Manipuri Morpheme Identification

Kishorjit Nongmeikapam[1] *Vidya Raj RK*[2] *Yumnam Nirmal* [2] *Sivaji Bandyopadhyay*[3]
(1) Department of Comp. Sc. & Engg., MIT, Manipur University, Imphal, India
(2) Department of Comp. Sc., Manipur University, Imphal, India
(3) Department of Comp. Sc. & Engg., Jadavpur University, Kolkata, India
kishorjit.nongmeikapa@gmail.com,vidyarajrk@gmail.com,
yumnamnirmal@gmail.com, sivaji_cse_ju@yahoo.com

ABSTRACT

The Morphemes of the Manipuri word are the real bottleneck for any of the Manipuri Natural Language Processing (NLP) works. It is one of the Indian Scheduled Language with less advancement so far in terms of NLP applications. This is because the nature of the language is highly agglutinative. Segmentation of a word and identifying the morphemes becomes necessary before proceeding for any of the Manipuri NLP application. A highly inflected word may sometimes consist of ten or more affixes. These affixes are the morphemes which change the semantic and grammatical structure. So the inflexion in a word plays an important role. Words are segmented to the syllables and are examined to extract a morpheme among the syllables. This work is implemented in the Manipuri words written with the Meitei Mayek (script). This is because the syllable formations are distinct comparing to the Manipuri written with Bengali script. The combination of 2-gram or bi-gram and the Standard Deviation technique are used for the identification of the morphemes. This system gives an output with the recall of 59.80%, the precision of 83.02% and the f-score of 69.52%.

KEYWORDS : Manipuri, Morpheme, Segmentation, Syllable, Bi-gram, Standard Deviation

1 Introduction

Manipuri or Meiteilon which is synonymous with the Manipuri language is a highly agglutinative Indian scheduled language. A single root word can sometimes have 10 or more affixes which are attached one after another. For example, different suffixes can be agglutinated to the root word ꯄꯨ(pu) in order to produce the word ꯄꯨꯁꯤꯟꯍꯟꯖꯔꯝꯒꯗꯕꯅꯤꯗꯀꯣ(pusinhənjərəmgədəbənidəko), which means "I wish (I) myself would have caused to bring in the article". Here 10 suffixes are being attached to a bound verbal root (Nonigopal, 2006). Separation of the morphemes in the word are as follows: ꯄꯨꯁꯤꯟꯍꯟꯖꯔꯝꯒꯗꯕꯅꯤꯗꯀꯣ = ꯄꯨ/pu + ꯁꯤꯟ/sin + ꯍꯟ/hən + ꯖ/jə + ꯔꯝ/rəm + ꯒ/gə + ꯗ/də + ꯕ/bə + ꯅꯤ/ni + ꯗ/də + ꯀꯣ/ko

Morphology is the identification, analysis and description of the structure of morphemes and other units of meaning in a language such as words, affixes, part of speech etc (Denial J et.al. 2008). Morphemes are the smallest conceptual meaningful component of a word that has semantic meaning.

In this work, first the syllables are identified which is followed by the morpheme identification among the syllables. It is not guaranteed that all the syllables are morphemes. Manipuri usages two script a borrowed Bengali script and the original Manipuri script which is also known as the Meitei Mayek. This work is implemented in the Meitei Mayek Manipuri words. This is because the syllable formations are distinct comparing to the Bengali Script Manipuri.

The paper is organised with the related works in section 2, word segmentation into syllables in section 3, the system design in section 4 which is followed by experiment and evaluation in section 5 and at the last section 6 conclusions is drawn.

2 Related work

Morphological works in Germanic and other European languages can be found in (Braschler et.al, 2004). The various stemming algorithms comparative analysis for nine European Languages is presented in the survey report by (Tomlinson, 2003).

Some of the other works on Indian Language can be found in the work of (Utpal et. al., 2002a) where unsupervised learning technique is implemented for highly inflectional language. Another work of (Utpal et. al., 2008b) deals about acquisition of morphology of an Indic language from text corpus. Oriya morphological analyser design is reported in (Mohanty et. al., 2008).

