Evaluation (MT-05) was part of an ongoing series of evaluations of human language translation technology.

BLEU measures translation accuracy according to the N-grams or sequence of N-words that it shares with one or more high quality reference translations. But it might be better to weigh more heavily those N-grams that are more informative – i.e., to weigh more heavily those N-grams that occur less frequently, according to their information value. This helps to combat possible gaming of the scoring algorithm, since those N-grams that are most likely to (co-)occur would add less to the score than less likely N-grams[24].

$$Info\,(w_1...w_n) = log_2 \left\langle \frac{the\,\#\,occurrences\,of\,w_1...w_{n-1}}{the\,\#\,occurrences\,of\,w_1...w_n} \right\rangle \tag{6.4}$$

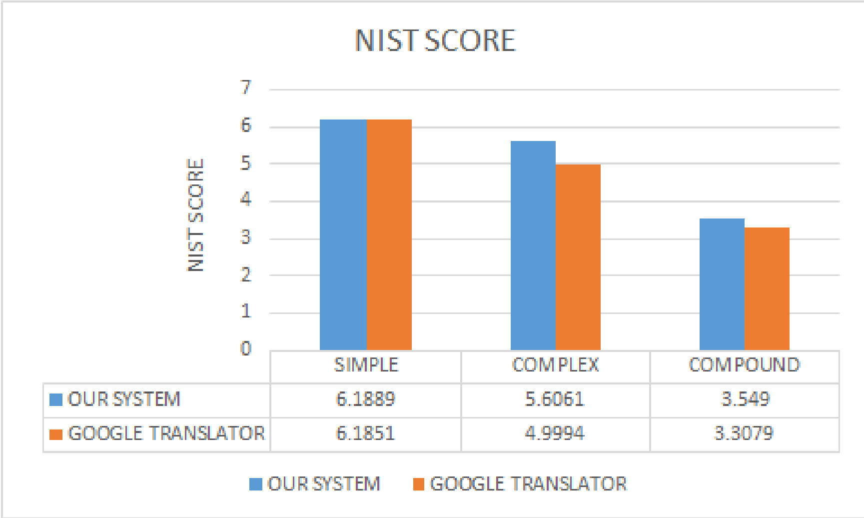

Figure 6.2 NIST scores: Our system vs Google translator

$$Score = \sum_{n=1}^{N} \left\{ \frac{\sum_{all\, w_1 \ldots w_n\, that\, co-occur} Info\,(w_1 \ldots w_n)}{\sum_{all\, w_1 \ldots w_n\, that\, co-occur\, in\, sys-output} (1)} \right\} exp \left\{ \beta log^2 \left[min \left\langle \frac{L_{sys}}{L_{ref}}, 1 \right\rangle \right] \right\} \tag{6.5}$$

where

β is chosen to make the brevity penalty factor = 0.5 when the no of words in the system output is $\frac{2}{3}$rds of the average no of words in the reference translation

$N = 5$

L_{ref} = the average number of words in a reference translation averaged over all reference translations

L_{sys} = the number of words in the translation being scored.

The NIST score comparison of our System to that of the Google translator is shown in Figure 6.2.

6.3.3 METEOR

METEOR stands for Metric for Evaluation of Translation with Explicit ORdering.It is based on the harmonic mean of unigram precision and recall , with recall weighted higher than precision. It also has several features that are not found in other metrics, such as stemming and synonymy matching, along with the standard exact word matching. The metric was designed to fix some of the problems found in the more popular BLEU metric, and also produce good correlation with human judgement at the sentence or segment level .It involves alignment of the reference text with the translated text.The the score is computed as

$$M = F_{mean}(1 - p) \tag{6.6}$$

where M denotes the score and p denotes the penalty computes as

$$p = 0.5\frac{c}{u_m^3} \tag{6.7}$$

where c is number of chunks of data , u_m is the number of unigrams that have been mapped.The METEOR score comparison of our System to that of the Google translator is shown in Figure 6.3.

6.3.4 Fluency and Adequacy

Fluency is a measure of the grammatical correctness of the sentence.Adequacy, on the other hand determines the degree of information conveyed by the translated text.Both measures are to be evaluated manually through a survey. The text translated by our system along with Google translation of the same text and the source text is given to a group of people who knew both the languages.The responses are shown in Figures 6.4 and 6.5.

