Vol. 7, No. 2 | July – Dec 2023

Prediction of Next Word in Balochi Language Using N-gram Model

Sharan Bashir[1]*, Sohail A. Sattar[1], Muhammad Umer Farooq[1], Saad Ahmed[2], Mustafa Latif[3]

Abstract:

Balochi Language is among the oldest languages, spoken by approximately 10 million people worldwide. The Balochi language has been spoken for a very long period. In comparison to other languages like English, Urdu, French etc. it has a research gap in Natural language processing (NLP). The next word prediction system is one of the techniques of NLP for suggesting standardization and corpus collection. This research aims to provide a next word prediction system and a corpus with no ambiguity for the Balochi language. N-gram model for the next word prediction has been utilized, i.e. Unigram, Bigram, Trigram, Quad-gram and so on. A trained model has been embedded in an application after being evaluated extrinsically and intrinsically. It plays a crucial role in typing through a keyboard and helps users to type faster. Additionally, it helps native users to have fewer typing errors in less time. The results of the research show that Five-gram model has the highest performance of 93% while Quad-gram model has 80% and Trigram model has 76% respectively.

Keywords: *NLP; N-gram Model; Word Prediction; Intrinsic evaluation; extrinsic evaluation; Laplace smoothing; Lidstone smoothing*.

1. Introduction

Balochi is among the oldest living languages of west Iranian languages; approximately 10 million people speak it as their first or second language in Pakistan, Iran, Afghanistan, Turkmenistan, and Gulf countries [1]. The language has three dialects Eastern, Western, and Southern. Its script is Latin (Roman), written from left to right, and Perso-Arabic, from right to left. The language has short and long vowels, both with consonants. Arabic script is preferable in writing, as most books and magazines are published in Arabic script. Nevertheless, there is hardly any work done to digitize the Balochi language. In recent times, writing has shifted to digital devices. Many tools for language intelligence have been made possible by natural language processing, including language translators, semantic analysts, spell checkers, word predictor etc.

Next word prediction is the primary feature of language in NLP. It facilitates swift and

[1]Department of Computer Science and information Technology, NED University of Engineering and Technology, Karachi, Pakistan
[2]Department of Computer Science, Iqra University Karachi
[3]Department of Software Engineering, NED University of Engineering and Technology, Karachi, Pakistan
Sharan Bashir: [sharanbaloch2018@gmail.com]

error-free typing. It suggests the most probable next word for the currently typed word. Moreover, it can be used for multiple devices like smartphones, tablets or laptops. On next word prediction, a lot of research have been conducted in different languages like Urdu, Sindhi, Hindi, Bangla, Hebrew, Kurdish, and Assamese. Unlike other languages, no significant research has been doneon Balochi language by using next word prediction system. Therefore, there is a need for research in Balochi language to implement the next word prediction to help users to type more frequently and accurately in this Language.

N-gram Language Model is a statistical language model broadly used to predict the next word. In this paper, N-gram Model is utilized for next word prediction in the Balochi language with the addition of Laplace smoothing to avoid data sparseness. N-gram aims to predict the next word according to its previous N-1 words. It relies on the combination of the words like Bigram two words pair, Trigram three words together, Quad-grams four words together and so on. The model was trained on the Balochi corpus according to its number of N-grams. Moreover, the corpus is based on 300,000 words approximately of western dialect taken from a Balochi novel.

The paper consists of the following sections: Section 2 deals with the literature review, the Balochi Language is discussed in Section 3 and methodology of the proposed model is mentioned in Section 4. The result and discussion are illustrated in Section 5. Finally, the conclusion and future work is drawn in Section 6.

2. Literature Review

The Word prediction system has been an essential practice in augmentative communication for over 20 years [2]. Early word prediction systems, introduced in the 1980s, were used as assistance for those who had difficulties in learning [3]. For the last few years predictive techniques on the previous word prediction have become the need of any language. Social media has a significant role in daily life, so each language has to cope with its requirements. Unlike the Balochi language, most work has been done for languages like Urdu, Sindhi, Bangla, Hindi, Kurdish etc.

S. Shahzadi et al. [5], 2013, have worked on the Urdu language keyboard for android mobile phones to have a word prediction function, which works on less memory with less processing speed by using bigram. M. Hassan et al. [4] 2018 also implemented a system for the Urdu language to predict the next word with reference to a current word using the Hidden Markov model. J. Mahar and G. Memon [6] have used N-gram for Sindhi Language next word prediction in a sentence by using its previous history. The Add-One smoothing method is also utilized to assign non-zero probabilities to those N-grams with zero possibilities for increasing N-grams' performance.

In a paper for the Bangla language, M. Haque et al. [7] applied unigram, bigram, Trigram, deleted Interpolation, and back-off models to complete the sentence automatically by having a word prediction system. However, the back-off model's accuracy is higher than other models, which is 63.50, unigram at 21.24%, bigram at 45.84%, Trigram at 63.04%, and deleted Interpolation at 62.86%. Habib et al. [8] have worked on the Bangla language using unigram, bigram, Trigram, back-off and linear Interpolation. The accuracy level of Trigram, linear Interpolation and back-off is the same; however, in accuracy and failure together, linear Interpolation is the most accurate in the word prediction process, is 77%. Mittra et al. [9] have applied the N-gram term frequency matrix to the total count of each stored term in the Bangla language. They also measured the semantic similarity of sentences after word prediction with the help of Word2Vector, and for the most probable word suggestion, they used the Stupid Back off model, which worked well for large N-grams.

Shah and Kshetra [10] have introduced Sn-Grams, i.e. "Syntactic N-grams" that follow the grammar while making a prediction. Likewise, two deep learning techniques have been utilized to predict the Hindi Language's [11] next word: Long Short Term Memory (LSTM) and Bi-LSTM. At the same time, their

accuracy rate is measured as 59.46 % and 81.07 %, respectively. Hamarashid et al. [12] implemented the N-gram model with the addition of the Stupid Back-off (SBO) algorithm to create the word prediction system that is based on two dialects of Kurdish, Kurmanji and Sorani. The system could not find the proper result to predict the next word each time. It decreases N-gram size, for example, from Trigram to bigram.