Morpheme detection will of great help for Manipuri which is a highly agglutinative Indian Language. It is because these morphemes are meaningful units and the affixation of these morpheme changes semantic and POS of a word. This identification may lead us to an easy way for Machine translation. So far, identification of syllabic unit for Manipuri words are reported in the work of (Kishorjit et. al, 2012a). The work of Manipuri morphological analyser is reported in (Sirajul et.al, 2004). Stemming works of Manipuri is also reported in (Kishorjit et. al.,2011b). Morphological driven Manipuri POS tagging are also reported in (Singh et. al., 2008).

3 Word Segmentation into syllables

Word Segmentation in this context refers to the process of division of words into phonemic units or syllables that forms the whole word. It is an operation for dividing words into unit syllables by automatic means (Kalyani et. al., 2009). It is also known as *syllabification* (Bernd, 1998).

Segmentation may be used in a variety of applications which need the unit prosodic cues of the words, or need the proper understanding of the structural construction of the words (Jean-Luc, 2007). The word segmentation of Manipuri follows the algorithms and the systems mention in (Kishorjit et. al., 2012b) with certain modifications.

3.1 A brief about Meitei Mayek characters

It will be better if the characters used in the Meitei Mayek are discussed in brief. The work is based on the 27 scripts approved by the State Government of Manipur. which is also published in (Kangjia, 2003). The characters used in Meiteilon (Manipuri language) can be classified into five different categories.

- Iyek Ipee : This character set consists of 27 letters which are mainly major consonants, out of which three are used to produce vowel sounds (ꯑ, ꯏ, ꯎ). This category is considered as major consonants in the sense that letters are used in their full form at the initial position of a syllabic unit. Moreover, associations with Cheitap Iyek are permitted with these characters only.
- Cheitap Iyek (Matras): These are associative symbols which can be found only in association with Iyek Ipee character sets. Association with Iyek Ipee characters follow a one to one relationship i.e. no two (or more) symbols is found to be associated with one letter in Iyek Ipee. Consecutive occurrence is also not permitted.
- Cheising Iyek (Numerals): This set contains the numeral figures and follow the decimal system.
- Lonsum Iyek: There are 8 characters in this set and these characters can be considered to be derivative form of 8 distinct letters in Iyek Ipee. In one sense, these letters can be regarded as half consonants as they cannot be associated with any symbols in Cheitap Iyek and cannot initiate formation of a syllable. This character set can only be present in the syllabic final position. Recurrence or clusters of these characters i.e. consecutive occurrence of these characters are also not permitted in the language.
- Khudam Iyek (Symbols): Usage of special characters is limited in this language and as such few symbols suffice the need in expression.

  Examples:

  '||' - Cheikhei (Full Stop)

  '.' - Lum Iyek (Sign of intonation) eg. ꯆ.ꯕ (*cha.ba* (to eat)) falling intonation and ꯆ꯬ꯕ (*cha.ba* (swimming/floating)) rising intonation.

  '_' - Apun Iyek (Sign of Ligature) eg. ꯆꯝꯄ꯭ꯔꯥ (*cham.pra* (lemon))

Other symbols are as internationally accepted symbols.

Before going deep into the syllabic structure it is important to discuss the usage of Apun Iyek for better understanding. This symbol is used to indicate clustering of Iyek Ipee characters to produce a combined sound.

ꯆꯝꯄ꯭ꯔꯥ	→	cəm.pra (lemon)
ꯆꯠꯈ꯭ꯔꯦ	→	cət.khre (gone)

Important thing to be noted is that clustering of more than two characters in Iyek Ipee is rare and Cheitap Iyek, if present (along with Apun Iyek), are found in association with the second Iyek Ipee character.