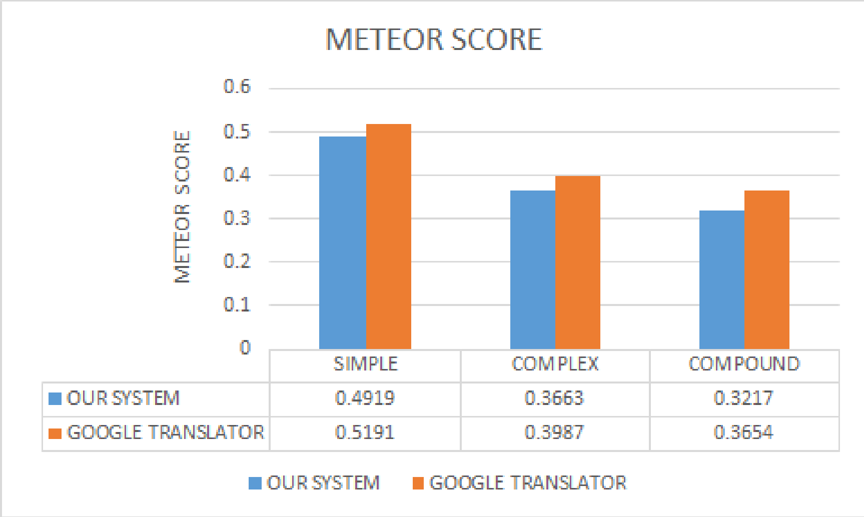

Figure 6.3 METEOR scores: Our system vs Google translator

Score	Description
4	perfectly correct sentence
3	Minor tense/Noun-verb flaws , Understandable
2	Major grammatical flaws , Understandable with effort
1	Sentence makes no sense

Table 6.1 Fluency score description

Score	Description
4	Full meaning conveyed
3	Most of the meaning conveyed
2	Noun and/or Tense information conveyed
1	No meaning at all

