International Journal of Applied Mathematics
Volume 38 No. 8s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

ANALYSIS OF CHARACTER BIGRAMS AND TRIGRAMS IN ASSAMESE LANGUAGE WITH AN APPLICATION TO PARTS OF SPEECH TAGGING

Dr. Diganta Baishya[1*], Dr. Rupam Baruah[2], Dr. Mousoomi Bora[3]

[1*]Jorhat Engineering College, dbaishya@gmail.com
[2]Jorhat Engineering College, rupam.baruah.jec@gmail.com
[3]The Assam Kaziranga University, mousoomi@gmail.com

Abstract:
Applications based on natural language processing (NLP), have uplifted the living standards of humans in recent times. However, the lack of annotated corpora and other digital resources poses some significant challenges to design a highly accurate NLP applications in most of the languages in this world. Bigrams and trigrams of characters are vital components of words and can be useful while designing NLP applications specially for languages with limited resources. This paper reports an analysis of the bigrams and trigrams of characters in Assamese language. It also presents a method to design parts of speech (POS) tagger for Assamese and proposes a framework to classify words as per their POS tags based on the frequency of character bigrams and trigrams. The paper also demonstrates the effectiveness of the method with the help of experimental results. The methods are applicable to any language, but experiments discussed here are based on Assamese language. Even though the accuracy is not very high, but the results are satisfactory in comparison to limited size of required training data.

Key Words and Phrases: Natural language processing, Low resource languages, Bigrams, trigram, N gram, POS tagging, Assamese

1. Introduction

One of the key developments in computing technologies is natural language processing (NLP). It is primarily concerned with the development of methods that enable computer systems to understand, process, and generate natural languages [1]. It focuses on applications, such as speech recognition, language translation, sentiment analysis, and emotion detection. Most of the NLP applications are available in some specific languages like English that are rich in digital resources. Such applications work with considerably high accuracy because the most of the popular methods perform better with large amount of training data. With limited research and limited available resources in digital form, the benefits are not visible in case of most of the languages spoken by a large population of people in the world. Therefore, it is very much important that researchers contribute towards developing methods to work with limited resources. Using the hidden characteristics of language features often prove to be useful in such cases. Bigram and trigram combinations of characters in a word represent very important characteristics of a word. This paper contributes by analyzing the frequency of bigrams and trigrams on a low resource Assamese language, which is a low resource language with limited resources in digital form. The paper also proposes a framework to implement a POS tagger based on the frequency of character bigrams and trigrams in a word. TPOS tagging is an essential phase for subsequent processing of text in many NLP applications.

2. Literature survey

Bigrams of characters and bigram-based process models have been used for text classification by researchers in recent times [2] [3]. N-gram-based word recognition methods have been used recently for Polish, which is a representative of West Slavic languages [4]. Researchers have also

Received: August 09, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

assessed the performance of using character trigrams for sentiment analysis and text classification in the Arabic language [5]. Research in the Assamese language has also been conducted recently to understand the effect of using characters and bigrams of characters with deep learning models in the Assamese language [6] [7]. The literature survey of POS tagging reveals some of the popular methods for POS tagging. Some of them are briefly outlined below:

Rule-Based Method: A rule-based POS tagger depends on already defined rules. These rules usually vary from language to language. Implementing such a rule-based approach requires linguistic experts.

Naive Bayes Classifiers: It is a probabilistic technique that uses probability theory and Bayes' theorem to determine the tag of elements. It helps to determine the possibility of a word belonging to a particular tag. Khurana et al. [8] reported that Naive Bayes Classifiers are excellent choices for tasks like POS tagging, segmentation, translation, word sense disambiguation, document identification, opinion mining, etc.

Hidden Markov Model: HMMs are like Legos when it comes to computational sequence analysis, as stated by Eddy [9]. It is a prevalent approach for POS tagging and speech recognition. The HMM uses three entities: outputs, transitions, and states. In the case of POS tagging, states are tags, outputs observed are the words, and transitions happen from one state to another. Martinez [10] stated that it is necessary to train such a model on a massive training dataset to obtain high accuracy. Hidden Markov Models are very popular among researchers involved in designing POS taggers. They perform better than rule-based methods and some other conventional machine learning approaches.