It is claimed that the corpus size for training purposes can acquire the accuracy of word prediction [13]. However, accuracy can also depend on various other factors like the methodology of prediction, speed of prediction, dictionary structure, user interface and several words suggested to the user [14].

In 2008 research was conducted by M. Herold et al. [15] to examine whether using next word prediction can improve typing speed and spelling accuracy. For the experiment, 80 students from grades 4-6 were selected who had difficulty with spelling. The students entered 30 words via an on-screen keyboard with the next word prediction ability and, without it, software. Surprisingly, an increase in spelling accuracy had arisen with the use of software with next-word prediction.

In 2020, Nandini et al. [16] discussed a word prediction system for the Kannada language by using Naïve Bayes. However, for model optimization, they implemented stochastic gradient descent. Ali Pourmohammad et al. [17] proposed Hidden Markov Models (hmms) for next word prediction in the Azeri language based on Natural language processing techniques.

A. Yazdani et al. [18] evaluated three measurement terms to find the efficiency in the next word prediction system using trigram as keystrokes, time reduction and text generation rates. The results show typing time reduction of 33.36% and 73.53% in the number of keystrokes.

In a study, M. Parekh and Y. Patel [19] applied a linear probability model for word completion. The results were shown to efficiently reduce keystrokes by about 51% and 0.019s. Using Impala, the 2-grams of Google N-grams dataset was used on Hadoop Distributed File System (HDFS).

Bhuyan and Sarma [20] have described two predictive models: a traditional model for general word prediction and an enhanced model for ambiguous word prediction. They used 6 grams for the improved model and quad-grams for the general model. In addition, they implemented Katz's back-off smoothing model to avoid zero probability of words. The accuracy of the enhanced model is 66.88%, and the failure rate is 29.17%. Moreover, the accuracy of the traditional model is 60.68%, and the rate of failure is 32.35%.

Nagalavi and Hanumanthappa [21] proposed an exponential interpolation language model, which combines the POS language model and the N-gram language, model. The model is to predict an aligned sequence of words in blocks of articles in e-Newspaper, and the model's accuracy is 98.8%.

Gosh, et al. [22] have introduced a collaborative filtering algorithm with the Pearson correlation coefficient (PCC) to calculate the word similarity for predicting the frequencies of a bigram, which are missed. Furthermore, they elaborated that filtering or smoothing can increase the efficiency in next word prediction.

Word prediction and auto-complete are very useful in Search Engines and hand-held devices like smartphones for typing purposes, and it helps reduce incorrect spellings and efforts of typing, increasing the communication rate [23]. Further, it can diminish the gap between fast and slow typing, and it also helps people with disability in typing [24]. Likewise, W. Tesema and D. Tamirat [25] have implemented a system to help disabled people in typing as the next word prediction. The accuracy and precision of the model were found to be 90% and 73.34%, respectively.

Nowadays, word prediction is not unique to a language. A system is introduced that can predict emoticons (highly used in social media to express their thoughts) for a text [26]. Also, Yogesh et al. [27] included punctuation marks

and semantic rules of the language in the word prediction model. Furthermore, K. C. Arnold et al. [28] have introduced a system for phrase suggestion in mobile communication with the help of the 5-gram model. Unfortunately, no research has been conducted on the next word prediction system for the Balochi Language. Only one study conducted for the Balochi language in computer science is based on Optical Character Recognition (OCR) [29].

So far according to literature review N-gram model is not used for Balochi Language, therefore in this paper, a novel system is developed, based on a combination of N-gram model and Laplace smoothing. The proposed system helps users to quickly type in the Balochi language with a high rate of keystrokes and without spelling mistakes. The system is a simple model effective for next word prediction requirements that is smart enough to suggest words without any need for grammatical rules that help to save time.

3. Balochi Language

Balochi is among the oldest languages in the world. But, there is no official estimation of the number of Balochi speakers. However, grim statistics say that balochi is the first language of at least 10 million people worldwide. These people belong to Pakistan, Iran, Afghanistan, Oman, UAE, Turkmenistan, East Africa and India [1]. Unlike other languages, it has a recent history of scripting. Balochi had no written language before 19th century, hence most historical events and tales were transmitted verbally.

Nevertheless, it is said that in mid of 18th century Osman Kalmati wrote a book containing Balochi poems. That book is kept in British Library, London [30]. Due to their proximity to Iran and the fact that certain Balochs had studied the Holy Quran, they developed an Arabic and Persian writing system that was compatible with Balochi. Later, after the British had taken control in the 19th century, they gave the Balochi language a Roman/Latin script. British priests taught their people Balochi and translated the Bible into Balochi [31].

Table I. Samples of Balochi Language Letters

Balochi Academy	Phon-emes	Words	Pronunciation	Meaning
ا	\a\	اپس	Aps	Horse
ب	\b\	باہوٹ	Bahot	To be in Custody
پ	\p\	پاد	Paad	Foot
ت	\t\	تو	Tou	You
ٹ	\ ṭ\	ٹپ	Ta,p	Mark
ج	\j\	جار	Jaar	Announcement
چ	\c\	چار	Chaar	See
د	\d\	دپ	Dap	Mouth
ڈ	\ḍ\	ڈڈ	Dadd	Hard
ر	\r\	رد	Rad	Wrong
ڑ	\ŕ\	مڑاہ	Marrah	To be shy
ز	\z\	زبگ	Zahg	Child
ژ	\ż\	ژند	Zhand	Tired
س	\s\	سر	Sar	Head
ش	\sh\	شام	Shaam	Evening/ dinner
ک	\k\	کار	Kaar	Work
گ	\g\	گٹ	Gatt	Busy
ل	\l\	لال	Laal	Red
م	\m\	مات	Maat	Mother
ن	\n\	نان	Naan	Bread
و	\w\	وہد	Wahd	Time
ہ (ھ)	\h\	ھشک	Hushk	Dry
ی	\i\	یک	Yak	One
ے	\e\	نودربرے	Nodarbare	A Student

After the creation of Pakistan, the Baloch scholars adopted the Perso-Arabic script for their language. In the early 1950s, Gul Khan Nasir, who is called the father of modern Balochi poetry, published his first poetry book, "Gulbang". Also, Sayad Zahoor Shah Hashmi, "The Father of Balochi", created complete guidance on Balochi language writing to give it a standardized orthography. He also wrote the first dictionary of Balochi, named as "Sayad Ganj".

