A Higher-Order N-gram Model to enhance automatic Word Prediction for Assamese sentences containing ambiguous Words

M.P. Bhuyan, S.K. Sarma

Abstract: Word prediction is a technique which tries to suggest the users’ words after knowing the few input letters of the user. This predictive model also tries to generate the future words or next words of a sentence by observing earlier words of the sentence. In this research, two problems are combined, one is word prediction and the next is handling of ambiguous words. A word prediction model predicts the future words of a sentence by using n-gram based model. In general, predictive models use unigram, bigram or trigram models to predict the next words. In case of sentences consisting of ambiguous words, the predictive model by using only bigram or trigram cannot perform well to predict the next words. To enhance this prediction for ambiguous words, maximum of six previous input words are observed and try to predict almost the exact words after the ambiguous words in those particular contexts. Different level of experiments are done and the results are compared for modified or enhanced prediction model with the traditional prediction model, improvement on accuracy and failure rate are found in the enhanced model. The accuracy of the Traditional Model is 60.68% on the hand the accuracy of the Enhanced Model is 66.88%. The failure rate of the Traditional Model is 32.35% and the Enhanced Model is 29.17%.

Keywords: N-gram, ambiguous words, word prediction.

I. INTRODUCTION

Assamese is an official language of India which is used in the north-east part of the country; the language is spoken by nearly 30 million people. Assamese language belongs to the Indo-European language family [1]. There are large numbers of newspapers, magazines, books, etc. are available in Assamese. But, the language is not getting very much focus.

In [2], authors have used N-gram language models like unigram, bigram, trigram, deleted interpolation and backoff model to generate prediction for Bangla language. They have used these models and tested at different levels with a corpus of size 0.25 million words to predict the next word in Bangla, 100 sentences are tested with length varies from 5 to 20 and the average accuracies are found: 21.24%, 45.84%, 63.04%, 63.50%, 62.86% for unigram, bigram, trigram, backoff and deleted interpolation model respectively.

II. RELATED WORKS

In [3], authors have said that word prediction and completion are helping the people with disability of typing and also automatic word prediction can help the new learners, students etc. by providing the correct word to complete the sentence. The n-gram models like Unigram, Bigram, Trigram, Backoff and Linear interpolation models are compared for accuracy and failure rate. Linear interpolation model is found better than the all other models and another experiment is done to find the optimum prediction list length, which is obtained as 7 along with the accuracy of prediction.

M.P. Bhuyan*, S.K. Sarma. Department of Information Technology, Gauhati University, Guwahati, India.

* Correspondence Author

Retrieved Manuscript Received on August 28, 2019.

Published By: Blue Eyes Intelligence Engineering & Sciences Publication

DOI: 10.35940/ijeat.F8706.088619

F8706088619/2019©BEIESP

2921
A Higher-Order N-gram Model to enhance automatic Word Prediction for Assamese sentences containing ambiguous Words

In [4], authors have used an N-gram model to predict word for Assamese language using Unigram, Bigram, Trigram, Quadrigram language models. Keystrokes saving values are tested for these models for both in-domain data and users’ data, and the maximum keystrokes saving is found 74.04% and 48.28% for in-domain data and users’ data.

In [5], authors have used a hybrid word suggestion model based on Naive Bayes and Latent Semantic Analysis (LSA) and the accuracy achieved is 44.2%. Naive Bayes method uses conditional independent variables which requires less memory and computation power, however Naive Bayes becomes worst because of knowledge loss due to independent assumption; LSA uses techniques to analyze text and try to find relationship among the words at different levels like paragraph, phrase etc. to store the relationship, a relationship table is designed and which stores the frequency of a word $w_i$ appeared in the context of the word $w_j$. To control the size of relationship table Singular Value Decomposition (SVD) method is used.

In [6], authors have described a predictive form fill-up system depending on the past history of the user so that the amount of time required to fill the form is reduced, model tries to predict sequence of entries of a form not only the current filed entry.

In [7], authors have proposed an exponential interpolation model which combines a Part Of Speech (POS) based language model and an n-gram based language model. They have found the improvement in keystroke saving, perplexity and also the hit rate over traditional n-gram and interpolation model. They have tested their model for Spanish, Portuguese and English. In all the three languages their proposed model has shown significant level of improvement in hit rate, keystrokes saving and perplexity.