Table 6.2 Adequacy score description

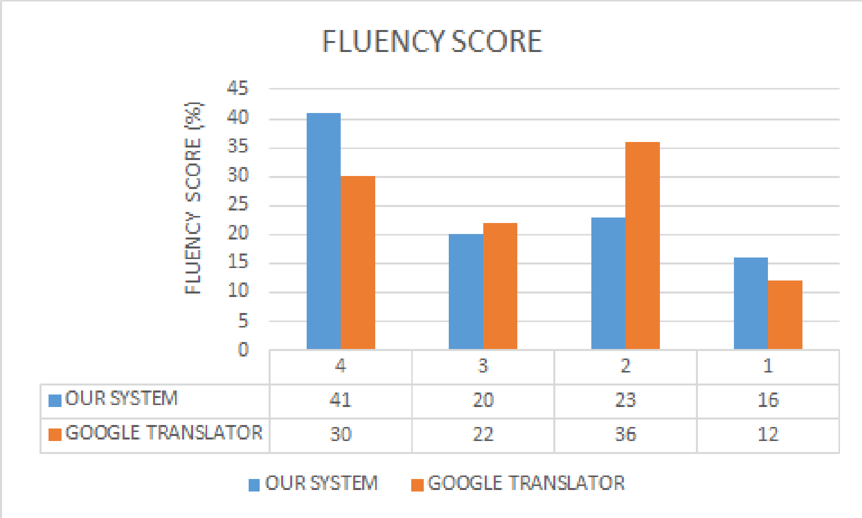

Figure 6.4 Fluency scores: Our system vs Google translator

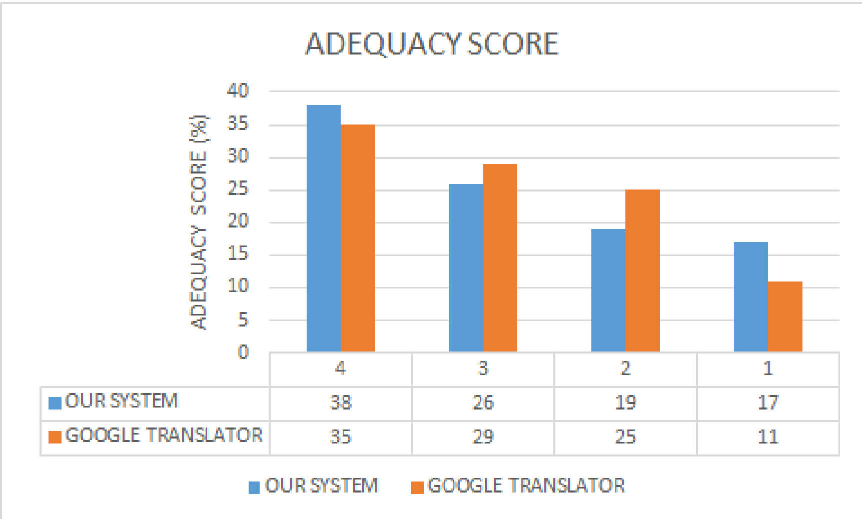

Figure 6.5 Adequacy scores: Our system vs Google translator

6.3.5 Evaluation of emotional speech

A survey was conducted to determine how well emotion was expressed in the final prosodically enhanced output sentences. Six recordings comprising of outputs expressing angry, happy and sad emotions were played and the corresponded responses were recorded.

Emotion	Percentage of total responses that accurately identified the emotion from modified voice
Anger	65%
Happiness	57%
Sadness	81%

Table 6.3 Survey Results on Identifying the Emotion present in Modified Voice

Emotion	Average score for the identified emotion given by responder on a scale of 1-5
Anger	3.6
Happiness	3.2
Sadness	4.1

Table 6.4 Scores for Emotion present in the Modified Voice

CHAPTER 7

CONCLUSION

7.1 SUMMARY

This system translates Tamil text to English text with UNL as interlingua and then voices the English text out with emotions.In case of a morphologically rich language such as Tamil, the use of a interlingua is better for translation.A custom built morphologically analyser is used to extract noun,tense and gender from the Tamil text. A POS Tagger was also created to classify the words in to noun,verb,object,adverb and adjective. Based on the POS tags and morphological structures rules are devised to convert the Tamil text to UNL with an enconverter.A Deconverter, with a newly developed algorithm, builds a simple English text from the UNL text with the help of SimpleNLG.

Then the emotion in the sentence is identified from the UNL and based on the identified emotion, the TTS output for the translated English text is modified to incorporate prosody on it. The results of evaluation showed that UNL is indeed efficient in translation.We have successfully added three emotions (sadness, happiness and anger) to the machine voice.

7.2 CRITICISM

Most of the erroneous translations are attributed to the faults in POS Tagger.POS Tagging, by nature is difficult in any free order language.POS Tags were assumed on words, on which no POS Tag could not be specified based on rules due to the non-availability of domain information and those words being void of any inflections. This becomes much more difficult because of WSD - word sense disambiguation where the same word has multiple meanings based on its POS tag. The number of POS Tags was also limited to noun, verb, pronoun, adverb and adjective.Identifying the Subject was indeed difficult in cases where subject had a description attached to it.

Exceptions also existed in Morphological analyser especially when there were named entities included as a part of the sentence.Identification of emotions was based on explicit word matching on keywords, which would turn to be faulty if those keywords does not exist in the sentence.Out of the 47 relations 35 were handled by the enconverter. Out of eight emotions, only three has been successfully implemented in voice modification.

7.3 FUTURE WORK

By increasing the word count in our Wordnet and also adding more domain information to the existing words, better accuracy could be achieved.Also, implementing Named Entity Recognition would aid in better POS Tagging.Few more POS Tags have to be added and corresponding rules need to be changed. Wordnet has to be improved with more words and word domain information needs to be incorporated for each word.Based on the prototype for these three emotions, a lot more emotions can be handled. This system is scalable with these improvements and can play a vital role in the field of Natural Language Processing.

APPENDIX A

Rules used in the system

A.1 DECONVERSION

The priority order for the relations are as follows

name	order	condition/connectors to add
seq	0	then
cnd	1	if
mod	2	-
plc	3	in/at (depending on domain)
dur	4	during
agt	5	should be followed by verb (mostly in root of UNL)
coa	6	and
obj	7	if any determiners
aoj	8	-
cnt	9	is
man	10	-
frm	11	from
to	12	to
gol	13	to
via	14	via
tmi	15	during/at
tmf	16	to/till
and	17	and
or	18	or
rsn	19	so

BIBLIOGRAPHY

[1] G. G. Chowdhury, "Natural language processing," *Annual review of information science and technology*, vol. 37, no. 1, pp. 51–89, 2003.

