

ROMAN URDU SENTIMENT ANALYSIS

Machine Learning Project Final Report



Submitted By

Abdul Samad 023-19-0128

Haseeb 023-19-0061

Muhammad Ismail 023-19-0140

Submitted to

DR SHER

Contents

Introduction	3
Problem Statement.....	3
Methodology.....	3
Pre-Processing Data	3
Dealing with Stop words	3
Tokenization.....	4
Vectorization	4
Encode the Labels	4
Model Selection	4
Dataset Discussion	5
Major Outcomes	6
Conclusion.....	7
References	7

Introduction

Sentiment analysis is a procedure that uses natural language processing (NLP) to automatically get responses, opinions, perspectives, and emotions from text, tweets, audio, databases, etc.

It determines if a particular portion of writing is positive, negative, or neutral.

The strategies used for text data typically concentrate on processing, looking up, or interpreting the factual data that is already present.

Sentiment classification has so many uses that are beneficial in businesses, marketing, and boosting product sales by recognizing the feedback given by clients.

Problem Statement

There are numerous E-Commerce websites in Pakistan. The reviews are the central part in their businesses. However, the reviews given by the customers are in Roman-Urdu almost ninety-nine percent of the time. Therefore, our project can be used by these types of businesses to classify their reviews and make important decisions about their products.

In addition, one use of our model can be in messaging applications to make a user aware about the sentiment of their written text, resulting in better communication between people.

Methodology

Pre-Processing Data

Firstly, we will pre-process the data because our data may contain lots of useless information like punctuation marks, etc. which may not give any sense while classification, therefore we remove all of these types of inputs, after that we will remove the empty instances in the dataset.

Dealing with Stop words

In order to get better results, we need to remove the common (most frequent) words. It is comparatively easy in English (i.e., removing the known stop words), however, for Roman-Urdu it is difficult, therefore, we

will try to find most used stop words in Urdu language. In last, we will remove those identified words from our dataset and continue with the project.

Tokenization

Here we will convert the sentences into tokens and these tokens will go through a vectorization process where they get a particular value for each token.

Vectorization

Here we will apply different vectorization techniques which help to assign a number to our tokens. We have many vectorization techniques like hash vectorizer, count vectorizer, etc.

Encode the Labels

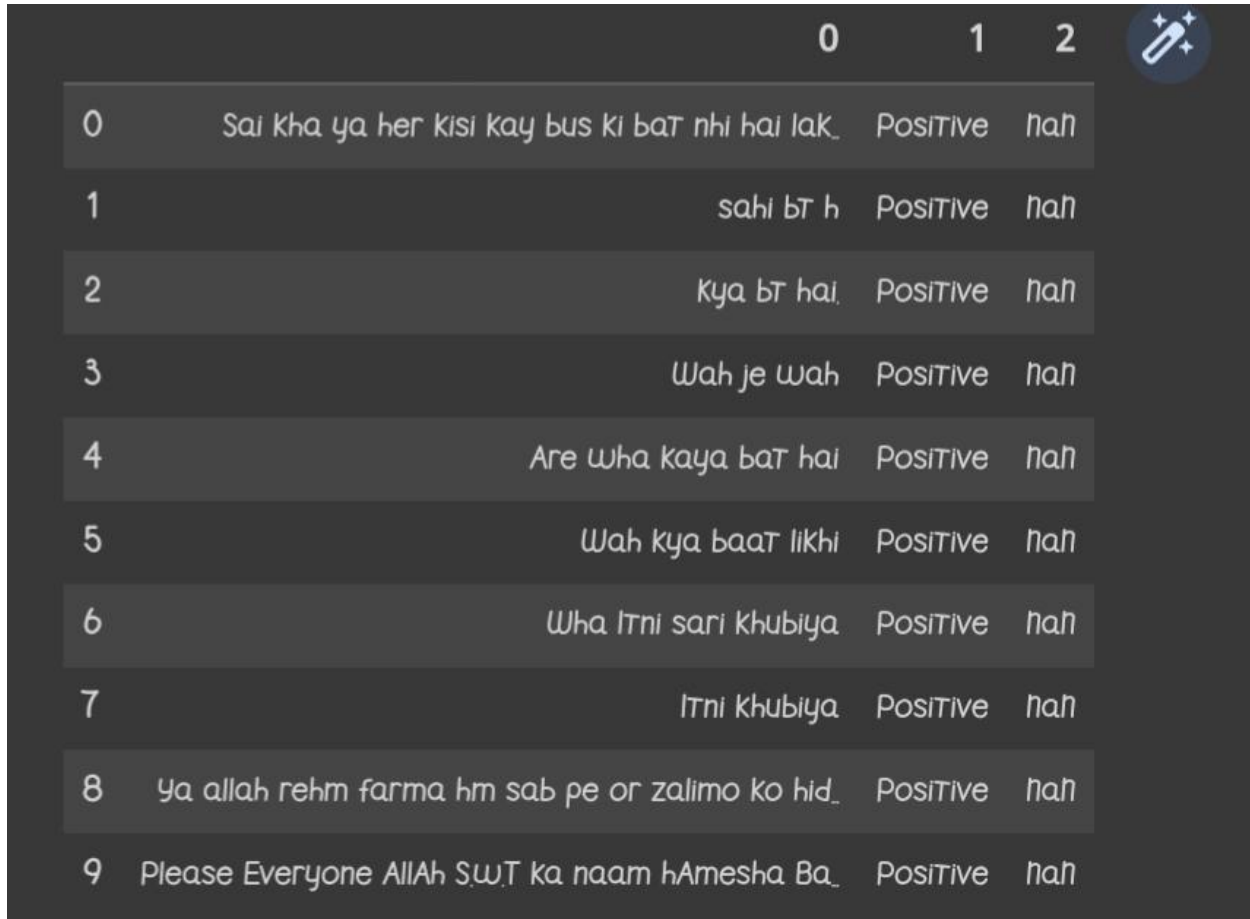
In this step we will simply assign labels to the output such as assign 0 to negative, 1 to neutral, and 2 to positive label because the model only understands numerical values so it's mandatory to assign these values to labels.

