Indian Journal of Science and Technology, *Vol 8(29), DOI:10.17485/ijst/2015/v8i29/72294, November 2015*
ISSN (Print) : 0974-6846
ISSN (Online) : 0974-5645

Text Processing for Developing Unrestricted Tamil Text to Speech Synthesis System

Vaibhavi Rajendran* and G. Bharadwaja Kumar

School of Computing Science and Engineering, VIT University, Chennai Campus, Tamil Nadu, India; vvaibavi@gmail.com

Abstract

In this Information and communication technology era, designing interactive computer systems that are effective, efficient, easy, and enjoyable to use is becoming increasingly important. Of the numerous ways explored by researchers to enhance Human-Computer Interaction, Text to Speech or Speech Synthesis affirms to be one such modality for developing better interfaces. The focal point here is to enhance the text processing module of Tamil speech synthesizer with an efficient and robust text normalizer and loan word identifier. Text normalization is performed on unrestricted Tamil text to convert non-standard words into standard words for the reduction of ambiguous utterances along the interim processing of the words. Loan words in Tamil text are identified in order to improve the pronunciation model of the Tamil speech synthesizer system. In this paper, we describe a 'semiotic classifier' based on decision list approach with which we are able to tackle many varieties of non-standard words. We also describe a 'loan/native word classifier' based on multiple linear regression which works efficiently even on shorter words of 3 syllables in length. In today's predominant Digital, Information-Communication Technology and Human-Computer Interaction era such profound text processors is imperative.

Keywords: Natural Language Processing, Tamil, Text Processing, Text-to-Speech (TTS), Unrestricted Text

1. Introduction

An extensive research has been carried out in developing Text-to-Speech (TTS) systems for languages such as English, Chinese, Japanese, Germany and also for Indian languages such as Hindi, Urdu, Gujarati, Telugu, and Tamil etc. A Natural Language Processing (NLP) module in an unrestricted TTS involves processing of the real text which is an intriguing task. Proper attention is required to perform text normalization in a way such that it enhances the readability of the TTS by decreasing the production of words with incorrect or unnatural pronunciation. In real text, many non-standard representations of words appear which can be termed as 'informal language' used for communication in social networking sites, blogs and other networking places. Processing informal text has become an increasingly popular research topic in recent years. Regardless of the size of the text corpus, there will always be tokens that do not appear and have unknown pronunciations. In general, texts are full of heteronyms, numbers, and abbreviations which require expansion for the phonetic representation. Hence Text normalization is commonly considered as a crucial as well as a language-dependent process[1]. India is very well known for its diversity not only for multiculturalism but also for multilingualism. It has twenty nine states, seven union territories, 22 national languages, 1162 other languages and dialects and almost all the religions of the world have adherents in the country[2]. Because of this, loan or borrowed words from different languages is a common thing in many languages of India. In today's information and communication technology era, one of the very common issues experienced by any application of automatic processing of the digital documents is that of multilingualism. Any linguistic text processing is completely dependent on the language of the input text. Tamil is one of the longest surviving Dravidian languages of India with very rich morphology. Also, unlike other Dravidian languages, Tamil Grapheme to Phoneme (G2P) conversion is a non-trivial task because of many language irregularities that can be observed between written

*Author for correspondence

to spoken form of the language. Hence, it is essential for a Tamil TTS system to find out the language of the word to be able to pronounce it in a proper manner.

In this paper, we concentrate on text normalization and loanword identification to improve the pronunciation of a Tamil TTS system. In the next section, we discuss about the Text Processing module which performs text pre-processing and normalization. In Section 3, we discuss about the loanword identification module built for improving the pronunciation of the TTS system and finally in the last section we have given the conclusion and future work.

2. Text Normalization

In general, a text normalization module of unrestricted text of a TTS system is indispensable to improve the intelligibility and naturalness, it includes the following steps:

- Pre-processing: Possible identification of text genre, character encoding issues, possible multi-lingual issues.
- Sentence Splitting: Segmentation of the document into a list of sentences.
- Tokenization: Segmentation of each sentence into a number of tokens.
- Semiotic Classification: Classification of each token as one of the semiotic classes of natural language, abbreviation, quantity, date, time etc.
- Verbalization: Conversion of non-natural language semiotic classes into words.
- Loanword Identification: Identifying the borrowed words from other languages to disambiguate the pronunciation.