In [8], authors have described the problem of large sized language models for on-device keyboards. These large sized language models would cause an overhead for the system to perform the computations. They have proposed on-device neural language model by optimizing the run time memory. Their model size is 7.40MB and the average prediction time is 6.47ms. They claimed that their proposed model was able to perform the computations. They have proposed on

In [9], authors have designed a tool called LuitPad to write Assamese efficiently. They have used two methods to complete a word; first one is by predicting the next character after knowing the previous characters and the second one is a unigram method which generates the current words using the knowledge of previous words. But their method is not able to produce the future words (next words) for the users.

In [10], authors have described an error detection and correction model for Assamese words using n-gram models, like unigram, bigram, and trigrams. They have designed a score based system for each character or letter of Assamese alphabets. In their model they have focused on the point of error in Assamese words and generate the relevant character(s) as suggestion. Their model is basically for the learners’ Assamese language and students of Assamese medium schools.

From the above discussion it is seen that the word prediction model are useful in any language and workings are going on not only the English language but also the local languages like Bangla and other languages, so it will be very much helpful if such types of accurate predictive models are designed for Assamese language. In addition, this research work will also enhance the work done by Bhuyan and Sarma in [4] which is the only work in Assamese to predict words while writing, and also, able to bring a new direction in the field of Assamese writing software or tools.

III. WORD PREDICTION MODEL

N-gram based models are used to predict the next word in a sentence, generally this N-gram model observe the previous all words and suggest the next word; for example: “*There is no student in the ____.*” here in this example the blank space can be filled up by the word ‘class’ or ‘college’ or ‘school’, etc. This process of filling the unknown word by human is a very common task, a human can fill such blank position by observing the previous words of the sentence and the same process of filling the blank space can be taught to machine but a machine does not have infinite memory, so the machine cannot remember all the previous words, so fraction of the previous words are used to predict the next word, for that, N-gram model is reduced to Unigram, Bigram, Trigram model etc. In addition, N-gram model is a probabilistic model which primarily depends on Hidden Markov Model (HMM).

In this research N-gram models like Unigram, Bigram, and Trigram are used to predict the next word, Bigram model is known as the 1st order Markov model and Trigram model is known as the 2nd order Markov model. Moreover, in this research to avoid the words having zero final probability, a smoothing technique called katz’s backoff model is used which helps to overcome the zero final probability of a sentence. In general, a smoothing technique plays a vital role in such predictive models.

Let us consider the following sentence and understand how the predictive model works:

sentence, = মই এজ লা তাৱা উত্তৰ ফানল যৰা (Moi Ejon Lora Uttar FalE JooAa Dekhiso meaning is I have seen a boy going towards North)

In the above example the word উত্তৰ (uttar) is ambiguous which means North in this context. Following process explains how the probabilities are calculated in the traditional word-prediction model.

**Unigram model:**

$$p(\text{sentence}_i) = p(মই) \times p(এজ) \times p(লা) \times p(তাৱা) \times p(উত্তৰ) \times p(ফানল) \times p(যৰা)$$

**Bigram model:**

$$p(\text{sentence}_i) = p(মই) \times p(এজ \mid মই) \times p(লা \mid এজ) \times p(তাৱা \mid লা) \times p(উত্তৰ \mid তাৱা) \times p(ফানল \mid উত্তৰ) \times p(যৰা \mid ফানল) \times p(দেখিয়া) \times p(যৰা)$$

**Trigram model:**

$$p(\text{sentence}_i) = p(মই) \times p(এজ \mid মই) \times p(লা \mid এজ) \times p(তাৱা \mid লা) \times p(উত্তৰ \mid তাৱা) \times p(ফানল \mid উত্তৰ) \times p(যৰা \mid ফানল) \times p(দেখিয়া \mid ফানল) \times p(যৰা)$$
In the modified or enhanced word prediction model if the system encounters an ambiguous word then from that point the model will look back maximum five previous contexts and using these five previous contexts words the system tries to generate the next suggestion after the ambiguous word. On the other hand, if the five previous words fail to produce the next word then the system reduces the context and uses four previous contexts words to generate the next word and so on. The probability evaluation process for the enhanced model is given below:

Trigram model in the enhance version:
\[ p(\text{sentence}) = p(\text{মই}) \times p(\text{এজ}) \times p(\text{উত্তর | মই এজ মই}) \] x \[ p(\text{ফানল | এজ মই মই}) \] x \[ p(\text{বাওল | এজ মই মই}) \] x \[ p(\text{সকল | এজ মই মই}) \]

In the above calculation, to predict the next word after the ambiguous word উত্তর (attar) all the previous context (quadrigram contexts) is searched and the next word ফানল (FauLe) is predicted with a higher probability in comparison with the earlier traditional word-prediction model. If this enhanced model fails then the prediction will go back to the traditional model by reducing the contexts. Actually, the enhancer is on the top of the traditional model to handle such type of ambiguous words. Following Algorithm-1 and Algorithm-2 describes the predictive model and enhanced module.