Neural Network and Deep Learning: The popularity of neural networks has recently achieved new heights in many fields, and POS tagging is also not an exception. Before 2010, most NLP tasks were performed using conventional techniques. However, the use of neural networks soon started changing the dynamics due to their capability of learning multilevel features.

However, these approaches cannot be applied effectively to low-resource languages (LRLs) due to the lack of quality digital texts available for training. Most of the new methods that are proposed in fields like machine translation (MT) work only with more training data [11] [12] [13]. King [14] reported that when these same techniques are used on LRLs with few resources, the results are often not satisfactory. It is important, therefore, to work on methods that require minimum labeled datasets so that they can be applied on LRLs. One of the popular LRLs, widely spoken across Assam and its neighboring states in India, is the Assamese language. Recently significant research works have been initiated for processing low-resource languages like Assamese [15] [16] [17] [18]. This paper focuses on attempts to achieve a better recognition rate for parts of speech tagging of words with limited digital resources for training the model. The method is applicable to any language, but the experiments discussed here are based on Assamese language.

3. ASSAMESE DATASET

A labelled Assamese dataset has been developed in-house from a large collection of 20 text files from different genres, including stories and novels, history, law, science, music, drama, monthly magazines, political science, reviews, and jokes. The Assamese dataset contains 5,421 sentences consisting of 81,244 words. The words are categorized into the following basic POS tags in Assamese: 'বিশেষ্য (Noun)', 'বিশেষণ (Adjective)', 'সর্বনাম (Pronoun)', 'অব্যয় (indeclinable)' and 'ক্রিয়া (Verb)'. The dataset reveals that most of the words in Assamese are the nouns. The verbs and adjectives are the next two most frequent tags.

- **The Vowels in Assamese:** There are eleven (11) vowels in Assamese. They are :

অ আ ই ঈ উ ঊ ঋ এ ঐ ও ঔ

Received: August 09, 2025

International Journal of Applied Mathematics
Volume 38 No. 8s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

- **The Consonants in Assamese:** The consonants in Assamese are:

ক	খ	গ	ঘ	ঙ
চ	ছ	জ	ঝ	ঞ
ট	ঠ	ড	ঢ	ণ
ত	থ	দ	ধ	ন
প	ফ	ব	ভ	ম
য	ৰ	ল	ৱ	
শ	ষ	স	হ	
ক্ষ	ড়	ঢ়	য়	
ৎ	◌ং	◌ঃ	◌ঁ	

4. Character bigrams and character trigrams in Assamese

Character bigrams are formed with sequences of two consecutive character symbols from the alphabet. Similarly, character trigrams are formed with sequences of three consecutive character symbols from the alphabet. Like many other languages, the Assamese alphabet consists of vowels, consonants, and numeric symbols. The frequency of character bigrams and trigrams serves as a vital parameter for any language, reflecting many hidden characteristics of the language. This can be helpful for word classification. The Table I illustrates this with the help of some of the common Assamese words along with their corresponding bigram and trigram character combinations.

Table-I: Bigram and Trigram of characters for ten common Assamese words

Assamese Word	**Character Bigrams**	**Character Trigrams**
অসম	অস, সম	অসম
মানুহ	মা, ান, নু, ুহ	মান, ানু, নুহ
ভাল	ভা, াল	ভাল
ঘৰৰ	ঘৰ, ৰৰ	ঘৰৰ
আকাশ	আক, কা, কাশ	আকা, কাশ
সূর্য	সূৰ, র্য	সূৰ্, ুর্য
চকু	চক, কু	চকু
শিক্ষা	শিক্ষ, ক্ষা	শিক্ষা

The Table-II below briefly describes the distribution of some of the character bigrams across basic Assamese POS tags.