2.1 Related Work

Standard Words (SW) have a specific pronunciation that can be phonetically described either in a lexicon or by the letter to sound rules. On the contrary, a Non-Standard Word (NSW) contains numerical patterns and alphabetical strings that do not have a regular entry in the lexicon and their pronunciation needs to be generated by a more complicated natural language process[3]. A NSW can be a number, acronym, abbreviation, URL, e-mail address, special character or loan word. The effort needed to verbalize the NSW is the biggest challenge confronted by a text normalizer.

The NSWs have been categorically represented as 23 classes and general techniques like n-gram language models, decision trees, decision lists, Weighted Finite-State Transducers (WFST), supervised and unsupervised techniques have been equipped to tackle them[4]. These approaches require a huge annotated corpus but building such a corpus is a laborious task and requires lot of resources[4,5]. The lack of large annotated corpus mounts a way to the hand engineered rules for normalization[6,7]. In many of the supervised and unsupervised learning techniques, certain aspects of normalization such as annotations[5,8,9], dealing of numbers, especially telephone numbers[7], homograph disambiguation and abbreviation expansion are still hand engineered.

Here are some posits which gives a comparison of Tamil NSW with English and Hindi:

- The numbering system pattern in Tamil is quite different from English. In English, according to the conventional numbering system, the number 493 is verbalized as "four hundred and ninety three", 523 is "five hundred and twenty three", 287 is "two hundred and eighty seven", as you can see the word formation follows a certain pattern. The numbers are converted into strings following the units, tens, hundreds position and so on in the increasing order of representation from right to left. From right, the digit in units place is given as the string representation of the digit itself (3 is "three" in 493), the digit in the tens place is represented with an addition of "ty" suffix with the digit (9 is nine+ty - "ninety" in 493) and the digit in the hundreds position is represented with the word "hundred" following the string representation of the digit (4 is four and followed by hundred - "four hundred" in 493). This is not the case always in Tamil numbering system; some distinct features exist in Tamil's conventional number-name representations. Although 523 – *"ஐந்நூற்றி இருபத்தி மூன்று"* (ainnURRi irubaththi munRu), 287 – *"இருநூற்றி எண்பத்தி ஏழு"* (irunURRi eNbaththi Ezu) are similar to the representations in English, 493 has a differently fashioned string formation, it corresponds to *"நானூற்றி தொன்னூற்றி மூன்று"* (nAnURRi thonnURRi munRu), where the unit's place and the hundreds place do follow the conventional numbering rules but, the digit 9 in the ten's place is termed as *"தொன்னூற்றி"* (thonnURRi) which when split has *"னூற்றி"* (nURRi) at the end. *"னூற்றி"* (nURRi) is an inflected form of the word *"நூற்றி"* (nURRi) which means "hundred". As you can see the digit 9 is in the ten's place, yet it is

suffixed with hundred, which is very untypical. Many such irregularities in Tamil number-name representations exist.

- In English, when we take the orthographic representation '1' its verbalization is always 'one' irrespective of the context. Some examples of the mapping include: 1 lakh – one lakh; 1 book – one book; and 1,2,3 – one,two,three. However, in the case of Tamil the same orthographic representation '1' is verbalized into either *'ஒரு'* (oru) or *'ஒன்று'* (onru) according to the context. When '1' is used as a quantifier it gets mapped as *'ஒரு'* (oru), the examples of such instances include: 1 *லட்சம்* (1 latcam) - *ஒரு லட்சம்* (oru latcam) and 1 *புத்தகம்* (1 puththagam) - *ஒரு புத்தகம்* (oru puththagam). When '1' is used in number series, phone numbers, account numbers and other places then it is verbalized into *'ஒன்று'* (onru), for an example 1, 2, 3 will be verbalized as *ஒன்று* (onru), *இரண்டு* (iraNdu), *மூன்று* (munRu).

After a thorough observation of many such facts, we have chosen a semiotic classifier based decision list for Tamil NSW normalization.

2.2 Semiotic Classifier

A Decision List is a type of decision tree used to implement hierarchical decision making. Although a decision list is the simplest approach among other decision tree approaches, it can be used to represent a wide range of classifiers[10,11]. In the present work, a semiotic classifier based decision list has been implemented for Tamil NSW normalization. Semiotic classifier helps to form taxonomy of NSWs. Table 1 lists the eight semiotic classes considered in the present work along with few examples.

