RESEARCH ARTICLE

Comparison of Machine Learning and Deep Learning Models for Part-of-Speech Tagging

Aftab Ahmad Khan[1], Wahab Khan[1], Muhammad Alamzeb Khan[1], Khairullah Khan[1], Fida Muhammad Khan[2,*], Atta Ur Rahman[3], Hazrat Bilal[4,*] and Islam Md Monirul[4]

[1] Department of Computer Science, University of Science and Technology Bannu, Bannu 28100, Pakistan
[2] Department of Computer Science, Qurtuba University of Science and Information Technology, Peshawar 25000, Pakistan
[3] Interdisciplinary Research Centers for Finance and Digital Economy, King Fahd University of Petroleum and Minerals (KFUPM), Dhahran, Saudi Arabia
[4] College of Mechatronics and Control Engineering, Shenzhen University, Shenzhen 518060, China

Abstract

The process of assigning grammatical categories, such as "Noun" and "Verb," to every word in a text corpus is known as part-of-speech (POS) tagging. This technique is widely used in applications like sentiment analysis, machine translation, and other linguistic and computational tasks. However, the unique features of the Pashto language and its limited resources present significant challenges for POS tagging. This study explores the critical role of POS tagging in the Pashto language by employing six popular deep-learning and machine-learning techniques. Experimental results demonstrate machine learning methods' effectiveness in capturing Pashto text's grammatical patterns. The evaluation is based on a well-curated and annotated dataset of Pashto text, meticulously compiled from diverse sources and enriched with POS tags, providing a reliable foundation for performance analysis. Among the tested algorithms, K-Nearest Neighbor (KNN) and Decision Tree achieved the highest accuracy rates, with 94.19% and 94.34%, respectively. Random Forest and Support Vector Machine (SVM) also delivered competitive results, exceeding the 90% accuracy threshold. Multi-Layer Perceptron (MLP), evaluated with various activation functions like ReLU and Tanh, achieved an accuracy of 87.25%, while Naïve Bayes, tested with different variants such as Multinomial NB and Gaussian NB, attained 83.33%. These results highlight the potential of machine learning techniques in overcoming the challenges associated with Pashto POS tagging.

Keywords: machine learning, part of speech tagging, morphological structure, grammatical features.

1 Introduction

In 1950, Allan Turing in a seminar proposed the idea that computers can understand Natural

Submitted: 13 March 2024
Accepted: 08 May 2024
Published: 21 June 2024

Vol. 1, **No.** 2, 2024.
10.62762/TACS.2024.493945

***Corresponding authors:**
✉ Fida Muhammad Khan
fida5073@gmail.com
✉ Hazrat Bilal
hbilal@mail.ustc.edu.cn

Citation
Khan, A. A., Khan, W., Khan, M. A., Khan, K., Khan, F. M., Rahman, A. U., Bilal, H., & Monirul, I. M. (2024). Comparison of Machine Learning and Deep Learning Models for Part-of-Speech Tagging. *ICCK Transactions on Advanced Computing and Systems*, 1(2), 106–116.

Languages and can communicate with Humans. From the beginning of the twenty-first century, this vision has begun to take on greater plausibility [1]. Natural language processing (NLP) is a subfield of computer science that involves computer programs that evaluates, synthesizes, and tries to understand one or more human languages. Text, spoken language, and keystroke will be used as input. The basic task of NLP is to translate one language into another, generate summaries, and manage dialogue with users as a part of information retrieval [2]. Some problems that require the processing of Natural Language have different relevance. In sentiment analysis, specific keywords are prioritized, whereas in machine translation, broader contextual words are sufficient [3]. In the present day and the future, NLP systems will undoubtedly play an important role in human-machine and machine-machine communication. NLP's role in advancing telecommunication and computer science cannot be underestimated. Recognition of speech, understanding languages, and language generation are included in the task of NLP [4]. We use NLP in daily life because our computers and smartphones use many software applications that use NLP applications. NLP is the combination of AI, Linguistics, and Statistics, and hence generates human-like responses [5]. Information is constantly produced in the form of Research articles, books, news, and others, and the system stores the information that needs NLP applications to retrieve such information [6]. Natural Language Understanding, which is a subpart of NLP, is divided into subparts, as their morphology, which includes Syntax, semantics, and pragmatics, where morphology defines the creation of words. Syntax means how the words are grouped to form a sentence, semantics refers to the expression of a sentence, and pragmatics stands for how a statement is used in different scenarios [7]. Natural Language Processing provides an interface for the users, where they can interact with the computer in their language instead of learning a special language to give commands to the computer [8].

Part of Speech tagging is the subarea of NLP where the text is divided into words and sentences, and the appropriate Part of Speech tag is assigned to each word [9]. Part of Speech tagging is considered a prerequisite to solving some linguistic problems, which may include Speech-to-text conversion, auto-completion, auto-correction, etc. Tagging words in a given sentence manually is a time-consuming process, but with advancements in NLP, now POS tagging is done automatically by using tagging algorithms [10]. Many words in a language like English are often ambiguous in Part-of-Speech tagging. Hence, a Part-of-Speech tagger is a system that assigns a part-of-speech category to words based on contextual information [11]. Assigning a correct Part of Speech tag to a selected word or sentence of any language and then generating new output based on the tagging data, significant efforts have been made for languages like English, German, Telugu, and Bangla, etc. but very little focus is done for the regional language of Pakistan like Schmid [12]. Part of Speech tagging is essential for Parsing and Grammar correction, etc. If we feed correct and unambiguous data to that system [13], it will perform even better, hence, we cannot ignore the importance of labeled data. The process of POS tagging is classified into three steps, i.e., making tokens, labeling tokens with correct tags, and disambiguation; words in the corpus are separated by using white spaces, and some words may have two or more tags based on morphology [14].

