

# **A Toolkit For The Urdu Language Using NLP**



By

NC Aleena Akbar  
NC Muhammad Irtaza Malik  
NC Muhammad Junaid  
NC Muhammad Faizan

Supervisor

Dr. Hamad Afzal

Submitted to the faculty of Department of Computer Software Engineering,  
Military College of Signals, National University of Sciences and Technology, Islamabad,  
in partial fulfillment for the requirements of B.E Degree in Software Engineering.

June 2023

In the name of ALLAH, the Most benevolent, the Most Courteous

## **CERTIFICATE OF CORRECTNESS AND APPROVAL**

*This is to officially state that the thesis work contained in this report*

*“An Urdu Toolkit”*

*is carried out by*

***Aleena Akbar, Muhammad Faizan, Muhammad Junaid &***

***Muhammad Irtaza***

*under my supervision and that in my judgment, it is fully ample, in scope and  
excellence, for the degree of Bachelor of Software Engineering at Military  
College of Signals, National University of Sciences and Technology (NUST),  
Islamabad.*

**Supervisor Name: Dr. HAMMAD AFZAL**

**Supervisor Signature:**

**Date:** \_\_\_\_\_

**Place:** \_\_\_\_\_

## **DECLARATION OF ORIGINALITY**

We hereby declare that no portion of the work presented in this Thesis has been submitted in support of another award or qualification in either this institute or anywhere else.

## **ACKNOWLEDGEMENTS**

Allah Subhan'Wa'Tala is the sole guidance in all domains.

Our parents, colleagues, and most of all our supervisor, Dr. Hammad Afzal  
without your guidance this wouldn't be possible.

And all the group members, who through all adversities worked steadfastly.

## **PLAGIARISM CERTIFICATE (Turnitin Report)**

This thesis has \_\_\_\_\_ similarity index. Turnitin report endorsed by Supervisor is attached.

NC Aleena Akbar  
000293511

NC Muhammad Faizan  
00000

NC Muhammad Irtaza  
00000

---

NC Muhammad Junaid  
00000

Signature of Supervisor

## **ABSTRACT**

This toolkit aims to address the growing concern of harmful and false content on various communication platforms. It provides three features for Urdu language users, namely Toxic Content Detection, Fake Content Detection, and Voice Cloning. The Toxic Content Detection feature uses machine learning algorithms and natural language processing to analyze text and voice notes for threat or abusive content. Similarly, the Fake Content Detection feature uses the same technologies to detect false or misleading information. Finally, the Voice Cloning feature uses speech synthesis technology and audio processing to generate a voice note containing text in a selected voice.

The toolkit's objective is to create a safer and more trustworthy online environment for Urdu language users. The need for such a tool arises from the increasing amount of harmful and false information being circulated on social media and other online platforms. The toolkit's innovative features can help detect such content and provide a way for users to generate trustworthy voice notes. This toolkit can be used by anyone who wants to ensure the content they are consuming or sharing is safe and authentic. Overall, this toolkit can contribute to promoting digital well-being and enhancing communication experiences in Urdu language communities.

# Table of Contents

CERTIFICATE OF CORRECTNESS AND APPROVAL.....	3
DECLARATION OF ORIGINALITY .....	4
ACKNOWLEDGEMENTS .....	5
PLAGIARISM CERTIFICATE (Turnitin Report) .....	6
ABSTRACT.....	7
LIST OF FIGURES .....	11
<i>Chapter 1</i> .....	12
INTRODUCTION .....	12
1.1	133
1.2	144
1.3	14
1.4	15
1.5 Deliverables.....	16
Software Requirement Specification:.....	16
Software Architecture Document:.....	16
Software Design Document: .....	16
Implementation Code Document: .....	17
Software Testing Document:.....	17
Final Project Report .....	17
1.6 Relevant Sustainable Development Goals.....	17
1.7 Structure of Thesis.....	18
<i>Chapter 2</i> .....	20
LITERATURE REVIEW.....	20
2.1 Deep Learning .....	22
2.2 Urdu Language .....	23
2.3 Fake/Toxic Content Detection with Deep Learning.....	24
2.4 Limitations and Areas of Improvement.....	25
2.5 Existing solutions and their drawbacks .....	26
<i>Chapter 3</i> .....	29
PROBLEM DEFINITION AND WORKING ELEMENTS.....	29
3.1 Working Elements and Techniques.....	30
3.1.1 Machine Learning Model.....	30
3.1.2 User Interface.....	30
3.1.3 Deep Learning.....	30
3.1.4 Programming Languages.....	30
<i>Chapter 4</i> .....	31
METHODOLOGY.....	31



4.1 Project Steps.....	31
4.2 Working Principle .....	31
4.2.1 Data Acquisition: .....	31
4.2.2 Preprocessing: .....	31
4.2.3 Fake Content Detection:.....	32
4.2.4 Abusive Content:.....	33
4.2.5 Voice Cloning: .....	34
<i>Chapter 5</i> .....	36
DETAILED DESIGN AND ARCHITECTURE.....	36
5.1 Architectural Design .....	36
5.2 Component Design.....	36
5.3 Decomposition Description.....	38
5.3.1 Data Flow Diagram .....	38
5.3.2 Structural Decomposition Diagram.....	39
5.3.3 Module Decomposition – Class Diagram.....	40
5.3.4 Process Decomposition – Use Case Diagram.....	41
5.4 Sequence Diagram: .....	41
5.5 Activity Diagram: .....	44
5.6 Interface: .....	44
5.7 Dictionary:.....	44
<i>Chapter 6</i> .....	46
User Interface.....	46
6.1 Mobile App .....	46
6.1.1 Important Features .....	47
6.1.2 How the App works.....	47
6.1.3 Implementation .....	48
6.2 Web App .....	60
6.2.1 How the App works .....	64
<i>Chapter 7</i> .....	70
IMPLEMENTATION AND TESTING.....	70
7.1 DataSet Collection: .....	70
7.2 Input .....	71
7.3 Machine Learning Model: .....	71
7.4 Testing.....	74
7.3 Tests and Discussions: .....	77
<i>Chapter 8</i> .....	80
CONCLUSIONS AND FUTURE WORK.....	80
8.1 Future Work .....	80
<i>Chapter 9</i> .....	82

REFERENCES..... 82

## LIST OF FIGURES

Figure 1: RNN, LSTM AND GRU.....	23
Figure 2: Bi-Directional Neural Network.....	25
Figure 3: Recurrent Neural Network.....	25
Figure 4: Architectural Design.....	36
Figure 5: Data Flow Diagram.....	38
Figure 6: Structural Decomposition Diagram.....	39
Figure 7: Use Case Diagram.....	40.
Figure 8: Class Diagram.....	40
Figure 9: Sequence Diagram.....	41
Figure 10: Activity Diagram.....	42
Figure 11: Main Screen.....	42
Figure 12: Features .....	44
Figure 13: Types of Input.....	48
Figure 13: Main Screen of Mobile App.....	49
Figure 14: Features of the App.....	50
Figure 15: Types of Inputs for Toxic and Fake Content Detection.....	51
Figure 16: Error message.....	53
Figure 17: Result message.....	55
Figure 18: Result message.....	59
Figure 19: Download and Play the Cloned Voice.....	61
Figure 20: Result Message for Fake Content Detection.....	65
Figure 21: Navigation Graph of the Mobile App.....	69
Figure 22: Website Home page.....	69
Figure 23: Contact Section.....	69
Figure 24: Footer Section.....	69
Figure 25: Speech Recognition model.....	71
Figure 26: RNN Model Architecture diagram.....	74

# *Chapter 1*

## **INTRODUCTION**

With the rise of technology and social media, there has been a significant increase in the amount of toxic and fake content circulating online. This can have serious negative effects on individuals and society as a whole.

In fact, a recent study by the Digital Rights Foundation found that over 60% of women in Pakistan have experienced online harassment, with much of it being in the form of toxic and abusive language.

To combat this issue, our app provides three powerful features to help protect and empower our users. Firstly, our Toxic Content Detection feature provides you with an analysis of whether the content contains toxic language or is offensive in nature. This can help you identify and avoid harmful content, and also provide feedback to others who may be using toxic language.

Secondly, our Fake Content Detection feature helps you identify and avoid false or misleading information, which has become increasingly common in today's digital landscape. This feature can analyze if the content is genuine or fabricated, helping you make informed decisions and avoid spreading false information.

Finally, our Voice Cloning feature allows you to generate a voice note containing text in the same voice as a given input voice note. This can be useful in situations where you need to create a message in someone else's voice, such as for accessibility or entertainment purposes.

We believe that our app can help make a positive impact in the fight against online toxicity and misinformation.

## 1.1 Overview

The Internet is one of the important inventions and a large number of people are its users. These people use this for different purposes. There are different social media platforms that are accessible to these users. Any user can make a post or spread the news through these online platforms. These platforms do not verify the users or their posts nor tell whether the uploaded content is toxic for some users or not.

With the rise of social media and the internet, communication has become more accessible than ever before. However, this accessibility has also led to the spread of toxic and fake content, which can have serious consequences. Toxic content, such as hate speech or cyberbullying, can cause emotional harm to individuals and even lead to violence. Similarly, fake content, such as misinformation or propaganda, can have a negative impact on society as a whole.

To address these issues, we have developed a toolkit that focuses on detecting and addressing toxic and fake content specifically for the Urdu language. Our toolkit utilizes cutting-edge technology to provide three powerful features designed to combat these issues.

The first feature is toxic content detection, which allows users to input text, text files, or voice notes and receive a warning if the content is deemed toxic. This feature uses advanced algorithms to analyze the content and identify any potentially harmful language or behavior.

The second feature is fake content detection, which works in the same way as the toxic content detection feature. This feature uses advanced algorithms to analyze the content and identify any potentially false or misleading information.

The third feature of our toolkit is voice cloning, which is designed to allow users to input a voice note and generate a new voice note containing text in that voice. This feature is particularly useful for individuals who may have difficulty speaking or communicating, as it allows them to generate a voice note in their own voice.

Overall, our toolkit is designed to help individuals and communities stay safe and informed in the digital age.

## **1.2 Novelty and Challenges**

- The development of a toolkit with the features of toxic content detection, fake content detection, and voice cloning for Urdu language poses some significant challenges and requires innovation. One of the primary challenges is developing algorithms that accurately detect toxic and fake content in Urdu language, as the language has a complex structure and nuances that may differ from other languages. Moreover, the availability of large datasets for training machine learning models is another challenge that needs to be addressed.
- Another challenge is developing a voice cloning feature for Urdu language, which requires developing deep learning models that can accurately mimic the voice of a specific speaker. This task requires substantial amounts of training data and a large computational infrastructure.
- Additionally, the toolkit must ensure the privacy and security of its users by protecting their data from potential misuse. Developing such an app in a language like Urdu also requires a good understanding of the cultural and societal norms of the Urdu-speaking population to avoid any unintentional offense.
- Overall, the development of a toolkit with the mentioned features for Urdu language requires a creative approach, advanced machine learning algorithms, and a comprehensive understanding of the language and cultural nuances.

## **1.3 Proposed Solution**

The proposed solution for the Urdu language-based toolkit, which has three

features, includes the use of various advanced technologies and algorithms to address the challenges of toxic and fake content detection and voice cloning.

For toxic content detection, we will be using natural language processing (NLP) techniques, machine learning algorithms, and deep learning models. We will train our model on a dataset of toxic comments and feedback to classify toxic content accurately. The model will take input in the form of text, text files, or voice notes, and provide output in the form of abusive, threat or normal label.

Similarly, for fake content detection, we will be using NLP techniques, machine learning algorithms, and deep learning models to detect fake news, reviews, and feedback. We will train our model on a dataset of fake and genuine content to classify fake content accurately. The model will take input in the form of text, text files, or voice notes, and provide output in the form of a fake or real label.

Voice cloning is the third feature of our app, and for this, we will be using text-to-speech (TTS) technology and deep learning models. We will train our model on a dataset of Urdu voice notes to generate a voice that sounds natural and human-like. The model will take input in the form of a voice note and text message, and provide output in the form of a voice note containing the text in that voice.

The proposed solution for our toolkit will be highly effective in detecting toxic and fake content and generating human-like voice notes in the Urdu language. It will help users to communicate more effectively and safely, without worrying about the authenticity and toxicity of the content.

## **1.4 Scope**

The proposed toolkit has a wide scope in the modern age of technology where the internet and social media have become an integral part of our daily lives. With the rise of online communication, the need to filter out toxic and fake content has become increasingly important. Moreover, the ability to clone voices can also have a significant

impact on various industries, including entertainment and customer service.

The toolkit's toxic content detection feature can be used to filter out hateful and harmful language in online communication, making it a safer and more positive space for individuals to interact. The fake content detection feature can help to identify and prevent the spread of misinformation and disinformation, which has become a growing concern in recent years.

The voice cloning feature has a wide range of potential applications, including creating personalized voice assistants, generating audio books in different languages, and dubbing films and television shows. With the increasing demand for audio content, the ability to generate a natural-sounding voice in different languages can be highly beneficial for various industries. Overall, the toolkit has a promising scope for making online communication safer and more efficient.

## **1.5 Deliverables**

### **Software Requirement Specification:**

The purpose of this document is to present a detailed description of toolkit. It will explain the purpose and features of the system, the interfaces of the system, what the system will do, its entire process, the constraints under which it must operate and how the system will react to external stimuli. This document is intended for both the stakeholders and the users.

### **Software Architecture Document:**

In this document, the overall architecture of the system is discussed, including the introduction of various components and subsystems. It is mainly supported by a system Architecture diagram which shows an insider's perspective of the system by describing the high-level software components that perform the major functions to make the system operational.



### **Software Design Document:**

The design document captures all our functional requirements and shows how they interact with each other conceptually. The low-level design also shows how we have been implementing how we are going to implement all of these requirements.

### **Implementation Code Document:**

The implementation code document provides details about the pseudo code for the application and project prototype.

### **Software Testing Document:**

This document has testing modules in which there are certain test cases which depicts the correctness and accuracy of the project.

### **Final Project Report**

This is the thesis report which compiles all the previous and current working for the project. Thesis report provides the whole summary for the project and also gives details about each and every aspect of the project starting from the introduction of the project, literature review, requirements leading to design discussions then testing and lastly future work and conclusion.

## **1.6 Relevant Sustainable Development Goals**

The proposed app has the potential to contribute towards several Sustainable Development Goals (SDGs) set by the United Nations. Firstly, it can help achieve

**SDG 3 - Good Health and Well-being** by detecting and preventing the spread of toxic content online that can harm mental health and well-being.

Secondly, it can contribute towards

**SDG 4** - Quality Education by promoting digital literacy and awareness about fake content and its impact on society.

Thirdly, it can aid in achieving

**SDG 5** - Gender Equality by detecting and preventing online harassment and hate speech, particularly against women and other marginalized groups.

Furthermore, the voice cloning feature of the app can have implications for

**SDG 8** - Decent Work and Economic Growth by promoting innovation and technology in the workforce.

It can also contribute towards

**SDG 9** - Industry, Innovation and Infrastructure by facilitating the development of new technologies and applications related to voice recognition and cloning.

Lastly, the app can support

**SDG 16** - Peace, Justice and Strong Institutions by preventing the spread of fake content and toxic messages that can incite violence and undermine social cohesion.

## **1.7 Structure of Thesis**

In summary, the thesis breakdown is as follows:

**Chapter 2:** Literature Review discusses the popular existing SE Schemes. Their comparisons and architectural models already exist relevant to the project's domain.

**Chapter 3:** Proposed Scheme presents the risks to our system and the scheme derived to tackle. It also presents the system specifications and working algorithm of our project.