The whole numbers are dealt by digit wise processing following the conventional numbering system of Tamil. The irregularity of the Tamil numbering system has been taken into consideration and accordingly rules have been formed. For decimal numbers, the number is split into two tokens, the integer part and the fractional part and then processed. The number series is one very interesting subclass of numbers, although finding all types of number series is quite combative, most of them can be processed by checking the place and number of occurrence of the special character "," between them. If the place of occurrence of "," and the number of digits in the number do not correspond to any standard format of the conventional numbering system, then it is taken to be a number

Table 1. Semiotic Class

S.No	Semiotic Class	Sub Class	Example
1	Numbers	Whole Numbers Decimal Number Series Phone Numbers Account Numbers	7423 23.8 14,1,6 9447873330 1004005678012
2	Date	With / With .	20/07/1988 20.07.2015
3	Time	With : With .	3:30 3.30
4	Alphanumeric	Tamil English	2015 ஆம் MH70
5	Abbreviations & Acronyms	Tamil English	ஐ.நா I.A.S
6	Money	Tamil English	ரூ.100 $.100
7	Special Characters	(given in Table 2)	1:2
8	Punctuations	. , ! ? : ; " ""	எங்கே?

series. If the numbers are confirmed to belong to the number series subclass, then the numbers are processed individually by using the whole number processor. The last subclass, in the number class is the telephone/account number which is very hard to identify. Hence, when the number is preceded with a "+" symbol or when the number has 10 digits or more it is taken to be a mobile number or an account number and is read as individual digits rather than as a whole number.

In a similar way, the date, time and alphanumeric classes are divided into sub tokens and processed according to the subclass category which is listed out in Table 1. Processing of the abbreviations and acronyms needs a prerequisite of a list containing the abbreviations and acronyms along with its expansions. The list of abbreviations and acronyms has been formed using wiktionary and Tamil newspapers (Dinamalar, Dinakaran and Dinamani) for both the sub classes - English and Tamil. If the abbreviation in the input text document is present in the list then the abbreviation is replaced with the corresponding expansion, otherwise it will be processed as separate letters later in G2P conversion phase.

In Tamil, *ரூ* is equivalent to "Rs.". The possible cases of money class are: $ - *"டாலர்"* (dAlar), Rs – "Rupees", and *ரூ* - *"ரூபாய்"* (rUbAy). Whenever a number is preceded with any of these notations the above mentioned corresponding expansion is replaced. However, there

are several exceptions. One such exception is that the expansion of *"ரூ.100 லட்சம்"* (Rs.100 lakh) will become *"நூறு ரூபாய் லட்சம்"* (hundred rupees lakh). Hence, post processing is carried out as shown in the Table 3 to get the correct expansion *"நூறு லட்சம் ரூபாய்"* (hundred lakh rupees).

The class for processing special characters contains several sub classes and hence is specified separately in Table 2. The context and the number of occurrence of the special character plays a vital role in helping us find out on how to process the character. For some special characters, English words transliterated into Tamil are replaced as its verbalized form during processing. For instance, the special character "&" is replaced with *"அண்டு"*, a transliterated form of "and" in Tamil. For example, *"வசந்த் & கோ"* is verbalized as *"வசந்த் அண்டு கோ"* (vasanth and co). There are several such instances of punctuations that have been handled in the present work.

Table 2. Processing of special characters

S.No	Symbol	S.No	Symbol
1	!	11	{ }
2	!=	12	+
3	" ' , ' "	13	" ' = ' "
4	#	14	:
5	$	15	;
6	%	16	,
7	&	17	.
8	*	18	/
9	( )	19	?
10	[ ]	20	-

Table 3. Post Processing

Example	Before Processing	After Processing
1 லட்சம் (1 latcam)	ஒன்று லட்சம் (onRu latcam)	ஒரு லட்சம் (oru latcam)
ரூ.9 லட்சம் (ru.9 latcam)	ஒன்பது ரூபாய் லட்சம் (onbadhu rUbAy latcam)	ஒன்பது லட்சம் ரூபாய் (onbadhu latcam rUbAy)
3.30	மூன்று புள்ளி முப்பது (mUnRu puLLi muppadhu)	மூன்று முப்பது (mUnRu muppadhu)
3:30	மூன்று க்கு முப்பது (mUnRu ikku muppadhu)	மூன்று முப்பது (mUnRu muppadhu)

Coming to the last semiotic category, certain punctuations are preserved and certain are omitted according to the context. If "?","!" and "." are used more than once after a word, they are preserved as it will help in integrating the prosodic component in the synthesized speech in the later phase[12].