Table-II: Distribution of Character Bigrams Over Assamese Tags

Bigram-POS	বিশেষ্য	বিশেষণ	সর্বনাম	অব্যয়	ক্রিয়া
অক	0.18	0.00	0.00	0.00	0.82
অখ	0.00	0.00	0.00	0.00	1.00
অগ	0.86	0.00	0.00	0.00	0.14
আশ	0.88	0.00	0.00	0.00	0.12
ইক	0.29	0.00	0.00	0.00	0.71
ইখ	0.10	0.00	0.50	0.00	0.40
ইস	0.50	0.00	0.33	0.00	0.17

Received: August 09, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

ইয়	0.00	0.00	1.00	0.00	0.00
কন	0.80	0.00	0.00	0.00	0.20
কৰ	0.27	0.67	0.04	0.00	0.01
কল	0.54	0.00	0.03	0.00	0.42
কশ	0.00	0.00	0.00	0.00	0.00
কষ	0.50	0.00	0.00	0.00	0.50
কস	0.93	0.00	0.00	0.00	0.07
কি	0.07	0.13	0.02	0.47	0.31
কা	0.64	0.11	0.02	0.09	0.14
কে	0.45	0.03	0.20	0.00	0.32
ষণ	0.94	0.00	0.00	0.00	0.06
ষত	0.68	0.00	0.00	0.00	0.32
ঁত	0.15	0.31	0.31	0.00	0.23

Table - II shows only a few selected samples of the possible bigram combinations of Assamese characters over POS tags. Similar analysis for other character combinations are also performed. Such a table contains critical information on words being assigned to a particular POS tag.

5. Methodology for Bigram trigram extraction and parts of speech tagging

Algorithms are proposed to extract the possible combinations of character bigrams and character trigrams. An ordered sequence of characters of a specified length can play a significant role in determining the POS tag of the word. Rules can be defined based on the frequency of bigram or trigram sequences of characters to determine the POS tag of a word. The following is a framework of N-gram-based algorithms that employs a character-level n-gram–based approach for predicting Parts of Speech (POS) in Assamese. The framework is effective even for low-resource languages where the size of annotated corpus is limited. The overall process consists of four key algorithms described below.

Algorithm 1: Character Bigram and Trigram Extraction
Objective: Generate all possible bigram and trigram character sequences from a given word.
Input: A word W consisting of n characters
Output: Lists of Bigrams(W) and Trigrams(W)
Procedure:

Step A. Initialize empty lists BIGRAMS and TRIGRAMS.
Step B. For i ← 0 to n – 2:
append substring W[i : i+2] to BIGRAMS.
Step C. For i ← 0 to n – 3:
append substring W[i : i+3] to TRIGRAMS.
Step D. Return BIGRAMS, TRIGRAMS.

For Example:
For W = “ঘৰৰ”,
BIGRAMS = [“ঘৰ”, “ৰৰ”],
TRIGRAMS = [“ঘৰৰ”].

Algorithm 2: N-gram → POS Fraction Distribution

Received: August 09, 2025

International Journal of Applied Mathematics
Volume 38 No. 8s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Objective: Build a table representing the fractional distribution of each n-gram (bigram or trigram) across POS tags.
Input:
- List of training words W[1..N]
- Corresponding POS tags T[1..N]
- N-gram length k (2 for bigram, 3 for trigram)

Output: Fraction table F where each row (n-gram) stores fractional probabilities across POS tags, total summation Σ_p F(n-gram, p) = 1.0.; Table F_2 for bigrams; F_3 for trigrams
Procedure:

Step A. Initialize count[n_gram][pos] ← 0.
Step B. For each (W_i, T_i) in training data:
For j ← 0 to len(W_i) – k:
ng ← W_i[j : j+k]
count[ng][T_i] += 1
Step C. For each ng in count:
total ← Σ_p count[ng][p]
For each POS p:
F(ng, p) ← count[ng][p] / total

Step D. Add column Total = Σ_p F(ng, p) ≈ 1.0.

Result: Table representing P(POS | n-gram), used later for inference.