**Chapter 4:** The methods, approaches, tools, techniques, algorithms, or other aspects of the solution are sufficiently discussed with sufficient details and supporting figures. It

also discusses our target market, Operating environment and some constraints, assumptions and dependencies.

**Chapter 5:** System Design presents system architectural design and other relevant UML diagrams to explain the design of the system.

**Chapter 6:** System Implementation discusses how the system is actually implemented. It includes pseudo code that presents the working of APIs, Demo outputs of APIs and User interface of the application and dashboard.

**Chapter 7:** A cognitive evaluation of the system using the test results. System testing is performed through a viable and relatable testing strategy to cover all the use cases of application.

**Chapter 8:** Future Direction discuss the possible directions available to grow زبان دست.

**Chapter 9:** A comprehensive list of references is cited using a standard format.

## *Chapter 2*

### **LITERATURE REVIEW**

A literature review for a project is a comprehensive analysis and evaluation of existing literature and research related to a specific project.

Literature review is a crucial step for development and transformation of the idea of a new product. Likewise, a detailed study of existing tools and applications relevant is mandatory.

#### **Introduction:**

The internet has given people a powerful platform to express themselves and share their thoughts with the world. However, it has also given rise to certain issues like toxic content, fake news, and misuse of personal information. This has created a need for tools that can detect such content and protect people from its harmful effects. The app we are discussing in this literature review aims to provide solutions to these problems through its three features: Toxic Content Detection, Fake Content Detection, and Voice cloning. In this literature review, we will explore the existing research on these topics and analyze how the proposed app compares to them.

- **Toxic Content Detection:**

Toxic content on the internet can take many forms, including hate speech, bullying, and harassment. Several studies have explored the use of machine learning algorithms to detect such content. For instance, a study by

Balamuralidhar et al. (2020) used a deep learning model to identify hate speech in social media posts. The model achieved an F1 score of 0.78, indicating its effectiveness in detecting toxic content. Another study by Fortuna et al. (2020) explored the use of natural language processing techniques to detect online harassment. The study used an ensemble of machine learning algorithms to achieve an accuracy of 90% in detecting harassment.

- **Fake Content Detection:**

Fake news and misinformation are significant problems on the internet, and several studies have explored ways to detect them. A study by Rahman et al. (2020) used a machine learning algorithm to identify fake news articles. The algorithm achieved an accuracy of 90%, demonstrating its effectiveness in detecting fake content. Another study by Mukherjee et al. (2019) explored the use of deep learning models to detect manipulated images. The study achieved an accuracy of 89% in detecting manipulated images, demonstrating the potential of deep learning in detecting fake content. People's opinions and their decisions are affected by the fake news in the US election of 2016 (Dewey, 2016). Different researchers are working for the detection of fake news. The use of Machine learning is proving helpful in this regard. Researchers are using different algorithms to detect the false news. Researchers in (Wang, 2017) said that fake news detection is a big challenge. They have used machine learning for detecting fake news. Researchers of (Zhou et al., 2019) found that the fake news is increasing with the passage of time. That is why there is a need to detect fake news. The algorithms of machine learning are trained to fulfill this purpose. Machine learning algorithms will detect the fake news automatically once they have trained.

- **Voice Cloning:**

Voice cloning involves generating synthetic speech that sounds like a

particular individual. Several studies have explored the use of deep learning algorithms to achieve this. For instance, a study by Taigman et al. (2017) used a deep neural network to generate high-quality speech from text input. The study demonstrated the potential of deep learning in voice cloning applications. Another study by Jia et al. (2021) explored the use of a deep-learning algorithm to generate speech in multiple languages. The study achieved high-quality speech synthesis in different languages, demonstrating the potential of deep learning in multilingual voice cloning.

Hence, the literature review of the toolkit consists of the following components:

## **2.1 Deep Learning**

It is a type of machine learning modeled after the human brain, with complex multi-layered neural networks that allow data to be passed between nodes. In our Interim report, we initially stated that we would be using a neural network model called convolute.

The use of deep learning, specifically neural networks, has shown promising results in fake/Toxic Content Detection applications. Neural networks have the ability to process complex and sequential data and learn patterns from datasets, which can be beneficial for accurate Fake/Toxic Content Detection. Several studies have utilized deep learning techniques, such as Recurrent Neural Networks (RNNs), Bi-Directional Deep Learning Models, and Long Short-Term Memory (LSTM) networks, in combination with the TensorFlow library, to develop a Fake/Toxic Content Detection application with high accuracy and real-time performance

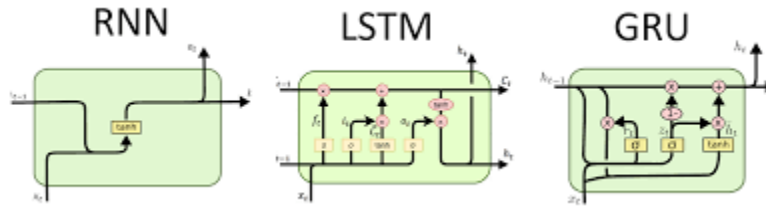


Figure 1: RNN, LSTM, AND GRU

```

✓ [2] import os
      import re
      import sys
      import numpy as np
      import pandas as pd

      import matplotlib.pyplot as plt
      %matplotlib inline
      from focal_loss import SparseCategoricalFocalLoss
      from tensorflow.keras.preprocessing.text import Tokenizer
      from tensorflow.keras.preprocessing.sequence import pad_sequences
      from tensorflow.keras.layers import RNN, GRU, LSTM, Dense, Input, Embedding, Dropout, Activation, concatenate
      from tensorflow.keras.layers import Bidirectional, GlobalAveragePooling1D, GlobalMaxPooling1D, dot, multiply
      from tensorflow.keras.models import Model
      from tensorflow.keras.callbacks import EarlyStopping, ModelCheckpoint
      from tensorflow.keras import initializers, regularizers, constraints, optimizers, layers
  
```

We will be developing this project for the Urdu Language, we have performed all the deep learning tasks with a python library called Keras. It was designed to simplify the most common tasks in machine learning and has extensive documentation, guides, and pre-existing projects that can be used to learn how to use the library.

## 2.2 Urdu Language

- Urdu is the most important language of South Asia. South Asia’s markets are growing rapidly and it is providing relevant content on the internet, making it a lucrative consumer market.
- Urdu grammar is as easy as English grammar. It consists of all the grammatical rules of English so people find it easy to learn Urdu. Learning Urdu depends upon the learner’s capability. You can get a grip on the Urdu language which is writing,

speaking, and reading in six months.

- Urdu is widely spoken in Pakistan and in many parts of India, Nepal, Bangladesh, the Middle East, and many other parts of the world. If you look at India, Urdu is spoken by large Muslim communities and in the cities like Delhi, Bhopal, Uttar Pradesh, and Hyderabad which are also known as the power hub of India.
- Urdu is the national language of Pakistan and one of the official languages of India.
- It is spoken by approximately 100 million people worldwide, making it one of the most widely spoken languages in the world.
- The Urdu language has a rich literary tradition, with works of poetry and prose dating back centuries.
- Urdu is a language that is often used in the media, including television, radio, and newspapers, and is also used in government and business.
- Urdu is also used as a medium of instruction in many educational institutions in Pakistan and India, from primary schools to universities.
- Urdu has been the subject of linguistic and literary studies, as well as research on its cultural and historical significance.
- The use of Urdu language is increasing globally with the rise of social media and the internet, creating new opportunities for its use in various fields.

### **2.3 Fake/Toxic Content Detection with Deep Learning**

Real-time and Accurate performance is a critical aspect of Fake/Toxic Content Detection applications, as these applications need to provide timely as well as accurate Detection. Deep learning techniques, particularly with the use of TensorFlow, have shown promising results in achieving real-time performance



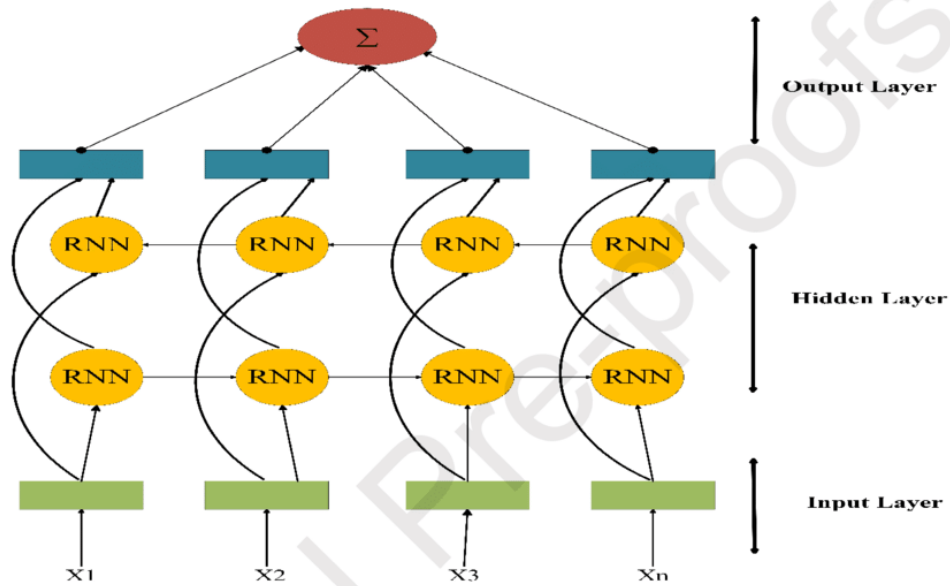


Figure 2: Bi-Directional Neural Network

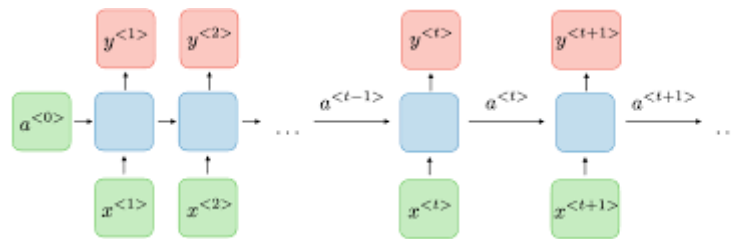


Figure 3: Recurrent Neural Network

## 2.4 Limitations and Areas of Improvement

- Despite the potential benefits, the proposed toolkit also has some limitations that need to be addressed. One of the main limitations is the accuracy of the detection algorithms. While the toolkit can detect toxic and fake content to some extent, it may not be able to catch all instances of such content. The accuracy of the voice cloning feature may also be limited by the quality of the input voice note and the text messages.
- Another limitation is the availability of the toolkit for only one language (Urdu).

This may limit its usability for users who speak other languages. To overcome this limitation, the toolkit could be expanded to include multiple languages, thus making it accessible to a wider audience.

- There is another limitation of availability of fake content and toxic content data in urdu language for AI. This is also the reason that we couldn't achieve the high accuracy in the deep learning models.
- Furthermore, the toolkit may not be able to handle complex and nuanced content, such as sarcasm or irony, which can be difficult to detect through automated means. This can lead to false positives or false negatives, reducing the effectiveness of the toolkit.
- To address these limitations, several areas of improvement can be considered. One potential area of improvement is the integration of natural language processing techniques, which can help the toolkit to better understand the context and nuances of language. This can improve the accuracy of the toxic and fake content detection features.
- Another potential area of improvement is the use of more advanced machine learning algorithms that can better handle complex and nuanced content. For instance, deep learning techniques such as Transformer based model used in ChatGPT-3, can be used to improve the accuracy of the toolkit.
- Lastly, the toolkit could benefit from the incorporation of user feedback mechanisms that allow users to report any inaccuracies or false positives/negatives. This can help to continuously improve the accuracy and effectiveness of the toolkit over time.

## **2.5 Existing solutions and their drawbacks**

- Existing solutions for the three features of the proposed toolkit - toxic content detection, fake content detection, and voice cloning - can be found in various commercial and open-source products. However, these solutions have some drawbacks and limitations that prevent them from providing accurate and reliable

results. In this section, we will discuss the drawbacks of some of the existing solutions for each of the three features.

- **Toxic Content Detection:** Several tools and services exist that claim to detect toxic content in text or voice messages. One of the most widely used services for this purpose is Google's Perspective API. This API provides a toxicity score for text input based on its analysis of various linguistic features such as grammar, syntax, and word choice. However, the API has been criticized for providing inaccurate results in Urdu language and being biased against certain groups of people. Additionally, the API is not suitable for analyzing audio or video input.
- Another existing solution is the open-source tool called Perspective Lite. This tool is a simplified version of the Perspective API that can be integrated into web applications. However, its accuracy is also limited, and it cannot handle audio or video input.
- **Fake Content Detection:** Fake content detection is another challenging task, and several tools and techniques have been developed to address it, but no work has been done in the Urdu language.
- One of the approach is to use machine learning models to detect fake content. These models are trained on a dataset of real and fake content and can classify new content based on its similarity to the training data. However, these models have limited accuracy, and their performance may degrade over time as new types of manipulations are introduced.
- **Voice Cloning:** Voice cloning is a relatively new technology that has gained popularity in recent years. Existing solutions for voice cloning typically use deep learning models to generate a synthetic voice that sounds like the original speaker. However, these models require a large amount of high-quality training data and can be computationally expensive.
- One of the most popular open-source tools for voice cloning is Tacotron 2, developed by Google. This tool uses a sequence-to-sequence model to generate speech from text input. However, its output can sound robotic or unnatural, and it may require additional post-processing to improve the quality.

- Another existing solution is the commercial product called Lyrebird, which claims to generate highly realistic synthetic voices. However, this product requires a significant amount of training data, and its pricing model is based on the number of generated voice minutes, which can be expensive for some users.
- In conclusion, while several existing solutions exist for the three features of the proposed app, they all have their limitations and drawbacks. Therefore, there is a need for a more accurate, reliable, and accessible solution to address these challenges.

## *Chapter 3*

### **PROBLEM DEFINITION AND WORKING ELEMENTS**

Fake news, toxic content, and voice cloning in the Urdu language is that these issues pose significant challenges for creating a safer and more trustworthy digital world. Fake news can lead to social unrest, economic instability, and even violence. Abusive content and hate speech can cause emotional distress, harm mental health, and lead to physical harm. Voice cloning can be used for fraudulent activities, such as creating fake audio messages or deep fakes. These issues are particularly challenging in Urdu, as there is a lack of tools and resources to detect and combat these problems effectively. Therefore, developing a toolkit tailored specifically to Urdu is essential for addressing these issues and creating a safer online environment for Urdu speakers.

The importance of addressing the issues of fake news, toxic content, and voice cloning in the Urdu language cannot be overstated. The spread of misinformation and hate speech can have significant negative consequences, such as social unrest and even violence. These issues erode trust in online communication and create an environment of fear and uncertainty. Voice cloning can be used for fraudulent activities, further damaging the trustworthiness of online communication. By addressing these issues, we can create a safer and more trustworthy digital world where people can communicate freely without fear of being misled or harmed. The development of a toolkit tailored specifically to the Urdu language can play a crucial role in addressing these issues, helping to create a safer and more inclusive online environment

### **3.1 Working Elements and Techniques**

The following components and techniques will feature the application towards its goal:

### **3.1.1 Machine Learning Model**

Our machine learning model will use algorithms to meet our application's needs. Our model will be trained with compatible data to allow us the use of deep learning and provide good results.

### **3.1.2 User Interface**

The goal of user interface design is to make it easy, efficient, and enjoyable for the users while working with the application. In this project, we have used HTML, CSS, and JavaScript (React) to design the layout of our web application's interface and Java for the android application.

### **3.1.3 Deep Learning**

Deep learning is a machine learning approach and it depends on neural networks that produce a very good accuracy. These deep neural networks require a huge dataset to learn and find a pattern between data. We made use of the Tensor Flow Library.