2.3 Post Processing

In certain contexts, post processing has been carried out to correctly disambiguate the NSWs. A Few examples where post processing is required have been shown in Table 3. The first case in Table 3, illustrates the need for post processing when the number "1" appears as a quantifier. The number 1 in general takes the verbal form *"ஒன்று"* (onru), conversely whenever 1 appears as a quantifier before a word it gets changed as *"ஒரு"* (oru), few instances are: when 1 appears before *"கோடி"* (kOdi), it is *"ஒரு கோடி"* (oru kOdi) and not *"ஒன்று கோடி"* (onru kOdi). This correct replacement of words based on the context is done by implementing a set of context sensitive rules, but for a limited set.

2.4 Results

A set of 412 sentences were collected from various sources such as daily newspapers in Tamil, magazines, Wikipedia and some Tamil websites. These 412 sentences were chosen manually in a way that most frequent NSWs occurring in real text are covered. The number of occurrences of NSW in each sentence is at least one NSW and a maximum of 6 to 7.

An Example sentence of our test data set with NSWs is given below:

அதேபோல 25.10.2010ல், 91-92, 92-93 ஆகிய ஆண்டுகளில் சசி எண்டர்பிரைசசும் வருமான வரிக்கணக்கைத் தாக்கல் செய்யவில்லை எனக் கூறி, 1997ல் வழக்கைத் தாக்கல் செய்தது வருமானவரித் துறை.

The normalized sentence after processing in the text normalization module:

அதேபோல, இருபத்திஐந்து பத்து இரண்டாயிரத்திபத்துல் தொன்னுற்றிஒன்று தொன்னூற்றிஇரண்டு, தொன்னூற்றி இரண்டு தொன்னூற்றிமூன்று ஆகிய ஆண்டுகளில் சசி எண்டர்பிரைசசும் வருமான வரிக்கணக்கதை தாக்கல் செய்யவில்லை எனக் கூறி, ஆயிரத்திதொல்லாயிரத்துதொன்னூற்றிஏழுல் வழக்கைத் தாக்கல் செய்தது வருமானவரித் துறை.

The NSW which were not processed are: some abbreviations and acronyms which were either not in the list or not written in standard format such as பி.எஸ்.கே., கி.மீட்டர் and some compound NSWs such as ரூ.12,290-தான். The compound NSW "ரூ.12,290 - தான்" (ru.12,290 dhAn) falls under nearly five classes as shown in Table 4. The processing of such NSW into correct sequence of standard words is difficult to achieve.

In order to measure performance of the semiotic classifier, the total number of sentences considered was 412 with 1236 NSWs on the whole. When these 1236 NSWs were tested, the classifier was not able to process only 26 NSWs. These 26 NSWs were mostly compound NSWs or abbreviations which were not present in the abbreviation list due to their rare usage or were written in non-standard form. Hence, the accuracy of semiotic classifier model is about 97.89% for the test set.

3. Loanword Identification

India is very well known for its diversity not only for multiculturalism but also for multilingualism. Hence, loanword identification is a predominant issue prevalent in all applications involving automatic processing of digital documents in Indian languages. Although dictionaries are impeccably convenient when it comes to pronunciation, it is very difficult to build pronunciation dictionary for the unrestricted text documents available on web.

A complication with respect to the text processing for a TTS system is the issue of loan words whose pronunciation is solely dependent on the language of origin of the word. Most of the Indian language scripts are phonetic in nature i.e. there exists a one-to-one correspondence between the orthography and pronunciation in these languages. Although Tamil script is also phonetic in nature, there are many exceptions. Unlike other Dravidian languages, Tamil grapheme to phoneme conversion is a non-trivial task because of many language irregularities that can be observed between written to spoken form of the language. Hence, finding the language origin of the word is essential. There also exists a need for both identifying the language of an unknown word and modeling its pronunciation according to the language. A successive investigation to be done is whether the words of another language origin should be pronounced using the rules and phonemes of that language or the language for which the TTS is being built[13].

Another major problem in loan word identification is going to be with the size of the text presented for identification. When the size of the text presented is long, identification of loan words is fairly easy, but when short text sequence is presented it becomes quite a challenge. Identification of loan words in short text segments is still a creasing problem. In this work, we are trying to identify if the word belongs to Tamil language or not. If the word does not belong to Tamil, we are categorizing it to be a loan word for which a different pronunciation model will be used.

Table 4. Categories of the Compound NSW "ரூ.12,290-தான் – (ru.12,290 dhAn)

Class	Sub Class	Reason for Semiotic Classification
Money	Tamil	Presence of "ரூ"
Alphanumeric	Tamil	Presence of numbers & Tamil alphabets
Special Character	Comma	Presence of ','
Special Character	Hyphen	Presence of '-'
Numbers	Whole Numbers	Presence of 12290

3.1 Related Work

A multilingual TTS system is one which can read texts of more than one language and produce a synthesized speech of the respective language. On the other hand, a polyglot TTS system can switch between languages as and when required even while reading a single language's text input document. This language switching is a vital requirement for reading multilingual digitized documents in today's world. To perform this language switching in between a document, language identification model is a pre-requisite[14]. A Polyglot TTS system is a monolingual TTS system which has the potential to switch between languages when it finds an inclusion of a loan word in the input text document. The capability of language identification and language switching is totally dependent on the training of the language identification model on the specific languages for the purpose of effective loan word identification.