Algorithm 3: POS Prediction from N-gram Distributions
Objective: Predict the most probable POS tag for an unseen word using the learned bigram and trigram fraction tables.
Input:
- Test word W
- Fraction tables F_2 (bigrams) and F_3 (trigrams)
- Set of POS tags POS
- Equal weights α = β = 0.5

Output: Predicted POS tag for W.
Procedure:

Step A. Extract B ← Bigrams(W) and, if len(W) ≥ 3, T ← Trigrams(W).
Step B. For each pos ∈ POS:
S_2[pos] ← Σ_{b∈B} log(F_2[b][pos] + ε)
If len(W) ≥ 3:
S_3[pos] ← Σ_{t∈T} log(F_3[t][pos] + ε)
Step C. If len(W) ≥ 3:
Score[pos] ← α·(S_2[pos]/|B|) + β·(S_3[pos]/|T|)
Else Score[pos] ← S_2[pos]/|B|.
Step D. Return POS* = argmax(Score[pos]).

Notes:
- ε is a small smoothing constant (e.g., 10^{-6}).
- If word length < 3, only bigrams are used.
- Both n-gram sequences contribute equally to the decision.

Algorithm 4: Evaluation and Accuracy Computation

Received: August 09, 2025

International Journal of Applied Mathematics
Volume 38 No. 8s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Objective: Measure model performance for different train/test splits by comparing predicted and actual POS tags.

Input:
- Training and test sets
- Predicted vs. actual POS labels

Output: Accuracy metrics for known, unknown, and overall words.
Procedure:

Step A. Count total words in train/test sets.
Step B. Separate **known** (seen in training) and **unknown** (unseen) test words.
Step C. For each category:
Compute Detected Correct and Detected Wrong.
Step D. Compute accuracies:
Unknown Accuracy = (Unknown Correct / Unknown Total) × 100
Known Accuracy = (Known Correct / Known Total) × 100
Overall Accuracy = (Total Correct / Total Test Words) × 100
Step E. Summarize results across multiple train/test ratios.

Table-III shows a summary of the proposed framework

Table -III: Summary of the Framework

Phase	**Algorithm**	**Role**
Pre-processing	Algorithm 1	Generate character bigrams/trigrams
Training	Algorithm 2	Build fractional n-gram→POS tables
Prediction	Algorithm 3	Determine most probable POS for unseen word
Evaluation	Algorithm 4	Compute accuracy and performance metrics

5. Results and Discussion

The framework is implemented on small subsets of the Assamese dataset to check how it performs with minimal training. The results are outlined in Table- IV.

Table-IV: Results

Metric	Performance	Performance	Performance	Performance
Train/Test Size Ratio	50/50	80/20	80/20	80/20
Size of Train Set (no. of words)	2100	3360	3360	6720
Size of Test Set (no. of words)	2100	840	840	1680
Total Unknown Words	950	258	258	995
Unknown Words Detected Wrong	745	193	178	524
Unknown Words Detected Correctly	205	65	80	471
Total Known Words	1150	582	582	685
Known Words Detected Wrong	447	170	144	101
Known Words Detected Correctly	703	412	438	584
Total Wrong Words	1193	363	321	624
Total Words Detected Correctly	907	477	519	1056
Unknown Accuracy	21.54%	25.18%	31.18%	47.37%
Known Accuracy	61.11%	70.72%	75.27%	85.29%
Overall Accuracy	43.21%	56.73%	61.73%	62.83%

The performance of the proposed framework of bigram and trigram character-based POS

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

taggers was found to be satisfactory. The experiments were conducted under four different train–test subsets of small datasets divided into training sets and testing sets with ratios of 50:50, 80:20, and 80:20, respectively. Each of the experiments evaluated the performance on both known and unknown words. Known words are the words that appear during training the model or during configuring parameters. Unknown words are the ones that are not encountered while training the model. The results indicate that the methods perform well for known words but struggle with unknown words, which is expected with limited resources in the case of low-resource languages. However, since character bigrams and trigrams of an unknown word may be parts of some known words used during training, their usage is helpful in determining the POS tag of a word. The framework shows promising results for character-level POS tagging in Assamese, even with limited labeled data. This can be integrated with other methods to obtain better results.