### **3.1.4 Programming Languages**

A programming language is designed to provide the medium of communication between user instructions and a machine. Therefore, programming language is to control the behavior of a machine, express and implement algorithms, and to create applications.

Following programming languages are utilized in our project for different aspects:

- Using JavaScript (React) for Web App.
- Using Python in model.
- Using JAVA for Android Application

# **METHODOLOGY**

## **4.1 Project Steps**

- Study and analysis the project idea from all aspects.
- Search and choose the best algorithm that applies our project idea.
- Modeling and simulating our system.
- Solve problems and errors in the system.
- Deploy and implement the system in the real world.
- Test the accuracy of the app and machine-learning model
- Evolution of the system in the future.

## **4.2 Working Principle**

The system work on the RNN bidirectional model. It takes input in the form of text or audio and processes it to deliver the required output. Regular updates and improvements to the system may be necessary to enhance its accuracy and usability.

Here is a detailed overview of the typical working principle for our application:

### **4.2.1 Data Acquisition:**

Our application accept data as text or audio. We can enter text or audio files or simply enter the desired data through the system interface

### **4.2.2 Preprocessing:**

The data is then processed. From the Audio we get the text using speech recognition model, after getting the text extract from the audio file or from he input text, it is tokenized using Tokenizer class, After the tokenizer is fitted on the text data, it can be used to convert the text data into sequences of integers using the text\_to\_sequences

method. This method takes a list of text data as input and returns a list of sequences of integers, where each integer represents a word in the vocabulary. The next step after converting the text data into sequences of integers is typically to pad the sequences to a fixed length using the `pad_sequences` function from the `tensorflow.keras.preprocessing.sequence` module in Python. Padding is necessary to ensure that all sequences have the same length so that they can be processed by machine learning models. The `pad_sequences` function takes the sequences of integers as input and pads them with zeros to a fixed length specified by the `maxlen` parameter, then this is used as a input in the model for training

#### **4.2.3 Fake Content:**

Our trained model for fake content identification is a combination of machine learning algorithms and natural language processing techniques. It is trained on a dataset of both real and fake news articles, which are labeled by human annotators to indicate whether they are real or fake. The dataset is used to train the model to identify patterns in the language, sentiment, and content of the articles that are associated with real or fake news.

The machine learning algorithm used in the model is typically a deep neural network that is trained using backpropagation to adjust the weights of the nodes in the network. The natural language processing techniques used in the model include text tokenization recognition.

Once the model is trained, it can be used to classify new news articles as either real or fake. This is done by inputting the text of the article into the model and using its predictive capabilities to determine the probability that the article is fake. The model outputs a score, which can be compared to a threshold value to determine whether the article is classified as real or fake.

The features used by the model to determine the probability that an article is fake include the language used in the article, the sentiment of the article, and the content of the article. For example, the model may identify certain words or phrases that are commonly



associated with fake news, or it may detect inconsistencies in the content of the article that suggest that it is not factual.

In order to improve the accuracy of the model, it may be necessary to continue to train it on new data and adjust its parameters as needed. Additionally, the model may need to be updated to account for new types of fake news or changes in the way that fake news is spread.

#### **4.2.4 Toxic Content:**

Work has been done in Toxic content in two parts as described below,

**Subtask A** focuses on detecting Abusive language using Twitter tweets in the Urdu language. This is a binary classification task in which participating systems are required to classify tweets into two classes, namely: Abusive and Non-Abusive

Abusive - This Twitter post contains some sort of abusive content.

Non-Abusive - This Twitter post does not contain any abusive or profane content.

**Sub-task B** focuses on detecting Threatening language using Twitter tweets in the Urdu language. This is a binary classification task in which participating systems are required to classify tweets into two classes, namely: Threatening and Non-Threatening.

Threatening - This Twitter post contains any threatening content.

Non-Threatening - This Twitter post does not contain any threatening or profane content.

We combined both tasks into one and used a single model to deal with it.

Our trained model for detecting abusive and threatening content in the Urdu language will work by analyzing the text data and using natural language processing techniques to identify the potentially harmful language. The model will use a combination of rule-based and machine-learning approaches to identify patterns and characteristics commonly associated with abusive and threatening language.

The model will be trained on a dataset of labeled text data that includes examples of abusive and threatening language. The data will be preprocessed to remove noise and irrelevant information and then fed into the model for training. During the training process, the model will learn to identify patterns and features that are indicative of abusive and threatening language, such as the use of derogatory terms, violent language, or direct threats.

Once the model is trained, it can be used to classify new text data as either abusive and threatening or normal content. The text data can be fed into the model through an API or a user interface, and the model will output a prediction of whether the text contains abusive and threatening language or not.

To ensure the accuracy and effectiveness of the model, it will be continuously updated and refined based on feedback and new data. This will involve retraining the model on new data and incorporating new features and techniques to improve its performance..

#### **4.2.5 Voice Cloning**

Our trained model for detecting voice cloning in Urdu language will work by analyzing the audio data and using signal processing techniques to identify potential anomalies or inconsistencies in the voice signal. The model will use a combination of rule-based and machine learning approaches to identify patterns and characteristics commonly associated with voice cloning.

The model will be trained on a large dataset of labeled audio data that includes examples of genuine and cloned voices. The audio data will be preprocessed to remove noise and irrelevant information, and then fed into the model for training. During the training process, the model will learn to identify patterns and features that are indicative of cloned voices, such as the presence of unnatural sounding pitch shifts, artifacts, and other signs of manipulation.

Once the model is trained, it can be used to classify new audio data as either genuine or cloned. The audio data can be fed into the model through an API or a user interface, and the model will output a prediction of whether the voice is genuine or cloned.

To ensure the accuracy and effectiveness of the model, it will be continuously updated and refined based on feedback and new data. This will involve retraining the model on new data and incorporating new features and techniques to improve its performance.

It's worth noting that working for languages like Urdu is a complex and evolving field, and the methodology for work may vary depending on the specific approach, techniques, and technologies used. The accuracy and performance of our models also depend on factors such as the quality of the data, the size and diversity of the training dataset, the choice of recognition algorithms, and the user interface design. Regular updates and improvements to the system may be necessary to enhance its accuracy and usability.

## DETAILED DESIGN AND ARCHITECTURE

### 5.1 ARCHITECTURAL DESIGN

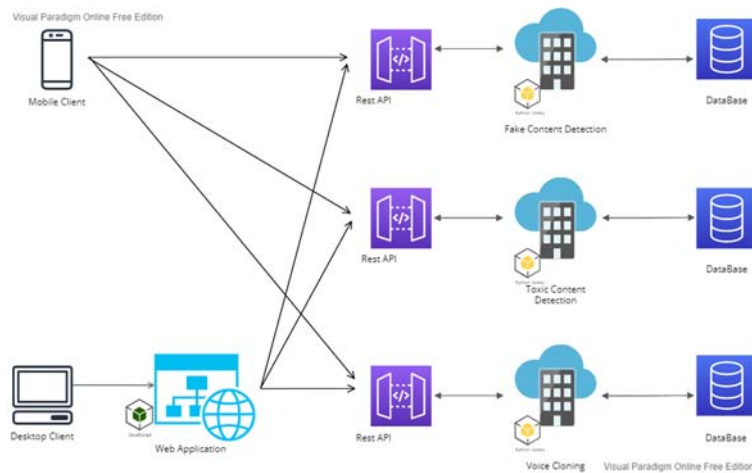


Figure 4: Architectural Design

The overall architecture of this project is micro services architecture. The project is divided into client side interface, API, micro services and the storage. The services will be running independently having their own database storage all the data communication will be done through API's from client to the micro services.

In this System the interface will be running on web browser as well as on mobile app.

### 5.2 COMPONENT DESIGN

#### Components

##### 1. Text Preprocessor

Purpose: This component is responsible for cleaning and normalizing and preparing text data before it is passed to other model components. The text is tokenized in the units of words. After the tokenizer is fitted on the text data, it can be used to convert the text data into sequences of integers using the `text_to_sequences` method. This method takes a list of text data as input and returns a list of sequences of integers, where each integer represents a word in the vocabulary. The next step after converting the text data into sequences of integers is typically to pad the sequences to a fixed length using the `pad_sequences` function from the `tensorflow.keras.preprocessing.sequence` module in Python. Padding is necessary to ensure that all sequences have the same length so that they can be processed by machine learning models. The `pad_sequences` function takes the sequences of integers as input and pads them with zeros to a fixed length specified by the `maxlen` parameter.

Inputs: Raw text data in the Urdu language.

Outputs: Preprocessed data used for model training.

## **2. Voice Cloner**

Purpose: This component is responsible for generating a realistic copy of a person's voice in the Urdu language based on a limited amount of sample audio.

Inputs: Sample audio of a person speaking in the Urdu language.

Outputs: Audio that sounds like the person speaking in the Urdu language.

## **3. Fake Content Detector**

Purpose: This component is responsible for identifying and labeling text that is intentionally misleading or untruthful in the Urdu language.

Inputs: Preprocessed text data in the Urdu language.

Outputs: Text data with fake content labeled in the Urdu language.

## **4. Toxic Content Detector**

Purpose: This component is responsible for identifying toxic or inappropriate content in

given text data. It should be able to classify text as toxic or non-toxic based on a set of predefined rules or a machine learning model.

Inputs: Preprocessed text data in the Urdu language .

Outputs: Text data with toxic content labeled in the Urdu language.

