MACHINE LEARNING

CASE STUDY

TOPICS

Speech to Text
Predictive Text

Romaan Halai 23030141025
Swapnil Kothari 23030141034
Rhea Shree S 23030141050

INDEX

INTRODUCTION

- In the fast-evolving domain of mobile technology, machine learning has emerged as a cornerstone, enabling substantial breakthroughs in user engagement and experience.

- Two of the most influential uses of machine learning in this domain are mobile voice-to-text and predictive text technology.

- Mobile voice-to-text technology allows users to translate spoken words into written text with astonishing precision, employing advanced speech recognition algorithms.

- Predictive text, on the other hand, boosts typing productivity by predicting the next word or phrase a user intends to enter, based on context and previous usage patterns.

- Both technologies rely heavily on machine learning models that are trained on vast datasets, enabling them to learn and adapt to the nuances of human language.

- Voice-to-text applications utilize deep learning models, particularly recurrent neural networks (RNNs) and convolutional neural networks (CNNs), to recognize and transcribe speech in real-time.

- These models must account for various accents, dialects, and speech impediments, making them highly complex and capable of continuous improvement as they process more data.

- Predictive text systems leverage natural language processing (NLP) techniques to interpret context and anticipate the user’s next input.

- These systems use techniques such as n-grams, long short-term memory (LSTM) networks, and transformer models to deliver accurate and contextually relevant suggestions.

- The integration of these machine learning technologies into mobile devices not only boosts user comfort and accessibility but also illustrates the transformational capacity of artificial intelligence in everyday applications.

WORKING OF PREDICTIVE TEXT

By recommending words or phrases that a user is likely to enter next, predictive text technology improves the typing experience. Natural language processing (NLP) methods and advanced machine learning models are used in this procedure. Here's a thorough explanation of how predictive text functions:

Gathering and Preparing Data:

- **Data Collection**: Large volumes of textual data are the basis of predictive text algorithms. This information may originate from a variety of sources, including posts on social media, emails, and texts, among others. The models for machine learning are trained using the gathered data.

- **Preprocessing**: Tokenization, which divides text into words or phrases, stop word removal (common terms like "the," "and," etc.), and stemming, which reduces words to their bases, are some of the preprocessing procedures the data goes through before being fed into models.

Machine Learning Models:

- **N-grams Model**: The n-grams model is one of the most basic models used in predictive text analysis. To forecast the following word, it examines the string of n words. In a trigram model with three variables, for instance, the sequence "I am going" could predict "to." The incapacity of n-grams models to capture context and long-range dependencies beyond n words limits their effectiveness.

Training the models

- **Supervised Learning**: In supervised learning, the models are trained with text sequences as the input data and next word predictions as the right outputs. In order to decrease the discrepancy between its predictions and the actual following words in the training set, the model learns by modifying its parameters.

- **Evaluation and Fine-Tuning**: To make sure the models perform effectively when applied to fresh, untested text, they are tested on distinct validation datasets following training. To improve performance, fine-tuning entails retraining the model and modifying the hyperparameters.

Prediction Process:

- **Context Analysis for Prediction Process**: As a user inputs, the predictive text system examines the current text input's context. In order to make predictions, it takes into account the words and sentences that have already been typed.

- **Making Predictions**: The model creates a list of potential future words or phrases based on the context. Finding the words with the highest possibility of appearing next entails calculating the probabilities of each word.

- **User Interaction**: The user is presented with these predictions by the system, usually as suggestions that are shown above the keyboard. After then, the user can choose a word that the system has suggested or just keep typing while the system keeps updating its predictions.

Continuous Learning and Adaptation:

- **User Input**: In order to increase accuracy, predictive text algorithms frequently take user input into account. For example, if a user chooses a suggestion again and over again, the algorithm picks up on this preference and modifies its predictions going forward.

- **Contextual Adaptation**: Over time, contemporary predictive text systems can adjust to the user's vocabulary, writing style, and often used expressions. The suggestions are more relevant and helpful as a result of this customization.