Several computational methods have been used earlier for language identification based on a variety of features. These language identification methods can be adopted for loan word identification with some changes. In case of a TTS system, the text sequence is a word which

could range from very few characters to many. Therefore, the challenge here is not only in finding the best method but also in consequently optimizing the parameters for identifying the loan word even from short text inputs.

Several distinct machine learning methods have been explored for this task such as: Bayesian classification[15], Relative entropy based classification[16], Decision trees[17], Neural Network[18], Centroid based classification[19], Multiple Linear Regression Classification[20], Support Vector Machines[21], Letter Weighting Approach using Neural Networks[22] and Artificial ants and k-means algorithms[23]. The Multiple Linear Regression (MLR) model has worked well for language identification among Indian languages[20].

In the present work, language identification technique has been adopted for our loan word identification problem to find whether the word belongs to Tamil language or not. The process of finding out the language of origin of the loan word further is not carried out. MLR model used for language identification in[20] has been modified and used for Tamil loan word identification. A MLR model is trained upon the n-gram syllable based language model. In Tamil language, as mentioned earlier the inclusion of loanwords from English, Sanskrit, Hindi and Telugu are more prominent. Such loanword inclusions can be of two categories:

3.1.1 Bilingual Word

The stem of the word may be of the foreign language with a conjugation of the base language. The word conjugation and the bilingual forms have been split and illustrated in Table 5.

Table 5. Example of few Bilingual word

Bilingual word	Word Conjugation	Non-Tamil stem word	Tamil Conjugate Form
அந்நாவலை (annAvalai)	நாவல் + ஐ (nAval) + (ai)	நாவல் (nAval)	அந், ஐ (an) , (ai)
அட்வைசரை (advaisarai)	அட்வைசர் + ஐ (advaisar) + (ai)	அட்வைசர் (advaisar)	ஐ (ai)
அட்வான்ஸாக (advAncAga)	அட்வான்ஸ்+ஆக (advAnce) + (Aga)	அட்வான்ஸ் (advAnce)	ஆக (Aga)
அகடமியில் (akatamiyil)	அகடமி+இல் (akatami) + (il)	அகடமி (akatami)	யில் (yil)

3.1.2 Full Loanword

These are words from other languages, which totally follow the foreign language morphology and pronunciation but are injected in the base language.

Loanwords which have a distinct morphological representation from the base language can be easily identified. But a loanword which follows a similar morphology to the base language is very difficult to identify.

3.2 Language Modeling for Loanword Classification

MLR is used for classification task in the present work with the classes being either "Tamil" or "non-Tamil". The regressor variables in this loan word identification case are a set of features obtained from the corpus. The values of the response variable during the training are hence discrete for samples of both the classes. Now, for the testing samples or for the words which are to be tested, only the feature set values which correspond to the regressor values are entered. The values which gets generated for the response variable makes us decide on whether the word falls under the class "Tamil" or "non-Tamil". For doing this classification we require a threshold value to help us classify between the classes. The threshold value has been manually taken for the testing samples and is not dependent on the training values. The threshold value depends on the test samples under consideration and is variable for each test data. Implementing the classifier according to MLR pertains to a series of steps:

- Extracting syllable based n-grams from corpus.
- Feature Set Selection.
- Training Set and Testing Set Generation.
- F-measure Calculation.

3.3 Feature Selection

Generally for identifying the language of text, we should be able to distinguish the language using some language specific features. When we have an entire document or paragraph it is very easy to identify the language. Identifying the language even from a sentence or a phrase is a comparatively easier task than identifying from a single word. The possibility of having language specific information which will help us to identify the language is much higher when the input is longer. When the input is a single word the complexities increase further. Hence the

feature that we select for the task of language identification should be such that it occurs within a word or is a word.

If words are taken to be the features it is practically not possible to store all the words in a language. Sometimes a bag of words approach is chosen for language identification task. In bag of words approach, only a specific list of words based on some linguistic exhibitory parameters are chosen. But this approach will be suitable only for language identification from a sentence or a phrase and is not applicable for a word. Hence words cannot be chosen to be features.