3.2 Syllabic unit structure in Meitei Mayek Manipuri

A hand craft structure is identified. This rule is used to identified the formation of a mono syllable Manipuri origin words without occurrence of consonant clusters. Basically, the structure consists of three cells: the initial position, intermediate position and the final position.

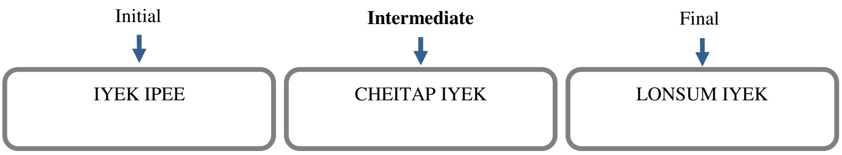

FIGURE 1 – The character set and positions of syllabic unit structure in Meitei Mayek Manipuri

3.2.1 Pattern of Character Sets in Different Positions in Mono Syllable

In this section, occurrences of different character sets in different position contributing to the formation of syllables are discussed along with exceptional cases.

3.2.1.1 Initial Position

The initial position should always be a character from Iyek Ipee. The conditions apply to every syllable in a Meiteilon origin word, regardless of exceptions.

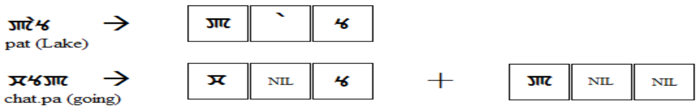

FIGURE 2 – Illustration of Initial Position with examples.

The first example is a mono syllabic word, in which the initial position is occupied by the character 'ꯄ' belongs to Iyek Ipee.

The second example is a word formed by two syllables where in both cases the character in initial position belongs to Iyek Ipee character set.

3.2.1.2 Intermediate Position

The intermediate position, generally, is occupied by the character set in Cheitap Iyek. This condition may not apply in case of consonant clustering and exceptional cases. Moreover, it is not necessary that this position be occupied.

In the first example shown in Fig 3.3, the occurrence of ' ` ' , a character in Cheitap Iyek in the intermediate position is illustrated, while in the second example the nonexistent state in this position is shown in **Figure 3.3**.

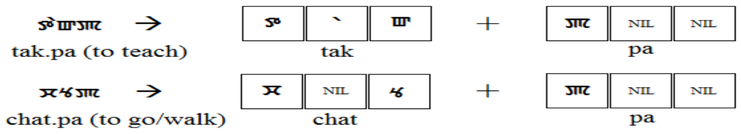
FIGURE 3 – Illustration of Intermediate Position with examples.

3.2.1.3 Final Position

Normally, the final position is filled in by characters from the Lonsum Iyek character set, but, as in the previous case, it is not a necessary condition. Syllables with nonexistent final position are common in the language.

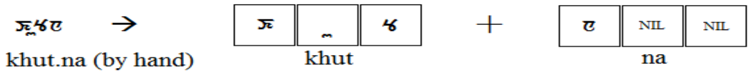
FIGURE 4 – Illustration of Final Position with examples.

In this example, the final position of the first syllable is occupied by Lonsum Iyek character ‘ ꯠ ’ whereas in the second syllable the position is empty.

3.2.1.4 Occurrence of Exceptional Characters in Final Position

In Meiteilon some exceptional characters belonging to Iyek Ipee are found to occur in the final position of a syllable. Special attention is to be given to each and every letter in this group because understanding the correct utilization of these letters is important for correct and proper command over the language.

One such character is the letter ‘ ꯏ ’(i). This letter is present in both Iyek Ipee character set as well as Lonsum Iyek. So, if it initializes a syllable, should it be considered as a major consonant, and if it finalizes the syllable, should it be considered as Lonsum Iyek? This issue is purely linguistic in nature. But from a computational view this is a character which has the capability to initiate as well as finalize syllable and its importance should be kept in mind while developing algorithm.

