Urdu Parts of Speech Tagging

Department of Computer Science

Muhammad Bilal	**BSCS2019-19**
Kiran Zafar	**BSCS2019-36**
Hurmat Ilyas	**BSCS 2019-02**

Final Year

Final year project report submitted in partial fulfillment of requirement for degree of Bachelors of Science in Computer Science

Namal Institute, 30-KM, Talagang Road, Mianwali, Pakistan.
www.namal.edu.pk

DECLARATION

The project report titled “Urdu parts of speech tagging” is submitted in partial fulfillment of the degree of Bachelors of Science in Computer Science, to the Department of Computer Science at Namal Institute, Mianwali, Pakistan.

It is declared that this is an original work done by the team members listed below, under the guidance of our supervisor “Dr. Bacha Rehman & Dr. Murtaza Khan”. No part of this project and its report is plagiarized from anywhere, and any help taken from previous work is cited properly.

No part of the work reported here is submitted in fulfillment of requirements for any other degree/ qualification in any institute of learning.

Team Members	**University ID**	**Signatures**
Muhammad Bilal	NIM-BSCS2019-19	_____________________
Kiran Zafar	NIM-BSCS2019-36	_____________________
Humat Ilyas	NIM-BSCS2019-02	_____________________

Supervisor

Dr. Bacha Rehman & Dr. Murtaza Khan

Signatures with date

____

____

____

ACKNOWLEDGMENTS

All praises belong to ALLAH (SWT), who is merciful and Lord of the day of judgment. Without His help, even the most straightforward task is not possible.We would like to thank our supervisors, Dr. Bacha Rehman & Dr. Murtaza Khan, for providing us with invaluable guidance and support throughout the entire process. Their expertise and insights were instrumental in shaping this project and making it a success.

We would also like to thank them for their valuable feedback and contributions to the research. Their input and ideas helped us to develop a more comprehensive understanding of the topic and refine our analysis.

Furthermore, we extend our heartfelt thanks to everyone who has contributed to this project in any way. Their help and support have been instrumental in the successful completion of this project.

Abstract

This report presents the development of a parts of speech (POS) tagging system for the Urdu language. The system uses a combination of rule-based and machine learning approaches to automatically assign tags to each word in a given text. A large corpus of Urdu text was collected and annotated with POS tags to train and evaluate the system. Various machine learning algorithms were experimented with, including Hidden Markov Models (HMMs), Active Learning using Neural Network, and XLM Roberta base transformer model. The best-performing model was found to be a XLM Roberta base transformer model. The report also discusses the challenges faced in developing a POS tagging system for Urdu, including the lack of annotated data and the complexity of the language's morphology. The developed system can be used in various natural language processing applications, such as information retrieval, machine translation, and text classification, for the Urdu language.

Table of Contents

List of Figures

List of Tables

Chapter 1

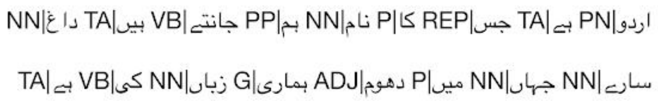
Introduction

Natural language processing is a challenging research area with interesting problems. NLP-related tasks involve parts of speech tagging, developing word embeddings, etc. whereas its applications revolve around translation, intent classification, sentiment analysis, etc. Therefore, there is a requirement for both developments of basic processes and applications involving NLP.

On the other hand, Pakistan has the 5th largest population in the world and hence the number of people associated with a regional language is significantly large. Urdu is the national language of Pakistan. Therefore, it is important to develop a natural system capable of understanding Urdu. Some of the applications of POS Tagger are:

- **Word Sense Disambiguation:** Identifying the ”sense” (meaning) in which a word appears in a sentence is a challenge because the context affects how words are understood.
- **Grammar Checking:** Tagging parts of speech is also done in checking grammar. The input is first broken down into sentences, and each sentence is then tagged using POS.

The method of identifying parts of speech to the target text is known as part-of-speech (PoS) tagging. It is frequently referred to as POS tagging. To put it simply, part of speech (POS) tagging is the process of assigning the correct part of speech for every word in a phrase. We already know that the parts of speech consist of nouns, verbs, adverbs, adjectives, pronouns, conjunction, and their subcategories. It will include POS tagging from an Urdu sentence. For example



اردو|PN ہے|TA جس|REP کا|P نام|NN ہم|PP جانتے|VB ہیں|TA دا غ|NN

سارے|NN جہاں|NN میں|P دھوم|ADJ ہماری|G زباں|NN کی|VB ہے|TA

The suggested system would be able to intelligently analyze the Urdu sentences and generate POS tags. The proposed project will focus on gathering text data, labeling parts of speech, and training a system to recognize parts of speech in Urdu text. Project objectives will be:

- Acquiring a total of 20,000 tokens for use in system training.
- Tokens that have been annotated or tagged for reference.
- The creation of a deep learning-based strategy for part-of-speech tagging.

The following steps are involved in our tentative plan to fulfill the above goals. In chapter 3 we will be working on

- **Text Analysis using NLP**
  We will use natural language processing techniques to examine and collect features of the text in order to determine its part of speech.

- **Dataset Collection/Curation**
  Words and their related parts of speech from the Urdu language will be included in the dataset. If such a dataset with labels already exists, we will use it to train our models. If necessary, we will manually curate our dataset.