Challenges and Future Directions:

- **Managing Ambiguity**: In situations where there are several plausible predictions, predictive text systems have to make their way through difficult situations. Research on how to handle these kinds of cases better is still under progress.

- **Multilingual Support**: In an increasingly globalized society, building models that can reliably anticipate text in several languages and switch between them fluently is a challenging but essential objective.

- **Privacy and Security**: It's critical to protect user data privacy while still producing precise forecasts. On-device processing and federated learning innovations are assisting in resolving these issues.

WORKING OF MOBILE SPEECH TO TEXT

Mobile speech-to-text technology converts spoken language into written text using advanced algorithms that process audio signals by capturing audio through a microphone, using a speech recognition engine to map sounds to text, and employing language models for accuracy.

Process:

- **Audio Capture:** The vibrations from the user’s audio are captured from the microphone
- **Acoustic Processing:** The captured audio is converted from analog to digital language through a converter.
- **Feature Extraction:** Key features of the audio signal after filtering the waves, such as pitch and tone, are extracted and converted into numerical data.
- **Speech Recognition:** A mathematical model matches the phonemes to well-known sentences, words, and phrases and translates them into text.
- **Natural Language Processing (NLP):** NLP techniques are often employed to improve accuracy, handle context, and understand nuances in speech**.**

Components:

- **Microphone:** The microphone records audio input from the user, and the quality of this recording can significantly impact the accuracy of the transcription.
- **Audio Processing:** The captured audio is processed to remove noise and improve clarity. This usually involves reducing background noise and amplifying the speaker's voice.
- **Speech Recognition Engine:** This is the primary component of speech-to-text systems, responsible for converting audio signals into text through multiple steps:
- **Feature Extraction:** The audio is divided into small segments called frames. These segments are analyzed to identify characteristics like pitch and loudness, creating a digital representation of the sound.
- **Acoustic Model:** This model translates the extracted audio features into phonemes, the basic units of sound in a language. It is trained using extensive datasets of speech recordings and their transcripts.

- **Language Model:** Utilizing linguistic context, this model predicts the likelihood of word sequences, helping to choose the most probable words by understanding grammar and typical usage.
- **Decoder:** This component integrates the outputs from the acoustic and language models. This process involves considering various word possibilities based on the detected sounds and selecting the most grammatically correct and contextually appropriate sequence.

Post-processing: The initial text output is refined to enhance readability and accuracy, involving tasks such as adding capitalization, punctuation, and correcting common transcription mistakes.

Feedback Loop: Some systems allow users to correct transcription errors, with these corrections feeding back into the system to improve its future accuracy.

APPLICATIONS

- Hands-free texting, messaging, and email composition.
- Aids individuals with disabilities, such as those with motor impairments or visual impairments.
- Applications that allow users to transcribe speech into text for notes, messages, or documents.
- Virtual assistants like Siri, Google Assistant, and Alexa use speech-to-text for voice commands and queries.

HOW TO BUILD A SYSTEM IN PYTHON

MOBILE VOICE TO TEXT IN PYTHON

Packages Used:

- speech_recognition
- pyttsx3
- matplotlib.pyplot
- pandas
- numpy

Code Snippets:

Function to recongise the audio:

```
def SpeakText(command):
    engine = pyttsx3.init()
    engine.say(command)
    engine.runAndWait()
```

Loop for listening audio and writing in the output:

```
    while True:
        with sr.Microphone() as source:
            print("Listening...")
            r.adjust_for_ambient_noise(source, duration=0.2)
            audio = r.listen(source)

        print("Recognizing...")
        MyText = r.recognize_google(audio)
        MyText = MyText.lower()

        if MyText:
            print("Did you say: ", MyText)
            SpeakText(MyText)
```

Audio Metrics Calculation:

```
        snr, rmse, bitrate = calculate_metrics(audio_data, sample_rate)

        data.append({'timestamp': timestamp, 'text': MyText, 'snr': snr, 'rmse': rmse, 'bitrate': bitrate})
        audio_data_list.append(audio_data)
```