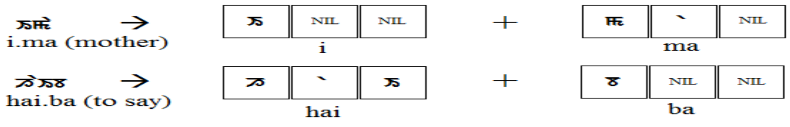
FIGURE 5 – Illustration of Exceptional cases with examples.

Similarly, the letters ꯋ,ꯌ,ꯎ,ꯑ are found to be used in both initial and final positions, but these letters do not have their derivative counterparts in Lonsum Iyek character set.

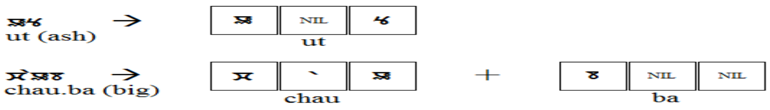

FIGURE 6 – Illustration of Exceptional cases with examples.

3.3 Complex Syllabic Structure

Majority of the syllable are of the basic structure. But occurrence of a slightly complex structure is found if the use of (_) Apun Iyek is found (mostly in close association with ꯌ, ꯔ, ꯋ).

In this case, a syllabic unit is divided into five cells, comprising of initial, final and three intermediate positions.

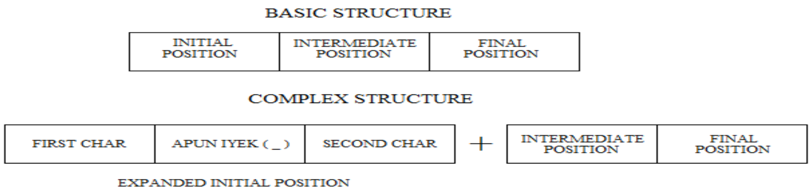

FIGURE 7 – Comparison of Simple and Complex Syllabic Structure.

In another sense, the first three cells can be thought as the expanded form of the initial position in basic structure. The last two cells (4th and 5th) are same as that of basic structure.

The first three cells i.e. expansion of the initial position, the second cell is meant to be occupied by the Apun Iyek (_) and the first and the third cell is filled in by characters from Iyek Ipee character set.

3.4 The Syllabic Pattern

Syllable can be divided into three parts; onset (beginning of a syllable, either a consonant or a semivowel), peak (nucleus of the syllable, vowels) and coda (sound which comes after the peak, generally consonants). In every syllable there must be a peak. But there may not be an onset or coda in Meiteilon syllabic system.

Referring to section 2.4.1 of (Yashawanta, 2000), the syllabic structure, the author has stated that the syllables can be of six forms.

Classification A

1. V
2. VC
3. CV
4. CCV
5. CVC
6. CCVC

where,
V = Syllabic peaks, vowels
C = Syllabic margins, consonants

In this process, syllables are segmented using a script based algorithm and thus the pattern observed also are on the basis of the script (Kishorjit et.al. 2012b). The patterns found are as follows;

Classification B

1. V
2. CV
3. C
4. VVV
5. CVC
6. CC
7. CVV
8. VV
9. VVC
10. VC
11. CCVC

where,
V = vowel characters
C = consonant characters

It must be noted that the previous classification A is based on linguistic approach and the classification B, on computational view.

It may seem like the later classification shows more variety of patterns, but it is not so. Both, classifications are one and same, the only difference lies in the approach. This is elaborated in (Kishorjit et. al., 2012b).

4 System design

The system design of this work can be divided into two parts. The first part is designed in order to identify the syllabic units of a word and the second part is the identification of morphemes from the syllabic units.

4.1 System design for word segmentation

The identification of the syllabic unit follows the same hand craft rule used in the work of (Kishorjit et. al., 2012b). Fig. 8 shows the details of the segmentation system. The first method (segmentor) provides the base foundation in which the Input File is read line by line and every line is tokenized and every tokens or word is provided for syllable extraction. Then from the stack, where the syllables for every word are stored, the unit/mono syllables are again written into the Output File.

