VFAST Transactions on Software Engineering
https://vfast.org/journals/index.php/VTSE@ 2024, ISSN(e): 2309-3978, ISSN(p): 2411-6246
Volume 12, Number 4, October-December 2024 *pp:158-166*

An NLP Approach to Predict and Suggest Next Word In Urdu Typing

Muhammad Hassan [1], **Saad Ahmed** [2], **Rohail Qamar** [1*], **Saman Hina** [3], **Hira Farman** [2]

[1]Department of Computer Science & Information Technology, NED University of Engineering & Technology, Karachi, Pakistan; [2]Department of Computer Science, IQRA University, Karachi, Pakistan; [3]Independent Research, London, England

Keywords: *Urdu Keyboard, Urdu Predictor, Rule-Based, Word Prediction, Stochastic Model, Unigram, Markov Model,*

Journal Info:
Submitted:
December 30, 2024
Accepted:
December 15, 2024
Published:
December 22, 2024

Abstract

The importance of fast speed typing is very important for computerization of contents in any language. Urdu which is a prominent language of south Asia also subjected to computerization and due to lack of resources available the process of computerizing the Urdu content has been hampered by the low speed in Urdu typing. Similarly high demand of Urdu content which needs to be digitized makes it more expensive. During this research we have worked on various aspects of Urdu language and discovered many limitations which exists which are creating hurdles in high-speed typing in Urdu language. As 35+ alphabets are in the Urdu language, the international ISO standard keyboards are only on English alphabets that are 25+ that make a quiet big difference of about 10 alphabets that means we have to press and hold SHIFT key while typing these 10+ alphabets that are wasting our time and slowing our speed of typing so we tried to solve this problem by keeping the standard along as they are. This paper is based on the word prediction and suggestion in Urdu Language (UL) based on a stochastic model, Hidden Markov Model is used to predict the next word, while Unigram Model was also used to suggest the current word and the next upcoming word, N-Gram Model was followed keeping N=2. Now, the biggest achievement in this Paper is POS tagging as each suggestion and prediction is also based upon Tagged words with a dataset of thousands of Tag combinations based upon frequency of occurrence is on test data. This tool is developed to implement this concept for Urdu Language (UL) and tested by regular and new URDU content writers to check their improvements in their typing speeds. We made some programs to let you type less and choose more.

***Correspondence author email address:** rohailqamar@cloud.neduet.edu.pk
DOI: 10.21015/vtse.v12i4.2011

1 Introduction

Urdu is one of the most prominent languages of the world that has around 70 million native speakers all around the globe [1]. Urdu is both the state language of the Islamic Republic of Pakistan as well as constituted as one of the scheduled languages of India. Though the word Urdu, itself is a Turkish word that means "horde" (Lashkar), Urdu is more closed to Hindi (in terms of grammar and phonology), Arabic and Persian (in terms of lexicon and script). Being the Natural Language of millions of people around the world, Urdu has been subject to computerization since the 1980s. The advancement of technology, globalization and the need for better computable linguistic artefacts has made it necessary to apply and study the Natural Language Processing techniques on Urdu. Although, a large number of alphabets, as compared to English (Table 1) in Urdu are found to be a major hurdle while using Urdu as the Input Language.

Table 1. URDU ENGLISH Keyboard

Language	Alphabets	Keyboard Keys	With Shift
English	26	26	-
Urdu	39+13	26	26

Word prediction and suggestion is one of the primary features of Natural Language Processing that can help in overcoming the aforementioned problem with the use of Urdu Language in modern-day IT works. As the basic drive behind the word prediction is to forecast the possible word in a given context or phrase while reducing the time and effort of typing, a smart, precise and effective Word Suggestion and Prediction model for the Urdu language is well needed. However, various Word Prediction models, such as eZiText, T9, iTAP, are available and used in the global IT industry, none of them is well-designed and light-weight enough to be used as the primary Part of Speech (POS) based Word Prediction and Suggestion utility in modern portable computational devices. This paper, therefore, presents a modern natural language processing tool, based on the previously researched word prediction and suggestion tool, that can easily predict and suggest the Urdu Language words and phrases while providing greater help to the technical community in developing modern-day computer applications with greater support for the language of millions i.e., Urdu. Below are the world's most spoken languages by number of native speakers, with data sourced from Ethnologue [14]. Table 2 shows the popular languages in 2024 year.