6. Conclusion

This paper analyzes bigram and trigram of characters in Assamese language. Since bigram and trigram characters are important features for word classification, their usage in Assamese language can be explored in future for many NLP applications. Bigram and trigrams also represent many hidden characteristics of a particular language. This paper proposes the idea of using character bigrams and trigrams to improve the effectiveness of Parts of Speech (POS) taggers. It can be used to address the limitations posed by limited annotated resources for low-resource languages like Assamese. The bigrams and trigrams of characters form the basic syntactic structure of a word. These structures are closely related to structures, meaning and other properties of a word. Therefore, proper use of them can enhance the efficacy of the NLP applications. In the future, the researchers can integrate other models with this framework for achieving better performance.

References:

1. Jurafsky, D. (2000). Speech & language processing. Pearson Education India.
2. Elghannam, F. (2021). Text representation and classification based on bi-gram alphabet. Journal of King Saud University-Computer and Information Sciences, 33(2), 235-242.
3. Foroozan Yazdani, S., Tan, Z., Kakavand, M., & Mustapha, A. (2022). NgramPOS: a bigram-based linguistic and statistical feature process model for unstructured text classification. Wireless Networks, 28(3), 1251-1261.
4. Wojcicki, P., & Zientarski, T. (2024). Polish word recognition based on n-gram methods. IEEE Access, 12, 49817-49825.
5. Alomari, D., & Ahmad, I. (2024). Exploring character trigrams for robust arabic text classification: a comparative analysis in the face of vocabulary expansion and misspelled words. IEEE Access, 12, 57103-57116.
6. Phukan, R., Baruah, N., Sarma, S. K., & Konwar, D. (2024). Exploring character-level deep learning models for pos tagging in assamese language. Procedia Computer Science, 235, 1467-1476.
7. Baishya, D., & Baruah, R. (2021). Highly efficient parts of speech tagging in low resource languages with improved hidden Markov model and deep learning. International Journal of Advanced Computer Science and Applications, 12(10).
8. D. Khurana, A. Koli, K. Khatter, and S. Singh, “Natural language processing: State of the art, current trends and challenges,” Multimedia tools and applications, vol. 82, no. 3, pp. 3713–3744, 2023.
9. S. R. Eddy, “What is a hidden markov model?” Nature biotechnology, vol. 22, no. 10, pp.

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

1315–1316, 2004.
10. R. Martinez, "Part-of-speech tagging," Wiley Interdisciplinary Reviews: Computational Statistics, vol. 4, no. 1, pp. 107–113, 2012.
11. D. W.OardandF.J.Och, "Rapid-response machine translation for unexpected languages," in Proceedings of Machine Translation Summit IX: Papers, 2003.
12. P. Resnik and N. A. Smith, "The web as a parallel corpus," Computational Linguistics, vol. 29, no. 3, pp. 349–380, 2003.
13. D. S. Munteanu and D. Marcu, "Improving machine translation performance by exploiting non-parallel corpora," Computational Linguistics, vol. 31, no. 4, pp. 477–504, 2005.
14. B. P. King, "Practical natural language processing for low-resource languages.," Ph.D. dissertation, Computer Science and Engineering in the University of Michigan, 2015.
15. Sarma, S. K., Deka, R., Baro, B., Deka, V., Deka, U., Rahman, M., ... & Kashyap, K. (2025). Challenges and Solutions in Developing Low-Resource Wordnets: Insights from Assamese and Bodo. GWC 2025, 249.
16. Gogoi, A., & Baruah, N. (2022). A lemmatizer for low-resource languages: Wsd and its role in the assamese language. Transactions on Asian and Low-Resource Language Information Processing, 21(4), 1-22.
17. Baishya, D., & Baruah, R. (2021, January). Improving Hidden Markov Model for very low resource languages: An analysis for Assamese parts of speech tagging. In 2021 11th International Conference on Cloud Computing, Data Science & Engineering (Confluence) (pp. 142-146). IEEE.
18. Baishya, D., & Baruah, R. (2024). Part-of-speech Tagging for Low-resource Languages: Activation Function for Deep Learning Network to Work with Minimal Training Data. ACM Transactions on Asian and Low-Resource Language Information Processing, 23(5), 1-31.

Received: August 09, 2025