Input to the program:

```
Say 'quit' to stop listening. Press any key to stop.
Listening...
Recognizing...
Did you say:  once logged in navigate to the project tab click on create my first project or add new project if you already have other projects enter your domain name and click create project
Listening...
Recognizing...
Did you say:  set up different tools under the project such as site audit position tracking on page seo checker and social media toolkit configure each tool based on your needs for example inside audit set the cross settings and in position tracking set the keywords you want to track
Listening...
Recognizing...
Did you say:  go to site audit under your project click start audit to analyse your website rave the audit report to find technical seo issues like broken links missing meta tags and crawl errors
Listening...
Recognizing...
Did you say:  quit
Quitting...

Process finished with exit code 0
```

PREDICTIVE TEXT IN PYTHON

Packages Used:

- ollections
- matplotlib.pyplot
- pandas
- seaborn

Code Snippets:

Predict Next Word Function:

```
  def predict_next_word(self, input_text):
      words = re.findall(r'\b\w+\b', input_text.lower())
      if not words:
          return []
```

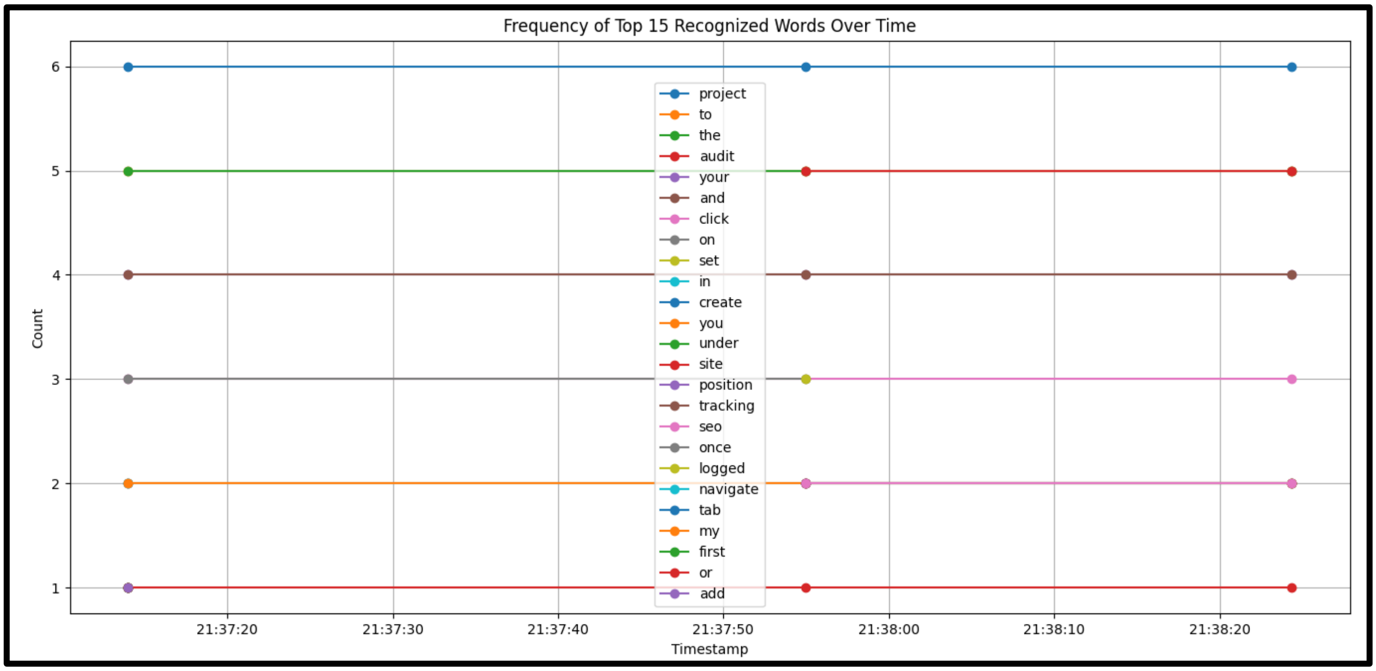
Calculate word frequency:

```
  def get_word_frequencies(self):
      frequencies = defaultdict(Counter)
      for current_word, next_words in self.model.items():
          frequencies[current_word] = next_words
      return frequencies
```