The other possibility is choosing features within a word. Generally, a word can be viewed as a sequence of characters, aksharas (in Indian language context), phonemes and syllables. The statistical language identification features include checking for the presence of certain characters, presence of certain words, presence of certain character n-grams and the frequency of character n-grams. These statistical features have been experimented with a number of analytic techniques compassing from semi automatic to fully automatic approaches. The accuracy of these methods generally decreases significantly when the text gets shorter.

As mentioned earlier the feature can be characters or phonemes or syllables or words. But we have chosen a syllable over the others for our loan word identification model adapted from the language identification model. The basic contrastive linguistic unit of any language is termed as a phoneme. Language identification from a single phoneme is a very absurd task as some phonemes are common for certain languages and they do not exhibit much of language specific information, but a sequence of phonemes can surely make a difference. Generally, trigrams of phonemes are good for providing some distinct language specific information. A syllable typically takes the C*VC* form, where a C is a consonant and V a vowel, on an average summing up to three phonemes. Evidently a syllable can capture language specific distinct information which can be used for language identification. Although commonality exists even on syllables between languages, the extent is comparatively lesser than phoneme commonality. To explain this hypothesis, let's take the domain of discourse to be Indian Languages, now if our task is to find out to which language the word with a syllable/zha/ belongs to, then the search space is immediately deduced to Tamil or Malayalam[24]. Hence syllables are powerful for the purpose of language identification as it helps us reduce the searching complexity both in terms of time and space.

The regressor variables or the independent variables of the MLR have been chosen after some experimentation from the corpus. Moreover the choice of features in language classification between two languages using MLR has already been experimented language-wise for short text segments for around 10 Indian languages[20]. After a little analysis using MLR on the training corpus, we found that the following features works well for the two class language identification problem in Tamil:

- Syllables which occur frequently in the word initial, medial and ending position in Tamil and as well the Non-Tamil corpus.
- Syllable bigrams which occur frequently in Tamil and the Non-Tamil corpus.
- Syllable trigrams which occur frequently in Tamil and the Non-Tamil corpus.
- Syllable fourgrams which occur frequently in Tamil and the Non-Tamil corpus.
- Syllable fivegrams which occur frequently in Tamil and the Non-Tamil corpus.

3.4 Training and Testing

In the present work, we have used syllable based n-grams in the language identification task. For building the n-grams used in loanword identification task, we have used three different text corpora. Firstly, text from the Tamil novel "Ponniyin Selvan" has been extracted which mostly has Tamil words. Secondly, Tamil text from the Wikipedia dump has been extracted that apparently contains a lot of other language words. The third corpus that has been used is the CIIL corpus, which contains data sourced from newspaper articles, recipes, books and textbooks. The corpus has been processed for removal of NSWs and only standard words are taken into consideration. A set of unique words have been extracted from the corpus and used for syllable based n-gram modeling.

Indian Languages follow a certain script grammar. A Finite State Machine (FST) has been used to depict the script grammar of Indian languages as in[20] and the same can be used to split the words into syllables. After syllabifying the corpus, we have romanized it for building the model using Multiple Linear Regression. In a similar manner, Telugu corpus of CIIL is subjected to language modeling. This corpus is required so as to help in identification of Tamil words by framing the language related feature set of Telugu and Tamil.

Let us consider on how the feature set matrix is constructed for training. For forming the feature set matrix, a word list, unigram list, unigram - start, middle and end list, bigram, trigram, fourgram, fivegram list are formed for both Tamil and the non-Tamil corpus. A list of unigrams in Tamil language is taken along with its frequency of occurrence and stored as the 'unigram list'. Redundant unigrams are discarded, so as to keep only a unique set of unigrams. Similarly a list of unique unigrams in non-Tamil language is formed along with its frequency of occurrence using the non-Tamil CIIL corpus. Same procedure is followed to form the other lists for both Tamil and non-Tamil corpus.

Another set of 3 lists are formed for each language for the unigrams. The unigrams occurring in the word initial, middle and ending position without redundancy along with its frequency of occurrence is listed as 'unigram-start list', 'unigram-middle list' and 'unigram-end list' respectively. Once all the lists are ready, a gramwise feature set matrix is constructed with the frequencies. The training is carried out for the combined Tamil and non-Tamil samples using this matrix as the regression values and a corresponding discrete value as the response values.