There has also been a Cyrillic script used for Balochi in addition to the Perso-Arabic

alphabet. In the 1980s, the Baloch of Turkmenistan invented this writing. However, the Cyrillic script was unable to gain international acceptance once the USSR was divided [32]. In May 2000, in a workshop at Uppsala University, the Latin script for the Balochi language was adopted again. Tab. I, presents Balochi alphabets of Arabic and Latin script.

3.1. Balochi Orthography

Sayad Hashmi [33] considers these excluded letters already have their alternative sounds like ص ،ث like س \s\, ذ ،ض ،ظ have alternative sounds as ز \z\, ع as ا \a\, غ as گ \g\ , ق as ک \k\ and ف as پ \p\. R Regarding standardization of language, academies like Balochi academy, Balochistan academy, Uppsala University and other Balochi language publishers use different alphabets of Arabic script. So, adding those letters to a Balochi keyboard also becomes necessary.

Thus far, in the world of technology, the Balochi language does not have proper research, and one of the reasons is the lack of adequate corpus. There is a need of non-ambiguous corpus for efficient word prediction system in the Balochi language to bring the language to the standard of other languages of the world.

4. Methodology

The proposed model is divided into three steps: in step one, data is preprocessed; in step two, the data is trained to generate N-grams as Unigram, Bigram, Trigram, Four-gram and so on, and in step three, the accuracy of the model is calculated through the test data by using perplexity and accuracy formula. These steps are elaborated further below and Fig. 1 represents the complete workflow of proposed model.

4.1. Data Pre-Processing

Data pre-processing is one of the primary and critical steps in processing the text to obtain pertinent data from the raw document. The text pre-processing steps are followed, which are required to gain an appropriate form of data for the model processing.

4.1.1. Corpus Collection

The whole data is based on a Balochi novel named “Bahisht o Doza” “بهشت ءُ دوزه” [34]. It is in a text file format that is easy to access. The corpus consists of 317590 words; it is divided into 80% for training purposes and 20% for testing. The total counting of words and sentences of corpus are presented Tab. 2.

Table II. Total Number of Words and Sentences in Corpus

	No of Words	**No of Sentences**	**Divided Percentage of corpus**
Training	253962	26511	80%
Testing	63629	6707	20%

As earlier explained, the Balochi language does not have a standard form. It has an ambiguous structure. One word has two different spellings, which makes the whole corpus confusing. This problem decreases the probability of the number of occurrences of the word. Therefore, it can directly affect the accuracy of the model is calculated through the test data by using perplexity and accuracy formula. These steps are elaborated further below and Fig. 1 represents the complete workflow of proposed model.