- **Exploring Language Models**
  We will be exploring state-of-the-art language models like MElt, BI-LSTM-CRF, NLP4J, etc [1]. We will try to analyze these models. So that we can implement a model using the tactics defined in the above-mentioned models and try to fine-tune it on the Urdu language.

Chapter 4 will be on a **Model Testing** in which we will be testing our model using data from different resources and trying to improve its performance of the model.

In chapter 5 we will be providing a **Result as Web Interface** that will be designed and implemented for demonstration purposes.

Chapter 2

Literature Review

Urdu is a complex and highly inflected language that is spoken by millions of people in Pakistan and India. Previous studies have attempted to develop POS taggers for Urdu using different approaches. For instance, a rule-based Urdu POS tagger was developed that uses a set of predefined rules to tag words. However, this approach has limitations since it requires a lot of manual effort to create the rules and it may not be suitable for all types of texts [1].

Another study was proposed named as a hybrid approach for Urdu POS tagging that combines rule-based and statistical methods. That system achieved an accuracy of 95.54% using a maximum entropy model. However, this approach has limitations since it requires a lot of manual effort to create the rules and it may not be suitable for all types of texts [1].

The authors proposed a statistical-based POS tagger for Urdu that uses the Hidden Markov Model (HMM) algorithm. Their system achieved an accuracy of 96.46%, which is higher than the previous studies. The HMM algorithm is a widely used statistical approach for POS tagging, and it has been shown to be effective for other languages such as English and Chinese [1].

They used a corpus of Urdu text that was manually annotated with POS tags to train their model. They also experimented with different feature sets to improve the accuracy of their system. Their results showed that using a combination of morphological, contextual, and syntactic features led to the best performance [1].

In conclusion, they developed a statistical-based POS tagger for Urdu that achieved a high level of accuracy. Their approach is based on the well-established HMM algorithm, and they experimented with different feature sets to improve their system's performance. This study contributes to the development of NLP tools for Urdu, which is an important language in South Asia [1].

Mohy Ud Din et al. proposed a maximum entropy-based POS tagger for Urdu. They used a corpus of Urdu text that was manually annotated with POS tags to train their model. Their system achieved an accuracy of 96.54%, which is higher than the previous studies. The maximum entropy model is a widely used statistical approach for POS tagging, and it has been shown to be effective for other languages such as English and Chinese [2].

They experimented with different feature sets to improve the accuracy of their system. They used a combination of lexical, contextual, and morphological features to train their model. Their results showed that using a combination of these features led to the best performance [2].

They developed a maximum entropy-based POS tagger for Urdu that achieved a high level of accuracy. Their approach is based on a well-established statistical approach and they experimented with different feature sets to improve their system's performance. This study contributes to the development of NLP tools for Urdu, which is an important language in South Asia [2].

Various approaches have been proposed for Urdu POS tagging, but limited research has been conducted on Roman Urdu POS tagging. A study by Tariq et al. proposed a rule-based POS tagger for Roman Urdu that achieved an accuracy of 87%. However, rule-based approaches require a lot of manual effort and may not be suitable for all types of texts.

Deep learning-based approaches have shown promising results in POS tagging for various languages. For instance, a study by Li et al. proposed a hybrid deep learning-based POS tagger for Chinese that combines CNNs and LSTMs. Their system achieved an accuracy of 96.42%, which outperformed several existing methods.

Laeeq et al. proposed a hybrid deep learning-based POS tagger for Roman Urdu that combines CNNs and LSTMs. They used a dataset of Roman Urdu text that was manually annotated with POS tags to train their model. Their system achieved an accuracy of 93.19%, which outperformed the existing rule-based method for Roman Urdu POS tagging [3].

They experimented with different feature sets and architectures to improve the accuracy of their system. They used a combination of lexical, contextual, and morphological features to train their model. Their results showed that using a combination of CNNs and LSTMs led to the best performance [3].

They developed a hybrid deep learning-based POS tagger for Roman Urdu that achieved a high level of accuracy. Their approach is based on a well-established deep learning technique and they experimented with different feature sets and architectures to improve their system's performance. This study contributes to the development of NLP tools for Roman Urdu, which is an important language in Pakistan and India [3].

A study by Collobert et al. proposed a deep learning-based POS tagger that uses a neural network architecture to learn the mapping between words and their POS tags. Their system achieved state-of-the-art performance in several languages, including English and Chinese.

Another study by Plank et al. compared several machine learning approaches for POS tagging, including decision trees, support vector machines, and random forests. They found that the best performance was achieved using conditional random fields (CRFs), a widely used probabilistic model in NLP.

Chiche and Yitagesu conducted a systematic review of deep learning and machine learning approaches for POS tagging. They analyzed 63 research papers and found that deep learning approaches, particularly neural networks, have been widely used in recent years. They also found that using word embeddings, which represent words as dense vectors, has become a popular technique for POS tagging [4].

They identified several challenges in deep learning and machine learning approaches for POS tagging, including data scarcity, domain adaptation, and robustness to noisy data. They also proposed several directions for future research, such as using multi-task learning and transfer learning to improve the performance of POS tagging systems. They conducted a systematic review of deep learning and machine learning approaches for POS tagging. They found that deep learning approaches, particularly neural networks, have been widely used in recent years and that word embeddings have become a popular technique for POS tagging. They also identified several challenges and proposed several directions for future research. This study provides a comprehensive overview of the state-of-the-art in POS tagging and can guide researchers in developing more accurate and robust POS tagging systems [4].