Model Selection

We will apply different types of classifier model which can classify multiple classes and helps to classify and evaluate their performance by checking their different metrics which helps us to select one of the best models.

Dataset Discussion

The dataset which we will use is named as Roman Urdu dataset. The dataset contains 3 columns and 20228 rows initially. The first and second column contains the sentences and sentiments respectively, the third column is empty (and will be dropped during the preprocessing phase).



	0	1	2
0	Sai kha ya her kisi kay bus ki baT nhi hai laK_	POSITIVE	nah
1	sahi bT h	POSITIVE	nah
2	Kya bT hai,	POSITIVE	nah
3	Wah je wah	POSITIVE	nah
4	Are wha kaya baT hai	POSITIVE	nah
5	Wah kya baat likhi	POSITIVE	nah
6	Wha ITni sari khubiya	POSITIVE	nah
7	ITni khubiya	POSITIVE	nah
8	Ya allah rehM farma hm sab pe or zalimo ko hid_	POSITIVE	nah
9	Please Everyone AllAh S.W.T ka naam hAmesha Ba_	POSITIVE	nah

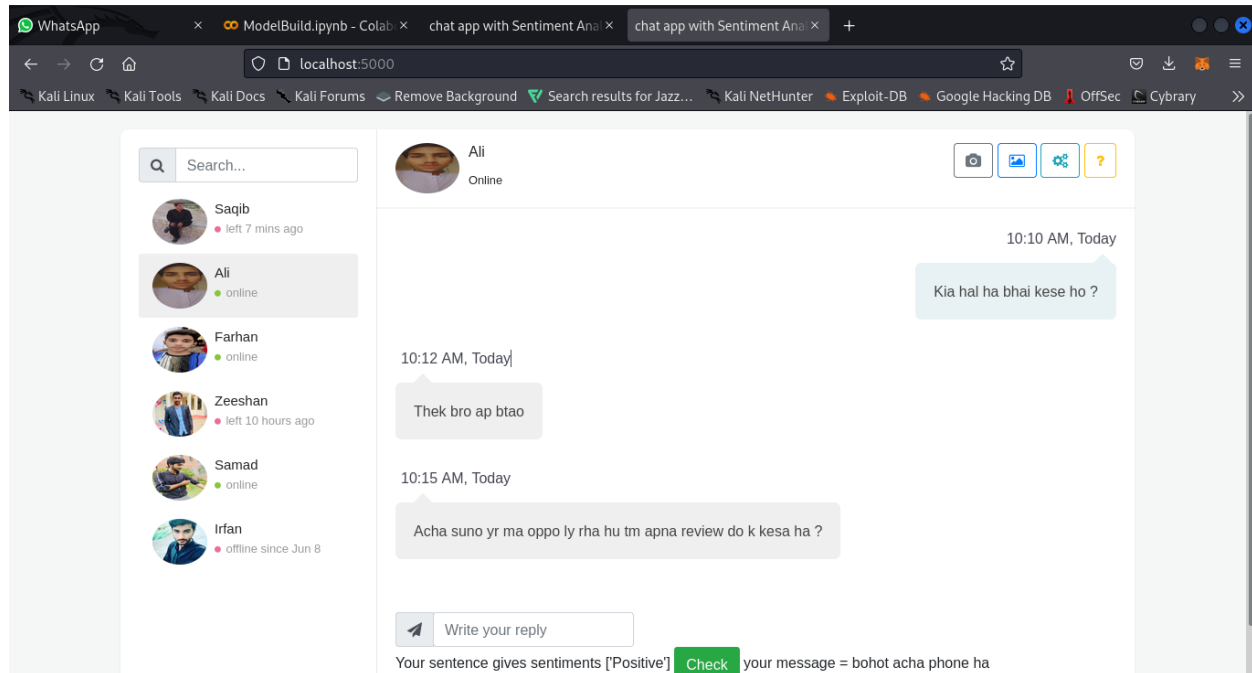
Here we can see our dataset contains the roman Urdu words along with their correct output class such as “khubiyan” is a positive word and we use the sentence “kia bt ha” to praise someone.