### 5.3 DECOMPOSITION DESCRIPTION

#### 5.3.1 Data Flow Diagram:

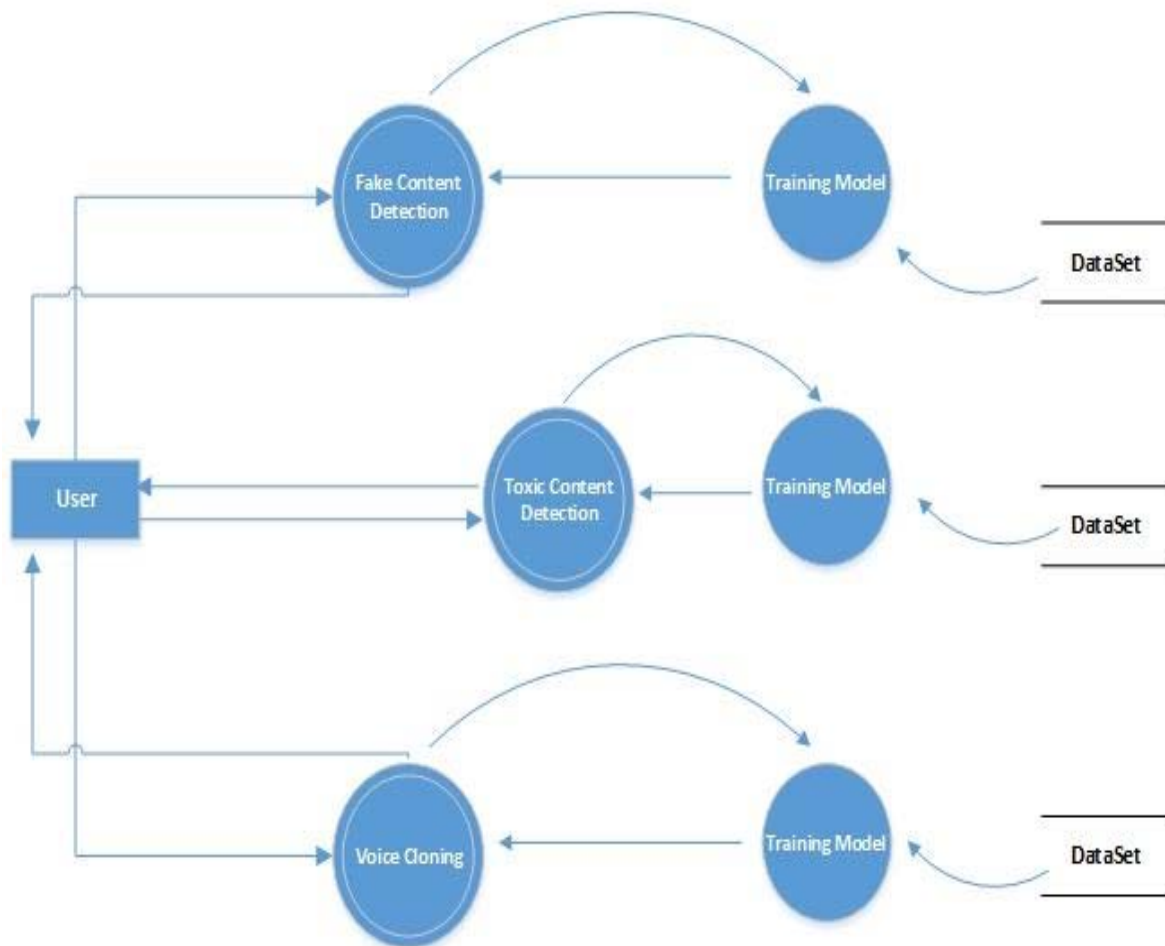


Figure 5: Data Flow Diagram

### 5.3.2 Structural Decomposition Diagram:

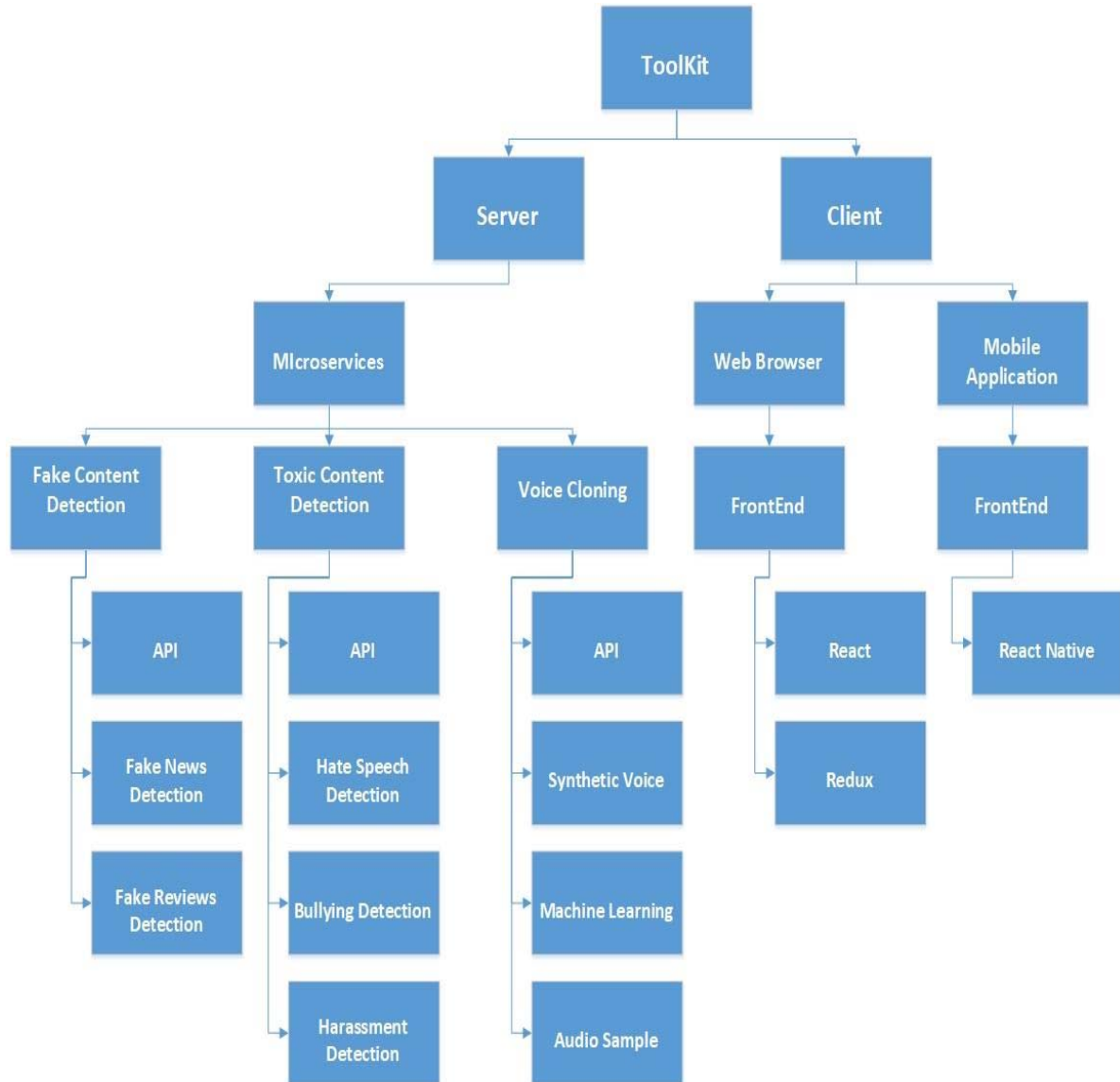


Figure 6: Structural Decomposition Diagram

### 5.3.3 CLASS DIAGRAM

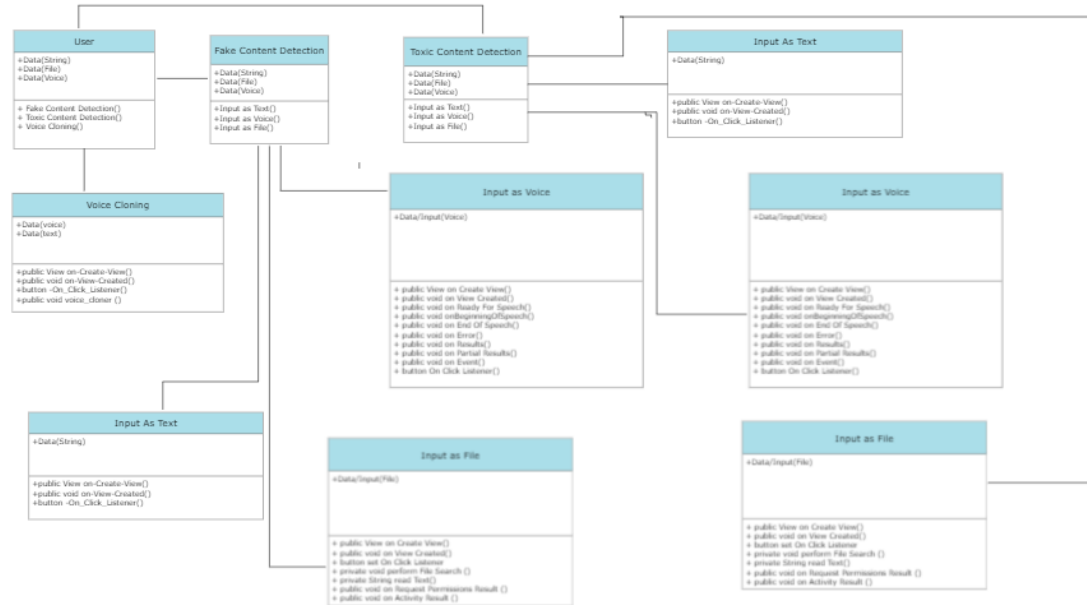


Figure 7: Class Diagram

### 5.3.4 USE CASE DIAGRAM



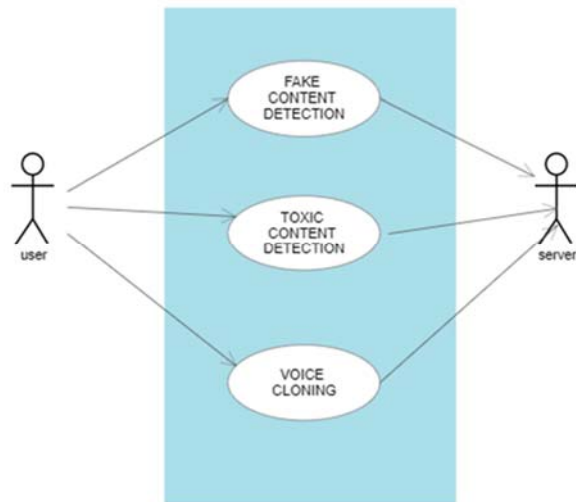


Figure 8: Use Case Diagram

## 5.4 SEQUENCE DIAGRAM:

- **Toxic Content Detection**

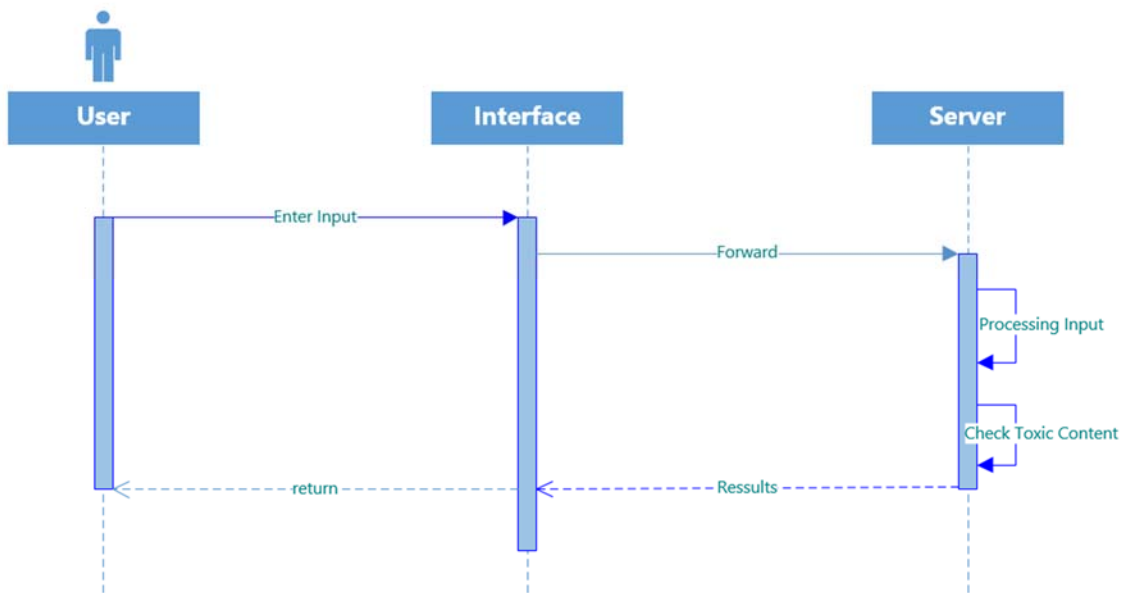


Figure 9: Sequence Diagram (Toxic Content Detection)

- **Fake Content Detection**

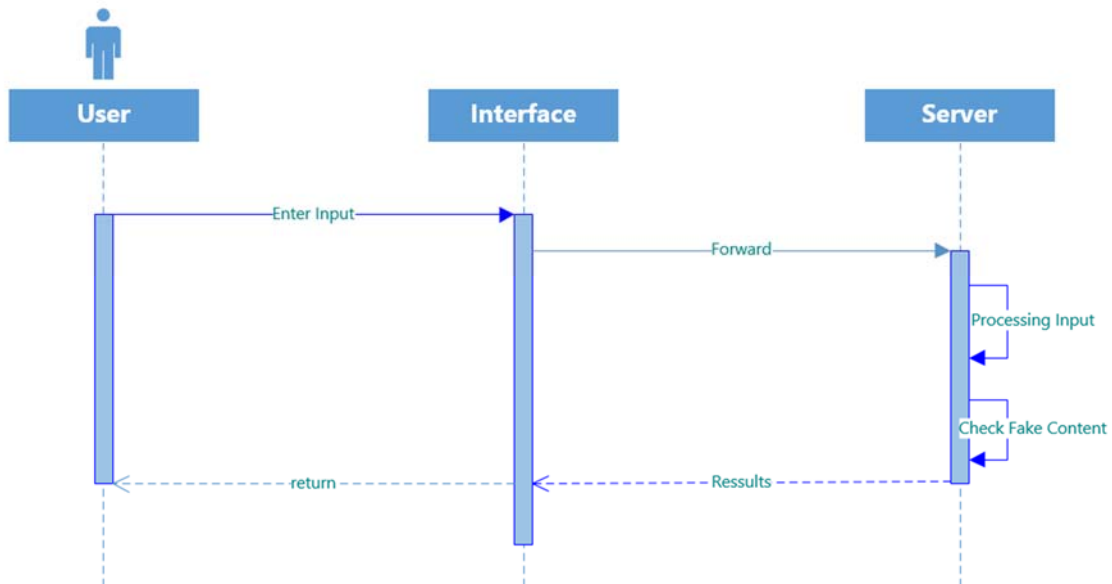


Figure 10: Sequence Diagram (Fake Content Detection)

- **Voice Cloning**

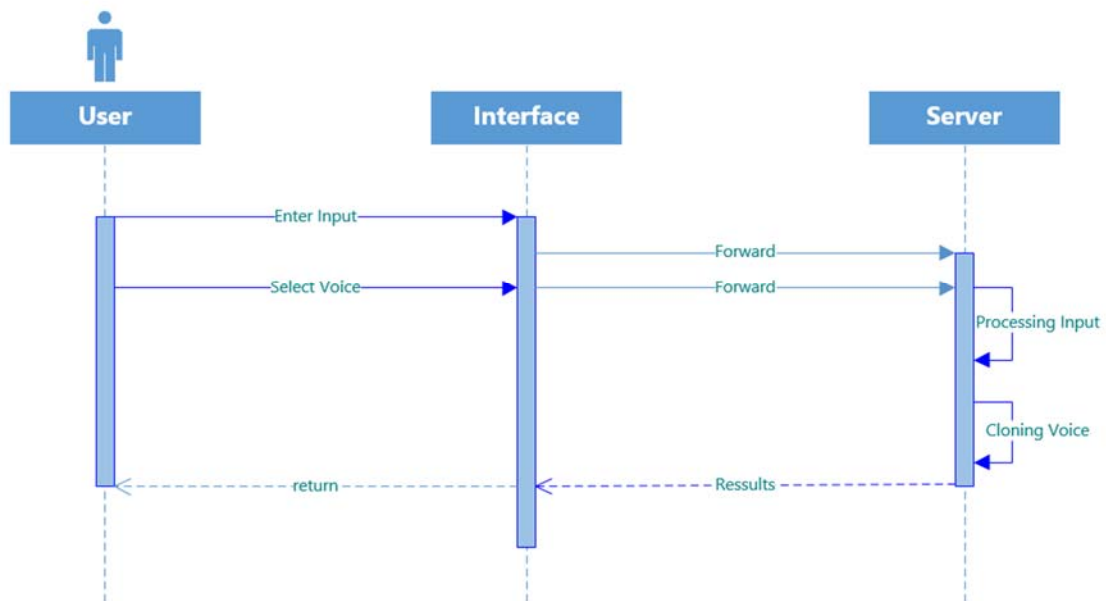


Figure 11: Sequence Diagram(Voice Cloning)

## **5.5 ACTIVITY DIAGRAM**

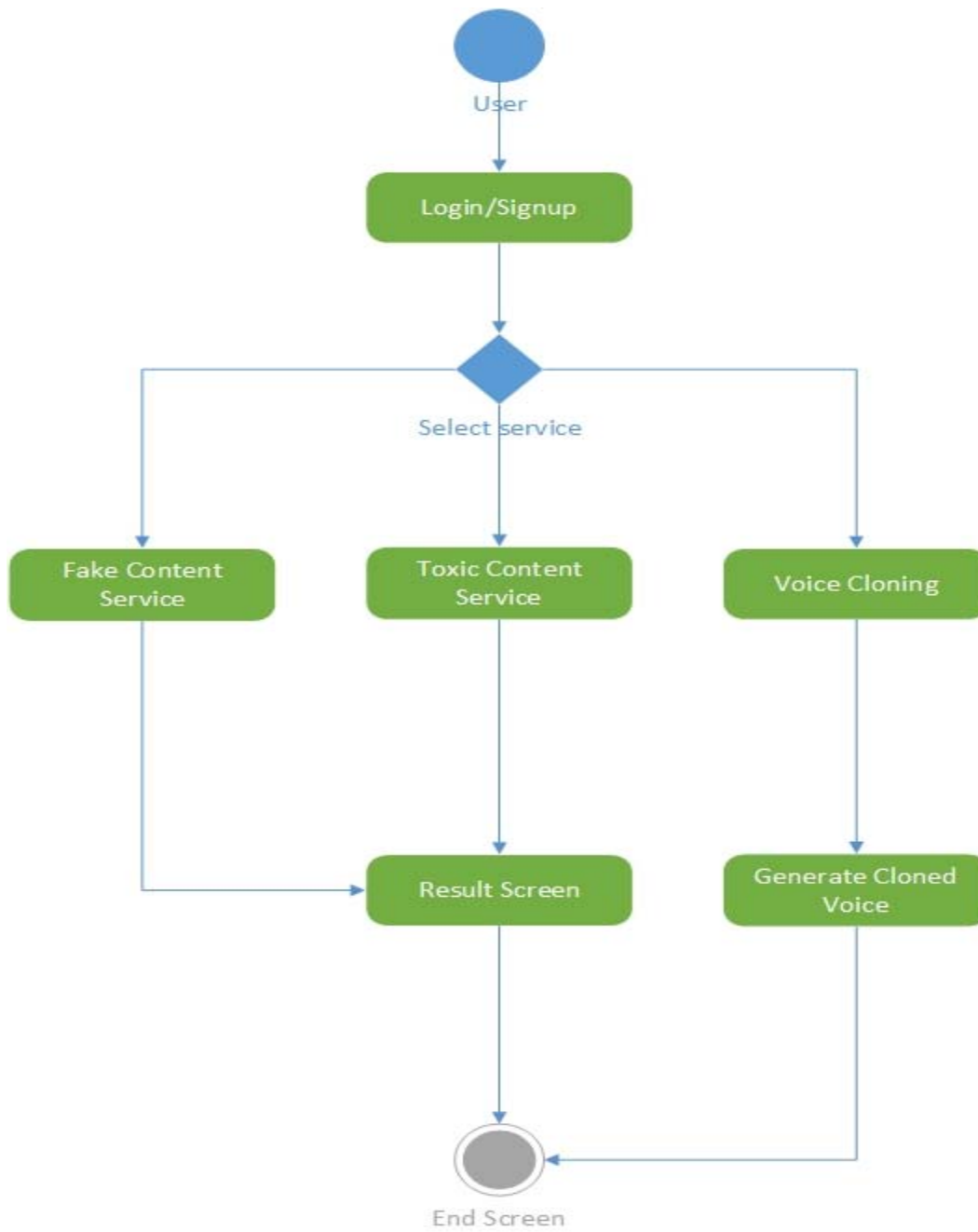


Figure 12: Activity Diagram

## 5.6 DICTIONARY

<b>Object</b>	<b>Attributes</b>	<b>Data Type</b>	<b>Description</b>
Article	Article text	String	This attribute contain the article to be processed
	Article Title	String	The title of the article
Comments	Comment text	String	This contain the comments
User	user name	String	The user name of the user
	user email	Email	The email of the user
Voice	Audio file	Audio	The audio file that is to be processed
	Sentence	String	The text of the sentence that is spoken in the audio clip
	Gender	String	The gender of the voice

### **USER INTERFACE**

The user interface is a graphical interface that will be developed to provide a user-friendly platform for users to access and interact with the toolkit. The interface will allow users to Input files, text, or voice recordings

#### **6.1 Mobile Application:**

To ensure that the application can be used easily by the target users it has to be easy to accessed and learn without the need for extensive explanation and instructions. This part of the report will justify why the application looks the way it does, why its functions are very visually oriented. The application design is built on the idea to provide a virtual colored overlay on mobile devices replicating physical ones already in existence. The design of the application will try to focus on simplicity as to make it as simple as possible for users to use.