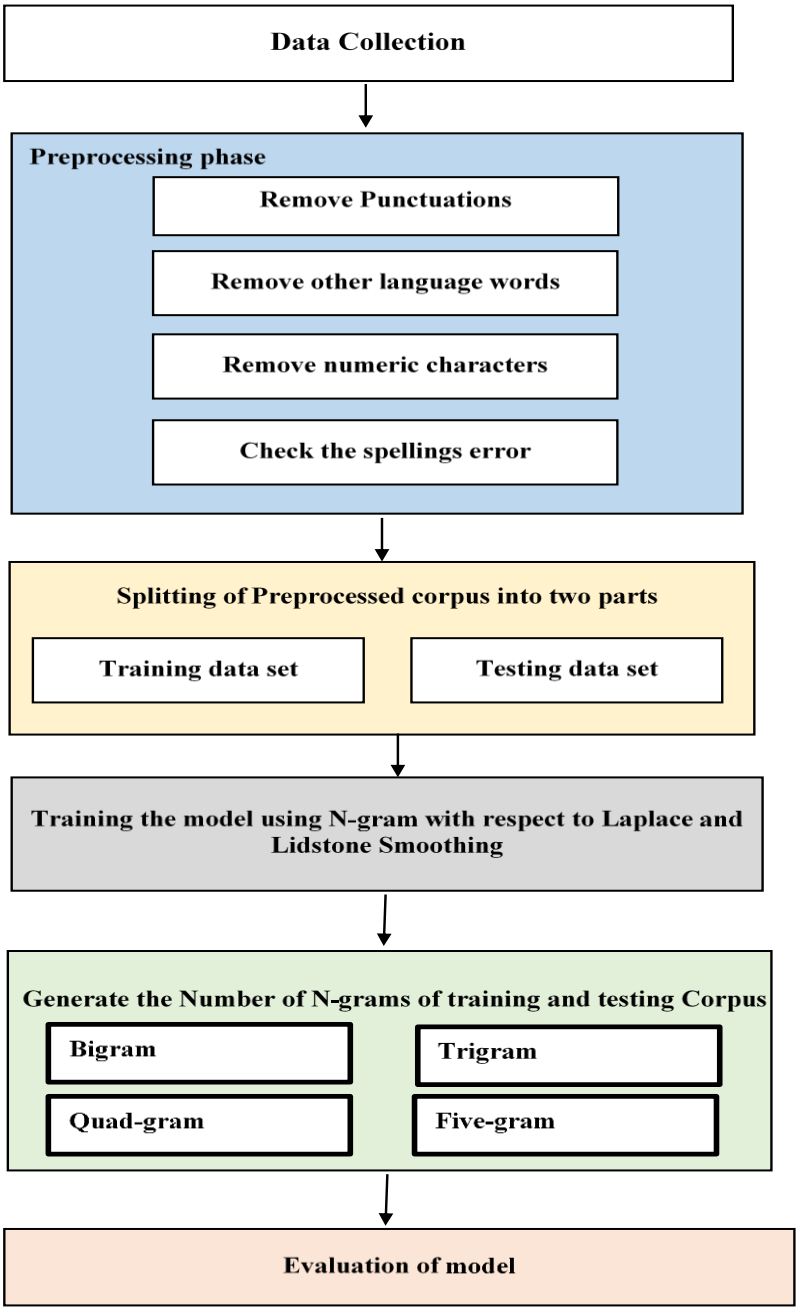

Fig.1. Methodology of the Proposed Model

Table III.: Example of Ambiguous words

Meaning	**Balochi Word**	**Balochi Word**	**Pronunciation**
Tell, say	گوش	گش	Gosh / Gwash
Sister	گوهار	گہار	Gohar
Think, thought	حیال	هیال	Hayal
History	راجدپتر	راج دپتر	Raj Daptar
You	شما	شُما	Shuma
Hani (Name)	هانی	حانی	Hani

In Tab. 3, it can be seen the word گوش that is pronounced as "Gwash" and the other one is گش that is pronounced as "Gosh", both the words are same but spoken in different dialects. The other word حیال and هیال as "Hayal", their pronunciation is same but spellings are different. As lack of language standardization the discussion whether to include ح in Balochi language or not. So those words which are started with H are sometimes written with ح and ھ. Due to these ambiguity the system considers these words different from one another.

4.1.1.Text Pre-Processing

There are several steps in text pre-processing, like noise removing, stemming and lemmatization, stop words removing, spell-checking etc.; each corpus is pre-processed according to the requirement of the model. Following are the steps that are required for the model before training.

- The first step is to clean the noise like words from other languages and numbers.
- Spellings of the words are checked to avoid ambiguity among words.
- The machine considers punctuation marks as words, which are not required in this research. After converting the docx file into txt format, the whole corpus is segmented into sentences with the use of (. ؟ ، !) As a separator. Then all other punctuation marks (: ؛ ' ") are removed from the corpus.
- The complete corpus is tokenized through NLTK library, those tokenized words are also considered Unigram.
- While testing the model there may occur words that are not seen in the training corpus those words are known as out of vocabulary words. These are tagged as Unknown <unk>.

Once the model is cleaned from the noice the next step is to train the model with respect to N-gram model that is proposed for the word prediction system.

4.2. N-Gram Language Model

Word predicting is a probabilistic task to check the appearance of the next word's probability with the current word. The main task of Language Models is to assign probabilities to the word sequences. One of the basic approaches is to implement word prediction on the N-gram Language model. It is a statistical model used to predict the possible word that follows the sequence of the given word [35]. The N-gram language model predicts the probability of a word occurrence with its previous word, also called the Markov assumption [7]. The computing task is done in Eq. (1).

$$WP=P(w|h) \qquad (1)$$

Where WP is word prediction, P is the probability of the current word w, and h is the history of the current word. Thus, the N-gram model predicts (n-1) words. Hence, the Markov assumption provides the technique to look at the current word's probability to its last term. The N-1 Markov assumption model measures the N-gram model. It assumes that the next word can only be predicted with the awareness of the previous N-1 word. The formula is obtained by using Markov Model with the N-gram model as shown in Eq. (2).

$$P(W_1^N) \approx \prod_{K=1}^{N} P(W_k|\, W_{k-1}) \qquad (2)$$

When N = 1, it is known as Unigram, that predicts how often a word has occurred in a

sentence sequence. When N = 2, it is known as Bigram, also called Markov first order. This is because it checks one prior word's probability with the current word. The general formula for the sequence of bigram to predict the next word conditional probability is mentioned in Eq. (3), and the generalized form is shown in Eq. (4).

$$P(W_n|W_{1,n-1}) \approx P(W_n \mid W_{n-1}) \quad (3)$$

$$P(W_N \mid W_1^{N-1}) \approx P(W_N \mid W_{N-N+1}^{N-1}) \quad (4)$$

However, the probability estimation is calculated using MLE, that is, Maximum Likelihood Estimation; it is done by taking the counts from the corpus and then normalizing them. After the normalization, the general formula is derived in Eq. (5).

$$P\ (W_n|W_{n-1}) = \frac{P(W_n|W_{n-1})}{P(W_{n-1})} \quad (5)$$

Thus when N = 3, it is known as a trigram that checks the probability of the next word with two previous words; it is called Markov second order and the general formula is presented in Eq. (6).

$$P(W_{1,n}) = P(W_n|W_{n-2},W_{n-1}) \quad (6)$$

After normalization, the formula obtained is shown in Eq. (7).

$$P(W_n|W_{n-2},W_{n-1}) = \frac{P(W_n \mid W_{n-2}, W_{n-1})}{P(W_{n-2},W_{n-1})} \quad (7)$$

Likewise, when N = 4, 5, 6 and so on, it checks the occurrence of the next word to three to four to five previous words as N-1.

The probability estimation of a sentence in Unigram model is as "منی نام هانی اِنت" "My name is Hani". Tab. 4 represents that how the Balochi language is pronounced, and the meaning of each word is presented in English.