Algorithm-1: Predictive Model

Step-1:
1.1 \( w_a = '' \) #empty string
1.2 Input a letter ‘L’ in \( w_i = w_a + L \).
1.3 Generate six most frequent words starting with ‘L’ in a suggestion list \( SL \) using \( \text{unigram_db} \).
1.4 If the users’ word appear in the \( SL \) then select the word \( w_i \) from \( SL \) list and \( w_i = w_a \) goto Step-2.
1.5 If the users’ word does not appear in the suggestion list goto Step-1.1 and input more letters or go to Step-2 skipping the suggestion part.

Step-2:
2.1 Using \( w_a \) obtained in Step-1 and the \( \text{bigram_db} \) the predictive model tries to generate the next word \( w_b \), display the suggestion list \( SL \) with six most frequent words in the context of \( w_a \).
2.2 If the users’ word appear in the \( SL \) list then select the word \( w_i \) from \( SL \) list and \( w_i = w_a \) goto Step-3.
2.3 If the users’ word does not appear in the suggestion list or the \( SL \) is empty then goto Step-1.1 and input more letters to form \( w_b \) after getting \( w_i \) goto Step-3.

Step-3:
3.1 Using \( w_a \) and \( w_b \) obtained in Step-1, Step-2 and the \( \text{quadrigram_db} \) the predictive model tries to generate the next word \( w_i \) display the suggestion list \( SL \) with six most frequent words in the context of \( w_a \) and \( w_b \).
3.2 If the users’ word appear in the \( SL \) list then select the word \( w_i \) from \( SL \) list and \( w_i = w_a \) goto Step-4.
3.3 If the users’ word does not appear in the suggestion list or the \( SL \) is empty then goto Step-1.1 and input more letters to form \( w_i \) after getting \( w_i \) goto Step-4.

Step-4:
Repeat Step-1 to Step-3 to form the sentence.

Following algorithm describes how the enhance model works along with the predictive model.

Algorithm-2: Enhancing procedure for the Ambiguous words

Step-1:
1.1 An ambiguous word \( w_i = W_{ambg} \) is detected while constructing a sentence.
1.2 Use \( <w_{1:5}, w_{1:4}, w_{1:3}, w_{1:2}, w_{1:1}, W_{ambg}> \) maximum six previous context by using \( \text{6gram_db} \) to get accurate next word(s) \( w_{i+1} \) after \( w_i = W_{ambg} \) in the \( SL \) list. If \( SL \) is empty then goto Step-1.3
1.3 Use \( <w_{1:4}, w_{1:3}, w_{1:2}, w_{1:1}, W_{ambg}> \) maximum five previous context by using \( \text{5gram_db} \) to get accurate next word(s) \( w_{i+1} \) after \( w_i = W_{ambg} \) in the \( SL \) list. If \( SL \) is empty then goto Step-1.4
1.4 Use \( <w_{1:3}, w_{1:2}, w_{1:1}, W_{ambg}> \) maximum four previous context by using \( \text{quadrigram_db} \) to get accurate next word(s) \( w_{i+1} \) after \( w_i = W_{ambg} \) in the \( SL \) list. If \( SL \) is empty then goto Step-1.5
1.5 Use \( <w_{1:2}, w_{1:1}, W_{ambg}> \) maximum three previous context by using \( \text{trigram_db} \) to get accurate next word(s) \( w_{i+1} \) after \( w_i = W_{ambg} \) in the \( SL \) list. If \( SL \) is empty then goto Step-1.6
1.6 Use \( <w_{1:1}, W_{ambg} \) maximum two previous context by using \( \text{bigram_db} \) to get accurate next word(s) \( w_{i+1} \) after \( w_i = W_{ambg} \) in the \( SL \) list. If \( SL \) is empty then goto Step-2.

Step-2: If unable to find the word \( w_{i+1} \) after \( w_i = W_{ambg} \) then goto Step-1 of Algorithm-1.
A Higher-Order N-gram Model to enhance automatic Word Prediction for Assamese sentences containing ambiguous Words

For the experimental purpose, a mixed set of corpus is used which contains sentences of ambiguous words as well as the other common sentences. The system is trained with a corpus of size nearly 250,000 words. If the prediction model is tested with new sentences the predictive models tries to predict 5 relevant words as the next word considering the previous context using bigram and trigram for unambiguous words and for ambiguous words quadrigram, 5-gram and 6-gram are also used as enhancement module.