[2] K. W. Church and R. L. Mercer, "Introduction to the special issue on computational linguistics using large corpora," *Computational linguistics*, vol. 19, no. 1, pp. 1–24, 1993.

[3] J. Lehrberger and L. Bourbeau, *Machine Translation: Linguistic characteristics of MT systems and general methodology of evaluation*, vol. 15. John Benjamins Publishing, 1988.

[4] L. Dugast, J. Senellart, and P. Koehn, "Statistical post-editing on systran's rule-based translation system," in *Proceedings of the Second Workshop on Statistical Machine Translation*, pp. 220–223, 2007.

[5] P. Koehn, F. J. Och, and D. Marcu, "Statistical phrase-based translation," in *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*, pp. 48–54, 2003.

[6] V. Kumar, R. Sridhar, S. Narayanan, and S. Bangalore, "Enriching spoken language translation with dialog acts," in *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics on Human Language Technologies: Short Papers*, pp. 225–228, 2008.

[7] D. Chiang, "A hierarchical phrase-based model for statistical machine translation," in *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pp. 263–270, 2005.

[8] A. Ramanathan, J. Hegde, R. M. Shah, P. Bhattacharyya, and M. Sasikumar, "Simple syntactic and morphological processing can help english-hindi statistical machine translation.," in *IJCNLP*, pp. 513–520, 2008.

[9] K. Bar, Y. Choueka, and N. Dershowitz, *An Arabic to English example-based translation system*. PhD thesis, 2007.

[10] Y. Lepage and E. Denoual, "Purest ever example-based machine translation: Detailed presentation and assessment," *Machine Translation*, vol. 19, no. 3-4, pp. 251–282, 2005.

[11] J. G. Carbonell, S. Klein, D. Miller, M. Steinbaum, T. Grassiany, and J. Frey, "Context-based machine translation," pp. 19–28, 2006.

[12] H. Uchida, M. Zhu, and T. Della Senta, *Universal Networking Language*. 2005.

[13] P. Kumar, *UNL Based Machine Translation System for Punjabi Language*. PhD thesis, Tharapur University, Patiala, 2012.

[14] D. Schwarz, "Data-driven concatenative sound synthesis," in *Proceedings of the COST G-6 Conference on Digital Audio Effects (DAFX-00), Verona, Italy,2009*.

[15] R. Carlson, T. Sigvardson, and A. Sjölander, "Data-driven formant synthesis," *KTH, Stockholm, Sweden, Progress Report*, vol. 44, pp. 121–124, 2002.

[16] J. Stark, B. Lindblom, and J. Sundberg, "Apex an articulatory synthesis model for experimental and computational studies of speech production," *TMH-QPSR*, vol. 2, pp. 45–48, 1996.

[17] K. Tokuda, T. Yoshimura, T. Masuko, T. Kobayashi, and T. Kitamura, "Speech parameter generation algorithms for hmm-based speech synthesis," in *IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 3, pp. 1315–1318, 2000.

[18] R. J. McAulay and T. F. Quatieri Jr, "Computationally efficient sine wave synthesis for acoustic waveform processing," June 26 1990. US Patent 4,937,873.

[19] A. W. Black and K. A. Lenzo, "Flite: a small fast run-time synthesis engine," in *4th ISCA Tutorial and Research Workshop (ITRW) on Speech Synthesis*, 2001.

[20] S. Sangeetha and S. Jothilakshmi, "Syllable based text to speech synthesis system using auto associative neural network prosody prediction," *International Journal of Speech Technology*, vol. 17, no. 2, pp. 91–98, 2014.

[21] T. Bowles and S. Pauletto, "Emotions in the voice: humanising a robotic voice," in *Proceedings of the 7th Sound and Music Computing Conference, Barcelona, Spain*, 2010.

[22] I. Manuel and Tolkappiyar, *Literary theories in Tamil: with special reference to Tolkappiyam*, vol. 8. Pondicherry Institute of Linguistics and Culture, 1997.

[23] R. Jayavasanth and R. Sridhar, "Tamil wordnet based on a hybdrid approach," *13th International Tamil Internet Conference*, pp. 120–124, 2014.

[24] G. Doddington, "Automatic evaluation of machine translation quality using n-gram co-occurrence statistics," in *Proceedings of the second international conference on Human Language Technology Research*, pp. 138–145, 2002.