1.1 Pashto Language

Pashto language belongs to the Indo-European languages, merely spoken in Pakistan and Afghanistan [10] and the people speaking Pashto are known as Pashtun or Pukhtun or Pashtun. About 37 million people in Pakistan and Afghanistan speak Pashto, moreover, some small communities in Iran, the United Arab Emirates, the Kingdom of Saudi Arabia, and the United Kingdom also speak Pashto [9]. Pashto is a low-resource language; due to this, a large and organized dataset is difficult to find [15] Researchers take a lot of time in collecting and organizing the dataset of the Pashto language. The Pashto language script is similar to Arabic and Persian and is written from right to left. There is no concept of lower and upper case in the Pashto language, it is not necessary to add a blank space between two just like in English. Sometimes the space insertion may generate errors or an ambiguous word [16].

As we mentioned earlier, Pashto is a Low-resource language; therefore, it did not gain more attention from the researchers, and hence, very little work was done on the Pashto Language. Previously, some researchers worked on the Pashto language by using various statistical and rule-based methods for POS tagging. A rule-based POS tagging is used by [7] which achieves an accuracy of 88%. A Bi-LSTM with CRF used by [9] and achieve 87.60% of accuracy. In this research study,

we focus on using machine learning and deep learning models for POS tagging of the Pashto language. These algorithms include Support Vector Machine, Decision Tree, Random Forest, K-NN, Multi-Layer Perceptron, and Naïve Bayes. The main objective of this research work can be applied to machine translation, sentiment analysis, and Pashto speech recognition. Additionally, Pashto is a low-resource language that is maintained and advanced in the field of computational linguistics.

This paper assesses various Machine Learning and Deep Learning models for POS tagging and offers a well-annotated dataset to alleviate the dearth of resources in Pashto. By enabling applications like sentiment analysis, machine translation, and text-to-speech systems, these contributions can aid in the development of NLP tools for Pashto, which are crucial for the preservation and advancement of the language in digital fields.

The Pashto language has unique grammatical features that make POS tagging challenging, including complex morphology, flexible word order, and the use of diacritical symbols. Additionally, other Pashto dialects add variations in syntax and vocabulary, including Quetta, Yousafzai, Kandahari, and others. These characteristics highlight the importance of this study and make Pashto POS categorization more challenging.

1.2 Challenges

Whenever someone develops a POS tagger for any language, it may face some challenges. The following are some challenges related to the Pashto language.

1.2.1 Ambiguity

Some words in the Pashto language can have different meanings according to sentence structure and context. As shown in Table 1, the Pashto word "کله "means "when," and the other meaning is "Village."

Table 1. Ambiguity example.

کوم	خوب/Sleeping	زه
	I am sleeping	
VB	NN	I-PR

1.2.2 Insufficient resources

Pashto is a low-resource language, and there is a lack of digital resources [17]. Pashto NLP researchers invest a lot of time and resources in data preparation and collection alone, since it is challenging to obtain large-scale corpora and well-structured datasets [15]. The lack of annotated datasets and the scarce application of deep learning and sophisticated machine learning techniques for Pashto point-of-sale tagging serve as research gaps that are addressed in this paper. This gap is notable because it hinders the development of NLP tools for Pashto, a low-resource language with rising computational demands.

1.2.3 Lack of Standardization

As shown in Table 2, Pashto has many dialects, this dialect leads to morphological, syntactical, and spelling variation.

Table 2. Ambiguity example.

اولدو	خوب/Dream	ما
	I saw a dream	
VB	NN	I-PR

1.2.4 Research Questions

The main research issue of this work is: How can machine learning and deep learning techniques be applied to accurately perform part-of-speech tagging for the Pashto language, considering its unique grammatical features and resource limitations? This question was used to assess the effectiveness of six approaches in addressing the challenges associated with Pashto POS tagging. Additional questions include:

1. How accurate and effective are machine learning versus deep learning models for Pashto part-of-speech tagging?
2. Which preprocessing techniques are most effective at handling the unique language traits and script variations of Pashto in POS tagging tasks?
3. Which machine learning or deep learning model is most suited to address the challenges caused by the lack of annotated Pashto datasets for POS tagging?

2 Related Work

Part of speech tagging started in the era of 1950s and 1960s and is performed for many languages, which are known as rich resource languages like English, French, Arabic Chinese, etc. In this article, we focus on PoS tagging for the Pashto language.

The earliest rule-based approach is used by [7], followed the EAGLE guidelines developed a tag

set consisting of 54 different tags, and achieved an accuracy of 88%. Another study was conducted by [7], where the authors introduce the sentence structure of the Pashto language, and describe that the sentence is in the form of Subject-Object-Verb or Object-Subject-Verb. In Pashto, many words have more than one meaning, and need attention, a study conducted by [9] do focus on that problem using CRF-LSTM and HMM models, where they achieve accuracies of 87 and 78% respectively. The neural network model BiLSTM was used by [10] with GloVE embedded model, which shows an accuracy of 97%.