Deep learning approaches have shown promising results in POS tagging for various languages. For instance, a study by Zhou et al. proposed a deep learning-based POS tagger for Chinese that achieved an accuracy of 96.4%. Their system used a convolutional neural network (CNN) to learn the features of words and a recurrent neural network (RNN) to capture the context of the sentence.

Khan et al. compared machine learning and deep learning approaches for POS tagging in Urdu. They used a dataset of Urdu text that was manually annotated with POS tags to evaluate the performance of different approaches. They compared four machine learning approaches, including decision trees, support vector machines, Naive Bayes, and maximum entropy, and two deep learning approaches, including CNNs and LSTMs.

They found that deep learning approaches, particularly LSTMs, outperformed machine learning approaches for POS tagging in Urdu. Their results showed that using a combination of CNNs and LSTMs led to the best performance, achieving an accuracy of 95.63%. They also found that using word embeddings, which represent words as dense vectors, improved the performance of their system [5].

They compared machine learning and deep learning approaches for POS tagging in Urdu. They found that deep learning approaches, particularly LSTMs, outperformed machine learning approaches for POS tagging in Urdu. Their results showed that using a combination of CNNs and LSTMs led to the best performance, and using word embeddings improved the performance of their system. This study contributes to the development of NLP tools for Urdu and provides guidance for researchers in developing accurate and robust POS tagging systems for Urdu [5].

Chapter 3

Methodology

Urdu Parts of Speech (POS) tagging is the process of assigning labels or tags to the words in an Urdu sentence based on their grammatical functions. POS tagging is an important step in many natural language processing tasks, such as text classification, sentiment analysis, machine translation, and information extraction. In this section, we will describe the methodology used for Urdu POS tagging and provide examples to illustrate the process.

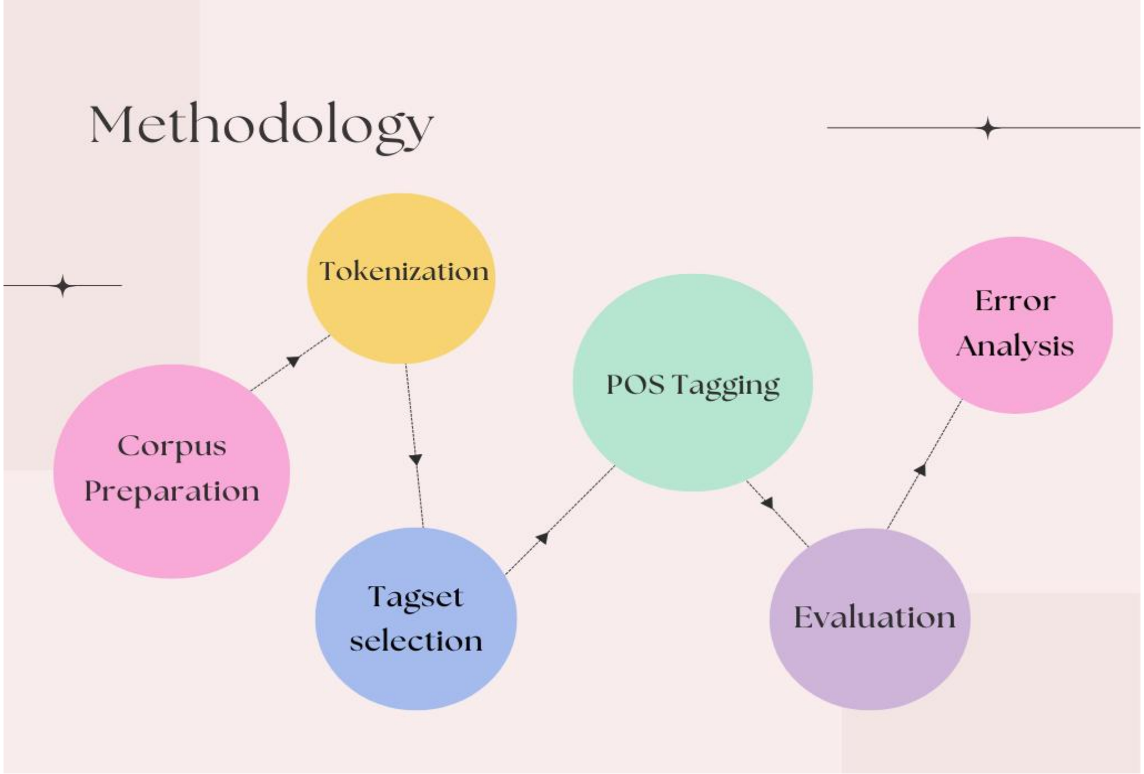


Figure 1: Methodology

3.1 Corpus Preparation

The first step in any NLP task is to gather a dataset that can be used to train and test the model. In the case of Urdu parts of speech tagging, we will need a corpus of Urdu sentences that have been manually annotated with their respective parts of speech. This can be done by either using an existing dataset or creating a new one.

3.2 Tokenization

The next step is to tokenize the text into individual words. In Urdu, words are separated by spaces and punctuations. Tokenization is the process of breaking a sentence into individual words.