Table IV. Balochi Words Pronunciation

Balochi Words	**Pronunciation**	**Meaning in English**
منی	Mani	My
نام	Naam	Name
هانی	Hani	Hani
انت	Int	Is

P(My name is Hani) = P (My) × P(name)× P(is) × P(Hani)

P (منی نام هانی اِنت) = P (منی) × P(نام) × P(هانی) x P(اِنت)

In the generalized form, the Unigram formula is given in Eq. (8).

$$P(Wi) = \frac{C(Wi)}{C(W)} \quad (8)$$

P is the probability of the current word wi, c is the count and w is the entire sentence,

$$P(\text{هانی}) = \frac{C(\text{هانی})}{C(\text{منی نام هانی اِنت})}$$

For the bigram model, the probability sequence of a sentence is,

P(My name is Hani) = P(My|<sos>) × P(name | My)× P(is |name) × P(Hani | is) × (</eof> | Hani)

P (منی نام هانی اِنت) = P(منی |<sos>) × P(| نام) × P(منی | هانی) x P(هانی |اِنت) x P(</eof> | اِنت)

<sos> represents the start of sentence, it is used in training to determine the start of the sentence. While </eof> represents the end of sentence, these are the padding sequences that are used separate one sentence probability from another. From the generalized formula, the Eq. (5) becomes,

$$P\ (\text{منی} \mid \text{نام}) = \frac{P(\text{منی} \mid \text{نام})}{P(\text{منی})}$$

For trigram model, the probability estimation is given as:

P(My name is Hani) = P (My|<sos><sos>) × P(name | My <sos>)× P(is | My name) × P(Hani | name is) × P(</eof> | is Hani) × P(</eof> | </eof> Hani)

P (منی نام هانی اِنت) = P(منی |<sos> <sos>) × P(نام | منی<sos>) × P(هانی| منی نام) x P(| اِنت) × P(نام هانی) × P(هانی اِنت | </eof>) × P(</eof> | </eof> اِنت)

Hence Eq, (7) becomes,

$$P\ (\text{هانی}|\text{نام منی}) = \frac{P(\text{منی نام}|\text{هانی})}{P(\text{منی نام})}$$

Thus, the N-gram model is a probabilistic sequential model that suggests the word according to its n number of previous words. It represents the nth order of Markov assumption that it depends on the preceding structure of the N-gram. However, in the above example, if the probability of a word "تئی" (your) is checked instead of the word "منی" (my) which never occurred in training due to multiplication and division, the probability becomes 0. That brings data sparsity, due to which the MLE model is unsuitable. To counter this problem, the smoothing technique is utilized.

4.2.1. Laplace Smoothing (Add-one Smoothing)

Maximum Likelihood estimation, the generalized form of N-gram, has the data sparseness problem. Due to that, the problem of zero probability occurs. However, the smoothing technique plays a vital role to avoid the sparsity of data which means looking ahead. Among various smoothing techniques, one of the simplest is the Laplace smoothing. The Laplace smoothing adds one to any number of N-gram before normalization.

By modifying Eq. (5) with Laplace smoothing, 1 is added to the count and Vocabulary V to its denominator, as represented in Eq. (9).

$$P_{add-1}(W_n|W_{n-1})=\frac{P(W_n, W_{n-1})+1}{P(W_{n-1})+V} \quad (9)$$

In spite of this, Laplace gives too much probability to unseen data instead of seen data. This problem can be avoided by using Lidstone smoothing.

4.2.2. Lidstone Smoothing

Lidstone smoothing is also named as Expected Likelihood Estimator. It adds a value (λ), smaller than and equal to 1 to the unseen data. It supposes that each n-gram has been seen λ times that is $0 < \lambda \leq 1$, Eq. (10) represents the formula of Lidstone,

$$P_{Lid}(W_1 \ldots W_n) = \frac{C(W1 \ldots wn)+\lambda}{N+B\lambda} \quad (10)$$

Where P is the probability of N-gram, C is the training count of N-gram, N is the total number of n-grams in training, and B represents the possible number of N-grams, and λ is the small positive number for unseen data to reduce sparsity.

4.3. Model Evaluation

The primary purpose of model evaluation is to determine a model's simplified accuracy on unseen future data. In this phase, the decision is taken on how well the model works. For example, the perplexity evaluation matrix and accuracy to evaluate the N-gram model are used.

4.3.1. Perplexity

Perplexity is the intrinsic evaluation metric used to determine how well the model predicts the next word. The model assigns the highest probability to the test data by that the model does not get confused it gets a good understanding of the model from the test set. It is the inverse of the probability which is assigned to the test set, as shown in Eq. (11).

$$PP(W) = \frac{1}{P(W_1, W_2, \ldots, W_N)^{\frac{1}{N}}} = \sqrt[N]{\frac{1}{P(W_1, W_2, \ldots, W_N)}} \quad (11)$$

The higher the probability, makes the perplexity lesser. The model performs well if it has lower perplexity. It is also known as cross-entropy, 2H (W), which means the average number of bits needed to encode one word. However, the perplexity of the words is encoded in those bits [36], like if the branching factor is 200. In that case, prediction is based on among these 200 predictions, which are the best to be chosen, which is shown in Eq. (12).

$$PP(W)=2H(W)=2^{-\frac{1}{N}P(w1,w2,\ldots,WN)} \quad (12)$$

In Eq. 12, H (W) = the average bits required for encoding one word, and 2H (W) = the average words which could be encoded by the use of H (W) bits.

4.3.2. Accuracy

The model's accuracy means the total number of predictions divided by the sum of correct and incorrect predictions. It has been calculated by embedding the model into an application then correct and inaccurate predictions are manually checked. However,

the task is time-consuming but worthy enough to estimate the result. After all, the user determines how accurate the model is. Eq. (13) represents the formula to calculate the model's accuracy.

$$Accuracy=\frac{numberofprediction}{correctprediction+incorrectprediction} \quad (13)$$

5. Result and Discussion

In the proposed system, approximately 300,000 words of Balochi book are divided into two parts 80% for training and 20% for testing. The most frequent Unigrams, Bigrams, Trigrams, Quad-grams and Five-grams generated from training corpus are illustrated in Figs. 2-6 respectively.

These N-gram counts are produced using training corpora, which implies smoothing to prevent data sparsity. Sparse data is a computational issue that is the phenomenon of having insufficient data in a dataset or data with value of zero. This issue is solved using smoothing techniques, where the zero value data are given a value of 1 for Laplace smoothing and a value of 0.5 (according to the requirement) for Lidstone smoothing.