The second method (extractor), when called by the first, takes a string parameter (word) and segments the word into unit syllables. Segmentation is done depending on the script based rules and the syllabic structures defined in this chapter. For every syllable extracted it is pushed down into a stack object defined for the particular word. The extraction process starts from the left most syllables, thus suitable for storing in a stack.

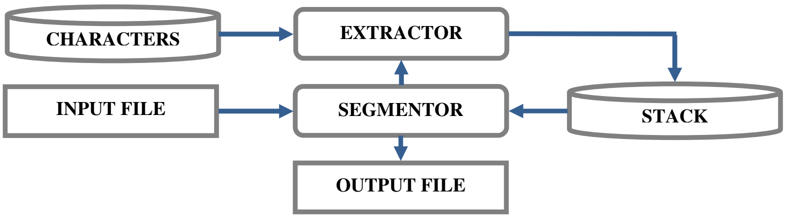

FIGURE 8 – System Diagram of Segmentation

The *Input File* is passed to the segmentation algorithm. The segmentation algorithm tokenizes the string in the file and calls the extractor algorithm passing each word as parameter. The extractor algorithm refers to the character list for checking the various conditions present and pushes down each extracted segment in the stack. A stack is used because the segmentation starts from the end of the word and LIFO (Last In First Out) principle is followed. When extraction is over, control is shifted to segmentation algorithm and it takes the segments from the stack and gives the output.

4.2 Morpheme identification system

With the principle that word consists of syllable or syllables and the syllable can be morpheme but not guaranteed that all the syllables will be a morpheme. For example in the word "unladylike", there are 4 syllables (un-, la-, dy- and like) but only 3 morphemes (un-, lady- and like). Taking a Manipuri word example ꯁꯌꯣꯛꯁꯪ/sa.yok.saŋ/ "*animal rearing place or zoo*", there are three syllables, out of which the first one ꯁ /sa/ "*animal*" is a free root (noun) and the second one is ꯌꯣꯛ /yok/ "*bring up/ rear*" is a bound root (verb) and the last one ꯁꯪ /saŋ/ "*house/shed*" is again a free root (noun). Together these three syllables form the word ꯁꯌꯣꯛꯁꯪ/sa.yok.saŋ/ meaning "*zoo/animal rearing place*" also altogether there are three morphemes.

4.2.1 Pre-processing and segmentation process

The system (Fig. 9) is feed with a text corpus, so pre-processing is very important and the process consists of cleaning of the file. By cleaning it means the correction of spellings, removal of unwanted characters and the grammatical errors. The cleaned file is fed as the input file. The input file is procured from local newspaper Huiyen Lanpao[1] which provides a Meitei Mayek version of the daily news. Approximately some well cleaned 13,045 words are used for the system input file.

The Cleaned file is fed as an input for the segmentation process. The segmentation process follows the same rule mention in section 4.1. The segmented file with syllables as output is used for the next module of morpheme identification.