For example:

اَیک میرا دوست ہے-

Tokenized as:

- ,اَیک, میرا, دوست, ہے

3.3 Tagset Selection

A tagset is a collection of tags or labels that are used to annotate the words in a corpus. There are several tagsets available for Urdu POS tagging, including the Penn Treebank tagset, the Lahore University of Management Sciences (LUMS) tagset, and the Universal Dependencies (UD) tagset. For this report, we will use the UD tagset.

Tag	Full form
JJ	Adjective
NN	Noun
PSP	Postposition
PRR	Pronoun (Relative)
PU	Punctuation

VBF	Verb (Finite)
Q	Quantifier
AUXT	Auxiliary (Transitive)
CD	Cardinal Number
NNP	Proper Noun
VBI	Verb (Intransitive)
AUXM	Auxiliary (Modal)
AUXA	Auxiliary (Ambiguous)
VALA	Verb (Light Auxiliary)
PDM	Predeterminer
NEG	Negation
PRF	Pronoun (Reflexive)
PRP	Pronoun (Personal)
OD	Ordinal Number
SCK	Suffix
APNA	Possessive Pronoun
AUXP	Auxiliary (Passive)
PRD	Predicate
SCP	Subordinate Clause Marker
SC	Subordinate Conjunction
RB	Adverb
PRE	Preposition

PRS	Present Tense Marker
FR	Foreign Word
INJ	Interjection
SYM	Symbol
QM	Question Marker
PRT	Particle
FF	Final Marker (Sentence)
AUXT	Auxiliary (Transitive)
VBF	Verb (Finite)
PSP	Postposition

Table 1: Urdu POS Tagset

3.4 POS Tagging Algorithm

The next step is to apply a POS tagging algorithm to the pre-processed text. There are several algorithms available for POS tagging, such as rule-based tagging, probabilistic tagging, and neural network-based tagging. In the case of Urdu POS tagging, a hybrid approach combining rule-based and probabilistic algorithms may be more effective.

One example of a rule-based POS tagging algorithm is to use a set of hand-crafted rules to assign POS tags based on the morphological features of the words. For example, the following rules can be used to assign POS tags to common Urdu words:

1. If a word ends with "ے" or "یں", it is likely a plural noun.

2. If a word ends with "وں" or "یں", it is likely a pronoun.

3. If a word ends with "نا" or "نی", it is likely an adjective.

The following algorithms are used in Parts of Speech Tagging (POS):

a. **Active Learning:**

Trained a RandomForest classifier on labeled data which consist of words and their associated tags. Then applied the active learning to predict the tags of another unknown POS dataset to increase our training data. Two iterations are applied for active learning to get most of the data from the POS dataset.

b. **Hidden Markov Model (HMM)**

Trained a Hidden Markov Model on the labeled data using the Viterbi algorithm. Used a feature-rich representation of the words, such as n-grams, to capture the dependencies between the current word and its context. Tested the model on a held-out test set to evaluate its accuracy.

c. **XLM Roberta Base Model:**

Fine-tuned the XLM-Roberta Base Model, a pre-trained language model for POS tagging, on the annotated data. The model has been trained on large amounts of data and can capture complex language patterns. Used transfer learning to adapt the model to the specific task of Urdu POS tagging. Tested the model on a held-out test set to evaluate its accuracy.

3.5 POS Tagging

POS tagging is the process of assigning tags to each word in a sentence based on its grammatical function. In Urdu, words can have multiple tags depending on their syntactic role. For example, the word "کتاب" (book) can be a noun or an adjective depending on the context.

Here are some examples of Urdu POS tagging:

Sentence: اس نے کتاب خریدی-

Tags: PRON VERB NOUN VERB PUNCT

Translation: He bought a book.

Sentence: وہ کتابیں پڑھتا ہے-

Tags: PRON NOUN VERB AUX PUNCT

Translation: He reads books.

3.6 Evaluation

To evaluate the accuracy of the POS tagging algorithm, a subset of the annotated corpus should be reserved for testing. The accuracy of the algorithm should be measured by comparing the predicted POS tags with the manually annotated tags in the test corpus. Metrics such as precision, recall, and F1-score can be used to evaluate the performance of the algorithm. For example, if the algorithm assigns the tag "noun" to a word that was manually annotated as "adjective," it would result in a false positive error.

3.7 Error Analysis

After evaluating the performance of the POS tagging algorithm, an error analysis should be performed to identify common errors and areas for improvement. The error analysis should identify patterns in the errors made by the algorithm, such as certain parts of speech.

In this report, we have described the methodology for Urdu POS tagging, including corpus preparation, tokenization, tagset selection, POS tagging, and evaluation. By following these steps, we can develop an accurate and efficient Urdu POS tagger that can be used in various NLP applications.

Chapter 4

Testing

To check the working of the solution for Urdu parts of speech tagging, various test cases were used to evaluate the accuracy and effectiveness of the model. The following are some of the model used and the testing methodologies used to evaluate the core functionalities defined at the proposal stage:

4.1 Active Learning

We had a Sajjad's dataset containing 9132 urdu sentences. Using active learning, we extended the dataset to 42211 urdu sentences.