Urdu is a widely spoken and national language of Pakistan. A study conducted by a family [18] achieved 95% accuracy in using SVM for their experiments. HMM with n-gram was used by [19] for the Urdu language and got 95% performance for the model used. A PoS tag is presented by [20] for the Arabic language, based on SVM and BiLSTM, and achieves 75 and 91% accuracy, respectively. The Arabic morphology is rich, and analysis of words can be done through various features like gender, voice, context, and PoS tagging [21], This study achieves accuracies of 97.6% and 98.1%. The study done by [22] for the Persian language used the Hidden Markov Model on homogenous and heterogenous Persian corpora and achieved 98.1% accurate results. The TnT-based HunPoS Hidden Markov Model (HMM) is used by [23] for the Persian language. HunPoS gives an overall accuracy of 96.9%. The study effort [9] uses the Long-Short-Memory (LSTM) Neural Network technology of the deep learning approach to reliably tag words that have several meanings in the sentence; hence, the same word has multiple POS tags. The CRF-BLSTM and HMM models are used by the authors to train the LSTM Neural Network and confirm its accuracy. They get 78% accuracy for HMM and 87% accuracy for CRF-BLSTM.

The literature study reveals the importance of POS tagging for Pashto Language in NLP and will be used in many applications like sentiment analysis and machine translation, etc., as Pashto has a rich morphology, various dialects, and less availability of well-annotated datasets, making the POS task challenging. This research will help to address the issues.

Part of speech tagging is important for machine translation and many more applications related to NLP; this research study is focusing on the development of a POS tagset for Urdu.

3 Methodology

The proposed methodology is composed of several machine learning and deep learning models to achieve the task of Part of Speech tagging for the Pashto language, the machine learning models we use in my experiments are Support Vector Machine (SVM), Random Forest (RF), Decision Tree (DT), K-Nearest Neighbor (KNN) classifier, Naïve Bayes and the deep learning model Multi-layer Perceptron (MLP). For more convenient and efficient results, two distance methods of the KNN classifier are used, which are the Jaccard and Euclidean distance methods. in Multi-Layer Perceptron, four activation functions, namely Tanh, Identity, ReLu, and Sigmoid, are included in these experiments, and four different variants of Naïve Bayes algorithm, ComplementNB, MultinomialNB, BernoulliNB, and GaussianNB, are also tested in the said experiments.

The following methods were picked because of their shown efficacy in text categorization tasks. While MLP with different activation functions was utilized to investigate nonlinear correlations in the data, SVM, RF, and DT are recognized for managing challenging decision limits. Because of their ease of use and effectiveness when working with small, low-resource datasets like Pashto, KNN and Naive Bayes (with its four variations) were included.

The unique contribution of this research study is to develop a well-annotated dataset, applying six different machine learning and deep learning models with their different variants and activation methods, and a comparison of these models.

3.1 Support Vector Machine (SVM)

Support Vector Machine (SVM) is used to find the optimal hyperplane that separates the data items in classes. SVM is suitable for classification and regression problems. In the case of the dataset, the item $\{(x_1, y_1), (x_2, y_2) \ldots (x_n, y_n)\}$ where X represents the feature set in the dataset and Y is the label.

3.2 K-Nearest Neighbor (KNN)

K-Nearest Neighbor (KNN) is a supervised machine learning algorithm that can be used for classification and Regression problems. KNN makes predictions based on similarity between data points. For classification problems, KNN assigns the most relevant label among the neighbors. For regression, it assigns the target by averaging the values. Different distance methods are used to calculate the distance between the neighbors, including:

3.2.1 Euclidean Distance

The Euclidean Distance method is a well-known distance method in geometry. This method is used to find the straight-line distance between the two data points in the Cartesian system. This method is most commonly used in KNN. If we have two data points (X_1, Y_1 and X_2, Y_2) in a 2D space, then the Euclidean distance is calculated as:

$$\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \tag{1}$$

3.2.2 Jaccard Distance method

The Jaccard Distance method in mathematics is used to find the dissimilarity between/Among the sets, and the method is based on the Jaccard Coefficient. This method uses distance ranges that are 0 and 1. If the distance range is 0, it indicates that the sets are identical; if the range is 1, it indicates that there is nothing common between the sets. The Jaccard method is also used in KNN for similarity measurement, it is used with categorical or binary data. For two data points, the Jaccard method is calculated as:

$$\text{Jaccard Distance} = \frac{A \cap B}{A \cup B} \tag{2}$$

This method is also extended to three, four, and other high dimensions. If we have three data points i.e, A, B, and C, then the Jaccard distance will be calculated as:

$$\text{Jaccard distance for AB} = \frac{A \cap B}{A \cup B} \tag{3}$$

$$\text{Jaccard distance for AC} = \frac{A \cap C}{A \cup C} \tag{4}$$

3.3 Decision Tree

A Decision Tree is a supervised Machine learning approach used for classification and regression problems, but most commonly, a Decision Tree is used for classification problems. It's called a decision tree because it takes conditions, and based on the condition, it grows like a tree, where Decision nodes and Leaf nodes are generated. The decision nodes take some condition and generate leaf nodes, leaf nodes represent the output of the tree. To select the most appropriate feature for the decision tree well well-known methods are used, namely Gini Impurity, Entropy, and Information Gain (IG).

3.4 Random Forest

Random Forest is another popular Machine learning algorithm; it's based on the Decision Tree with the introduction of randomness to improve the performance and decrease overfitting problems. Random Forest is a versatile algorithm and has many application areas, including health care, image identification and recognition, finance, etc. It combines the output of many decision trees and generates a single output, hence called a Random Forest. The accuracy of this model increases when the number of decision trees increases and prevents overfitting.