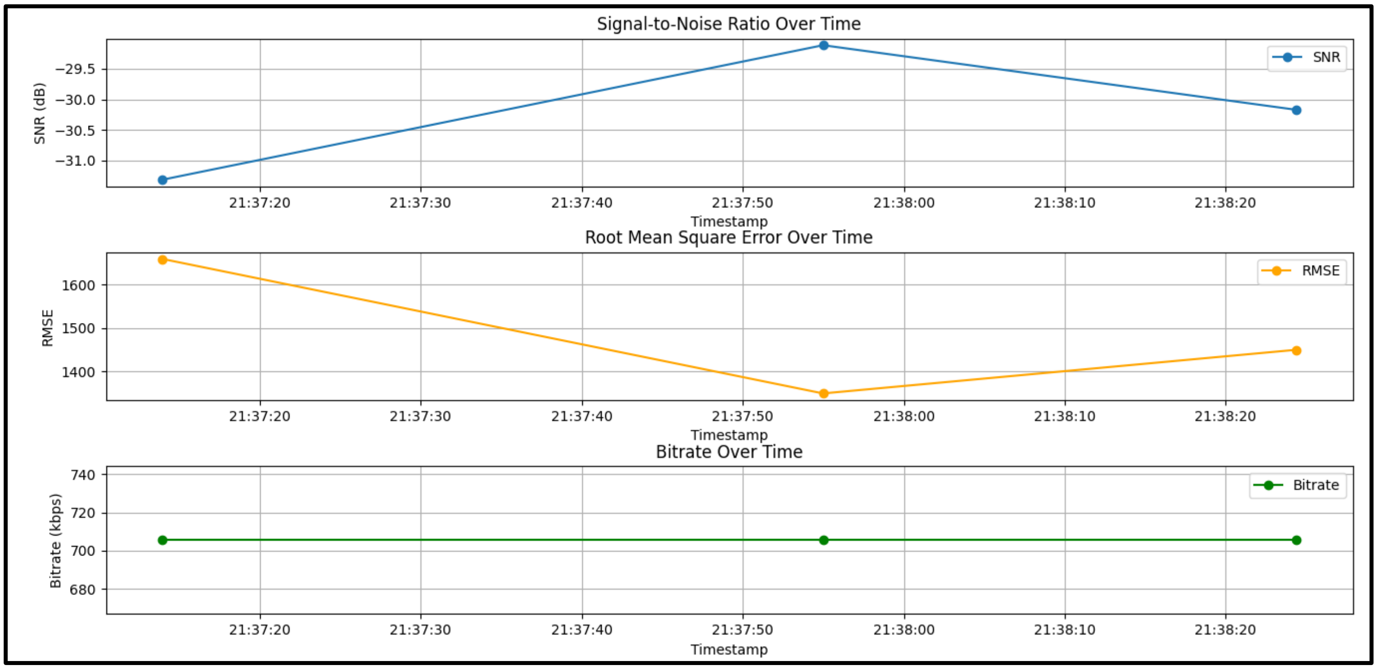
Use of Corpus file:

```
def load_corpus(file_path):
    with open(file_path, 'r', encoding='utf-8') as file:
        return file.read()
```