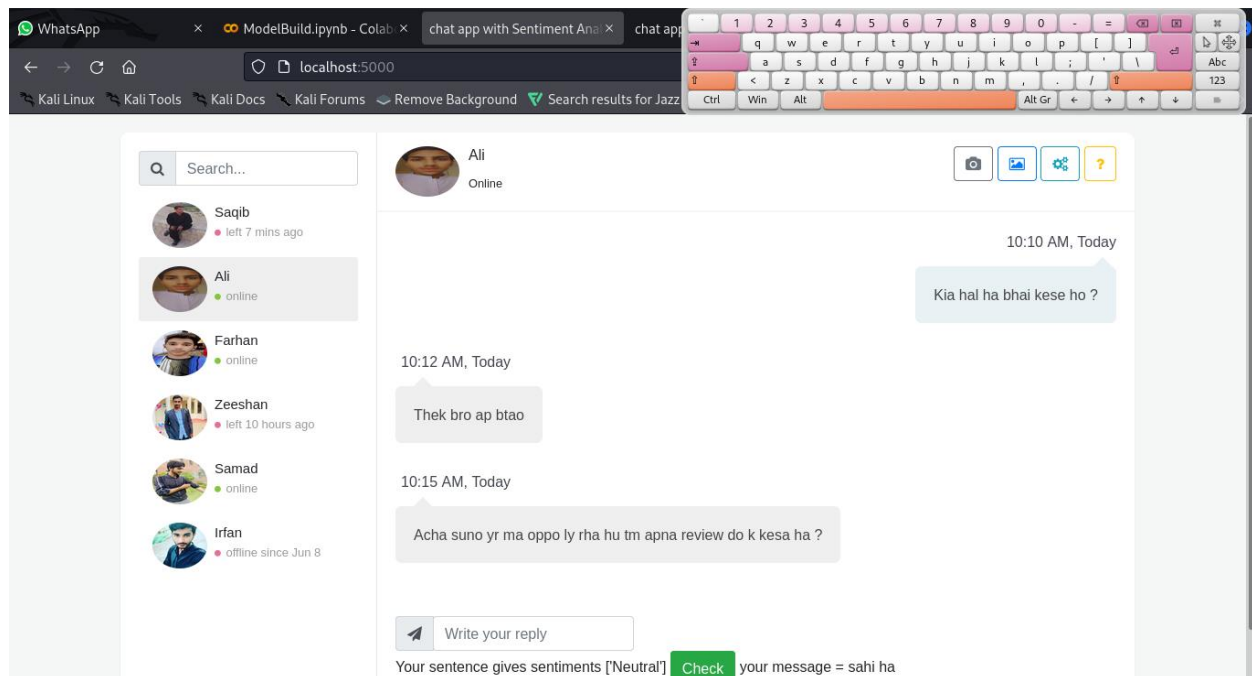
As we see we have different number of records in different classes and neutral class contains more records then other.

Major Outcomes

The expected outcome from this project is that we will be able to apply machine-learning knowledge and theory in practice. Also, we will become familiar with the online communities and environments related to machine learning and artificial intelligence. Furthermore, we will be able to make working machine-learning applications.

Screenshots





Conclusion

We have developed a machine learning model that processes text to classify the text into three classes.

Positive, negative and neutral

Our project is based on **Sentiment Analysis from Roman-Urdu Text** that it is an intuitive application and an interesting opportunity to grasp the essential concepts of machine-learning especially Natural Language Processing

References

<https://archive.ics.uci.edu/ml/datasets/Roman+Urdu+Data+Set>

<https://www.youtube.com/watch?v=LE3NfEULV6k&t=24s>