Table 2. Top 10 Languages in 2024

Rank	Language	Native speakers (in millions)
1	English	1500
2	Chinese	1100
3	Hindi	608.8
4	Spanish	559.5
5	Standard Arabic	332.5
6	French	311.6
7	Bengali	278.2
8	Portuguese	263.8
9	Russian	255.4
10	Urdu	237.9

2 LITERATURE REVIEW

Within the past few years, several models and frameworks have been derived and developed for word prediction and suggestion. The majority of these models have used stochastic models to predict the probable outcomes (word) by estimating random variables (last word or alphabet). Khan et.al. (2009) developed the first Urdu Language Virtual Keyboard with features like word prediction [2]. The primary technique used by these researchers is based on the character frequency analysis of Urdu corpus. To optimize the layout of the keyboard, Monte Carlo Simulation with simulated annealing of word lists from Urdu corpus is used.

Another, research carried by [3] proposed a much faster and better technique for word prediction in the Urdu language. The researchers used the Bigram Model with pre-defined Urdu and English corpus to predict words, as the prime focus of the study was Roman Urdu. The resulted tool was fast enough to be adopted for portable devices [4] developed a similar prediction system for the Italian language, using

N-gram and Lexicon methods. The Italian language is similar to Urdu as it is an inflected language where the word predicted is more dependent on context.

Unlike their predecessors, [5] avoided the syntactic parsing and adopted Sparse Matrix, surface features and semi-supervised techniques to predict the whole sentence, rather than the word. The results from the study reveals that the specificity is much more useful in the prediction of words and sentences. Another study by [6] was conducted for the development of the word prediction in South Asian languages, used the stochastic model. The study reveals that the use of the stochastic model along with the large corpus is a better choice for reducing the chance of word misprediction [7] Studied the problems and issues related to the modern word prediction approaches on both single words as well as phrase. Using the Fussy Tree Model, the researchers anticipated a probabilistic driven model to predict more accurate words and phrases.

Some of the applications of language modelling are sentiment analysis and next-word suggestion or prediction. Also, some work has been done on Urdu in other areas such as sentiment analysis [13] but the area of next-word prediction largely remains unexplored. The detailed review of previous literature on the topic identifies the need for a new Urdu language word and phrase prediction tool using stochastic models and fussy tree models.

3 Data Collection

Data is the most prominent aspect of any research or study, particularly the ones that are associated with artificial learning and prediction. Urdu is a language that has 39 basic alphabets and 13 extra characters [8], and around 254165 words in its dictionary [9]. Furthermore, the globalization and increasing number of Urdu speakers as their second language have affected the total word count in the Urdu language. In short, a huge amount of loan words has also been drawn into Urdu vocabulary over the course of time. Besides having various loan words from other native languages, another key feature that need to be considered during this research, of the Urdu language is the presence of diacritics. The diacritics in Urdu has also exaggerated the possible word counts of the language.

Part-of-Speech (POS) tagging plays a crucial role in natural language processing. This study introduces the first large-scale POS-tagged corpus and a BiLSTM-based POS tagger for Shahmukhi (Western Punjabi). A 0.13-million-word balanced corpus from 14 text domains was annotated using a newly devised Shahmukhi POS tagset and tagging guidelines. Multi-step evaluation ensured consistency, achieving 95.35% inter-annotator agreement and a Kappa coefficient of 0.94. The BiLSTM tagger outperformed TreeTagger and Stanford POS tagger, with an f-score of 96.11% and accuracy of 96.12%. Transfer learning with Word2Vec and ELMo further improved results, demonstrating promise for low-resource, morphologically rich languages [17].

Lemmatization extracts the root form of a word, aiding NLP tasks like Information Retrieval, Machine Translation, and Plagiarism Detection. While rule-based lemmatizers exist for high-resource languages, limited efforts have targeted South Asian languages like Urdu, a morphologically rich and low-resource language. This study presents a novel Urdu lemmatization system using a dictionary lookup approach. Key contributions include developing a benchmark Urdu corpus, exploring the relationship between POS tags and lemmatization, and standardizing lemmatization methods. Experiments demonstrated a peak accuracy of 76.44%, highlighting its potential to improve Urdu NLP applications [18].