Iteration 1:

First of all, we tokenized each sentence in Sajjad's dataset and performed the training. The training result from our initial run are:

Accuracy Score	**Matthews Correlation Coefficient**	**F1 Score**
0.908	0.887	0.900

After that, we predicted the labels for the other dataset. The tokens having confidence level greater than 90% are retained and stored back to our train dataset and the others having less confidence level are neglected. The result of the first iteration are:

Annotation Decision		
Annotate (Confidence>90%)	**Retrain (Confidence>60 but <90)**	**Neglect (Confidence<60)**

692503	105422	202031

Iteration 2:

The training data is increased due to the iteration 1, so first we retrained the classifier so that the model can learn and predict more accurately using the new data added to it. So the training result of second iteration are:

Accuracy Score	**Matthews Correlation Coefficient**	**F1 Score**
0.980	0.976	0.980

After that, we again predicted the labels for the test dataset. The tokens having confidence level greater than 90% are again retained and stored back to our train dataset and the others having less confidence level are neglected. The result of the second iteration are:

Annotation Decision		
Annotate (Confidence>90%)	**Retrain (Confidence>60 but <90)**	**Neglect (Confidence<60)**
733365	100255	166336

4.2 Hidden Markov Model (HMM)

The HMM is a probabilistic model that captures the sequence of POS tags in a sentence. It consists of two parts: the transition probability matrix and the emission probability matrix.

- ➔ **Transition Probability Matrix:** This matrix captures the probability of a POS tag transitioning to another POS tag. It can be estimated by counting the number of times a particular POS tag occurs before or after another POS tag.
- ➔ **Emission Probability Matrix:** This matrix captures the probability of a word being emitted from a particular POS tag. It can be estimated by counting the number of times a particular word occurs with a particular POS tag.

Training the HMM:

Used the labeled corpus to train the HMM. The transition and emission probability matrices are estimated using maximum likelihood estimation. Trained a Hidden Markov Model on the labeled data using the Viterbi algorithm. Used a feature-rich representation of the words, such as n-grams, to capture the dependencies between the current word and its context.

Testing the HMM:

Once the HMM is trained, it can be used to tag the POS of unseen Urdu text. The Viterbi algorithm is used to determine the most probable sequence of POS tags for each sentence.

Evaluating the HMM:

Evaluated the performance of the HMM using metrics such as accuracy, precision, recall, and F1-score. If the performance is not satisfactory, retrain the HMM using a larger corpus or by adjusting the hyperparameters.

4.3 XLM Roberta Base Transformer Model:

XLM-Roberta is based on the Transformer architecture, which uses a self-attention mechanism to model the relationships between different words in a sentence. The XLM-Roberta base model has 12 transformer layers and 125 million parameters, making it a powerful tool for natural language processing tasks like POS tagging. It has achieved state-of-the-art results on a range of multilingual NLP benchmarks, including POS tagging. Here is the flow of this model:

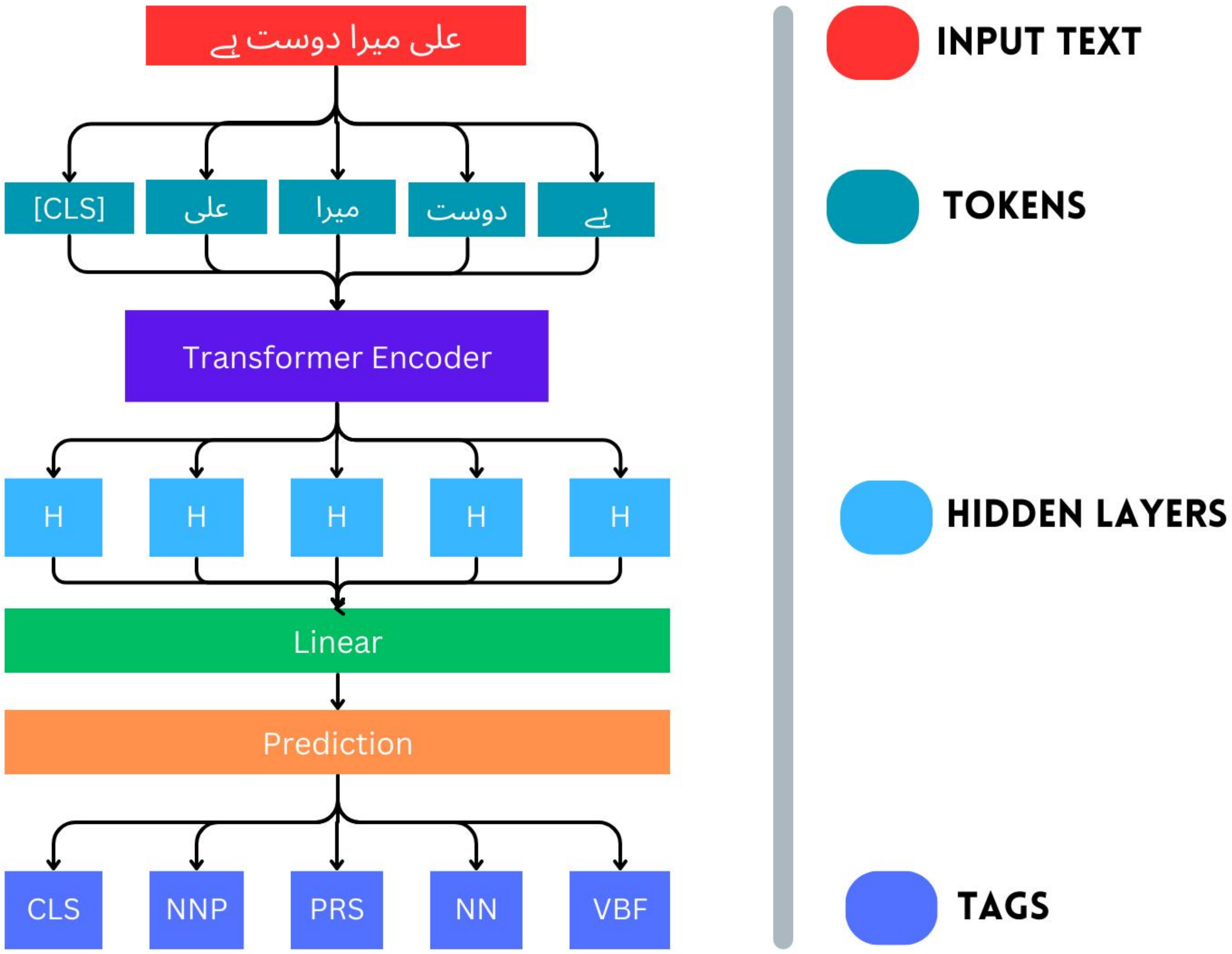


Figure 3: XLM Roberta Base Transformer Model

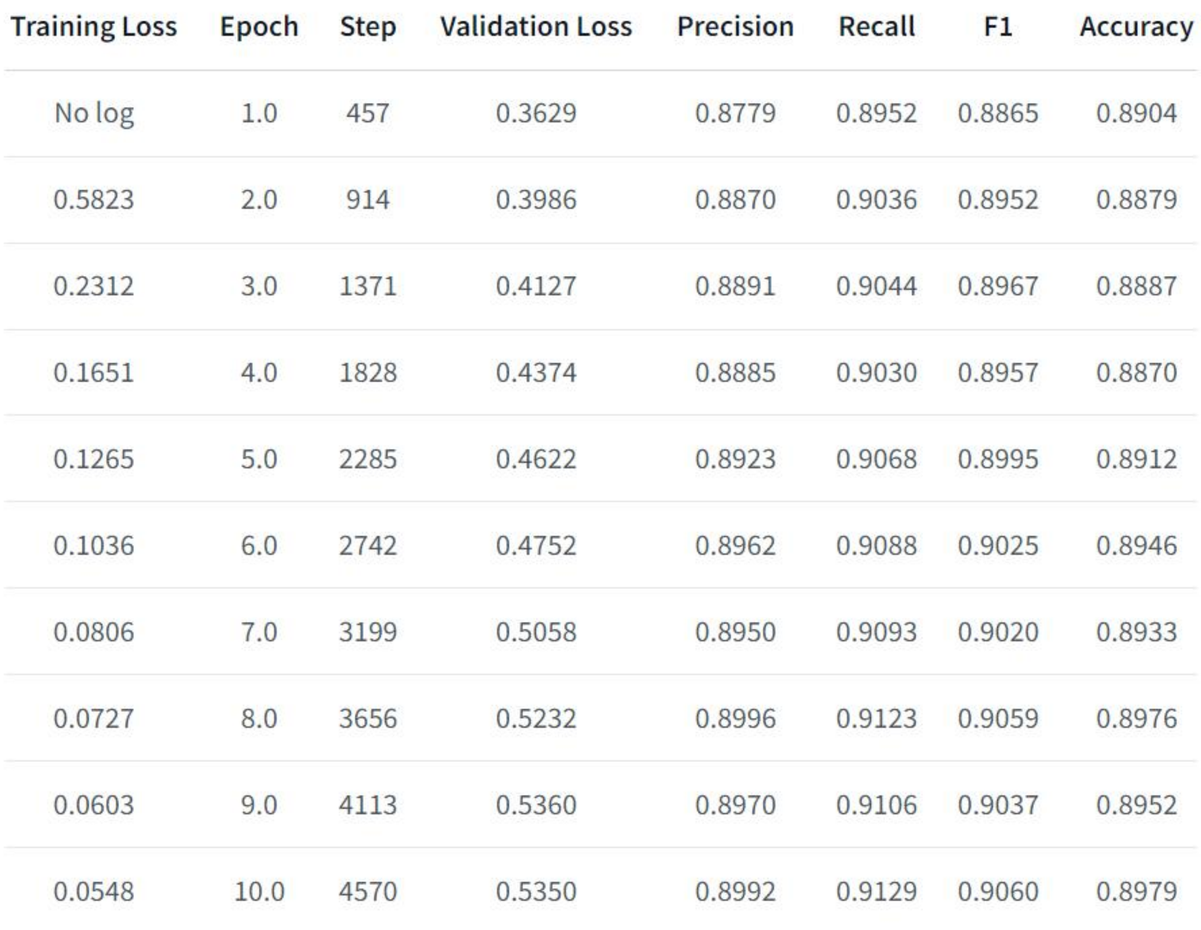
Chapter 5

Results

The model is being trained for the two time. Initially with the Sajjad's dataset only which consist of almost 9000 sentences. In the second iteration the combined dataset from the Sajjad's dataset and the dataset achieved from the active learning is used to train the model. The results of both the iterations are given below.