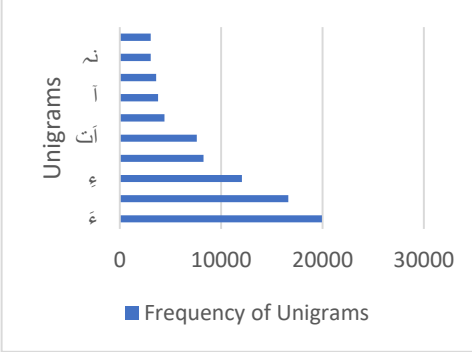

Fig. 2. Most Frequent Words in Unigrams from training corpus

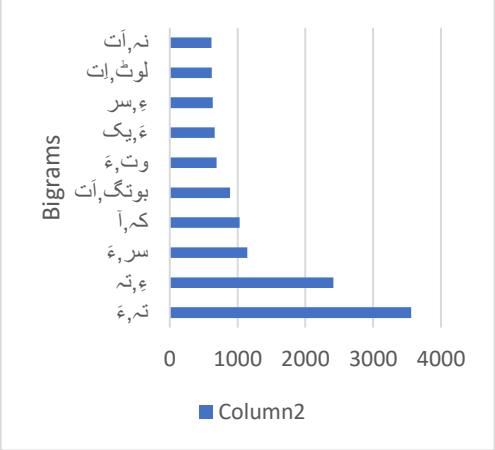

Fig. 3. Most Frequent Words in Bigrams from training corpus

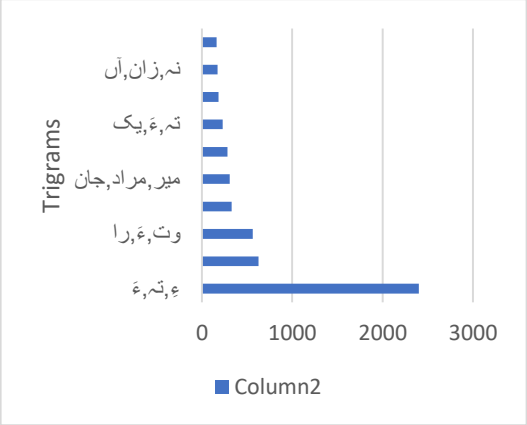

Fig. 4. Most Frequent Words in Trigrams from training corpus

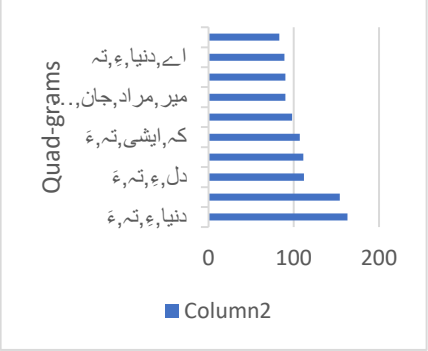

Fig. 5. Most Frequent Words in Quad-grams from training corpus

Fig. 6. Most Frequent Words in Five-grams from training corpus

5.1. Intrinsic Evaluation Result

For various N-gram models, the perplexity (i.e., intrinsic evaluation) result is obtained using Laplace and Lidstone smoothing, as shown in Tab. 5 and Fig. 7. Lidstone smoothing employs a 0.5 value for unseen words in the testing corpus since it outperforms Laplace by a quite margin. Laplace, however, employs one value for the testing corpus's unseen data. After applying Laplace and Lidstone smoothing, the difference in Bigram perplexity is 42, trigram is 56, quad-gram difference is 42, and five-gram difference is 35. The average difference is 43.75, which indicates that the N-gram performs 43.75 times better than Laplace smoothing when Lidstone smoothing is being used. It demonstrates how much more effectively Lidstone smoothing decreases data sparsity. This benefits in lowering word prediction errors.

Table V. Perplexity results after Laplace and Lidstone Smoothing

Model	**Perplexity (Laplace)**	**Perplexity (Lidstone)**	**Difference in Perplexity**
Bigram	419	377	42
Trigram	359	303	56
Quad-gram	312	270	42
Five-gram	275	240	35
Average			43.75

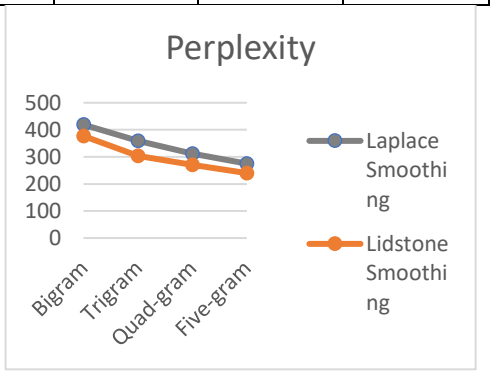

Fig. 7. Perplexity measurements through Laplace and Lidstone smoothing

5.2. Extrinsic Evaluation Result

Further, the trained model is embedded in an application to predict the next word (i.e. extrinsic evaluation). The following conditions are applied in the application:

- When a user enters a word, it uses the bigram model and predicts one word to the next.
- When the user enters two words, it uses the trigram model and predicts two words to the next word.
- When a user enters three words, it uses the Quad-gram model and predicts three words to the next word.

5.2.1. User Interface

The user interface helps to interact with the model in the real world, which is considered a final product. A user checks the model through the user interface to see whether the result is correct. The user enters one word; after pressing space, it predicts the next most common word. The following figures represent the user interface; Figs. 8-10 show the model embedded in an application that how it is suggesting the next possible word.