Domain shift occurs when training data differs from application data, reducing model accuracy. In Natural Language Processing (NLP), this challenge is particularly significant for predictive typing systems, such as predicting and suggesting the next word in Urdu typing. To address this, domain adaptation adjusts pre-trained models to target domains, often using unsupervised methods due to the scarcity of labeled target data. This study introduces Cosine Generative Adversarial Network (CosGAN), a source-free domain adaptation approach integrating GANs with cosine embedding loss to tackle unsupervised domain adaptation challenges. CosGAN's simplified

two-step training process achieves results comparable to state-of-the-art methods. Experiments on benchmark datasets demonstrate its superior accuracy and generalization, making it suitable for applications such as predictive Urdu typing, robotic vision, and other automation tasks. However, challenges like GAN-induced instability and overfitting remain areas for further improvement [19].

3.1 Methods for Data Collection

As a vocabulary develops over a period of time and several factors are found to be affecting it, the collection of an exclusive yet extensive vocabulary of any language is a time taking work. In order to collect the data for the corpus of the Urdu language, several special programs are developed. These programs provided greater results in extracting Urdu words and phrases.

3.1.1 Dictionaries

The very first method or approach, adopted for data collection is the dictionary. Different Urdu dictionaries, particularly the ones that have already established databases, were collected, searched and saved. These dictionaries not only provided a greater amount of data, around 85000 words in a single database but also helped in extracting already sorted data. Moreover, the dataset collected from the dictionary provided with the greater ability to identify the probability and nature of the words in terms of the POS while using them to tag and train the artificial intelligence for improved predictions.

3.1.2 Urdu Word Extractor

Another tool that was developed and used for extracting both words and phrases was Urdu Word Extractor. This tool grabbed different Urdu documents, such as novels, poetry, articles and books, read them word by word and exploded each word into a separate text (.txt) file. Furthermore, the tool also provided the pair of words by splitting phrases into (N/2) and (N-1) subgroups and helped them tagged into relevant POS as per the grammar of the Urdu language.

3.1.3 URL-based Urdu Word Extractor

In order to extract Urdu words and phrases present online and in order to calculate the frequency of recurring of these words and phrases, a URL-based Urdu Word Extractor was used. This small tool takes the URL of Urdu blogs or websites and extracts Urdu words and their count in a separate file so that the probability of the recurring of single-words, double-words (bigram) and probability of the use of pertinent POS can easily be calculated.

3.1.4 POS Tags List

As for the categorizing of the Urdu words into respective POS, a well-established POS tag list was required. To overcome this problem, the database of Urdu POS Tagger from Essential Urdu Linguistic Resources was retrieved [10].

3.1.5 Analysis and Refining of Data Collected

The data collected through the aforementioned tools and approaches were then refined and analysed for further improvement. For this purpose, all the words collected were brought in a single file. A thorough search for clearing any duplicate values was done, resulting in a database of Urdu words that have all unique values. The second step taken for refining the collected data was the removal of diacritics. Although diacritics are an essential part of Urdu words, the general public usually doesn't use diacritics in daily works, therefore, words having diacritics such as "Zair, "Zabar", "Paesh" were also removed from the collected database. The third step was the POS tagging of the words according to their definition and context. For this purpose, the words and phrases collected was then run against the POS tag list of Centre of Language Engineering and tagged accordingly.

3.1.6 Collected Data Count

A total of 396024904 Urdu words are collected. For this purpose, around 55000 different sources were used. After extracting, collecting, analysing and refining data, 418195 unique Urdu words are found. Similarly, for double-words or bigram round 500000 double words

with higher recurring frequency were collected (Table 3). These two datasets are the biggest ever collected in any research regarding Urdu Language processing. Around 8000 tagged Urdu words along with their frequency in phrases were collected using the Centre of Language Engineering POS tags database.