**Software Used:** Android Studio Electric Eel Version

**Language Used:** Java

**Color Scheme:**

Bronze #CD7532 - FOR UPLOAD,CLEAR ,BACK AND NEXT BUTTON

Gray #808080 – FOR DISPLAY RESULT BUTTON

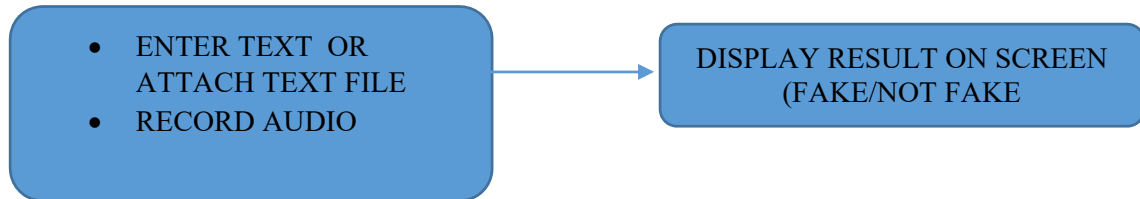
Dark Green #0C3B2E –FOR THE BACKGROUND

Lighter Shade of Green #6D9773 -For the Navbar

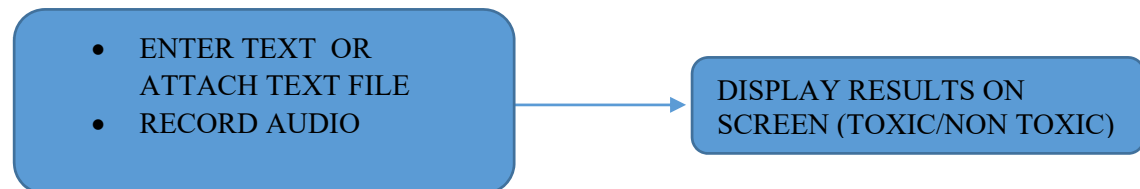
## 6.1.1 IMPORTANT FEATURES OF THE APP

‘A tool kit for Urdu Language using NLP’ performs following key functions:

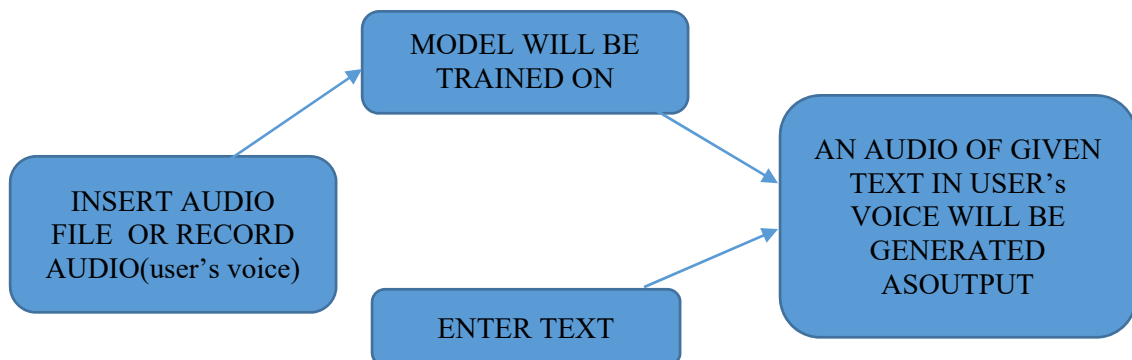
1. Detect fake content in text or audio.



2. Detect any toxic content in text and audio.



3. Voice Cloning to create a clone of an individual’s specific, unique voice.



## 6.1.2 HOW THE APP WORKS

Main Screen opens and contains a logo, a Navbar and a “NEXT” button that leads to next Page/Screen.

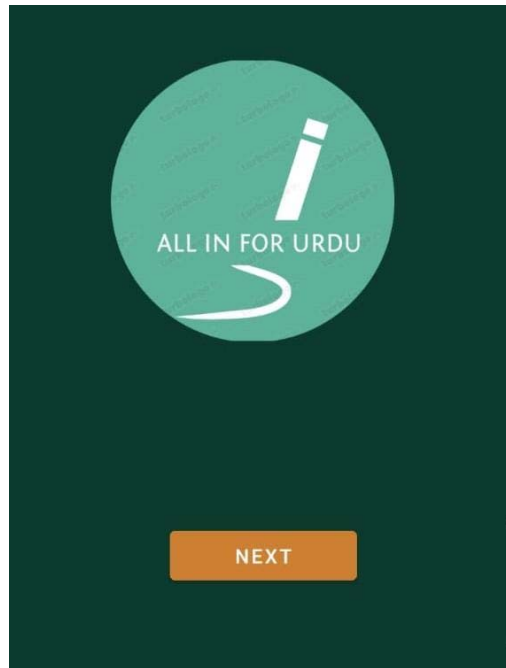


Figure 13: Main Screen of Mobile App



1. Three options are provided here which are the Main features of our App i.e

- Fake Content Detection
- Toxic Content Detection
- Voice Cloning

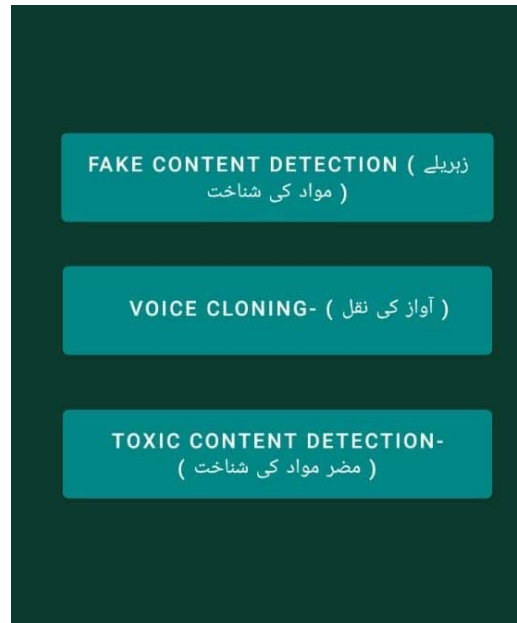


Figure 14: Features of the App

1. On Selecting Option, “**TOXIC CONTENT DETECTION**”, user will be directed to Screen where he’ll be provided with three options as follows:

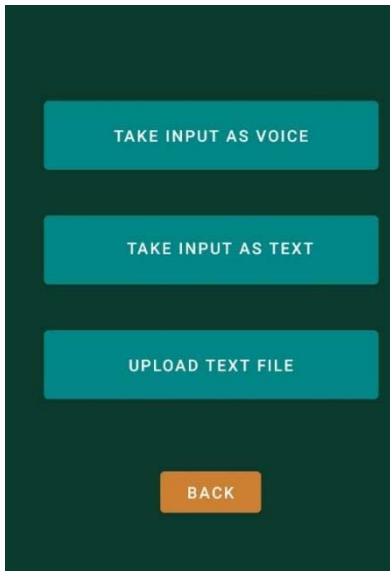
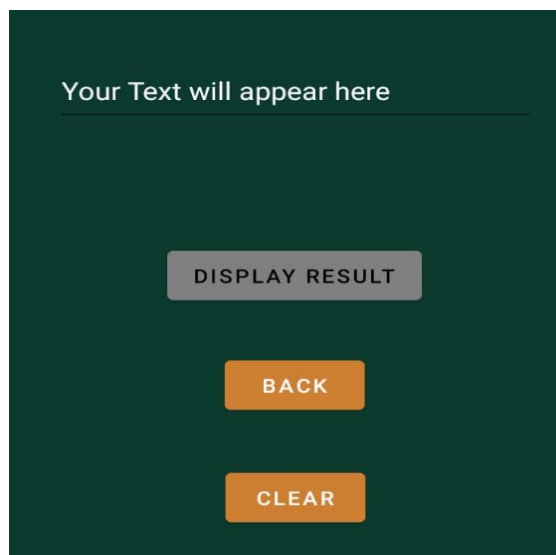


Figure 15: Types of Inputs for Toxic and Fake Content Detection

- **CASE -1**

*Input as Text*-where user will able to give input as text.



**Description:**

1. Back Button will get back to previous screen.
2. Clear Button will clear the Text
3. On clicking **Display** Result Button,

4. If no Text is Entered, a pop-up will appear as:

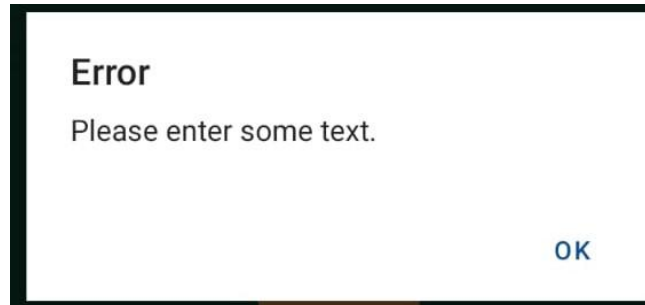


Figure 16: Error message

5. Else if the text is entered, It'll be classified as **TOXIC/NON TOXIC** and a pop-up will appear as

Text you entered is Toxic  
متن زہریلہ ہے۔

Figure 17: Result message

- **CASE-2**

***Input as Voice***-User will be able to provide the recording as input

**Description:**

1. User will be asked for the permission to record voice first using:

```
(ActivityCompat.requestPermissions(requireActivity(), new  
String[]{Manifest.permission.RECORD_AUDIO},  
PackageManager.PERMISSION_GRANTED);
```

Note in the Manifest file:

```
<uses-permission android:name="android.permission.RECORD_AUDIO"/>
```

2. Once the Permission is Granted, On First Selecting the Option “Voice”, the recording will start. Pressing it again will stop the recording and the text will appear on the TextView.
3. Once the text appears on the TextView, rest of the steps will be followed as in case-1.

- **CASE-3:**

User will be able to upload Text File as input

**Description:**

1. To be able to upload TextFile as input, first the Permission to excess the internal Storage is required.

```
ActivityCompat.requestPermissions(requireActivity(), new  
String[]{Manifest.permission.READ_EXTERNAL_STORAGE},  
PackageManager.PERMISSION_GRANTED)
```

Note In the Manifest File

```
<uses-permission  
android:name="android.permission.WRITE_EXTERNAL_STORAGE"/>
```

2. Once the Permission is Granted, clicking on “Upload” Button will lead to the storage files where user will be able to select .txt file.
3. Text from the .txt file will appear on the TextView as soon as you select the file.
4. Once the text appears on the TextView, rest of the steps will be followed as in case-1.

2. On Selecting Option, “**VOICE CLONING**”, user will be directed To Screen where he’ll be asked to provide
  1. Voice in which he wants the text to be cloned
  2. Text he wants to cloned
  3. Result

User has to provide either of two inputs i.e

1. Audio Recording and Text to be Cloned
2. Audio File and Text to be Cloned

- **CASE-1:**

User provides Input as Text and Audio Recording

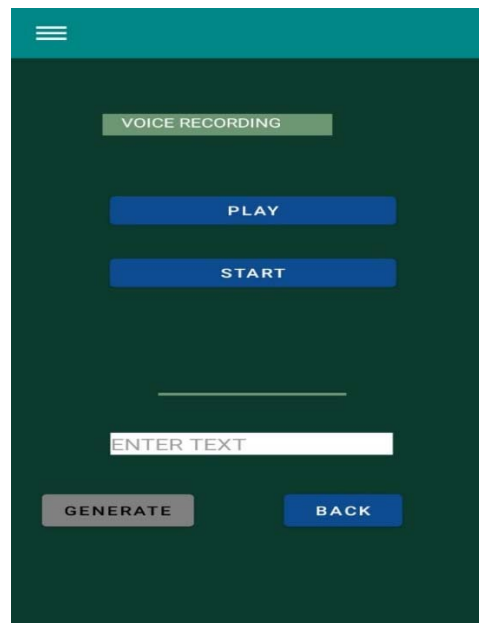
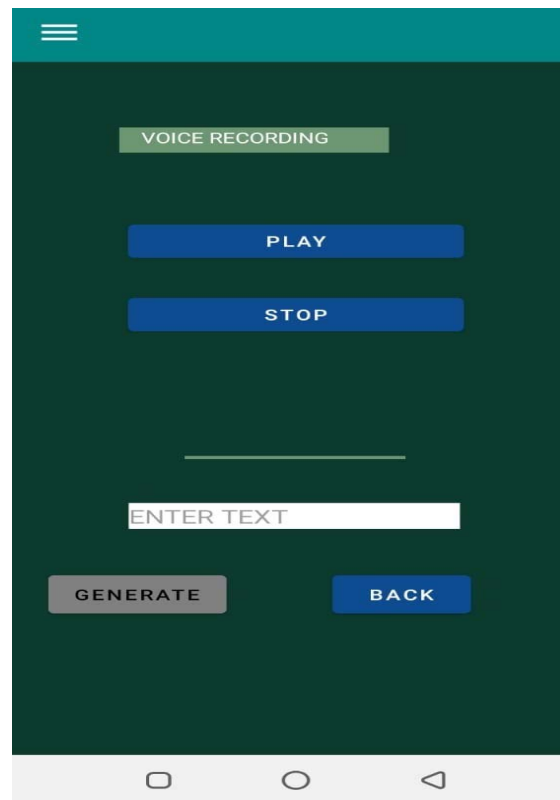


Figure 18 Voice Cloning

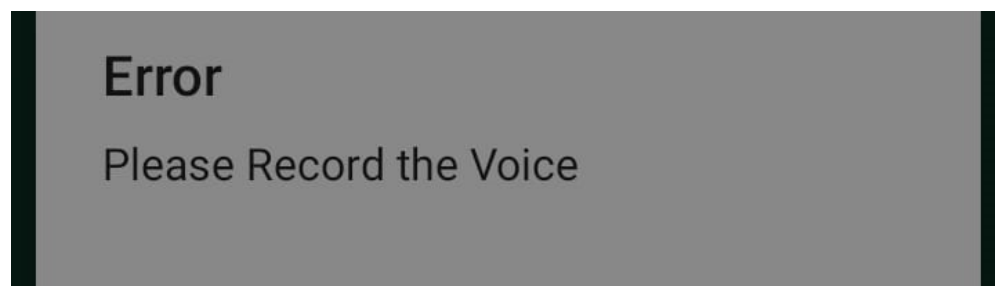
## Description:

1. On Pressing the “Start” Button, Recording will start and “Start” button will become “STOP” button. On pressing it, the Recording will stop. You can listen to recorded audio by Pressing “PLAY” button.