[1] http://www.hueiyenlanpao.com/meeteilon/

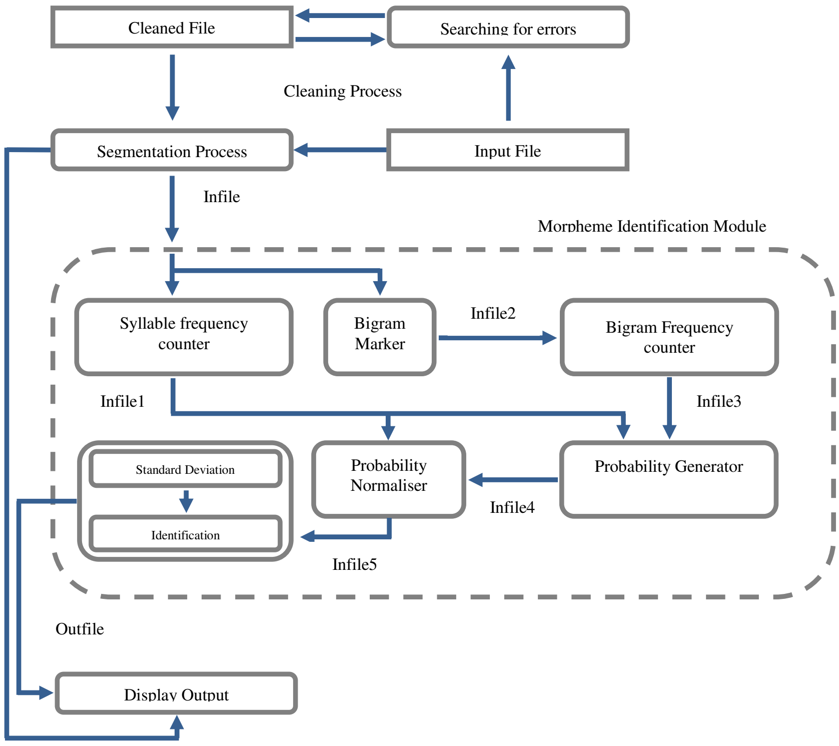

FIGURE 9 – System design

4.3 Morpheme identification module

In the morpheme identification module the syllables are used for frequency finding and other statistical findings. The Bigram model is used in order to measure the relative association between each syllable present in a word and find their corresponding probability values. This probability values are normalized so as to bring uniformity in the data and prevent redundancy and duplication.

The output of Bigram probability is implemented with Standard deviation.

4.3.1 N-gram and Bigram

In order to start a brief discussion of N-gram, the concept of conditional probability is necessary. In probability theory, the **conditional probability** of **A** given **B** is the probability of **A** if **B** is known to occur (or have occurred). It is commonly denoted as $\boldsymbol{P(A/B)}$, and sometime $\boldsymbol{P_B(A).P(A/B)}$ can be visualized as the probability of event **A** when the sample space is restricted to event **B** (Freund et. al., 2006).

Mathematically, it is defined for $P(B) \neq 0$ as :

$$P(A/B) = \frac{P(A \cap B)}{P(B)}$$

An ***n*-gram model** is a type of probabilistic language model for predicting the next item in such a sequence in the form of a **(n-1)** order Markov model.In the fields of computational linguistics and probability, an ***n*-gram** is a contiguous sequence of *n* items from a given sequence of text or speech. The items in question can be phonemes, syllables, letters, words or base pairs according to the application. *n*-grams are collected from a text or speech corpus. Mathemetically, it is defined as :

Given a string of words ($w_1, w_2, w_3, \ldots\ldots, w_n$), we can compute the probability of the complete string using n-gram. If we consider each word occurring in its correct location as an independent event, we can represent this probability as follows:

$P(w_1, w_2, w_3, w_4, \ldots\ldots, w_n)$

This can be further decomposed by the chain rule of probability as follows:

$P(w_1^n) = P(w_1)P(w_2|w_1)P(w_3|w_2)P(w_4|w_3) \ldots\ldots\ldots\ldots\ldots P(w_n|w_1^{n-1})$

$= \prod_{k=1}^{n} P(w_k \mid w_1^{k-1})$

And in order to compute the probability like $P(w_n \mid w_1^{n-1})$, we have to approximate the probability of a word given all previous words.

An *n*-gram of size 1 is referred to as a "unigram"; size 2 is a "bigram", size 3 is a "trigram" and so on. In this system it has restricted to the **bigram** model which approximates the probability of a word given all the previous words $P(w_n \mid w_1^{n-1})$ by the conditional probability of the preceding word $P(w_n \mid w_{n-1})$

As an example, instead of computing the probability

***P*(rabbit | Just the other day I saw a)**

We approximate it with the probability:

P (rabbit | a)

Implementing in the syllabic level, for example for the word ꯁꯥꯌꯣꯛꯁꯪ/sa.yok.saŋ/ "*animal rearing place or zoo*", the three syllables are ꯁꯥ /sa/ "*animal*", ꯌꯣꯛ /yok/ "*bring up/ rear*" and ꯁꯪ /saŋ/ "*house/shed*". And the sample Bigram probability notation is:

P(ꯌꯣꯛ|ꯁꯥ), P(ꯁꯪ|ꯌꯣꯛ), etc.