Two slight variations are followed for forming the regression matrix of the test data. Firstly, testing is done word wise and not n-gram wise. Secondly, the frequencies were replaced with a count value. Once the regression matrix for the test set is generated, each word in the test data will correspond to a total of 14 features as mentioned in the feature selection. Next, we compute the scores of response variables using MLR. From the scores generated, we have chosen a threshold value that varies in accordance with the length of word in n-gram wise. The words having the score above the threshold are considered to be Tamil otherwise non-Tamil class. The accuracy of the model is calculated with the help of the precision and recall parameters, in turn compounding to the f-measure.

The parameters are defined as:

Recall (R) = (CC/TW) * 100
Precision (P) = (IC/TW-U) * 100
F-measure = 2PR/(P+R)

Here, CC is count of correctly classified words, IC is count of incorrectly classified words, TW is Total words in the test sample and U is count of unclassified words.

3.5 Results

The language identification module has been tested with words of trigram, fourgram and fivegram in length, the input text segment has been as short as 3 characters. In the literature, the maximum shortest word that has been tested is of around 5 characters in length[25]. The fact that our loan word identification model works well for a very short input size of 3 characters length proves the novelty of the model.

The model when tested for trigram words around 500 resulted in an accuracy of 85.68%, for fourgram words around 1130 the accuracy was 91.71% and for fivegram words around 1300, the accuracy was 96.18%. From the accuracy it is quite evident that the accuracy increases as the input text segment's length increases.

Few examples from the trigram test data along with the score of the word, the predicted class and the original class are given in Table 6. The first column contains the word, the second column contains the score generated after running the MLR, the third and the final column contains the classification of the word in to either T or NT class, where T stands for Tamil and NT for non-Tamil class. The test data sample in the Table 6 contains pure Tamil words and some loan words. In the predicted class column, two incorrect predictions can be seen. The first IC is for the word அநீதீ (aNIdhi) which is actually a Tamil word but has been classified incorrectly as NT, the second IC is for the word காலேஜை (kAlEjai) which is a loan word but conjugated along with a Tamil morpheme. The word காலேஜை can hence be termed to be a bilingual word which is a result of the language mixing phenomena and as discussed earlier identifying such bilingual words is very difficult. We can see that the other loan words and Tamil words in the sample data is Classified Correctly.

Among the factors to be noted the most important is that even input as short as 3 characters in length

Table 6. Example of test data classification

Word	Score	Predicted Class	Original Class
கிராஃபை	0.000076	<NT>	<NT>
அநீதீ	0.000076	<NT>	<T>
ஜஷேடோ	0.000126	<NT>	<NT>
கிரீஸை	0.000246	<NT>	<NT>
கிளாஸி	0.000382	<NT>	<NT>
காலேஜை	0.000546	<T>	<NT>
உரையை	0.000620	<T>	<T>
இனிது	0.000736	<T>	<T>
சவலை	0.000797	<T>	<T>
ஏரியை	0.000974	<T>	<T>

is classified. We should also bear in mind that the complexities of the language mixing phenomena affect the performance of the model. Overcoming the reflection of the language mixing phenomena can help in improving the performance of the model.

But only a syllable based n-gram model may not be able to shoulder against these performance deteriorating factors, a hybrid system may help in increasing the performance. Some of the information which may be taken in to the feature set in future for the hybrid system could be probabilities of various character combinations, occurrence of diacritics and special characters, more exploration of syllable characteristics and morphology and syntax. Making use of a morphological analyzer along with the existing model could be of more advantage as the complexities prevalent in the developed approach mostly evolve around words which are either inter-lingual homophones or words which seem to have some inter-lingual morphological similarity.

4. Conclusion and Future Work

A text processing module for a Tamil Text to Speech System with text normalization and loan word identification have been discussed. The 'semiotic classifier' based decision list approach for text normalization is able to tackle many varieties of NSW but some exceptions still do exist. These exceptions are due to the enthralling facts of Tamil language's morphological richness. Verbalizing a NSW by analyzing the semiotic class is itself a challenging task, if the NSW happens to be a compound NSW then the complexity increases further. In addition to this if the NSW's are subjected to suffix addition due to the highly agglutinative and inflectional nature of Tamil language then the processing becomes highly complicated and it discerns the processing efficiency of the semiotic classifier. Next, the 'loan/native word classifier' based on multiple linear regression performs well even on shorter words of 3 syllables in length with an average accuracy of 91%. The hindrance in classifying the loan words is due to inter-lingual homophones and homographs and both are very strenuous to be identified from text. The performance of loan word classifier is sensitive for variable test samples and it can be increased further if we can develop a system to deal with inter-lingual homophones and homographs.