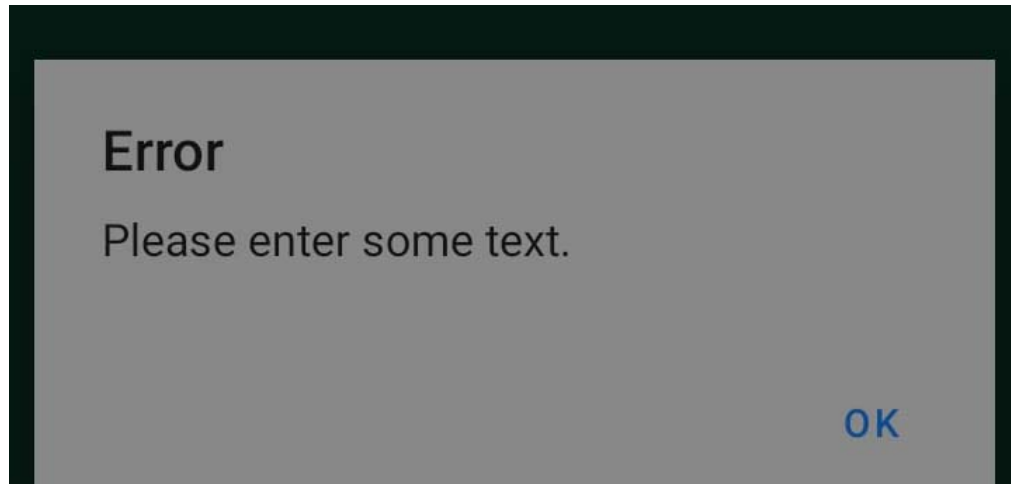


## 2. ON PRESSING “GENERATE” Button,

- If Recording is Empty, a pop-up will be generated as:



- If Text is not Entered, a pop-up will appear as:



- Else , a pop-up will appear as:

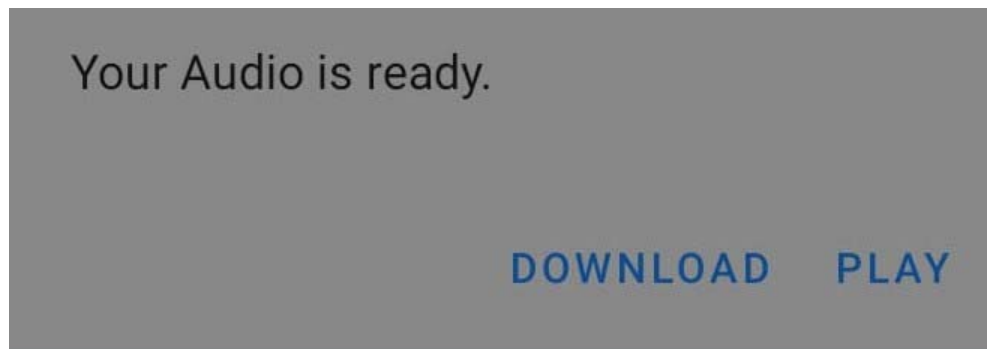
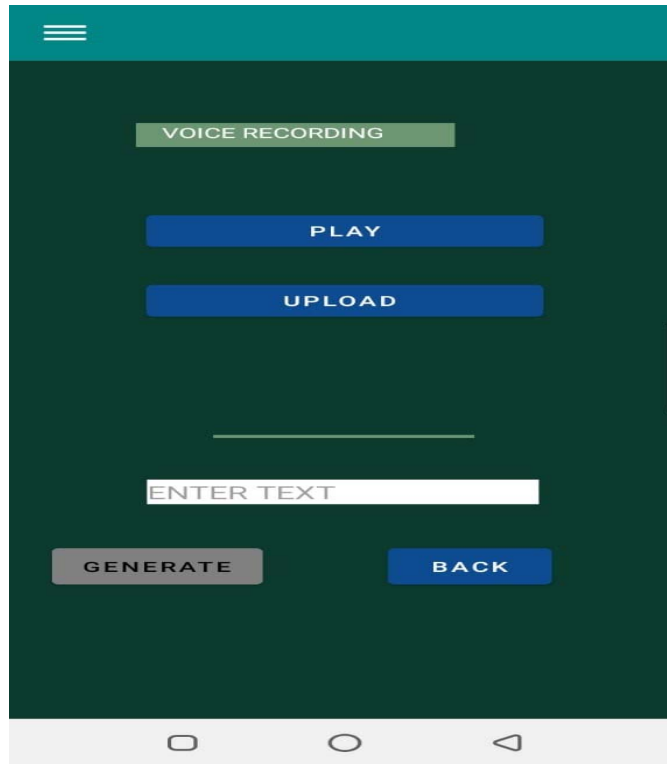


Figure 19 Download and Play the Cloned Voice

You can play as well as Download the Cloned Audio

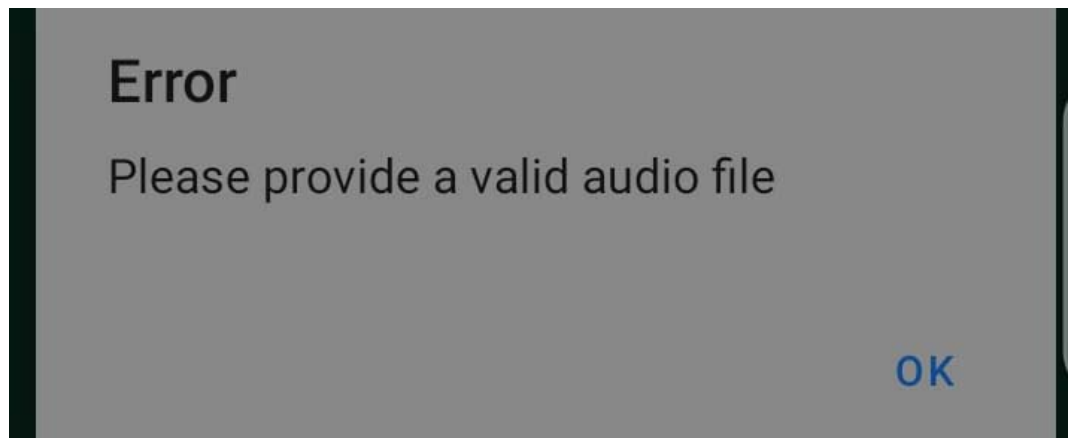
- **CASE-2:**

User provides Input as Text and Audio File.



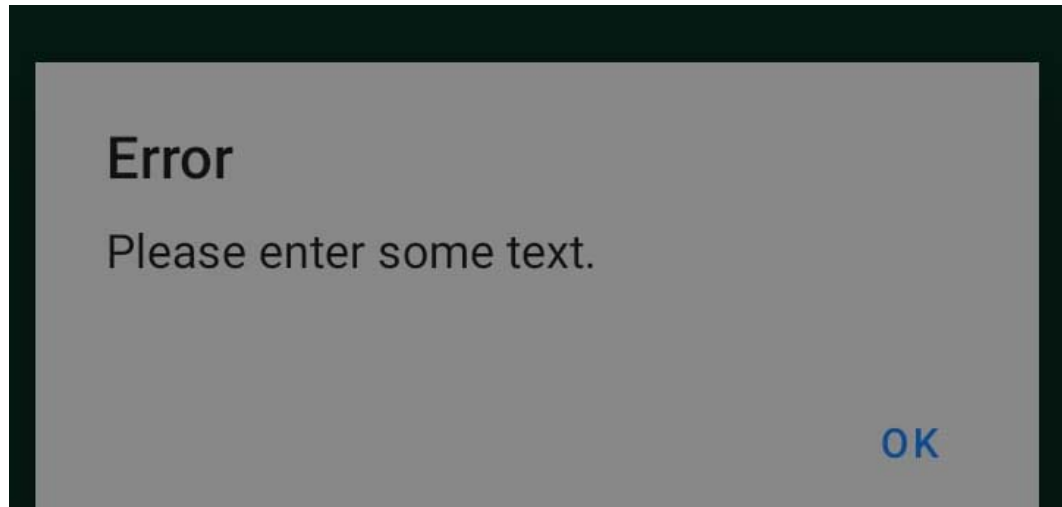
**Description:**

1. On Pressing the “UPLOAD” Button, Audio will be uploaded  
You can listen to uploaded audio by Pressing “PLAY” button.
2. ON PRESSING “GENERATE” Button,
  - If file is Empty/doesn't contain Voice:

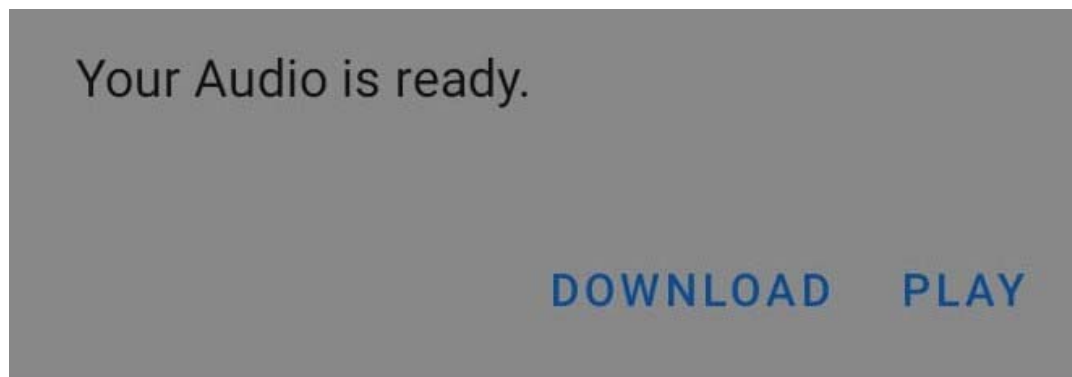


- If Text is not entered , a pop-up will appear as:





- Else, a pop-up will appear as:

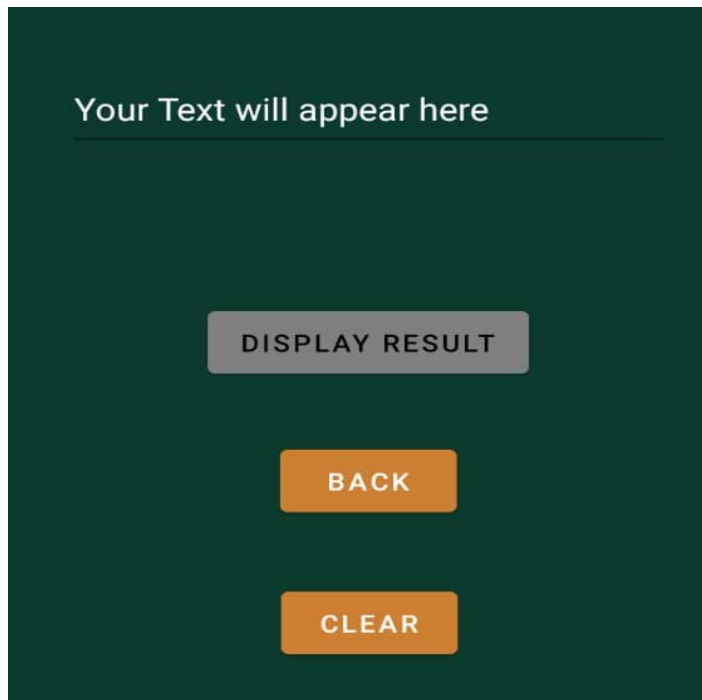


You can play as well as Download the Cloned Audio

3. On Selecting Option, “**FAKE CONTENT DETECTION**”, user will be directed to Screen where he’ll be provided with three options as follows:

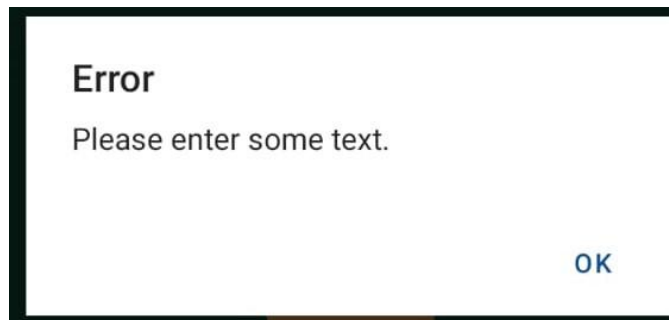
- **CASE -1**

- Input as Text*-where user will able to give input as text.



**Description:**

1. Back Button will get back to previous screen.
2. Clear Button will clear the Text.
3. On clicking Display Result Button,
4. If no Text is Entered, a pop-up will appear as:



Else if the text is entered, it'll be classified as FAKE or Not FAKE and a popup will appear as



The text you entered is fake.  
دیا گیا متن مہلک ہے۔

Figure 20 Result Message for Fake Content Detection

- **CASE-2**

Input As Voice-User will be able to provide the recording as input.

**Description:**

1. User will be asked for the permission to record voice first using:

```
(ActivityCompat.requestPermissions(requireActivity(), new  
String[]{Manifest.permission.RECORD_AUDIO},  
PackageManager.PERMISSION_GRANTED);
```

Note in the Manifest file:

```
<uses-permission android:name="android.permission.RECORD_AUDIO"/>
```

2. Once the Permission is Granted, On First Selecting the Option “Voice”, the recording will start. Pressing it again will stop the recording and the text will appear on the TextView.
3. Once the text appears on the TextView ,rest of the steps will be followed as in case-1.

- **CASE-3:**

User will be able to upload Text File as input

**Description:**

1. To be able to upload **TextFile as input**, first the Permission to access the internal Storage is required.

```
ActivityCompat.requestPermissions(requireActivity(), new  
String[] {Manifest.permission.READ_EXTERNAL_STORAGE},  
PackageManager.PERMISSION_GRANTED)
```

Note In the Manifest File

```
<uses-permission  
android:name="android.permission.WRITE_EXTERNAL_STORAGE"/>
```

2. Once the Permission is Granted, clicking on “Upload” Button will lead to the storage files where user will be able to select .txt file.
3. Text from the .txt file will appear on the TextView as soon as you select the file.
4. Once the text appears on the TextView, rest of the steps will be followed as in case-1.

### 6.1.3 IMPLEMENTATION:

The implementation will be explained here such as how the application is built, important code snippets and list of important functions used. The application UI screen is coded in XML while the main functions are coded with Java.

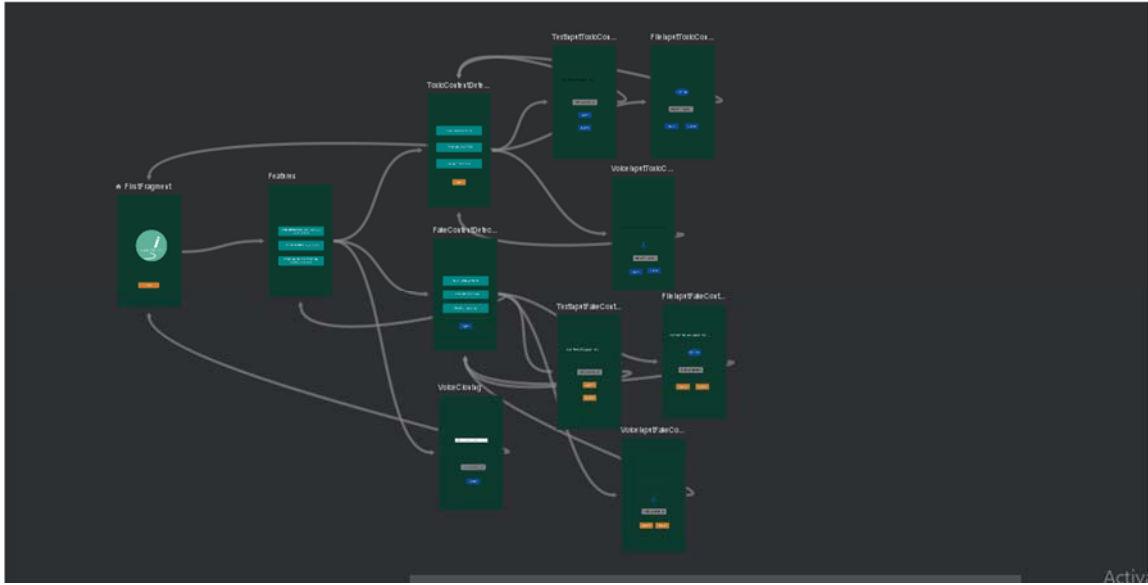
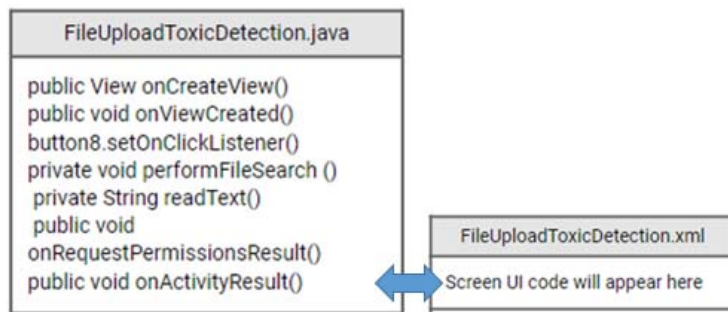


Figure 21: Navigation Graph of the Mobile App

Sample Diagrams where xml files will take inputs and Java File will process.