IV. RESULT AND DISCUSSION

In the following Table-I a portion of the list of Assamese ambiguous words and their meanings are given which are used in the predictive model as part of the corpus data.

<table>
<thead>
<tr>
<th>Words</th>
<th>Ambiguous words</th>
<th>Meaning-1</th>
<th>Meaning-2</th>
</tr>
</thead>
<tbody>
<tr>
<td>W01</td>
<td>উত্তর (uttar)</td>
<td>answer</td>
<td>north</td>
</tr>
<tr>
<td>W02</td>
<td>আদি (aadi)</td>
<td>starting</td>
<td>etc.</td>
</tr>
<tr>
<td>W03</td>
<td>কাপ (kol)</td>
<td>banana</td>
<td>factory</td>
</tr>
<tr>
<td>W04</td>
<td>ঘর (ghar)</td>
<td>house</td>
<td>quadrangle</td>
</tr>
<tr>
<td>W05</td>
<td>আঙ্ক (ongko)</td>
<td>numeric values, maths</td>
<td>an act of drama</td>
</tr>
<tr>
<td>W06</td>
<td>ঘর (ortho)</td>
<td>meaning</td>
<td>money</td>
</tr>
<tr>
<td>W07</td>
<td>আলাপ (asol)</td>
<td>the root form</td>
<td>the end portion of a wearing cloth</td>
</tr>
<tr>
<td>W08</td>
<td>আচার (aachar)</td>
<td>behavior</td>
<td>pickle</td>
</tr>
<tr>
<td>W09</td>
<td>কালা (kola)</td>
<td>cultural related</td>
<td>deaf</td>
</tr>
<tr>
<td>W10</td>
<td>কাঠি (kaati)</td>
<td>seventh month of Assamese calender</td>
<td>inclined</td>
</tr>
<tr>
<td>W11</td>
<td>পদ (pod)</td>
<td>designation</td>
<td>the act of singing vocal music</td>
</tr>
<tr>
<td>W12</td>
<td>মালিক (maalik)</td>
<td>God</td>
<td>owner</td>
</tr>
<tr>
<td>W13</td>
<td>ফল (fol)</td>
<td>result</td>
<td>fruit</td>
</tr>
<tr>
<td>W14</td>
<td>গৌণ (goon)</td>
<td>characteristics</td>
<td>multiple</td>
</tr>
<tr>
<td>W15</td>
<td>মুখ (moor)</td>
<td>head</td>
<td>boundary</td>
</tr>
<tr>
<td>W16</td>
<td>কাপ (cup)</td>
<td>trophy</td>
<td>cup</td>
</tr>
<tr>
<td>W17</td>
<td>কালি (kaani)</td>
<td>opium made for smoking</td>
<td>cloths</td>
</tr>
<tr>
<td>W18</td>
<td>কালি (kobi)</td>
<td>poet</td>
<td>A vegetable</td>
</tr>
</tbody>
</table>

The probability values and the improvement of the sentence probability values in number of times are shown in the above Table-II. To visualize the outcome and the performance more clearly, the improvement result obtained in the experiment are plotted graphically as shown in the above Fig.1. For the sentences: S01, S03, S11, S13, and S18 the improvement is relatively higher than the other sentences. The sentence S13 is showing the highest improvement among the all sentences with the ambiguous word 'ফল' (Fol means Fruit) and the present trained corpus. The sentences: S02, S04, S07, S08, S09, S12, S14, S15 the enhanced model is showing little moderate result and finally for few ambiguous words for the sentences: S05, S06, S10, S16, S17 the enhanced-prediction-model is unable to improve the performance, there are two reason for this non-performance, either the word is not found in the corpus or the position of the word in the sentence (i.e. the context for the word is not more than the trigram). In such situation the enhanced-prediction-model will fall to the trigram model, recalling the concept, this enhanced-prediction-model is working on the top of the traditional trigram model so once
the enhanced-prediction-model fails the predictive system will work like the traditional trigram model.

B. Performance Measure-2

The performance of the predictive model can be evaluated using accuracy and failure rate of prediction.

\[ \text{Accuracy} = \frac{(L+1)-i}{L} \times 100\% \quad (1) \]

Where \( L \) is the length of the prediction list and \( i \) is the position where the intended word is appeared.

\[ \text{FailureRate} = \frac{F}{P} \times 100\% \quad (2) \]

Where \( F \) is the number of times system fails to predict the intended word out of \( P \) predictions [3]. The same 18 sentences are tested in this experiment and the results are shown in the Table -III.