3.5 Multi-Layer Perceptron (MLP)

Multi-Layer Perceptron (MLP) is a neural network that comes into the category of deep learning algorithms. MLP is considered the base or building block for many other artificial neural networks and can be used for classification, pattern recognition, and regression problems. The structure of an MLP consists of nodes that may be arranged in multiple layers. These layers are divided into three categories, i.e., the Input layer, hidden layers, and Output layer. The data is fed to the input layer of the Network with some weights and bias values, which may be passed to the hidden layer (which may be one or more in number), and finally, the output is displayed on the output layer. Two methods, forward and Backward propagation, are used to train the model. In forward propagation, the data is fed to the network through the input layer, and the output is generated layer by layer. In backward propagation, the error is calculated backward, and to remove the error, the Weight and bias values are modified. Different activation functions are used in MLP for processing the input.

3.5.1 ReLU Activation Function

ReLU is among the most often used activation functions. It has the following formula:

$$f(m) = \max(0, m) \tag{5}$$

Stated differently, it outputs the input value if it is negative and zero otherwise. The "dying ReLU" problem, in which neurons can become stuck during training and cease updating their weights if they persistently produce negative values, can occur even if the ReLU function adds non-linearity.

3.5.2 Tanh Activation Function

Another popular activation function that shifts input values into the $[-1, 1]$ range is Tanh (Hyperbolic). The

formula:

$$f(Y) = \frac{e^{Y} - e^{-Y}}{e^{Y} + e^{-Y}} \tag{6}$$

Tanh has the advantage of being zero-centered and introduces non-linearity, both of which can help with model optimization and mitigate some of the problems with ReLU.

3.5.3 *Sigmoid (Logistic) Function*

The sigmoid activation function assigns input values to a range $(0, 1)$. It may be calculated using the formula

$$f(c) = \frac{1}{1 + e^{-c}} \tag{7}$$

Despite being widely used in the past, the sigmoid function has many disadvantages, such as disappearing gradients, which can make training deep networks difficult.

3.6 Naïve Bayes

Naïve Bayes is a popular machine learning algorithm, based on the Bayes Theorem of probability. It is called "naïve" because it considers that all the features should be independent. This algorithm is widely used for classification problems due to its efficiency and simplicity. It's also known as a Probabilistic classifier. Other variants of Naïve Bayes classifier are Multinomial Naïve Bayes, Gaussian Naïve Bayes, Bernoulli Naïve Bayes, and Complement Naïve Bayes. The Naïve Bayes algorithm is based on Bayes' theorem of probability, mathematically expressed as

$$P(H|E) = \frac{P(E|H)^{*}P(H)}{P(E)} \tag{8}$$

3.7 Proposed Dataset

A carefully annotated dataset with words matched to their POS tags is needed by academics to perform part-of-speech tagging efficiently. Because there isn't a standardized dataset for the Pashto language, we created PashtoPoSTags, our dataset that we annotated with the help of Pashto language specialists by combining data from several sources. To make it easier to apply Machine Learning algorithms correctly, we then converted the dataset into CoNLL format. Sentences are indicated by blank lines to indicate gaps between sentences in the CoNLL format, which divides the data into two columns: one for Pashto words and the other for their matching POS tags. The data was put through several preprocessing steps to ensure its quality and fit for the models. Tokenization was used to separate the text into discrete words, normalization was employed to manage variations in Pashto script, and missing values were dealt with by removing incomplete entries. The dataset was ensured to be clean and ready for analysis by taking these steps.

3.7.1 *Data Pre-processing*

Before using data for the experiment, the data must be pre-processed and prepared for further processing. The following pre-processing steps are involved. The dataset is presented in both TXT and CSV formats, as shown in Figure 1.

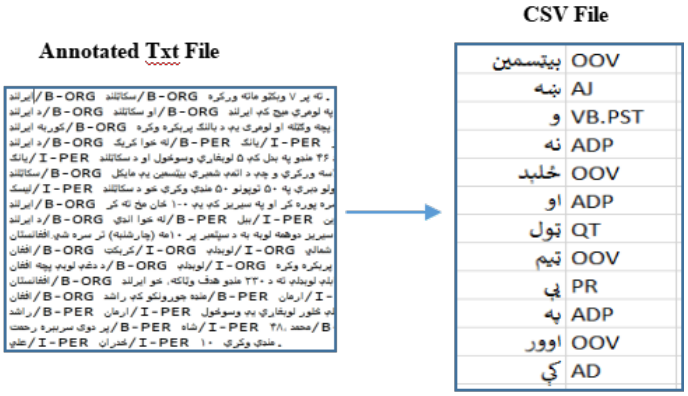

Figure 1. The dataset in TXT and CSV file formats.

Character Encoding The dataset we built consists of words and their corresponding POS tags, which are alphabets, numbers, and symbols, converted into Binary (stream of 0's and 1's) to understand by a computer. For this, we use the "UTF-8" character encoding technique. Table 3 illustrates the symbols excluded from the dataset.

Data cleaning Removing unnecessary symbols and letters is obligatory because it can cause ineffective results, as we arrange the dataset in CoNLL format, hence blank spaces are there in the dataset as well, which are known as NaN values in Python. So, the following symbols that are not part of the language are removed during this step. As shown in Table 4, the unnecessary symbols and characters are removed during the data cleaning process to ensure the dataset is properly formatted for further analysis.

