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 focuses in the digital world, i.e. the resources of Assamese language are not easily available to process electronically. Importance of the word prediction model is to increase the productivity or throughput of a user who has typing speed below the average speed so that the user should not annoyed with the computer-system to type his text.

Word Sense Disambiguation (WSD) is a technique to resolve the lexical level of ambiguity of a word i.e. the word which has multiple meaning. It is found that every language has some words which have multiple meanings, because of having multiple meanings the traditional predictive model with low context memory will predict some unrelated words with higher probability and the actual word may not be predicted at the first time. This research aims to find the exact next words after the ambiguous word with high probability. Let us consider an Assamese word ‘আদি(aadi)’

Sentence1 = (তঁরে কথাটি কোনো আদি অজ্ঞ নাই | (Teor Kothar Konu Aadi Onto Nai ), In English the sentence is: His words do not have any beginning and ending here the ambiguous word আদি(aadi) is used as starting.

Sentence2 = লব্ধা আদি (গৌরাঙ্গিতি আদিনেই হল ইয়াস উত্তরাধ্য | (Lora Aaru Sowali aadiEi hol Iyaar UdaHoron), In English the sentence is: Boys and girls etc. are the example of this. Here the ambiguous word আদি(aadi) is used as etc.

Assamese is a morphologically rich language and free word order, so, it is very difficult to process the language. There are many critical grammar rules in the language that are very difficult to follow while writing. That is why it is necessary to have a prediction model in Assamese to assist the users.

The rest of the paper is organized as follows: Section 2 describes the related works, Section 3 describes the word prediction model, Section 4 analyzes and discusses the experimental result and Section 5 concludes the work and highlight the future direction of this research.

II. RELATED WORKS

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.

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.

Revised Manuscript Received on August 28, 2019.

* Correspondence Author
M.P. Bhuyan*, Department of Information Technology, Gauhati University, Guwahati, India.
S.K. Sarma, Department of Information Technology, Gauhati University, Guwahati, India.
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, appeared in the context of the word w. 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 show improvement in comparison with the existing methods in terms of keystrokes savings and word prediction rate.

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 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:

\[
\text{sentence}_{i} = \text{ঝি এজল লৰা উত্তর ফানলে যোৰা দৰিদ্রা। (Moi Ejon Lora Uttar FalE JooAa Dekhiso meaning is I have seen a boy going towards North)}
\]

In the above example the word ’উত্তর’ (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:**

\[
p(\text{sentence}_{i}) = p(আজ) \times p(এজ) \times p(লরা) \times p(উত্তর) \times p(ফানলে) \times p(যোৰা) \times p(দৰিদ্রা)
\]

**Bigram model:**

\[
p(\text{sentence}_{i}) = p(আজ | এজ) \times p(এজ | লরা) \times p(লরা | উত্তর) \times p(উত্তর | ফানলে) \times p(ফানলে | যোৰা) \times p(যোৰা | দৰিদ্রা)
\]

**Trigram model:**

\[
p(\text{sentence}_{i}) = p(আজ | এজ | লরা) \times p(এজ | লরা | উত্তর) \times p(লরা | উত্তর | ফানলে) \times p(ফানলে | যোৰা | দৰিদ্রা)
\]
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:
\[
p(\text{sentence}) = p(\text{মই}) \cdot p(\text{েজ} | \text{মই}) \cdot p(\text{এজ} | \text{েজ} \text{মই}) \cdot p(\text{এজ} | \text{েজ} \text{মই}) \cdot p(\text{এজ} | \text{েজ} \text{মই})
\]

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 ফালে (FaaLe) 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 \( w_u = " " \) #empty string
1.2 Input a letter ‘L’ in \( w_u = w_u + L \).
1.3 Generate six most frequent words starting with ‘L’ in a suggestion list SL using unigram_db.
1.4 If the users’ word appear in the SL list then select the word \( w_i \) from SL list and \( w_u = w_i \) 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 \( w_u \) obtained in Step-1 and the 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 \( w_u \).
2.2 If the users’ word appear in the SL list then select the word \( w_i \) from SL list and \( w_u = w_i \) 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 \( w_b \) goto Step-3.