Table 3. URDU Words Data Collection

Words	Files	Unique Words	Double Words
396024904	55000	418195	50000

4 Procedure and Algorithm

As the POS tagger is dependent upon two main datasets POS tags collected from the Centre of Language Engineering database and Tag Bigram list created on the calculation of recurring frequencies of POS words in other datasets. The Tag Bigram list was created using a simple algorithm of tokenising each word in the phrases of the input dataset and looping through it to properly mark each word in the phrase with the correct POS tag from the POS tag dataset or Word Tag List (WTag) from Centre of Language Engineering. Each word in the Tag Bigram list was put along with other most used word combinations or on behalf of the recurring frequencies. In order to predict the next word in the phrase, the typed word is searched in the Tag Bigram List and the next word with the highest frequency is predicted. A detailed overview of the POS Tagger Prediction Algorithm is as follows.

Different states and variations describe parallel conditions, activities and system functions that are considered to achieve the primary goal of POS Tagger Prediction are as shown in Figure 1.

We ended up with an optimized version of our algorithm that will fulfil our need of output with the best utilization and artificial intelligence with working behind the screen also to make the intelligence better and better for our front-end program to get more professional experience with this application.

The Diagrammatical view of the optimized version of our Word Predictor algorithm shown in Figure .2 that will fulfils our need of output with the best consumption and artificial intelligence with working behind the screen which we mentioned above.

Algorithm 1. POS Tagger

Step 1. Get the current word and last word POS tag. If it is the first word or the last word is non-alphabetical character like space comma etc the take S0 as the last POS tag.

Step 2. Now search the current word in WTag file and get a list of same word but with multiple different POS tags and their usage frequencies.

Step 3. Now with the last POS tag, run a loop with new current word suggested POS tag list over Tag Bigram file and get a list of combination of previous POS tag and current Word POS tag and get the higher frequency POS Tags combination.

Step 4. Then after getting the higher frequency POS Tags list, use the second POS tag as the predicted POS tag for the current word.

The Diagrammatical view of the optimized version of our Word Suggester algorithm shown in Figure. 3 that will fulfils our need of output with the best consumption and artificial intelligence with working behind the screen which we mentioned above.

5 Results and Discussion

Using the Stochastic Model, n-gram, bigram and POS tagger, this research successfully developed a working Urdu language word and phrase predictor. By integrating the power of artificial intelligence, the proposed too not only make use of the predefined datasets of the Urdu corpus but also develops a runtime dictionary for future use. The use of the POS Tagging approach enhanced the features of the previously developed Urdu word predictor. The ability to autosuggest a number of options for both the word and the phrase in a given context and to choose between them with a single click or keystroke guarantees the feasibility of this tool to be integrated into any portable device.

In a global challenge conducted by [11] the top winner got 45wpm by hitting 226 keystroke that make an average of 5 keystroke to type word as shown in the table below (Table 4).