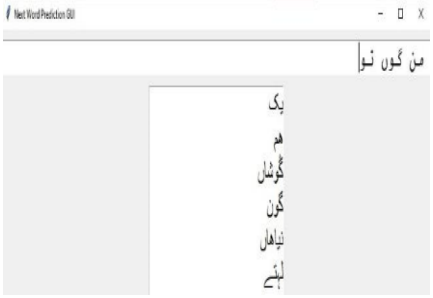

Fig. 8. Predicts the next word based on trigram

Fig. 9. Predicts the next word based on quad-gram

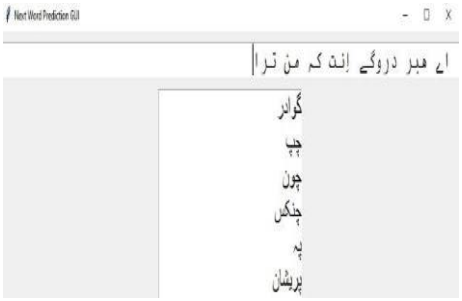

Fig. 10. Predicts the next word based on five-gram

From the above application, 100 sentences are generated for five-gram, quad-gram, Trigram, and bigram to determine the accuracy. The accuracy of Bigram is 68%, Trigram is 76%, Quad-gram is 80 %, and Five-gram is 93%. Tab. 6 and Fig. 11 represent the accuracy of these models.

Table VI. Accuracy generated from training corpus

Model	**Accuracy**
Bigram	68%
Trigram	76%
Quad-gram	80%
Five-gram	93%

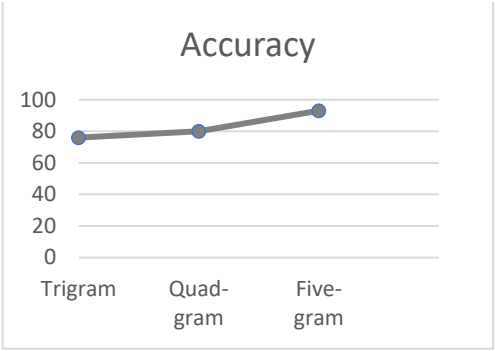

Fig. 11. Measurement of Accuracy

5.2. Result Analysis

A large corpus of approximately 3 lac words was utilized to build the Next Word Prediction System for the Balochi Language. The corpus was cleaned from noises like punctuation marks, numbers, and other languages letters that are not the required characters to be predicted. As Balochi language is unstandardized the spell-checking of each word was done manually to avoid the ambiguity of the data and this is represented in Tab. 4. However, N-gram Model was used to analyze the frequency of the words. N-gram Model is beneficial as it depends on the frequency of terms. The results from the perplexity (intrinsic evaluation metric) and accuracy (extrinsic evaluation metric) showed that the accuracy of Five-gram model is much better than other models. It has a perplexity result of 240, and its accuracy is 93%.

This research determined that Five-gram is a suitable model for the system. Moreover, it can also generate new sentences that are reliable for real-world application**.**

6. Conclusion and Future Work

The word prediction implemented in the Balochi language with the help of the N-gram

model is an innovative study. In the area of NLP, the Balochi language hardly has any research. Further, the Balochi language does not have a large corpus to do a wide range of research. A corpus of 317590 words is created with no ambiguity. The N-gram model is used to build a word prediction system with that Balochi corpus. The N-gram model has shown promising results in predicting the next word in the Balochi language. To avoid the sparsity of data, Lidstone smoothing has been used, which improved perplexity has compared to Laplace smoothing. The trained model has been embedded in an application to achieve the model's accuracy. However, the result of the five-gram model is 93%, and Quad-gram has 80% of the prediction system.

The accuracy of the model mainly depends upon the training corpus. Therefore, we must have a non-ambiguous corpus to achieve a precise result. Nevertheless, the feature has brought ease in typing for native users; it will benefit Balochi language users.

Nevertheless, many other smoothing techniques, such as the Back-off model, Good turning model, KneserNey smoothing etc., are included in future work to check whether it gives better results in perplexity. Furthermore, a word prediction based on POS tagging can enhance the efficiency of the word prediction system. It can also be great to include the corpus of other Balochi dialects to broaden the service area. This research will help develop a plan to complete the sentence according to the grammar and correct the spelling.

REFERENCES

[1] C. Jahani, “A Grammar of Modern Standard Balochi Language,” in Acta Universitatis Upsaliensis, 1st ed., Upsala, Sweden, 2019.

[2] G. W. Lesher, B. J. Moulton and D. J. Higginbotham, "Effects of N-gram order and training text size on word prediction," In Proceedings of the RESNA'99 Annual Conference, pp. 52-54. 1999.

[3] M. Ghayoomi and E. Daroodi, "A POS-Based Word Prediction System for the Persian Language," in Advances in Natural Language Processing., Springer, Berlin, Heidelberg, 2008.

[4] S. Shahzadi, B. Fatima, K. Malik and S. M. Sarwar, "Urdu Word Prediction System for Mobile Phones," World Applied Sciences Journal, vol. 21, no.1, pp. 1260-1265, 2013.

[5] M. Saeed, A. Nawaz, K. Ahsan, S. Jabeen, K. Islam, M. Hassan and F. A. Siddiqui, "Effective Word Prediction in Urdu Language Using Stochastic Model," SJCMS, vol. 2, no.2, pp. 38-46, 2018.

[6] J. Mahar and G. Memon, "Probabilistic Analysis of Sindhi Word Prediction using N-Grams," Australian Journal of Basic and Applied Sciences, vol. 5, no.5, pp. 1137-1143, 2011.

[7] M. Haque, M. Habib and M. Rahman, "Automated Word Prediction in Bangla Language Using Stochastic Language Models," International Journal in Foundations of Computer Science & Technology (IJFCST), vol. 5, no.6, pp. 67-75, 2015.

[8] T. M. Habib, A. Al-Mamun, S. M. Rahman, S. M. T. Siddiquee and F. Ahmed, "An Exploratory Approach to Find a Novel Metric Based Optimum Language Model for Automatic Bangla Word Prediction," International Journal of Intelligent Systems and Applications, vol. 2, no.2, pp. 47-54, 2018.