4.3.2 Standard deviation

Standard deviation is the degree of variation in the probability distribution. It expresses the amount of dispersion from the average (mean).

The mean, denoted by μ, of a distribution or set of random variables X is given by:

$$\mu = \frac{\sum_{i=1}^{n} x_i}{n}$$ Where, n = total elements in a population/sample

The standard deviation of a probability distribution is given by the positive square root of its variance. Variance is the square of summation of the difference between the probability values and the mean value divided by the population size.

Variance,

$$\sigma^2 = \frac{\sum_{i=1}^{n} x_i - \mu}{n}$$

Standard Deviation,

$$\sigma = \sqrt{\frac{\sum_{i=1}^{n} x_i - \mu}{n}}$$

Standard deviation (σ) and mean (μ) of the probability distribution is calculated for the given syllables. In Standard Deviation, the domain range is defined as the distance between one σ difference from μ on both negative and positive values. And this range is used to determine whether a syllable is a morpheme or not. In our case only the positive values of μ is considered because negative value is meaningless in our case. It is because the syllabic unit bigram frequency probability outputs do not have negative values.

5 Experiment and evaluation

A total of 13,045 word forms were supplied. The outcome was 35,840 syllables and 972 of them were distinct syllables. In order to evaluate the experiment, the system used the parameter of Recall, Precision and F-score.

Recall,

$$R = \frac{total\ no\ of\ correct\ morpheme\ given\ by\ the\ system}{total\ no\ of\ possible\ correct\ morpheme\ in\ the\ text}$$

Precision,

$$P = \frac{total\ no\ of\ correct\ morpheme\ given\ by\ the\ system}{total\ no\ of\ morpheme\ given\ by\ the\ system}$$

F-measure,

$F = \frac{(\beta^2+1)PR}{\beta^2 P+R}$ **;** where, β is one, which means precision and recall are given equal weight.

The Fig. 10 shows the graph of the syllables with the probability values. It can be observed that the ranges between **0.01** and **0.08** are clustered together. This indicates that there is some consistency in this range and as mention above only these positive values range is considered even though in the standard deviation it has a range both comprises of negative and positive values. It is because negative value is meaningless since it's calculated with the syllabic unit bigram frequency probability outputs which do not have negative values.

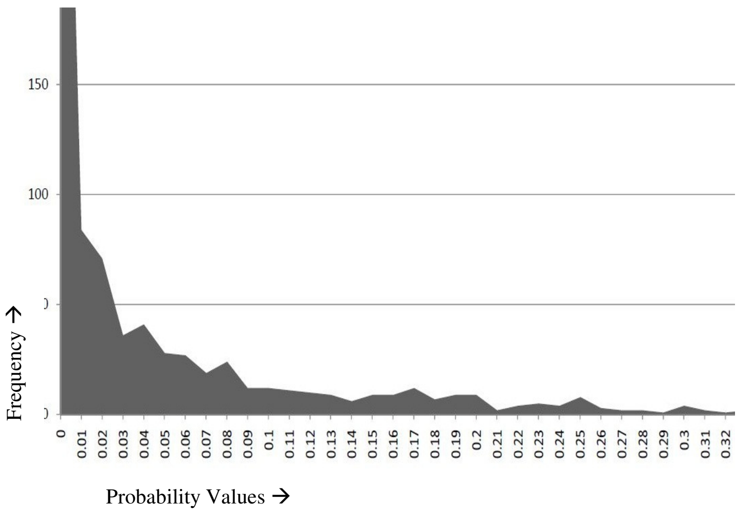

FIGURE 10 – Standard Deviation

The measurement is done using the evaluation method of recall, precision and f-score mention above. So, the system gives a Recall of **59.80%**, Precision of **83.02%**, and F-score of **69.52%.**

6 Conclusion

The system designed is a morpheme identifier not a morphological analyser, so the job of this system is to identify the morpheme from a set of syllables. A syllable segmentation algorithm is applied and the output syllables are verified whether it's a morpheme or not. In the identification, the concept of standard deviation is implemented. There is no such report of identifying Manipuri language morphemes using syllable or syllabic unit inputs.