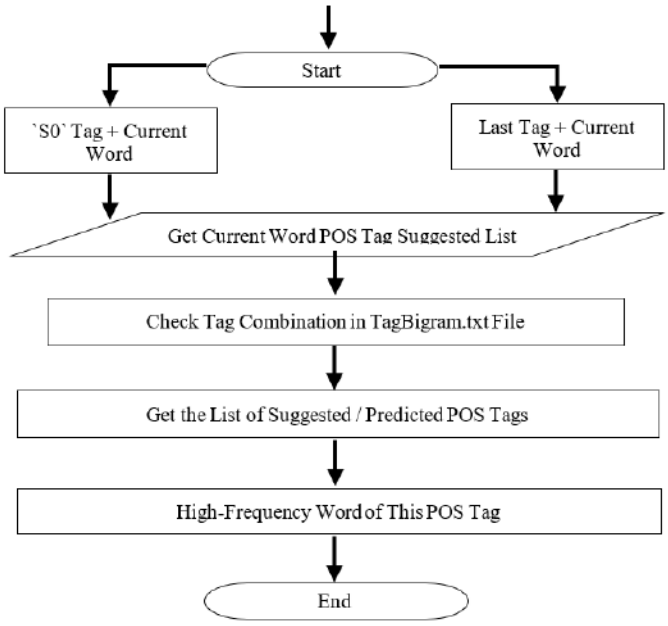

Figure 1. Flow Diagram of Different Aspects of POS Tagger Algorithm

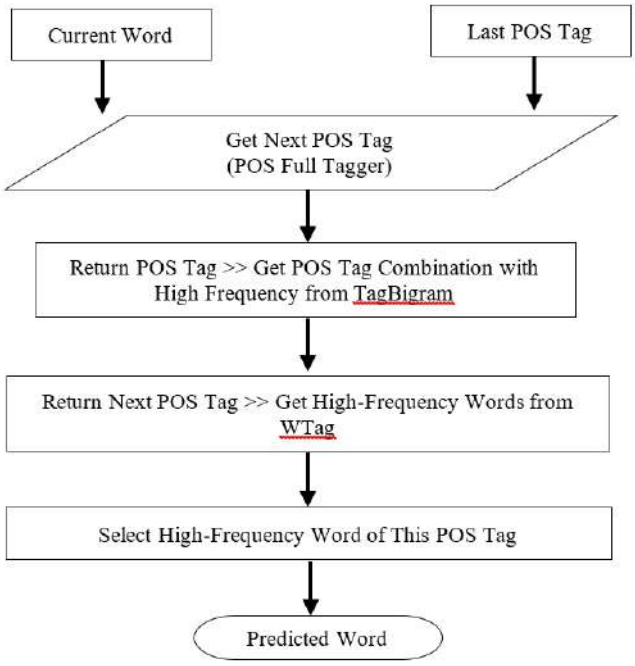

Figure 2. Diagrammatical View of Predictor Algorithm

The results conducted on a closed domain showed an approximate typing speed of 67 wpm achieved by hitting only 152 keystrokes, including the selection from the list. It can be concluded that the typing speed improved by up to 50%, and keystrokes decreased by up to 30%. Notably, this keystroke count corresponds to a 65 wpm test. Additionally, the count of two-keystroke combinations was also reduced.

Algorithm 2. Suggester and Predictor

Step 1. Get the current word and the last word along with their POS tags. If the first word or the last word is a non-alphabetical character like space comma etc. then consider "S0" as the last POS tag.
Step 2. Now search for the current word in WTag and get a list of the same word but with multiple different POS tags and their usage frequencies.
Step 3. Hit POS tagger when user hit a key to type anything.
Step 4. If the last character is the word, then it means the user is still trying to complete the word then get this half type word till the last space or non-alphabetical character.
Step 5. Now hit the WTag to get the list of matched completed words along with POS tags and frequency.
Step 6. With the last POS tag, run a loop with the new current word suggested POS tag list over Tag Bigram file and get a list of all the possible combinations of the previous POS tag and current Word POS tag. Consider the higher frequency POS Tags combination.
Step 7. After getting the higher frequency POS Tags list, use the second POS tag as the predicted POS tag for the current word.
Step 8. Else If the last character is the space, then it means the user has completed the word then get his word.
Step 9. Now run the complete POS Tagger() with the last POS tag and the current word.
Step 10. Get the last word POS tag to hit the Tag Bigram list to get the next word POS tag prediction by matching the last POS tag and the list of predicted current POS tags to get the high-frequency list.
Step 11. Now get the predicted next POS tag list and hit back the WTag to get the words with matching the Tag and top frequency holders.