Table-III: Accuracy and Failure rate of the prediction models

<table>
<thead>
<tr>
<th>Sentence</th>
<th>Ambigous words</th>
<th>Accuracy</th>
<th>FailureRate</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Traditional Model</td>
<td>Enhanced Model</td>
</tr>
<tr>
<td>S01</td>
<td>উত্তর</td>
<td>57.16</td>
<td>72.34</td>
</tr>
<tr>
<td>S02</td>
<td>আদি</td>
<td>62.41</td>
<td>65.23</td>
</tr>
<tr>
<td>S03</td>
<td>কল</td>
<td>54.23</td>
<td>64.38</td>
</tr>
<tr>
<td>S04</td>
<td>ঘৰ</td>
<td>56.71</td>
<td>61.28</td>
</tr>
<tr>
<td>S05</td>
<td>অংক</td>
<td>62.35</td>
<td>62.35</td>
</tr>
<tr>
<td>S06</td>
<td>অংক</td>
<td>73.14</td>
<td>73.14</td>
</tr>
<tr>
<td>S07</td>
<td>আচল</td>
<td>61.41</td>
<td>68.25</td>
</tr>
<tr>
<td>S08</td>
<td>আচল</td>
<td>62.17</td>
<td>66.74</td>
</tr>
<tr>
<td>S09</td>
<td>কলা</td>
<td>59.31</td>
<td>63.49</td>
</tr>
<tr>
<td>S10</td>
<td>কালি</td>
<td>58.71</td>
<td>58.71</td>
</tr>
<tr>
<td>S11</td>
<td>পাড</td>
<td>54.57</td>
<td>73.22</td>
</tr>
<tr>
<td>S12</td>
<td>মাজিক</td>
<td>58.88</td>
<td>64.45</td>
</tr>
<tr>
<td>S13</td>
<td>ফল</td>
<td>61.12</td>
<td>75.41</td>
</tr>
<tr>
<td>S14</td>
<td>শুধু</td>
<td>61.14</td>
<td>65.24</td>
</tr>
<tr>
<td>S15</td>
<td>সদ্ধ</td>
<td>58.41</td>
<td>62.39</td>
</tr>
<tr>
<td>S16</td>
<td>কৃষ</td>
<td>72.48</td>
<td>72.48</td>
</tr>
</tbody>
</table>

From the above experiment, it is seen that the enhanced model is performing better than the traditional model for both accuracy and the failure rate. For the sentences: S05, S06, S10, S16, S17 both the traditional and enhanced models are showing similar accuracy and failure rate, in reality, for these sentences prediction system is unable to enhance the accuracy and failure rate, this is due not having sufficient context or the ambiguous words may not present in the corpus. From the Fig.2 it can be clearly visualized that the accuracy of the enhanced model is higher than or equal to the traditional model and from Fig.3 it can be easily identified that failure rate of the enhanced model is lower than the traditional model.
V. CONCLUSION AND FUTURE WORK

Word prediction model will help the people who type in Assamese by increasing the productivity of the users. Moreover, the model will also help the differently-abled persons. From the above experiments it is clearer that the enhanced model is able to improve the probability of prediction, accuracy and the failure rate in most of the cases and also this enhanced-prediction-model is an additive to the original model so it will not degrade the performance if it fails to improve the performance. Thus, the Enhanced Model will definitely purify the prediction result for the ambiguous words.

In the future, the same ambiguous words can be tested by placing the word at different positions in the sentences where they can be placed without losing the meaning of the sentence, this placing can be done in Assamese because of its free word order property and the performance of the system can be evaluated again. For all the ambiguous words more number of sentences can be included in the corpus. For both the two models time required to generate the prediction for relatively longer sentences can be evaluated. Lastly, to measure the real-time performance of the system computer typist of Assamese language shall be asked to test the present predictive model.

REFERENCES


AUTHORS PROFILE

M.P. Bhuyan is a research scholar in the department of Information Technology, Gauhati University, Guwahati, India. He did BE in Computer Science & Engineering and MTech in Information Technology. His research areas are Artificial Intelligence, Natural Language Processing, and Machine Learning. Email: mpratim250@gmail.com

S.K. Sarma is currently working as a professor in the department of Information Technology, Gauhati University, Guwahati, India. He is a computer scientist and published various research papers in renowned conferences and journals. His research areas are Natural Language Processing, Artificial Intelligence, Machine Learning, Computer Networks, etc. He has guided many PhD students.