We can make use of the syllable feature set as an inception to the subsequent modules in a TTS. We can develop a pronunciation model by mapping the graphemes to syllables; work has already been done on segmenting syllables for developing a syllable based Tamil TTS[26]. Further duration and intonation modeling for increasing naturalness in a syllable based TTS can be explored with this similar kind of syllable feature set[27]. Hence this syllable level features will exhilarate the development of a thriving syllable based Tamil TTS.

5. References

1. Bigi B. A multilingual text normalization approach. 5th Language and Technology Conference: The 2nd LRL Workshop; 2011.
2. Gupta PS. Linguistic diversity and economic disparity: An issue for multiculturalism in India. The International Journal of Diversity in Organizations, Communities and Nations; 2009.
3. Xydas G, Karberis G, Kouroupertroglou G. Text normalization for the pronunciation of non-standard words in an inflected language. Methods and Applications of Artificial Intelligence. Springer Berlin Heidelberg; 2004. p. 390–9.
4. Sproat R, et al. Normalization of non-standard words. Comput Speech Lang. 2001; 15(3):287–333.
5. Pennell D, Liu Y. Toward text message normalization: Modeling abbreviation generation. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP); 2011.
6. Raj AA, et al. Text processing for text-to-speech systems in Indian languages. SSW; 2007.
7. Sproat R. Lightly supervised learning of text normalization: Russian number names. Spoken Language Technology Workshop (SLT). IEEE; 2010.
8. Ramakrishnan AG, Kaushik LN, Narayana L. Natural language processing for Tamil TTS. 3rd Language and Technology Conference (LTC); 2007.
9. Cook P, Stevenson S. An unsupervised model for text message normalization. Proceedings of the workshop on computational approaches to linguistic creativity. Association for Computational Linguistics; 2009.
10. Panchapagesan K, et al. Hindi text normalization. 5th International Conference on Knowledge Based Computer Systems (KBCS); 2004.
11. Zhou T, et al. A three-stage text normalization strategy for Mandarin text-to-speech systems. 6th International Symposium on Chinese Spoken Language Processing, ISCSLP'08; 2008.
12. Brody S, Diakopoulos N. Cooooooooooooooollllllllllllll! !!!!!!!!!!!!!: Using word lengthening to detect sentiment in microblogs. Proceedings of the conference on empirical

methods in natural language processing. Association for Computational Linguistics; 2011.
13. Llitjos AF, Black AW. Knowledge of language origin improves pronunciation accuracy of proper names. INTERSPEECH; 2001.
14. Romsdorfer H, Pfister B. Text analysis and language identification for polyglot text-to-speech synthesis. Speech Communication. 2007; 49(9):697–724.
15. Dunning T. Statistical identification of language. Computing Research Laboratory, New Mexico State University; 1994.
16. Sibun P, Reynar JC. Language identification: Examining the issues; 1996.
17. Hakkinen J, Tian J. N-gram and decision tree based language identification for written words. IEEE Workshop on Automatic Speech Recognition and Understanding. ASRU'01; 2001.
18. Tian J, Suontausta J. Scalable neural network based language identification from written text. IEEE International Conference on Acoustics, Speech, and Signal Processing, Proceedings ICASSP'03. 2003; 1.
19. Kruengkrai C, et al. Language identification based on string kernels. IEEE International Symposium on Communications and Information Technology, ISCIT. 2005; 2.
20. Murthy KN, Kumar GB. Language identification from small text samples. J Quant Ling. 2006; 13(1):57–80.
21. Bhargava A, Kondrak G. Language identification of names with SVMs. Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics. Association for Computational Linguistics; 2010.
22. Ng C-C, Selamat A. Improved letter weighting feature selection on arabic script language identification. First Asian Conference on Intelligent Information and Database Systems, ACIIDS; 2009.
23. Amine A, Elberrichi Z, Simonet M. Automatic language identification: An alternative unsupervised approach using a new hybrid algorithm (IJCSA). 2010; 7(1):94–107.
24. Vatanen T, Vayrynen JJ, Virpioja S. Language identification of short text segments with N-gram models (LREC); 2010.
25. Dey S, Murthy H. Unsupervised clustering of syllables for language identification. 20th Proceedings of the European Signal Processing Conference (EUSIPCO); 2012.
26. Natarajan VA, Jothilakshmi S. Segmentation of continuous Tamil speech into syllable like units. Indian Journal of Science and Technology. 2015; 8(17).
27. Shreekanth T, Udayashankara V, Chandrika M. Duration modelling using neural networks for Hindi TTS system considering position of syllable in a word. Procedia Computer Science. 2015; 46:60–7.