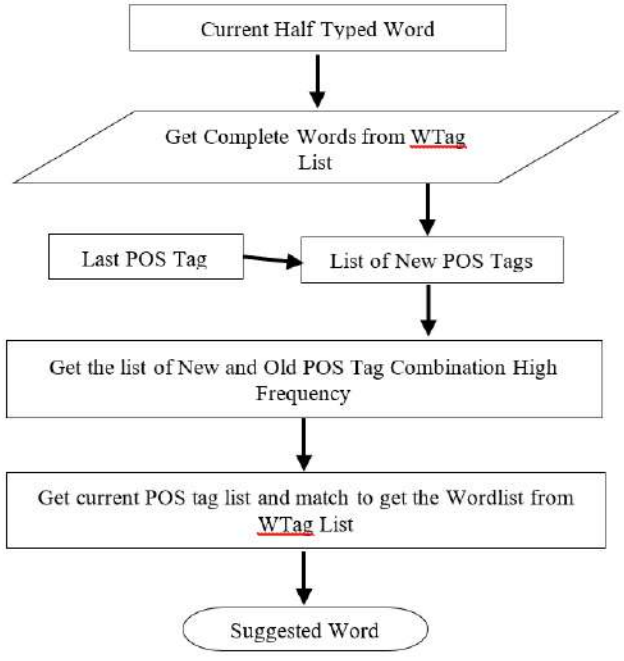

Figure 3. Diagrammatical View of Suggester Algorithm

Table 4. URDU Word Typing Result

Result	WPM	Keystroke	Keystroke Word (Avg)	Result
Without This	45	226	5.02	
With This *	67	152	2.23	50% Fast

*Note: * Based on a continuous text in a close domain.*

Furthermore, the development of .dll (Dynamic Link Library) using C, .jar (Java ARchive) Java and .js (JavaScript) ensure the usability of the tool in almost every platform out there. The lack of support of diacritics and Roman Urdu words, however, decreases the efficiency of the tool and requires immediate implementation of improved statistical approaches and greater Word corpus. Anyhow, the developed tool is surely a good contribution in the augmentation of the use of Urdu language in the IT industry, as it showed greater accuracy in suggesting and predicting words and phrases in a given context.

6 Conclusion

Urdu is one of the most spoken languages of the world that has over 200 million speakers around the globe. The advancement of technology has thus required the adoption of the Urdu language in the IT industry. Unlike other languages, Urdu have a much complex set of grammar and alphabets. This intricacy requires special approaches while developing tools like word and phrase predictors for the Urdu language. With the use of the Stochastic Model, n-gram, bigram and POS tagger and artificial intelligence this research develops an astonishing word and phrase prediction and suggestion tool that increases the typing speed. Although the research and experiments did not contain a huge amount of dataset to cover every grammatical context of the Urdu language, the outcomes of the research are surely overwhelming.

As it is noted that the quality and the quantity of the corpus and well-defined dictionaries employed with the language model made a vivacious impact on the precision of the suggestion and prediction of words, in future, bigger data sets and models such as BERT [15] or other transformer-based [16] models will be used to extend the circle up to Urdu Grammar, Roman Urdu and Urdu Translation. Since the main purpose of this research is to increase the ease of use of the Urdu language in the IT industry, a number of better approaches will also be pursued to bring the Urdu language nearer to other European and East Asian languages.

Author Contributions

Muhammad Hassan: Conceptualization, Methodology, Software. **Saad Ahmed**: Supervision, Data curation, Writing- Original draft preparation. **Rohail Qamar**: Visualization, Investigation. **Saman Hina**: Software, Validation. **Hira Farman**: Writing-Reviewing and Editing

Compliance with Ethical Standards

Declare any potentially competing interests, financial or otherwise see the example It is declared that all authors don't have any conflict of interest. It is also declared that this article does not contain any studies with human participants or animals performed by any of the authors. Furthermore, informed consent was obtained from all individual participants included in the study.

Acknowledgement

Special thanks to the Department of Computer Science & Information Technology, NED University of Engineering and Technology, as they are the primary supporter in terms of technical as well as theoretical provision.

Funding Information

Not applicable.

References

[1] The Editors of Encyclopaedia Britannica, "Urdu Language | History, Script, & Words," *Encyclopaedia Britannica*. [Online]. Available: https://www.britannica.com/topic/Urdu-language.

[2] M. A. Khan, M. A. Khan, and M. N. Ali, "Design of Urdu Virtual Keyboard," presented at the *Conference on Language & Technology*, 2009. Accessed: Oct. 23, 2021. [Online]. Available: https://www.semanticscholar.org/paper/Design-of-Urdu-Virtual-Keyboard-Khan-Khan/d385649378ab0f4ec68535e836fb1226930ce340#paper-header.

[3] S. Shahzadi, B. Fatima, K. Malik, and S. M. Sarwar, "Urdu Word Prediction System for Mobile Phones," *World Applied Sciences Journal*, vol. 22, no. 1, pp. 113–120, 2013, doi: 10.5829/idosi.wasj.2013.22.01.142.