These could be a boon for this highly agglutinative language since the complexity of Machine translation could be solved up-to a great extend. Other statistical method can also be implemented to improve the output.

References

Bernd Mo¨bius (1998): Word and syllable models for German text-to-speech synthesis, In *Proceedings of the Third International Workshop on Speech Synthesis*, Jenolan Caves, Australia, pages 59–64.

Braschler, M. and Ripplinger, B. (2004): How Effective is Stemming and Decompounding for German Text Retrieval? *Information Retrieval*, Vol. 7, (3-4), pages 291-306.

Daniel Jurafsky, James H. Martin, 2008. *"Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition"*, Dorling Kindersley (India) Pvt. Ltd., licensees of Pearson Education in South Asia.

Freund's E. John, Irwin Miller and Marylees Miller (2006). "*Mathmatical Statistics with Applications*", Dorling Kindersley (India) Pvt. Ltd., licensees of Pearson Education in South Asia.

Rouas, (2007): Automatic prosodic variations modeling for language and dialect discrimination, *IEEE Transaction on Audio, Speech and language Processing*, Vol. 15, 6, pages 1-8.

Kalyani, N. and Sunitha, K.V.N. (2009): Syllable analysis to build a dictation system in Telugu language, *International Journal of Computer Science and Information Security*, Vol.6, No. 3, pp. 171-176.

Kangjia Mangang, Ng. (2003). *"Revival of a closed account; A Brief History of Kanglei Script and the Birth of Phoon (zero) in the World of Arithmetic and Astrology"*, Sanamahi Laining Amasung Punsiron Khupham, Imphal, India.

Kishorjit N., Bishworjit S., Romina M., MayekleimaChanu Ng., Sivaji B., 2011. *"A Light Weight Manipuri Stemmer"*, In: The Proceedings of NCILC, Cochin, India.

Kishorjit N., Vidya Raj RK., Imocha Singh O. and Sivaji B., 2012. *"Automatic Segmentation Of Manipuri(Meiteilon) Word Into Syllabic Units"*, International Journal of Computer Science and Information Technology (IJCSIT). ISSN – 0975 – 3826

Mohanty, S., Santi, P.K. and Adhikary, K.P.D. (2004): Analysis and Design of Oriya Morphological Analyser: Some Tests with OriNet. In: Proceeding of symposium on Indian Morphology, phonology and Language Engineering, IIT Kharagpur.

Nongthombam Nonigopal Singh, 1987. *"A Meitei Grammar of Roots and Affixes"* (Unpublished Thesis), Manipur University, Canchipur, Imphal, India 795003.

Singh, T. D. and Sivaji B. (2008): Morphology driven Manipuri POS tagger, In *Proceedings of IJCNLP-08 workshop on NLP for Less Privilege Languages*, India, pages 91-98.

Sharma, U., Kalita, J. and Das, R. (2002): Unsupervised learning of morphology for building lexicon for a highly inflectional language. In ACL SIGPHON, pages 1–10.

Utpal Sharma, Jugal K. Kalita, and Rajib K. D. (2008): Acquisition of morphology of an Indic language from text corpus, *ACM Transactions on Asian Language Information Processing (TALIP)*, vol. 7, Issue 3, August 2008.

Tomlinson, S. (2003): Lexical and Algorithmic Stemming Compared for 9 European Languages with Hummingbird SearchServerTM, In CLEF 2003, pages 286–300.

Yashawanta Singh, Ch. (2000). "*Manipuri Grammar*", Rajesh Publications New Delhi – 110002.