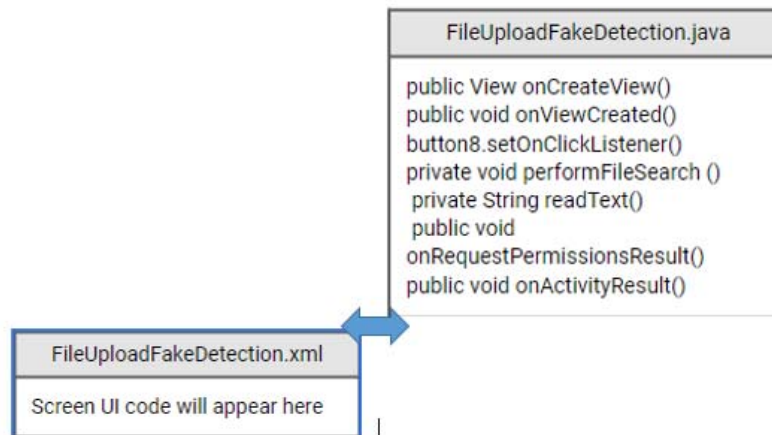


Figure 25

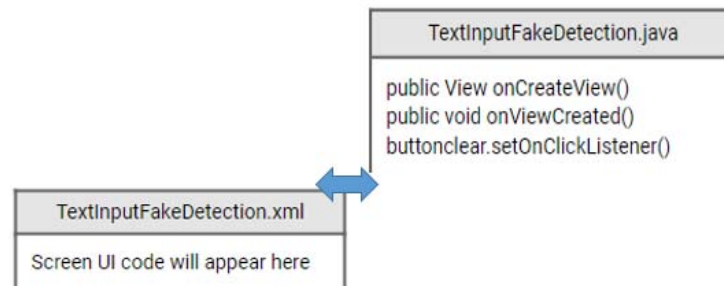


Figure 26

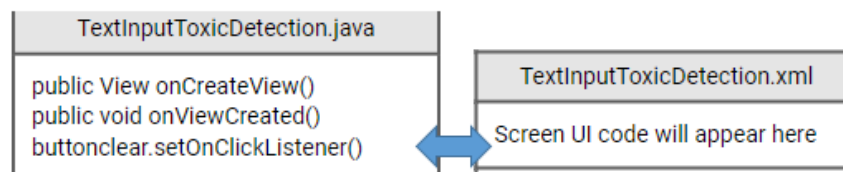


Figure 27

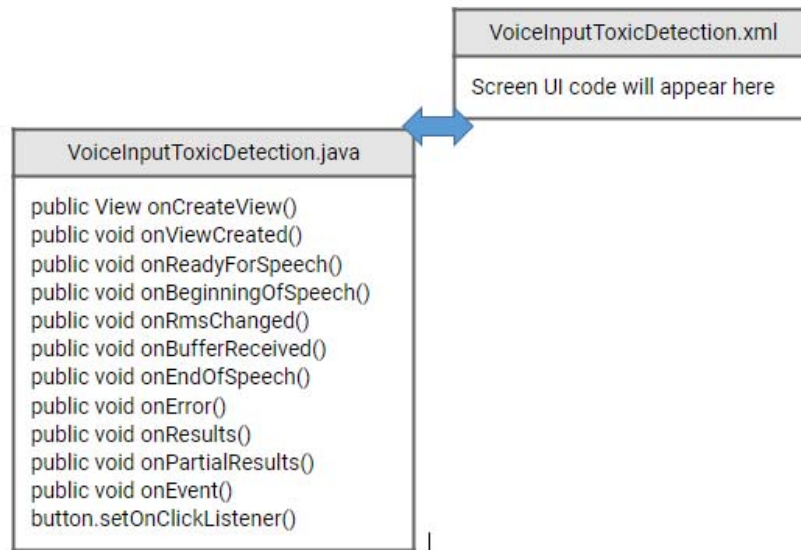
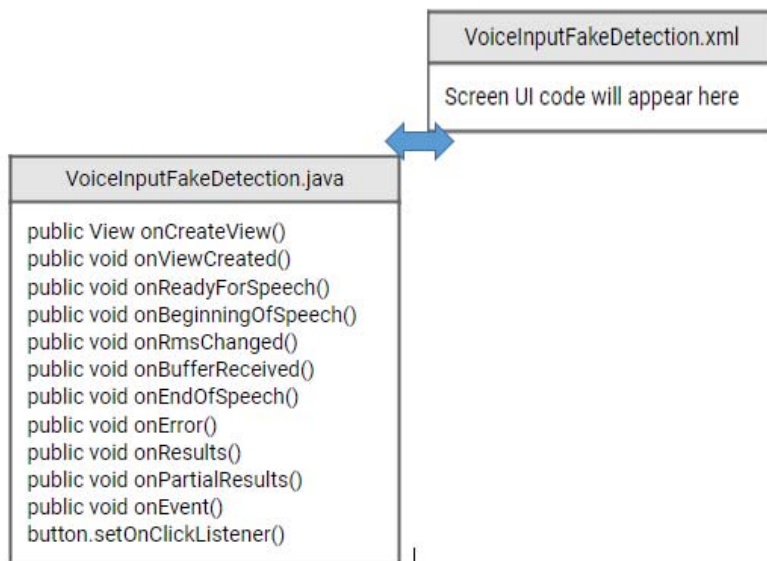
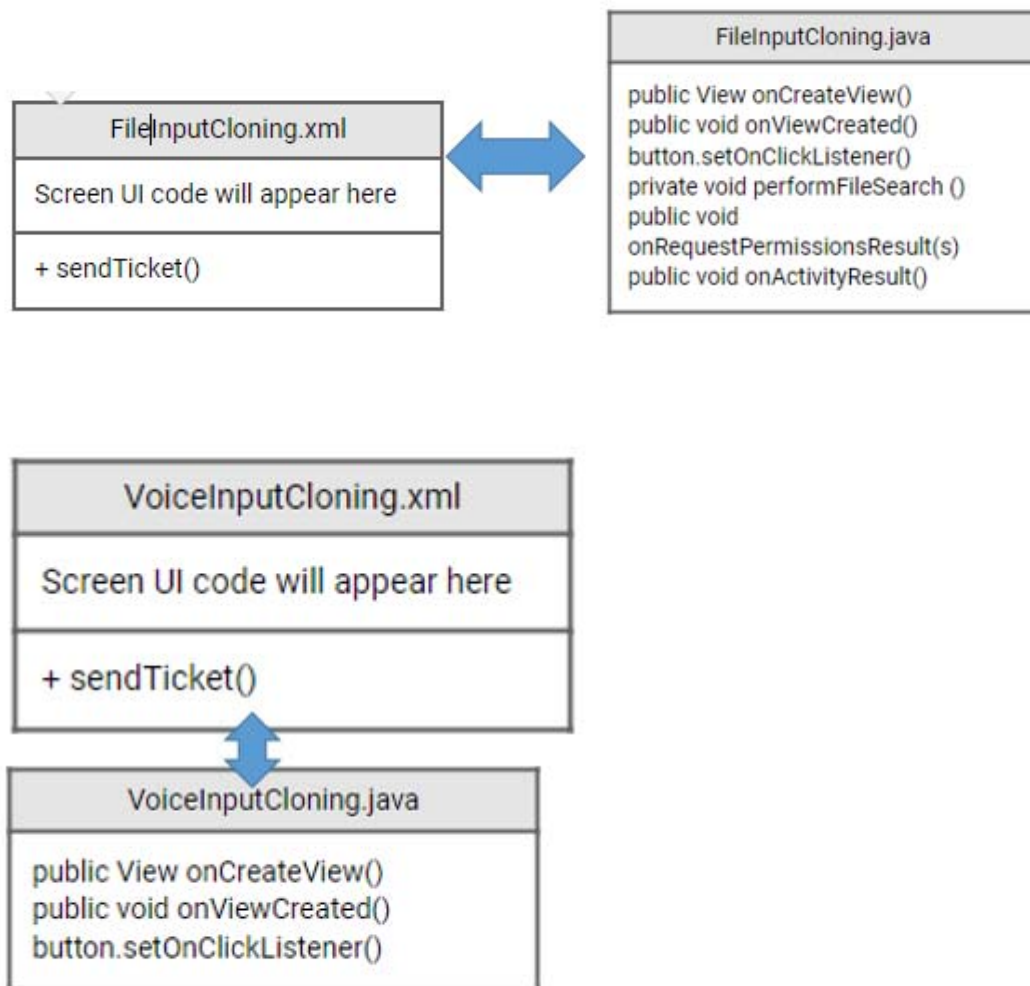


Figure 28





## 6.2 HOW THE WEB APP WORKS

1. Initially the user will navigate to our Landing Page where the user can see two sections, the first one is the Header which includes a user profile menu, a logo and the navbar link to move smooth on the website. And the second section is the main body section where we provide information to the user.

The important point is that, the header is position fixed, which can be seen at any level of



website

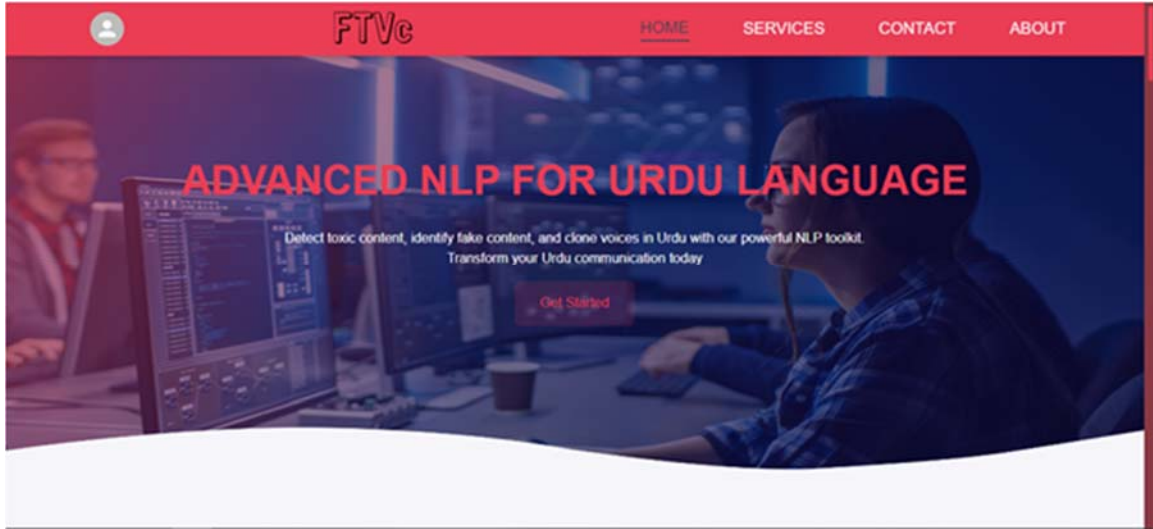


Figure 22: Website Home page

2. And when the user clicked on any of the services, he can move to that services pages and can used that. Like this one

## Toxic Content

## Toxic Content Detection

Enter Text Input:

اپنا متن یہاں درج کریں۔

Submit

Upload Voice Clip:

Choose File No file chosen

Submit

Output:

Result:

آپ کی دی گئی آواز میں کوئی زہریلا مواد نہیں ہے۔

**Fake Content**

## Fake Content Detection

Enter Text Input:

اینا متن یہاں درج کریں۔

Submit

Upload Voice Clip:

Choose File No file chosen

Submit

Output

Result:

آپ کا دیا ہوا متن جعلی نہیں ہے۔

Voice Cloning

## Voice Cloning

Enter Text Input:

ایدا متن یہاں درج کریں۔

Submit

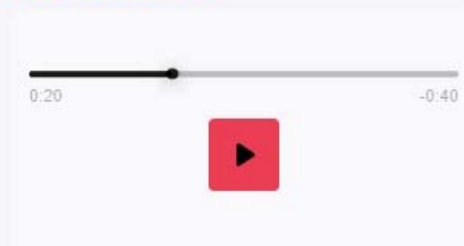
Upload the Voice to be Cloned:

Choose File No file chosen

Submit

Output:

Generated Cloned Voice Clip :



Contact Section

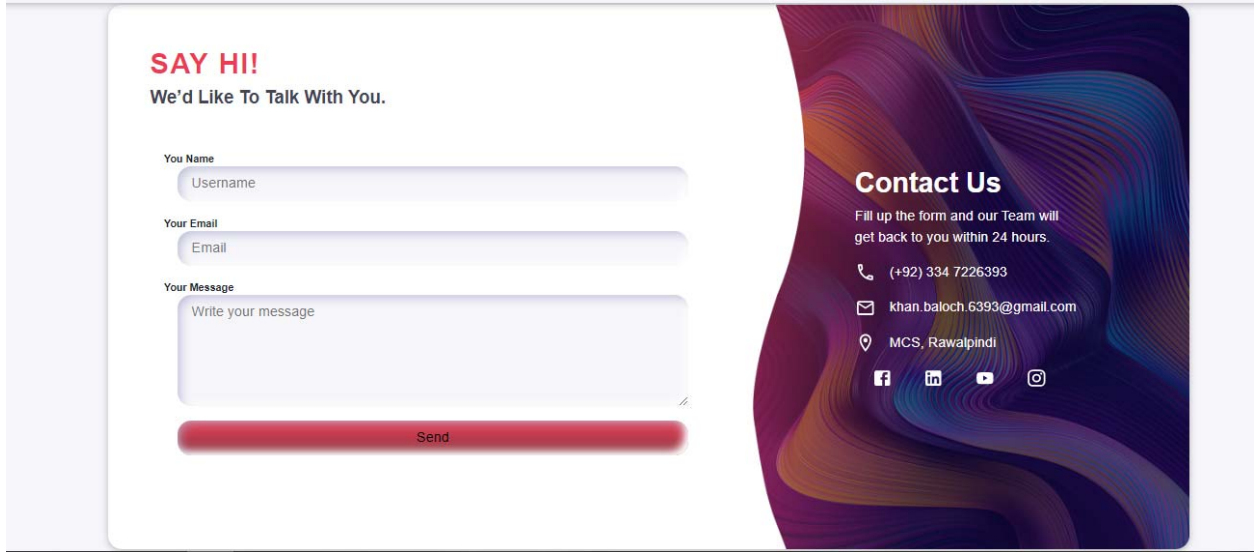


Figure 23: Contact Section

3. Also, we have included a Footer section at the bottom of every page for better user experience

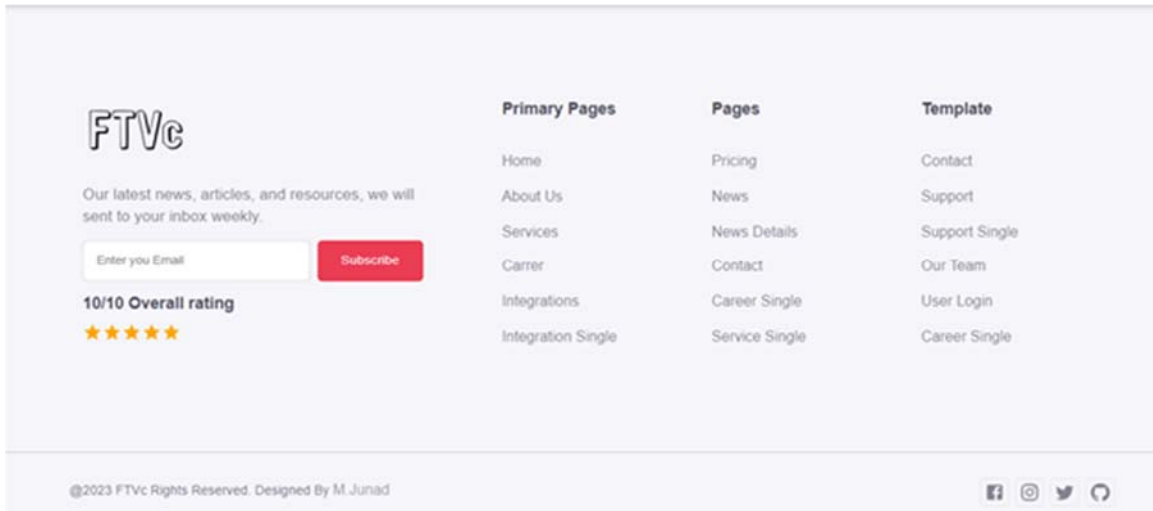


Figure 24: Footer Section

# TESTING AND IMPLEMENTATION

## 7.1 DATA SET COLLECTION:

### Fake Content

We took this data set from GitHub link: <https://github.com/MaazAmjad/Urdu-Fake-news-detection-FIRE2021>