[4] C. Aliprandi, N. Carmignani, P. Mancarella, and L. Pontecorvo, "An Inflected-Sensitive Letter and Word Prediction System," *International Journal of Computing & Information Sciences*, vol. 5, no. 2, pp. 79–85, 2007. Accessed: Oct. 23, 2021.

[5] J. J. Li and A. Nenkova, "Fast and Accurate Prediction of Sentence Specificity," presented at the AAAI Conference on Artificial Intelligence, 2015. Accessed: Oct. 23, 2021. [Online]. Available: https://www.semanticscholar.org/paper/Fast-and-Accurate-Prediction-of-Sentence-Li-Nenkova/69f5a7032605a88e7bed7bf0c9c2218c5e3f2512.

[6] Md. M. Haque, Md. T. Habib, and Md. M. Rahman, "Automated Word Prediction in Bangla Language Using Stochastic Language Models," *International Journal in*

Foundations of Computer Science & Technology, vol. 5, no. 6, pp. 67–75, 2015, doi: 10.5121/ijfcst.2015.5607.

[7] A. Nandi and H. V. Jagdaish, "Effective Phrase Prediction," in *33rd International Conference on Very Large Databases*, 2007, pp. 219–230.

[8] BBC - Languages, "A Guide to Urdu - The Urdu alphabet," *www.bbc.co.uk*, 2014. [Online]. Available: https://www.bbc.co.uk/languages/other/urdu/guide/alphabet.shtml. Accessed: Oct. 23, 2021.

[9] Urdu Dictionary Board, "Urdu Lughat - Published Volumes," *udb.gov.pk*, 2009. [Online]. Available: http://udb.gov.pk/Matbooaat.php. Accessed: Oct. 23, 2021.

[10] Center for Language Engineering, "Urdu Parts of Speech (POS) Tagset," Center for Language Engineering, 2013. Accessed: Oct. 23, 2021. [Online]. Available: https://www.cle.org.pk/Downloads/langproc/UrduPOStagger/Urdu%20POS%20Tagset%200.3.pdf.

[11] URDU Typing Test by 10fastfingers.com. Accessed: Oct. 23, 2021. [Online]. Available: https://10fastfingers.com/typing-test/urdu/top50.

[12] M. Hassan *et al.*, "Effective Word Prediction in Urdu Language Using Stochastic Model," *Sukkur IBA Journal of Computing and Mathematical Sciences*, vol. 2, no. 2, pp. 38–46, Sep. 2018, doi: https://doi.org/10.30537/sjcms.v2i2.304.

[13] N. Mukhtar, M. Abid Khan, N. Chiragh, S. Nazir, and A. Ullah Jan, "An intelligent unsupervised approach for handling context-dependent words in Urdu sentiment analysis," *Transactions on Asian and Low-Resource Language Information Processing*, vol. 21, no. 5, pp. 1–15, 2022.

[14] Data sourced from Ethnologue. https://www.ethnologue.com/insights/ethnologue200/. Accessed: Nov. 15, 2024.

[15] M. V. Koroteev, "BERT: A review of applications in natural language processing and understanding," *arXiv preprint arXiv:2103.11943*, 2021.

[16] A. Vaswani *et al.*, "Attention is all you need," in *Advances in Neural Information Processing Systems*, 2017.

[17] A. Tehseen, T. Ehsan, H. B. Liaqat, A. Ali, and A. Al-Fuqaha, "Neural POS tagging of shahmukhi by using contextualized word representations," *Journal of King Saud University - Computer and Information Sciences*, vol. 35, no. 1, pp. 335–356, Dec. 2022, doi: 10.1016/j.jksuci.2022.12.004.

[18] S. Shaukat, M. Asad, and A. Akram, "Developing an Urdu Lemmatizer Using a Dictionary-Based Lookup Approach," *Applied Sciences*, vol. 13, no. 8, p. 5103, Apr. 2023, doi: 10.3390/app13085103.

[19] L. F. Naz, R. Qamar, R. Asif, M. Imran, and S. Ahmed, "Robot Vision over CosGANs to Enhance Performance with Source-Free Domain Adaptation Using Advanced Loss Function," *Intelligent Automation Soft Computing*, vol. 0, no. 0, pp. 1–10, Jan. 2024, doi: 10.32604/iasc.2024.055074.