**Step-3:**
3.1 Using \( w_u \) and \( w_b \) obtained in Step-1, Step-2 and the trigram_db the predictive model tries to generate the next word \( w_u \) display the suggestion list SL with six most frequent words in the context of \( w_u \) and \( w_b \).
3.2 If the users’ word appear in the SL list then select the word \( w_i \) from SL list and \( w_u = w_i \) 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 \( <w_{i-5}, w_{i-4}, w_{i-3}, w_{i-2}, w_{i-1}, W_{ambg}> \) maximum six previous context by using 6_gram_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 \( <w_{i-4}, w_{i-3}, W_{i-2}, W_{i-1}, W_{ambg}> \) maximum five previous context by using 5_gram_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 \( <w_{i-3}, W_{i-2}, w_{i-1}, W_{ambg}> \) maximum four previous context by using 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 \( <w_{i-2}, w_{i-1}, W_{ambg}> \) maximum three previous context by using 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 \( <w_{i-1}, W_{ambg}> \) maximum two previous context by using 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.
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.

| Words | Ambiguous words | Meaning-1 | Meaning-2 |
|-------|-----------------|-----------|-----------|
| W01   | उत्तर (uttar)   | answer    | north     |
| W02   | आदि (aadi)     | starting  | etc.      |
| W03   | कल (kol)       | banana    | factory   |
| W04   | गधर (ghar)     | house     | quadrangle|
| W05   | आंगको (ongko)  | numeric values, maths | an act of drama |
| W06   | अर्थ (ortho)    | meaning   | money     |
| W07   | आँग (asol)     | the root form | the end portion of a wearing cloth |
| W08   | आचार (aachar) | behavior  | pickle    |
| W09   | कोला (kola)    | cultural related | deaf |
| W10   | काद (kaati)    | seventh month of Assamese calender | inclined |
| W11   | गो (pod)        | designation | the act of singing vocal music |
| W12   | मालिक (malik)  | God       | owner     |
| W13   | फल (fol)       | result    | fruit     |
| W14   | गोन (goon)     | characteristics | multiple |
| W15   | मूर (moor)      | head      | boundary  |
| W16   | कप (cup)       | trophy    | cup       |
| W17   | कलिक (kaani)  | opium made for smoking | cloths |
| W18   | कबि (kobi)     | poet      | A vegetable|

A. Performance Measure-1

18 sentences are tested for different ambiguous words which are shown in Table-I. From the table it is clearly visible that the Enhanced Model is able to increase the sentence probability. On the other hand, the Traditional Model is treating these ambiguous words as the other common words.

| Senteces | Ambiguous words | Sentence Probability (Traditional Model) | Sentence Probability (Enhanced Model) | Improvement (no. of times) |
|----------|----------------|------------------------------------------|---------------------------------------|---------------------------|
| S01      | उत्तर           | 4.243E-5                                | 2.354E-4                              | 5.55                      |
| S02      | आदि            | 5.241E-5                                | 7.533E-5                              | 1.44                      |
| S03      | कल             | 1.342E-6                                | 4.284E-6                              | 3.19                      |
| S04      | गधर            | 3.021E-5                                | 6.124E-5                              | 2.03                      |
| S05      | आंगको          | 4.102E-6                                | 4.102E-6                              | 1.00                      |
| S06      | अर्थ             | 5.231E-7                                | 5.231E-7                              | 1.00                      |
| S07      | आचार           | 3.248E-5                                | 7.102E-5                              | 2.19                      |
| S08      | आदि            | 2.701E-6                                | 6.705E-6                              | 2.48                      |
| S09      | कल             | 8.745E-6                                | 1.637E-5                              | 1.87                      |
| S10      | काद             | 2.341E-5                                | 2.341E-5                              | 1.00                      |
| S11      | गो              | 5.241E-5                                | 2.143E-4                              | 4.09                      |
| S12      | मालिक          | 6.208E-6                                | 7.214E-6                              | 1.16                      |
| S13      | फल             | 4.219E-5                                | 2.874E-4                              | 6.81                      |
| S14      | गोन            | 7.125E-5                                | 8.245E-5                              | 1.16                      |
| S15      | मूर             | 4.433E-6                                | 6.714E-6                              | 1.51                      |
| S16      | कलिक          | 7.112E-6                                | 7.112E-2                              | 1.00                      |
| S17      | कबि            | 2.401E-7                                | 2.401E-7                              | 1.00                      |
| S18      | कबि            | 7.651E-6                                | 3.214E-5                              | 4.20                      |

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 ‘फल’ (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.