Dataset Splitting The dataset was divided into training, validation, and testing sets in a standard 70/15/15 ratio to ensure accurate model evaluation. Accuracy was the primary evaluation factor used to compare model performances consistently across all

Table 3. Symbols excluded from the dataset.

Punctuation Marks	Special Characters	Other Symbols
Question Mark (?)	Semicolon (;)	Ellipsis (...)
Exclamation Mark (!)	Hyphen (-)	Ampersand (&)
Comma (,)	Apostrophe (')	Dollar Sign ($)
Period/Full Stop (.)	Quotation Marks (" ")	Percent Sign (%)
Colon (:)	Parentheses ( )	Plus Sign (+)
Minus Sign (-)	Equal Sign (=)	Greater Than (>) and Less Than (<) Signs
At Sign (@)	Asterisk (*)	And others

Table 4. Example of data cleaning.

Original Sentence	Cleaned Sentence	Tokenized Sentence	POS Tags
زه کور ته ځم	زه کور ته ځم	زه، کور، ته، ځم	[PR, NN, ADP, VB]

experiments. These steps ensured that the models were trained and evaluated systematically.

Lexicon/Lexemes/tokenization. After data cleaning, the dataset is split into tokens. in training, each token consists of a single word with a corresponding POS tag. When testing, the word is checked by the machine learning algorithms and assigned an appropriate POS tag.

"PashtoPoSTags" is a CoNLL-formatted dataset with two columns: one for words and another for the associated POS tags. To guarantee linguistic diversity, the dataset includes 32000 samples from various sources, such as news articles, books, and social media posts. For reproducibility and ease of integration with POS tagging techniques, this structure provides a standard format.

4 Experiments

The models mentioned in section 3 are applied by conducting the following pattern for experiments. The dataset we prepared for the experiment, named "PashtoPoSTags," is used in our experiments. To feed data into a deep learning or machine learning model, the dataset must be split into separate training and testing subsets. Within this framework, the PashtoPoSTags dataset is partitioned so that 75% of the data is set aside for training, 15% is set aside for testing purposes, and the last 15% is for validation.

Among all the methods we examined in our trials, the Decision Tree machine learning model produced the best accuracy of 94.34% when we used the dataset we assembled as input. This result highlights how well Decision Trees can identify trends in data, which makes it easier to classify POS tags for Pashto words in an effective manner.

In this experiment, we look at two distance approaches that provide two distinct accuracies, and the KNN algorithm demonstrates robustness. While the Euclidean Distance approach yields an accuracy of 93.97%, KNN with the Jaccard Distance method generates an accuracy of 94.14%. The selection of nearest neighbors has a big impact on the KNN algorithm's output; in certain situations, selecting fewer nearest neighbors yields good results, while in other situations, selecting more nearest neighbors may yield good results. We use a random selection process to see how the KNN reacts in my experiments. we noticed that when the number of nearest neighbors is changed, the KNN yields varied results. In the end, we choose to apply the KNN algorithm for both distance approaches at the K=3 number since the KNN performs well at this K value. Figure 2 illustrates the comparison between Decision Tree and K-Nearest Neighbor (KNN) models.

The method used to estimate the distance between the data points is another important factor that could influence the result. This distance can be measured using many techniques, such as the Jaccard, Manhattan, Cosine, and Euclidean distance approaches, as we indicated in the Introduction section. Every approach for measuring distance has a unique outcome. In these investigations, we used two distance methods: the Euclidean and the Jaccard, and each yielded result that were different from the other.

While the SVM produces results with an accuracy of 90.00%, the Random Forest tests yield results with an accuracy of 90.28%. Even while Decision Tree

Table 5. Details of feature analysis for different machine learning models.

Model	Feature Analyzes	Detail
KNN	N-Gram, Euclidean Distance	K=5, Euclidean Distance method
Decision Tree	N-Gram, Hierarchical split	Gini Index
SVM	Word Embedding, Hyperplanes	Kernel: SVC, Regularization parameter (C): 1.0
Random Forest	Ensemble learning method	Entropy
MLP	Activation Functions	Relu, Tanh, Sigmoid, Identity. Two layers 50, 100
Naïve Bayes	N-Gram	Variants include: Gaussian, Bernoulli etc.

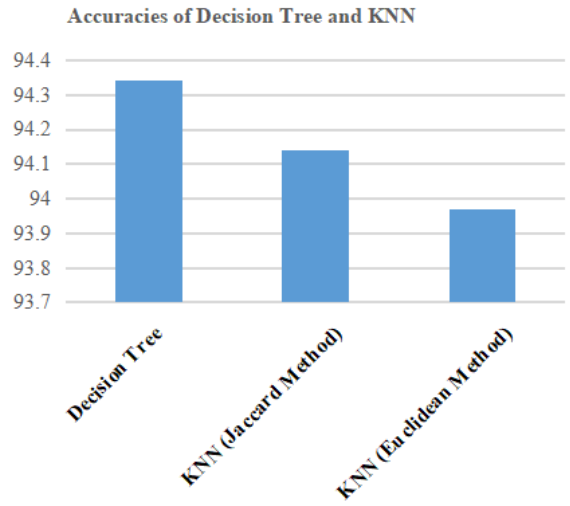

Figure 2. Comparison of Decision Tree & KNN.