[9] T. Mittra, L. Islam and D. C. Roy, "Prediction of Semantically Correct Bangla Words Using Stupid Backoff and Word-Embedding Model," in 2nd International Conference on Applied Information Technology and Innovation (ICAITI), Denpasar, Indonesia, pp. 66-70, 2019.

[10] N. Shah and N. Khetra, "Syntactic Word Prediction for Hindi," IJSART, vol. 3, no.3, pp. 1191-1195, 2017.

[11] R. Sharma, N. Goel, N. Aggarwal, P. Kaur and C. Prakash, "Next Word Prediction in Hindi Using Deep Learning Techniques," in International Conference on Data Science and Engineering (ICDSE), Patna, India, pp. 55-60, 2019.

[12] H. Hamarashid, S. Saeed and T. Rashid, "Next word prediction based on the N-gram model for Kurdish Sorani and Kurmanji," Neural Computing and Applications, vol. 33, no.5, pp. 4547-4566, 2020.

[13] Y. Hacohen-Kerner and I. Greenfield, "Basic Word Completion and Prediction for Hebrew," in String Processing and Information Retrieval, Springer, Berlin, Heidelberg, pp. 237-244, 2012.

[14] G. S. Mahi and A. Verma, "PURAN: Word Prediction System for Punjabi Language News," in Data Management, Analytics and Innovation, Proceedings of ICDMAI, Springer, Singapore, 2019, pp. 383-400.

[15] M. Herold, E. Alant and J. Bornman, "Typing speed, spelling accuracy, and the use of word-

prediction," South African Journal of Education, vol. 28, no.1, pp. 117-134, 2008.

[16] Nandini, P. Hamsaveni and P. Charunayana, "Hybrid Machine Learning based Kannada Next Word Prediction," International Research Journal of Engineering and Technology (IRJET), vol. 7, no.7, pp. 5605-5608, 2020.

[17] A. Pourmohammad, M. Gulami, J. Mahmudov, Y. Aliyev and R. Akberov, "The First Azeri (Azerbaijani) Language Next Word Predictor," Information Systems and Signal Processing Journal, vol. 5, no.1, pp. 1-4, 2020.

[18] A. Yazdani, R. Safdari, A. Golkar and S. R. N. Kalhori, "Words prediction based on N-gram model for free-text entry in electronic health records," Health Information Science and Systems, vol. 7, pp. 1-6, 2019.

[19] M. Parekh and Y. Patel, "Word Completion and Word Prediction using Probabilistic Model," 2019.

[20] M. P. Bhuyan and S. K. Sarma, "A Higher-Order N-gram Model to enhance automatic Word Prediction for Assamese sentences containing ambiguous Words," International Journal of Engineering and Advanced Technology (IJEAT), vol. 8, no.6, pp. 2921-2926, 2019.

[21] D. Nagalavi and M. Hanumanthappa, "A Model to Predict Words in the Sentence to Identify an Aligned Sequence of Article Blocks in e-Newspaper," International Journal of Advanced Research in Computer and Communication Engineering, vol. 5, no.2, pp. 160-164, 2016.

[22] S. Ghosh, H. S. Rana and R. Tomar, "Word Prediction using Collaborative Filtering Algorithm," IJCTA, vol. 9, no.22, pp. 115-122, 2016.

[23] R. Makkar, M. Kaur and D. V. Sharma, "Word Prediction Systems: A Survey," Advances in Computer Science and Information Technology (ACSIT), vol. 2, no.2, pp. 177-180, 2015.

[24] M. Bhuyan and S. Sarma, "An N-gram based model for predicting of word formation in Assamese language," Journal of Information and Optimization Sciences, vol. 14, no.2, pp. 427-440, 2019.

[25] W. Tesema and D. Tamirat, "Enhancing the Text Production and Assisting Disable Users in Developing Word Prediction and Completion in Afan Oromo," Journal of Information Technology & Software Engineering, vol. 7, no.2, pp. 1-4, 2017.

[26] R. Mahte, R. Nair, V. Nair, A. Pillai and P. M. Kulkarni, "Emoticon Suggestion with Word Prediction using Natural Language," International Research Journal of Engineering and Technology (IRJET), vol. 07, no.5, pp. 3104-3108, 2020.

[27] Y. Sharma, J. S. Bindra, K. Aggarwal and N. Dahiya, "Word Prediction and Sentence Completion," IJSRD - International Journal for Scientific Research & Development, vol. 7, no.3, pp. 744-747, 2019.

[28] K. C. Arnold, K. Z. Gajos and A. T. Kalai, "On Suggesting Phrases vs. Predicting for Mobile Text Composition," in Symposium on User Interface Software and Technology, Tokyo, Japan, pp. 603-608, 2016.

[29] G. J. Naseer, A. Basit, I. Ali and A. Iqbal, "Balochi Non Cursive Isolated Character Recognition using Deep Neural Network," International Journal of Advanced Computer Science and Applications, vol. 11, no.4, pp. 717-722, 2020.

[30] "Baask," [Online]. Available: http://baask.com/diwwan/index.php?Topic=4381.0.

[31] Z. Baloch, "بلوچی راست نبیسَی," Raeesi Chaap o Shing Jah, 2015.

[32] P. Kokaisl and P. Kokaislová, "The Ethnic Identity of Turkmenistan's Baloch," Asian Ethnology, vol. 78, no.1, pp. 181-196, 2019.

[33] S. Hashmi, "بلوچی سیاهگ ءِ راست نبیسی," in Sayad Hashmi Academy, Karachi, Pakistan, 1964.

[34] M. A. Badini, "بهشت ءُ دوزه," in New College Publication, Quetta, Pakistan, 2013.

[35] D. Jurafsky and J. H. Martin, "Speech and Language Processing: An introduction to natural language processing," in Prentice Hall, 1st ed., New Jersey, NJ, USA, 2000.

[36] C. Campagnola. "Perplexity in Language Models", 2020, [Online]. Available: https://towardsdatascience.com/perplexity-in-language-models-87a196019a94.