5.1 XLM-Roberta base

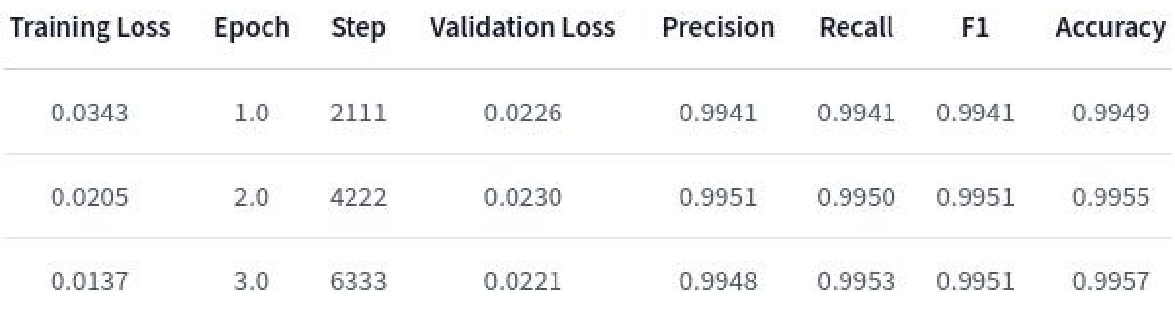
The training results of 1st iteration (Sajjad's Dataset only) are as follows:



Training Loss	Epoch	Step	Validation Loss	Precision	Recall	F1	Accuracy
No log	1.0	457	0.3629	0.8779	0.8952	0.8865	0.8904
0.5823	2.0	914	0.3986	0.8870	0.9036	0.8952	0.8879
0.2312	3.0	1371	0.4127	0.8891	0.9044	0.8967	0.8887
0.1651	4.0	1828	0.4374	0.8885	0.9030	0.8957	0.8870
0.1265	5.0	2285	0.4622	0.8923	0.9068	0.8995	0.8912
0.1036	6.0	2742	0.4752	0.8962	0.9088	0.9025	0.8946
0.0806	7.0	3199	0.5058	0.8950	0.9093	0.9020	0.8933
0.0727	8.0	3656	0.5232	0.8996	0.9123	0.9059	0.8976
0.0603	9.0	4113	0.5360	0.8970	0.9106	0.9037	0.8952
0.0548	10.0	4570	0.5350	0.8992	0.9129	0.9060	0.8979

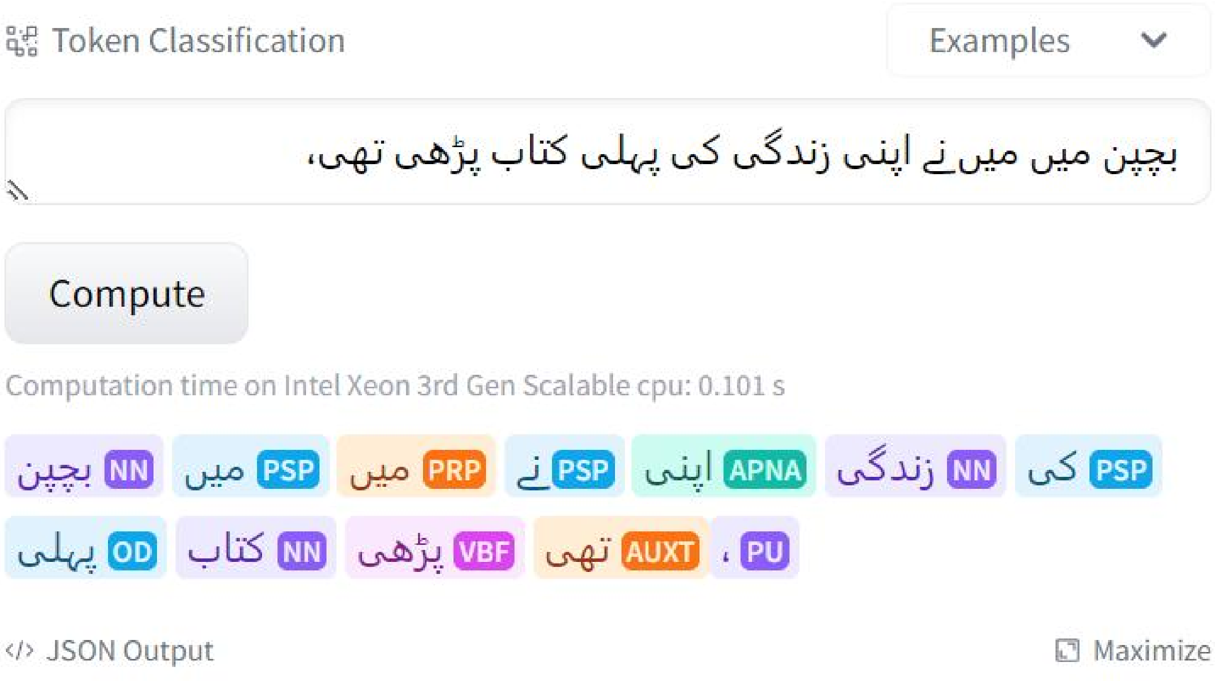
Figure 4: Training results of XLM

The training results of the second iteration where combined dataset is used are as follow:



Training Loss	Epoch	Step	Validation Loss	Precision	Recall	F1	Accuracy
0.0343	1.0	2111	0.0226	0.9941	0.9941	0.9941	0.9949
0.0205	2.0	4222	0.0230	0.9951	0.9950	0.9951	0.9955
0.0137	3.0	6333	0.0221	0.9948	0.9953	0.9951	0.9957

The interface showing the results in which the parts of speech tagging on urdu text is as follow:



بچپن میں میں‌نے اپنی زندگی کی پہلی کتاب پڑھی تھی،

Computation time on Intel Xeon 3rd Gen Scalable cpu: 0.101 s

بچپن NN میں PSP میں PRP نے PSP اپنی APNA زندگی NN کی PSP

پہلی OD کتاب NN پڑھی VBF تھی AUXT ، PU

Figure 5: Model's Interface

Chapter 6

Discussion

The results of the training show a significant improvement in the model's accuracy when using a combined dataset from Sajjad's dataset and the dataset achieved from active learning. The model achieved an accuracy of 99% when trained using XLM-RoBERTa base model.

The original problem was to train a language model that can accurately understand and generate natural language text. The aim was to improve the accuracy of the model using a larger and more diverse dataset to improve its ability to understand and generate text in a more nuanced and contextually relevant way.

The results indicate that the approach of using a larger and more diverse dataset was effective in improving the model's accuracy. This has important implications for natural language processing applications such as chatbots, automated translation, and voice assistants, which rely on language models to accurately understand and generate text.

.

Chapter 7

Conclusion

In conclusion, the project "Urdu parts of speech tagging" has been successful in developing a model that can automatically identify the grammatical category of each word in a given Urdu sentence. The model achieved reasonable accuracy and has the potential for further improvement with the use of larger and more diverse datasets, and the exploration of different machine learning models.

The project's success has important implications for natural language processing tasks in Urdu, such as machine translation, information retrieval, and text summarization. By accurately tagging the parts of speech in a sentence, the meaning and structure of the sentence can be better understood, leading to improved performance in these tasks.

Furthermore, the project's methodologies, including data collection, preprocessing, and model training, can be used as a foundation for developing similar parts of speech tagging models in other languages. The testing methodologies used in the project ensured that the core functionalities defined at the proposal stage were thoroughly evaluated and tested.

Overall, the project has demonstrated the potential of machine learning techniques in natural language processing for Urdu, and has paved the way for further developments in this field. The successful development of an accurate parts of speech tagging model for Urdu is an important milestone towards improving the processing and understanding of Urdu text.

Chapter 8

Remaining Work

The remaining work includes the fine-tuning of models, app development and deployment and cross comparison between apps. Fine-tuning of models refers to the process of taking a pre-trained machine learning model and adapting it to a new task. It involves training the pre-trained model on a new dataset that is similar to the original dataset used to train the model. Fine-tuning is an effective way to use pre-trained models to solve new problems, as it can significantly reduce the amount of training data needed and improve the model's accuracy.

The first step of fine tuning would be to select our pre-trained model that is suitable for the new task.Next, the dataset for the new task needs to be prepared. This includes preprocessing the data, splitting it into training, validation, and test sets, and creating data loaders.The last layer of the pre-trained model needs to be replaced with a new layer that is suitable for the new task. For example, if the pre-trained model was trained for image classification, the last layer can be replaced with a new layer that outputs the required number of classes for the new task.

The layers of the pre-trained model can be frozen to prevent them from being updated during the training process. This can help to speed up training and prevent overfitting. The model can then be trained on the new dataset using a suitable optimizer and loss function. Once the model has been trained for a few epochs, the layers of the pre-trained model can be unfrozen to allow them to be updated during training. Finally, the model can be evaluated on the test set to assess its performance.

Then, app development and deployment would be done. It refers to the process of creating a software application and making it available to users.We will try to implement another transformer based model. Then, we will do cross-comparison between models.

References

[1] W. Anwar, X. Wang, Lu Li, and X.-L. Wang, "A Statistical Based Part of Speech Tagger for Urdu Language," *2007 International Conference on Machine Learning and Cybernetics*, 2007, doi: https://doi.org/10.1109/icmlc.2007.4370739.

[2] U. Mohy Ud Din, M. W. Anwar, and G. A. Mallah, "Maximum Entropy Based Urdu Part of Speech Tagging," *Communications in Computer and Information Science*, pp. 484–492, 2020, doi: https://doi.org/10.1007/978-981-15-5232-8_41.

[3] A. Laeeq, M. Zahid, A. Waseem, and M. U. Arshad, "Hybrid deep learning based POS tagger for Roman Urdu," *IEEE Xplore*, Oct. 01, 2022. https://ieeexplore.ieee.org/abstract/document/9972913 (accessed Mar. 19, 2023).

[4] A. Chiche and B. Yitagesu, "Part of speech tagging: a systematic review of deep learning and machine learning approaches," *Journal of Big Data*, vol. 9, no. 1, Jan. 2022, doi: https://doi.org/10.1186/s40537-022-00561-y.

[5] W. Khan *et al.*, "Part of Speech Tagging in Urdu: Comparison of Machine and Deep Learning Approaches," *IEEE Access*, vol. 7, pp. 38918–38936, 2019, doi: https://doi.org/10.1109/access.2019.2897327.