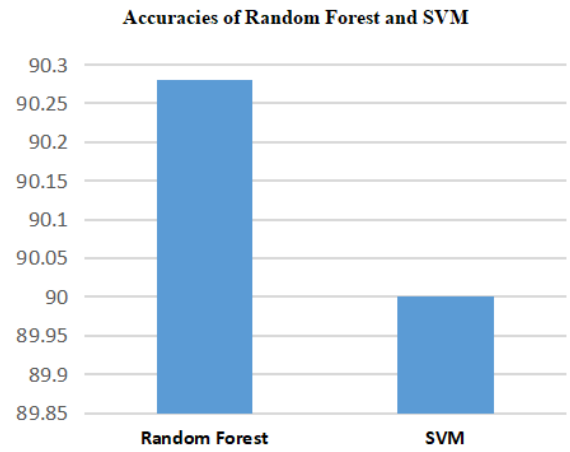

Figure 3. Comparison of Random Forest & SVM.

and KNN (with both distance methods) have slightly higher accuracy than RF and SVM, these algorithms nevertheless provide competitive performance for Pashto language POS tagging. While Random Forest blends several decision trees to get the output, making it a useful method for some applications, SVM is good at distinguishing data points in high-dimensional space. Figure 3 shows the comparison of Random Forest and Support Vector Machine (SVM) in terms of accuracy.

The accuracy of the Multilayer Perceptron (MLP) varies based on the activation functions that are used. Popular activation functions include Tanh, identity, sigmoid, and ReLU, and various accuracies are obtained. More specifically, the accuracy is 86.65% when using the sigmoid and ReLU activation functions. With the Identity activation function, the accuracy is 86.99%, which is marginally higher. Out of all the activation functions studied, the Tanh function yields the highest accuracy of 87.25%.

The importance of choosing the right activation function for maximizing the Multilayer Perceptron's (MLP) performance is highlighted by these results. Tanh is the most appropriate option among the tested activation functions for improving accuracy in this POS tagging activity. This proves that activation functions are essential for optimizing the performance of the MLP for certain uses, like part-of-speech tagging. Determining the Multilayer Perceptron's (MLP) hidden layer configuration, that is, how many layers and how many neurons are in each layer, is a crucial step in optimizing the MLP. I chose a design with two hidden layers for our investigation, the first with 100 neurons and the second with 50 neurons. It's important to note that changing these settings has a substantial impact on both the algorithm's overall accuracy and the effectiveness of each activation function used. Through a series of studies examining various hidden layer configurations and neuron counts, we concluded that the best performance may be achieved by using two hidden layers, each with 100 and 50 neurons.

The MLP model was trained with a batch size of 32, which is commonly chosen to balance computational efficiency and model generalization. The Pashto dataset's modest size was effectively handled by this batch size, which also ensured steady gradient updates throughout training. This specific feature shows how careful consideration was given to enhancing model performance. Figure 4 compares the performance of

different activation functions used in the Multi-Layer Perceptron (MLP) model.

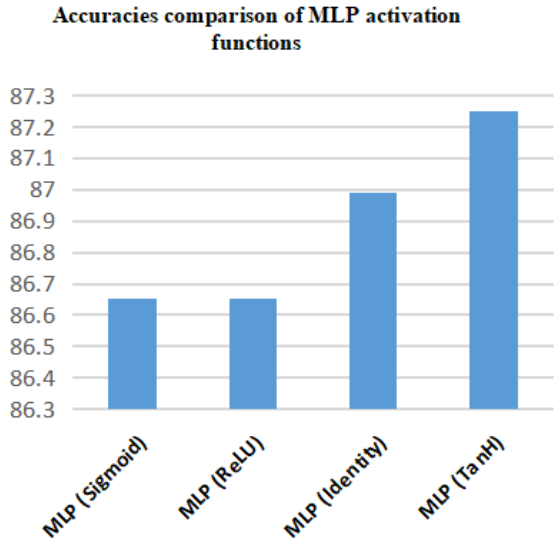

Figure 4. Comparison of MLP activation functions.

There exist numerous iterations of the Naive Bayes algorithm, and their respective accuracy rates vary based on the given circumstances. we employed several different variations in my experiment, such as Gaussian Naive Bayes (GaussianNB), Bernoulli Naive Bayes (BernoulliNB), Complement Naive Bayes (ComplementNB), and Multinomial Naive Bayes (MultinomialNB). In contrast, the accuracy produced by Multinomial Naive Bayes is 80.33%; Gaussian Naive Bayes is 63.98%; Complement Naive Bayes is 83.96%; and Bernoulli Naive Bayes is 73.45% of the course. These results indicate that using a different form of the Naïve Bayes algorithm for the same example can yield different results. Figure 5 compares the performance of Naive Bayes and its different variants.

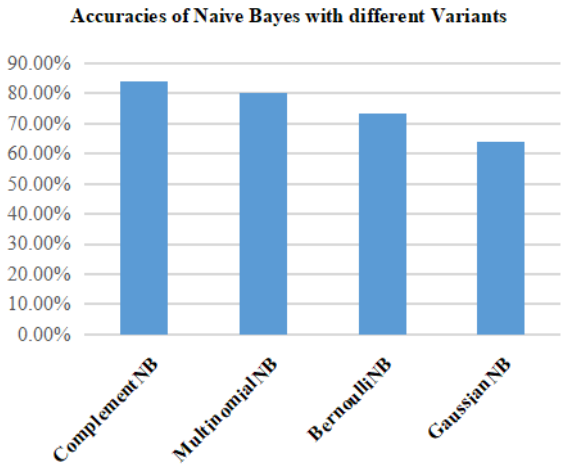

Figure 5. Comparison of Naive Bayes and its different variants.