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

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**

| Sentences | Ambiguous words | Accuracy | Failure Rate |
|-----------|-----------------|----------|--------------|
|           |                 | Traditional Model | Enhanced Model | Traditional Model | Enhanced Model |
| S01       | উত্তর           | 57.16     | 72.34        | 32.36           | 22.41          |
| S02       | আদি            | 62.41     | 65.23        | 30.12           | 30.12          |
| S03       | কল             | 54.23     | 64.38        | 35.22           | 31.47          |
| S04       | ঘর             | 56.71     | 61.28        | 32.54           | 28.74          |
| S05       | অংক            | 62.35     | 62.35        | 31.25           | 31.25          |
| S06       | অর্জন          | 73.14     | 73.14        | 27.35           | 27.35          |
| S07       | আচল           | 61.41     | 68.25        | 32.39           | 29.14          |
| S08       | আচার          | 62.17     | 66.74        | 34.56           | 28.34          |
| S09       | কলা           | 59.31     | 63.49        | 30.14           | 31.55          |
| S10       | কাজি           | 58.71     | 58.71        | 36.54           | 36.54          |
| S11       | পথ             | 54.57     | 73.22        | 35.14           | 24.75          |
| S12       | মঙ্গলিক        | 58.88     | 64.45        | 31.24           | 31.24          |
| S13       | ফল             | 61.12     | 75.41        | 35.21           | 21.14          |
| S14       | শুধু           | 61.14     | 65.24        | 31.25           | 30.74          |
| S15       | মূখ                      | 58.41     | 62.39        | 33.76           | 30.81          |
| S16       | কাণ                      | 72.48     | 72.48        | 28.36           | 28.36          |

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 cleared 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

1. G C Goswami. 1982. “Structure of Assamese”, Gauhati University, Assam Guwahati, Assam.
2. M. Haque, M. T. Habib, and M. M. Rahman, “Automated word prediction in Bangla language using stochastic language model,” International Journal in foundation Computer Science & Technology (IJFCAST), vol. 5, Nov. 2015.
3. Md. Tarek Habib , Al-Mamun Abdullah, Rahman Md. Sadekur, Shah Md. Tanvir Siddique and Ahmed Farruk, “An Exploratory Approach to Find a Novel Metric Based Optimum Language Model for Automatic Bangla Word Prediction,” International Journal of Intelligent Systems and Applications (IJISA). Vol. 10, pp 47-54, February 2018.
4. M. P. Bhuyan & S.K. Sarma, “An N-gram based model for predicting of word-formation in Assamese language,” Journal of Information and Optimization Sciences, 40:2, 427-440, DOI: 10.1080/01286203.2019.1580883.
5. H.X. Goulart, M.D. Tosi , D.S. Gonçalves, R.F. Maia, & G.A. Wachs-Lopes, “Hybrid Model For Word Prediction Using Naive Bayes and Latent Information,” CoRR, abs/1803.00985.
6. L. Troiano, C. Birtolo, & R. Armenise, Soft Comput (2017) 21: 1583. https://doi.org/10.1007/s00500-015-1870-7.
7. D. C. Cavalieri, S. E. Palazuelos-Cagigas, T. F. Bastos-Filho, and M. Sarcinelli-Filho, “Combination of language models for word prediction: an exponential approach,” IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 24, no. 9, pp. 1481–1494, 2016.
8. Yu Seunghak, Nilesk Kulkarni, Haejun Lee, and Ji-hie Kim, “On-device neural language model based word prediction,” in COLING 2018, The 27th International Conference on Computational Linguistics: System Demonstrations, Santa Fe, New Mexico, August 20-26, 2018, Dongyan Zhao, Ed. 2018, pp. 128–131, Association for Computational Linguistics.
9. N. Saharia, K. M. Konwar, “LuitPad: a fully unicode compatible Assamese writing software,” in Proceedings of the 2nd Workshop on Advances in Text Input Methods (WTIM 2), Mumbai, India, 2012, pp. 79-88.
10. M P Bhuyan and S K Sarma, “A statistical model for automatic Error Detection and Correction of Assamese Words”, International Journal of Recent Technology and Engineering (IJRTE), vol.8, issue: 2, July 2019, pp: 6111-6116.