The "Bend the Truth" dataset contains news in six different domains: technology, education, business, sports, politics, and entertainment. The real news included in the dataset was collected from a variety of mainstream news websites predominantly in Pakistan, India, the UK, and the USA. These news channels are BBC Urdu News, CNN Urdu, Express-News, Jung News, Noway Waqat, and many other reliable news websites. The fake news included in this dataset consists of fake versions of the real news in the dataset, written by professional journalists. More details on the data collection are provided in the paper.

### Toxic Content

We took the data set from the following source link: <https://ods.ai/competitions/urdu-hack-soc2021>

The dataset we collected was divided into to sub task described below

**Subtask A** contains Twitter tweets in the Urdu language. The data was labeled into two classes, namely: Abusive and Non-Abusive

Abusive - This Twitter post contains some sort of abusive content.

Non-Abusive - This Twitter post does not contain any abusive or profane content.

**Sub-task B** contains Twitter tweets in the Urdu language. The data was labeled into two classes, namely: Threatening and Non-Threatening.

Threatening - This Twitter post contains any threatening content.

Non-Threatening - This Twitter post does not contain any threatening or profane content.

we combined these two dataset into single dataset which is labeled into three classes, namely: Normal( having value 0 ), Abusive( value 1 ) and Threat( value 2)

## Voice Cloning

We took the data set from the following source link:

<https://commonvoice.mozilla.org/en/datasets>

## 7.2 INPUT:

### Voice-to-text conversion

We used following approach to for audio input

```
import speech_recognition as sr

r = sr.Recognizer()
audio_file = sr.AudioFile('file.wav')
with audio_file as source:
    audio = r.listen(source)
    try:
        text = r.recognize_google(audio, language='ur')
        print("You said : {}".format(text))
    except:
        print("Sorry could not recognize what you said")
```

You said : اسلام علیکم اسناد جی کیا حال ہے ٹھیک ہے مزے میں اور سناؤں آپ کیسے ہیں طبیعت ٹھیک ہے

Figure 25: SpeechRecognition model

### Text and File

The Input can be give in the form of text or a document file or a voice

## 7.3 MACHINE LEARNING MODEL:

The machine learning model is a core component of the toolkit, and it will be trained on labeled data to detect fake news and abusive content in Urdu language. The model will be designed to analyze the language used in news articles, social media posts, and other online content to identify patterns and markers of misinformation and hate speech. The model will use natural language processing techniques to analyze and classify the text data, allowing it to detect and flag potential instances of fake news and abusive content.

We used the RNN model for this. Our model contains the following layers

### **Fake Content Model Architecture**

**Embedding:** This layer performs word embedding, which maps each word in the input sequence to a vector of fixed size. The Embedding layer takes as input the integer-encoded input sequence and returns a tensor of shape (batch\_size, max\_len, embed\_size), where batch\_size is the number of samples in each batch and embed\_size is the dimensionality of the embedding vector, that is 300 and max\_len is 250

**Bidirectional:** This layer wraps a GRU (gated recurrent unit) layer in a bidirectional manner. This means that the input sequence is processed in both forward and backward directions by two separate GRU units. The output of this layer is a tensor of shape (batch\_size, max\_len, 128) where 64 comes from the number of GRU units, and  $2 * 64 = 128$  comes from the bidirectional processing.

**GlobalMaxPooling1D:** This layer computes the maximum of the elements in the temporal dimension of the input tensor, which reduces the tensor's dimensions from (batch\_size, max\_len, 128) to (batch\_size, 128).

**Dense:** This layer is a fully connected layer with a single neuron and a sigmoid activation function. The output of this layer is a tensor of shape (batch\_size, 1).

### **Toxic Content Model Architecture**

**Embedding:** This layer performs word embedding, which maps each word in the input sequence to a vector of fixed size. The Embedding layer takes as input the integer-encoded input sequence and returns a tensor of shape (batch\_size, max\_len, embed\_size), where



batch\_size is the number of samples in each batch and embed\_size is the dimensionality of the embedding vector, that is 300 and max\_len is 250

**Bidirectional:** This layer wraps a GRU (gated recurrent unit) layer in a bidirectional manner. This means that the input sequence is processed in both forward and backward directions by two separate GRU units. The output of this layer is a tensor of shape (batch\_size, max\_len, 192) where 96 comes from the number of GRU units, and  $2 * 96 = 192$  comes from the bidirectional processing.

**GlobalMaxPooling1D:** This layer computes the maximum of the elements in the temporal dimension of the input tensor, which reduces the tensor's dimensions from (batch\_size, max\_len, 192) to (batch\_size, 192).

**Dense:** This layer is a fully connected layer with a single neuron and a softmax activation function. The output of this layer is a tensor of shape (batch\_size, 3).

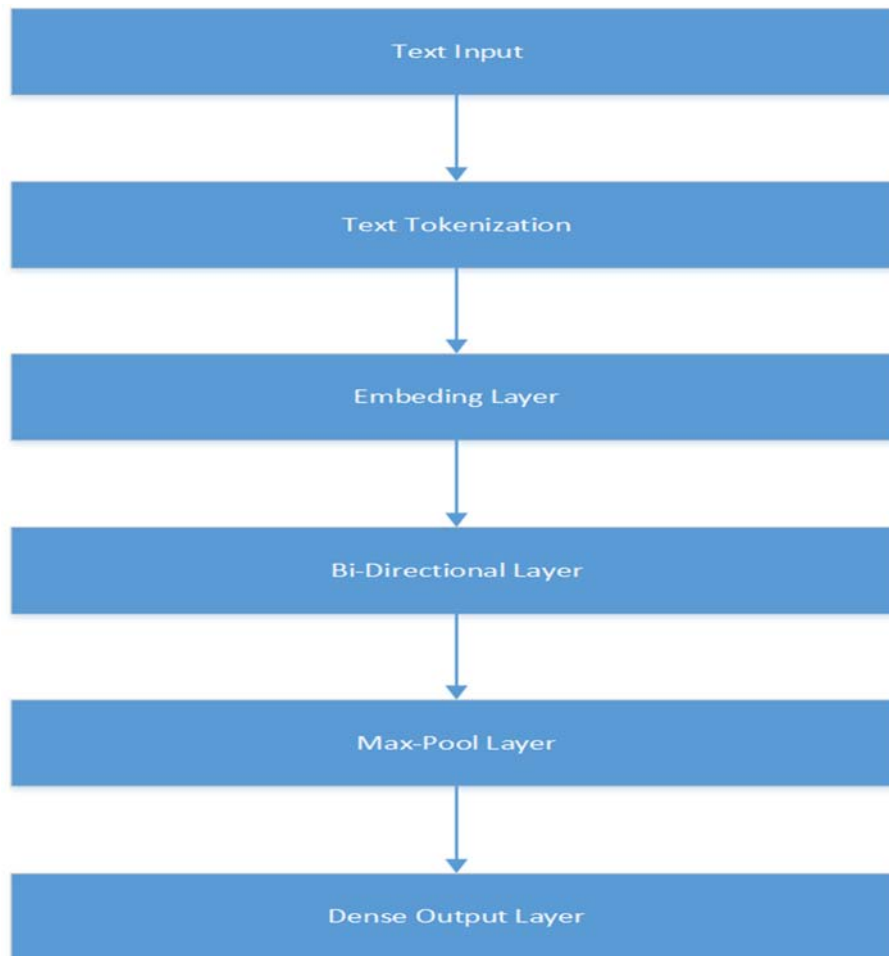


Figure 26: RNN Model Architecture diagram

## 7.4 TESTING

Unit testing: This type of testing involves testing each individual component of the toolkit, such as functions and methods, to ensure that they work correctly. It helps identify bugs and defects at an early stage of development, making it easier to fix them.

1. Integration testing: This type of testing involves testing how different components of the toolkit work together. It helps identify issues that arise due to the interaction between different components.
2. System testing: This type of testing involves testing the toolkit as a whole, ensuring that it meets the project requirements and performs as expected. It involves testing

the toolkit in different environments and under different conditions to identify any issues.

3. Performance testing: This type of testing involves testing the performance of the toolkit under different loads and stress levels. It helps identify any performance issues and ensures that the toolkit can handle the expected load.
4. User acceptance testing: This type of testing involves testing the toolkit with end-users to ensure that it meets their needs and is user-friendly. It helps identify any usability issues and ensures that the toolkit is easy to use.
5. The testing process should be continuous and iterative, with each iteration building on the previous one. The development team should also prioritize testing and ensure that it is integrated into the development process. Testing should be automated wherever possible to reduce the manual effort required and improve the reliability of the testing process.

In summary, testing is a critical part of developing a toolkit for addressing fake news, abusive content, and voice cloning in the Urdu language. It helps ensure that the product meets the project requirements, is reliable, and performs as expected. Different types of testing should be performed throughout the development process to identify and fix any issues.

### **Different Models for Fake Content Detection**

<b>Models</b>	<b>Accuracy</b>	<b>F1-Score</b>
Bi-Directional RNN	83%	77%
RNN	60%	--
LSTM RNN	65%	--

CNN	73%	--
ADA Boost	74%	46%
Random Forest	66%	42%
Decision Tree	72%	53%
Logistic regression	74%	42%

### **Different Models for Fake Content Detection**

<b>Models</b>	<b>Accuracy</b>	<b>F1-Score</b>
Bi-Directional RNN	90	80%
RNN	72%	46%

LSTM RNN	76%	56%
CNN	67%	45%
ADA Boost	71%	48%
Random Forest	69%	47%
Decision Tree	73%	56%
Logistic regression	74%	53%

## 7.5 TESTS AND DISCUSSION

### Test Cases:

Following are the test cases performed

#### Case 1 Audio/txt file to text Conversion

<u>Test Case Id</u>	<u>001</u>
<u>Test Performed By</u>	<u>Muhamad Faizaan</u>
<u>Objective</u>	<u>Voice-to-Text Conversion/ Getting Text from the File</u>
<u>Prerequisite</u>	<u>None</u>
<u>Steps</u>	<u>Use a Microphone to enter audio data</u>
<u>Data</u>	<u>Audio</u>
<u>Result</u>	<u>Success</u>

#### Case 2 Fake Content Detection

<u>Test Case Id</u>	<u>002</u>
<u>Test Performed By</u>	<u>Muhammad Irtaza</u>
<u>Objective</u>	<u>Fake News</u>
<u>Prerequisite</u>	<u>You have opened the Fake content screen</u>
<u>Steps</u>	<u>Enter Input</u>
<u>Data</u>	<u>Text or Audio Data</u>
<u>Result</u>	<u>Success</u>

### Case 3 Toxic Content Detection

<u>Test Case Id</u>	<u>003</u>
<u>Test Performed By</u>	<u>Aleena Akbar</u>
<u>Objective</u>	<u>Toxic Content</u>
<u>Prerequisite</u>	<u>Toxic Content Screen opened</u>
<u>Steps</u>	<u>Enter input.</u> <u>Press Enter</u>
<u>Data</u>	<u>Audio or Text Data</u>
<u>Result</u>	<u>Sucess</u>

### Case 4 Voice Cloning

Test Case Id	004
Test Performed By	Muhamad Faizaan
Objective	Voice Cloning
Prerequisite	None
Steps	Enter sufficient audio Sample
Data	Audio Recording
Result	Pending

### **CONCLUSIONS AND FUTURE WORK**

#### **8.1 FUTURE WORK**

There are several potential areas of future work that can be done on the toolkit for addressing fake news, abusive content, and voice cloning in the Urdu language. Some of these include:

- **Continuous improvement:** The toolkit can be continuously improved by incorporating new machine learning algorithms, natural language processing techniques, and other advanced technologies to enhance its capabilities and accuracy.
- **Expansion to other languages:** The toolkit can be expanded to support other languages such as Punjabi, and Sindhi which are written in same script, thereby increasing its usefulness and impact.
- **Integration with social media platforms:** The toolkit can be integrated with social media platforms to automatically detect and flag fake news and abusive content in real time.
- **Collaboration with media organizations:** Media organizations can collaborate with the developers of the toolkit to use its capabilities to fact-check news articles and other content.
- **User feedback and usability testing:** The toolkit can be further improved by incorporating user feedback and conducting usability testing to ensure that it is user-friendly and effective.
- **Detection of deep fakes:** The toolkit can be expanded to detect deep fakes, which are highly realistic manipulated videos or audio that can be used to spread misinformation and fake news.



Overall, there is a lot of potential for future work on the toolkit, and continued development and improvement will help combat the spread of fake news, abusive content, and voice cloning in the Urdu language and beyond.

## *Chapter 9*

### **REFERENCES**

1. Detecting Fake News using Machine Learning: A Systematic Literature Review 1Alim Al Ayub Ahmed -  
<https://arxiv.org/ftp/arxiv/papers/2102/2102.04458.pdf>
2. Abdullah-All-Tanvir, Mahir, E. M., Akhter S., & Huq, M. R. (2019). Detecting Fake News using Machine Learning and Deep Learning Algorithms. 7th International Conference on Smart Computing & Communications (ICSCC), Sarawak, Malaysia, Malaysia, 2019, pp.1-5  
<https://doi.org/10.1109/ICSCC.2019.8843612>
3. Google's Perspective API  
<https://pypi.org/project/perspective/>
4. TACOTRON 2: HUMAN-LIKE SPEECH SYNTHESIS  
TACOTRON 2: HUMAN-LIKE SPEECH SYNTHESIS FROM  
<https://NIX-UNITED.COM/BLOG/NEURAL-NETWORK-SPEECH-SYNTHESIS-USING-THE-TACOTRON-2-ARCHITECTURE-OR-GET-ALIGNMENT-OR-DIE-TRYIN/>
5. Overview of Abusive and Threatening Language Detection in Urdu at FIRE 2021  
<https://ceur-ws.org/Vol-3159/T4-1.pdf>