The underlying assumptions of each version and the characteristics of the dataset are what cause the variations in the accuracy of Naive Bayes variants. The assumption of continuous data and a normal distribution in Gaussian Naive Bayes may not be well adapted to the dataset's discrete word-based characteristics. On the other hand, because Complement Naive Bayes is designed for imbalanced datasets, it is more ideal for controlling the word distributions in Pashto POS tagging.

The Naive Bayes algorithm's results demonstrate how versatile it is, as different variants produce different results on the same dataset. Despite this flexibility, it performs the worst out of the six tested algorithms. This demonstrates that the algorithm can handle a wide range of data distributions, but it also shows that it is not as effective as other approaches in this specific experiment. As shown in Table 5, the different machine learning models used in this study vary in their feature analysis methods and configuration settings, such as the KNN's use of Euclidean distance and Decision Tree's reliance on Gini Index.

5 Conclusion

In this research study, we examine different machine learning and deep learning algorithms for Pashto language Part of Speech Tagging (POST), which are Decision Tree, Support Vector Machine, Random Forest, Multi-Layer Perceptron with different activation functions like Tanh, Sigmoid, ReLu and Identity, K-Nearest Neighbor by examining two distance methods Jaccard and Euclidean and Naïve Bayes with its four different variants. Among all these algorithms, we assess Decision Tree and KNN with the Jaccard and Euclidean methods, generating the highest accuracies of 94.34%, 94.19%, and 93.9%, respectively. However, this research needs more focus on some parameters, such as the size and diversity of the dataset, and implementing deep learning and ensemble methods, which may produce different results in this case. KNN and Decision Trees performed better because they can manage non-linear decision boundaries and successfully adjust to the structure of the Pashto dataset. Naïve Bayes performed badly because its strong independence assumptions are not as well adapted to the complex relationships present in language data. Notwithstanding its power, MLP's accuracy was reduced due to the tiny quantity of the dataset, which may have limited its ability to fully comprehend the patterns in Pashto POS tagging. Future research might include including such an analysis or displaying the decision tree structure to

enhance the results' interpretability. This could offer valuable insights into how the model gives priority to particular language features.

Data Availability Statement

Data will be made available on request.

Funding

This work was supported without any funding.

Conflicts of Interest

The authors declare no conflicts of interest.

Ethical Approval and Consent to Participate

Not applicable.

References

[1] Kibble, R. (2013). Introduction to natural language processing. *London: University of London*.

[2] Ballan, L. (2003). Natural language processing.

[3] Galassi, A., Lippi, M., & Torroni, P. (2020). Attention in natural language processing. *IEEE transactions on neural networks and learning systems*, *32*(10), 4291-4308. [Crossref]

[4] Joshi, A. K. (1991). Natural language processing. *Science*, *253*(5025), 1242-1249. [Crossref]

[5] Zaman, F., Maqbool, O., & Kanwal, J. (2024). Leveraging bidirectional lstm with crfs for pashto tagging. *ACM Transactions on Asian and Low-Resource Language Information Processing*, *23*(4), 1-17. [Crossref]

[6] Fanni, S. C., Febi, M., Aghakhanyan, G., & Neri, E. (2023). Natural language processing. In *Introduction to artificial intelligence* (pp. 87-99). Cham: Springer International Publishing. [Crossref]

[7] Chopra, A., Prashar, A., & Sain, C. (2013). Natural language processing. *International journal of technology enhancements and emerging engineering research*, *1*(4), 131-134.

[8] Mihalcea, R., Liu, H., & Lieberman, H. (2006, February). NLP (natural language processing) for NLP (natural language programming). In *International Conference on intelligent text processing and computational linguistics* (pp. 319-330). Berlin, Heidelberg: Springer Berlin Heidelberg. [Crossref]

[9] Haq, I., Qiu, W., Guo, J., & Peng, T. (2023). The Pashto corpus and machine learning model for automatic POS tagging. [Crossref]

[10] Haq, I., Qiu, W., Guo, J., & Tang, P. (2023). NLPashto: NLP toolkit for low-resource Pashto language. *International Journal of Advanced Computer Science and Applications*, *14*(6). [Crossref]

[11] Khan, H. A., Ali, M. J., & Hanni, U. E. (2020, November). Poster: A novel approach for pos tagging of pashto language. In *2020 First International Conference of Smart Systems and Emerging Technologies (SMARTTECH)* (pp. 259-260). IEEE. [Crossref]

[12] Schmid, H. (1994). Part-of-speech tagging with neural networks. *arXiv preprint cmp-lg/9410018*. [Crossref]

[13] Rajper, R. A., Rajper, S., Maitlo, A., & Nabi, G. (2021). Analysis and comparative study of POS tagging techniques for national (Urdu) language and other regional languages of pakistan. *SINDH UNIVERSITY RESEARCH JOURNAL (SCIENCE SERIES)*, *53*(04).

[14] Naz, F., Anwar, W., Bajwa, U. I., & Munir, E. U. (2012). Urdu part of speech tagging using transformation based error driven learning. *World Applied Sciences Journal*, *16*(3), 437-448.

[15] Khanam, M. H., & Murthy, K. M. (2014). Part-of-speech tagging of urdu in limited resources scenario. *International Journal on Recent and Innovation Trends in Computing and Communication*, *2*(10), 3280-3285.

[16] Rabbi, I., Khan, A. M., & Ali, R. (2009). Rule-based part of speech tagging for Pashto language. In *Conference on Language and Technology, Lahore, Pakistan*.

[17] Rabbi, I., Khan, M. A., Ahmad, R., & Ali, R. (2016). Theoretical Analysis of Pashto Phrases for the Creation of Parser.

[18] Alharbi, R., Magdy, W., Darwish, K., AbdelAli, A., & Mubarak, H. (2018, May). Part-of-speech tagging for Arabic Gulf dialect using Bi-LSTM. In *Proceedings of the eleventh international conference on language resources and evaluation (LREC 2018)*.

[19] Sajjad, H. (2007). Statistical part of speech tagger for Urdu. *Unpublished MS Thesis, National University of Computer and Emerging Sciences, Lahore, Pakistan*.

[20] Anwar, W., Wang, X., Li, L., & Wang, X. L. (2007, August). A statistical based part of speech tagger for Urdu language. In *2007 international conference on machine learning and cybernetics* (Vol. 6, pp. 3418-3424). IEEE. [Crossref]

[21] Habash, N., & Rambow, O. (2005, June). Arabic tokenization, part-of-speech tagging and morphological disambiguation in one fell swoop. In *Proceedings of the 43rd annual meeting of the association for computational linguistics (ACL'05)* (pp. 573-580).

[22] Okhovvat, M., & Bidgoli, B. M. (2011). A hidden Markov model for Persian part-of-speech tagging. *Procedia Computer Science*, *3*, 977-981. [Crossref]

[23] Seraji, M. (2011). A statistical part-of-speech tagger for Persian. In *NODALIDA 2011, Riga, Latvia, May 11–13, 2011* (pp. 340-343).

Aftab Ahmad Khan is a computer science lecturer at the Government Degree College Mamash Khel. He graduated from the University of Science & Technology, Bannu, with a Bachelor of Science in Computer Science and an MS in Artificial Intelligence and Data Science. His research interests include Data science, machine learning, and deep learning. (Email: aftabaak7@gmail.com)

Dr. Wahab Khan is a Lecturer of Computer Science, at the University of Science & Technology, Bannu. He did his PhD at the International Islamic University Islamabad. His areas of Interest are Deep Learning and Natural Language Processing, he published many research articles in well-reputed journals and is a member of IEEE-Access. (Email: wahabshri@gmail.com)

Muhammad Alamzeb Khan received his B.Sc Electrical (Telecommunication) Engineering degree from the University of Science & Technology Bannu, Pakistan, in 2018. His research project was "Ultrasonic Blind Walking Stick" for visually disabled people using sensors to detect obstacles, water and alerts ahead. He completed his master degree from department of computer science, University of Science and Technology Bannu in 2023. His research interests include Machine learning, Deep learning, NLP, IoT, network security, cloud computing and big data. (E-mail: alamzebkhan07@gmail.com)

Khairullah Khan received the Ph.D. degree in information technology from University Teknologi PETRONAS, Malaysia, in 2012, where he worked on machine learning for the automatic detection of opinion targets from text. He is currently Professor with the Department of Computer Science, University of Science and Technology, Bannu, Pakistan. (Email: khair@ustb.edu.pk)

Fida Muhammad Khan is pursuing a Ph.D degree in Computer Science at Qurtuba University of Science and Information Technology, Peshawar, Pakistan. He did his MS in Computer Science at the University of Science and Technology, Bannu, Pakistan. His research interests include Data Mining, Cybersecurity, Machine Learning, Deep Learning, Blockchain and Natural Language Processing (NLP). (Email: fida5073@gmail.com)

Atta Ur Rahman received the MS degree in Computer Science from the University of Science and Technology Bannu in 2018, and the Ph.D. degree in Computer Science from Ghulam Ishaq Khan Institute of Engineering Sciences and Technology (GIKI) in 2022. He worked as an Assistant Professor from (2023-2024) at Riphah Institute of System Engineering (RISE), Riphah International University Islamabad, Pakistan. Currently, he joined King Fahd University of Petroleum and Minerals, Saudi Arabia, as a Postdoctoral Researcher. He has more than 20 publications in various reputed journals and conferences including IEEE Transactions. His research interest includes Human-computer Interaction, Artificial Intelligence in healthcare, and Federated learning for privacy preserving. (Email: attaur.rahman@kfupm.edu.sa)

Hazrat Bilal received his MS degree in Control Science and Engineering in 2018 from Nanjing University of Science and Technology, Nanjing, China, and his PhD degree in Control Science and Engineering from the University of Science and Technology of China, Hefei, Anhui, respectively. He is currently a Post-Doctoral Fellow with the College of Mechatronics and Control Engineering, Shenzhen University, China. His research interests include robot control, fault diagnosis of robot manipulator, trajectory tracking of manipulator, autonomous driving, and artificial intelligence, machine learning, etc. (Email: hbilal@mail.ustc.edu.cn)

Islam Md Monirul received the M.E. degree in Information and Communication Engineering from the Southwest University of Science and Technology, Mianyang, China, in 2021. He is currently pursuing the Ph.D. degree in Optomechatronics Engineering and Applications with the College of Mechatronics and Control Engineering, Shenzhen University, Shenzhen, China. His research interests include battery modeling, lithium-ion batteries, energy storage system, renewable energy, networked control systems. (Email: islammdmonirul@email.szu.edu.cn)