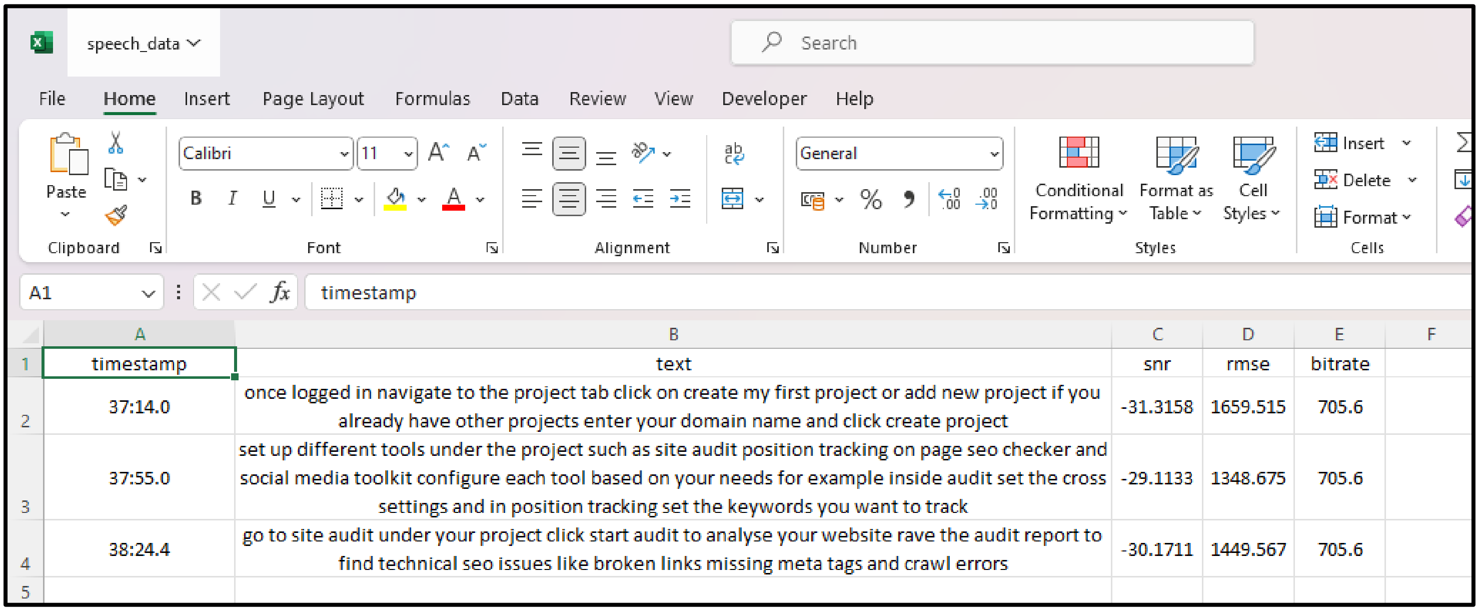
ANALYSIS



Word Frequency in the audio



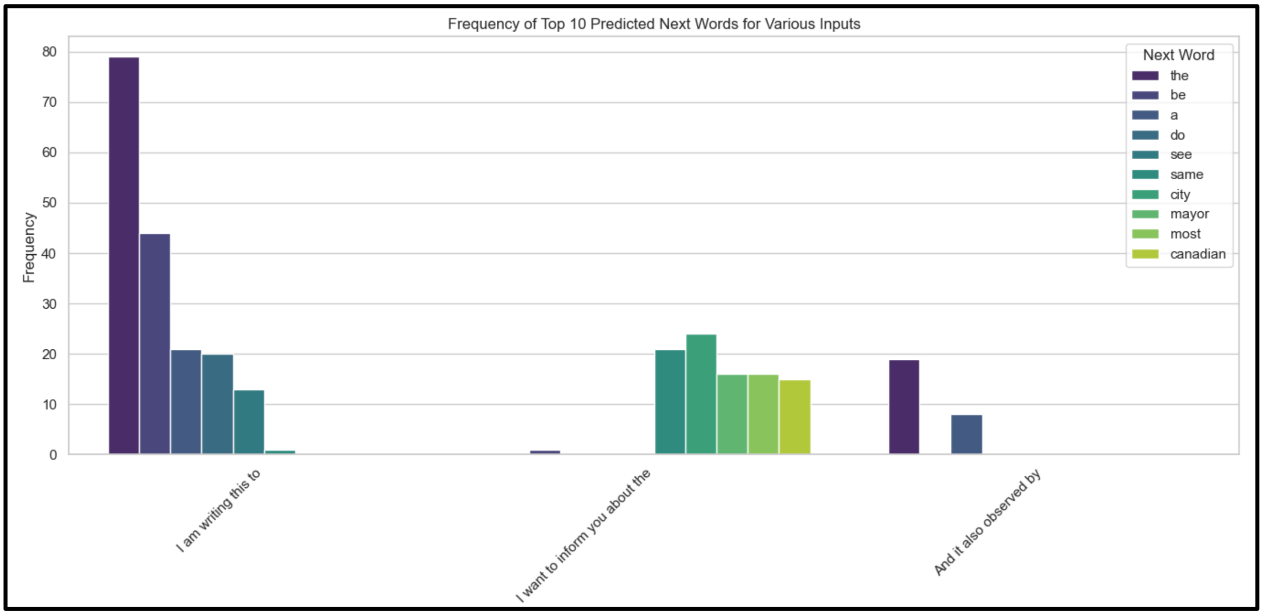
Audio Metrics Display



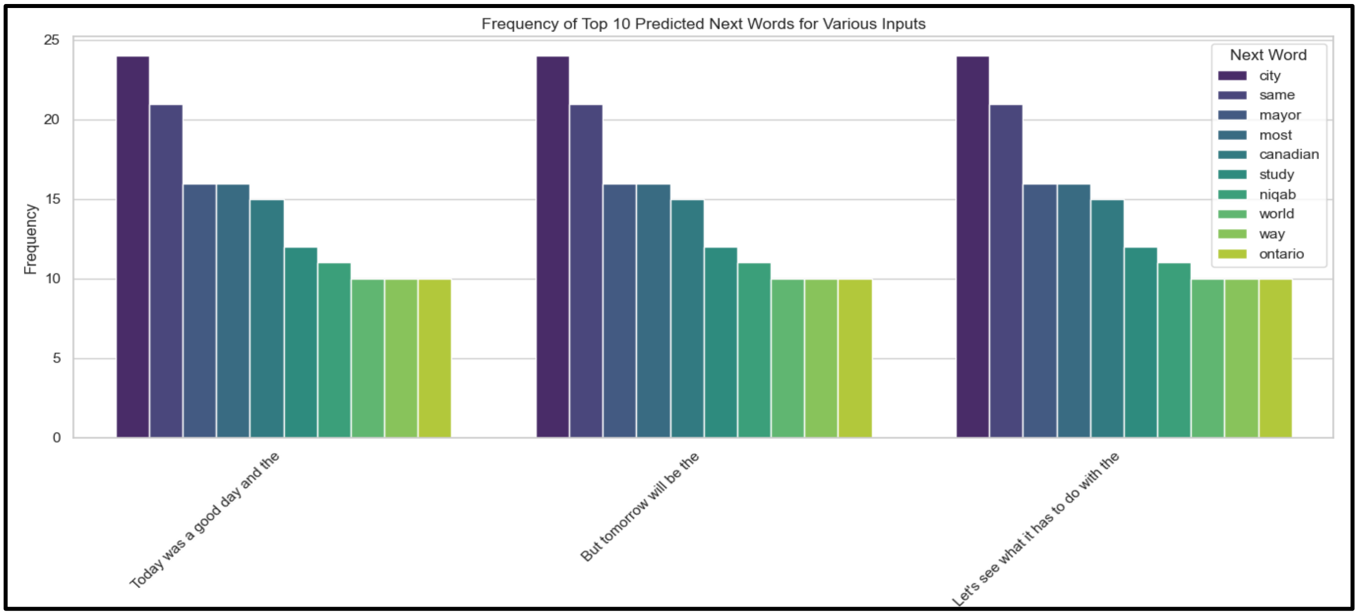
timestamp

	A	B	C	D	E	F
1	timestamp	text	snr	rmse	bitrate	
2	37:14.0	once logged in navigate to the project tab click on create my first project or add new project if you already have other projects enter your domain name and click create project	-31.3158	1659.515	705.6	
3	37:55.0	set up different tools under the project such as site audit position tracking on page seo checker and social media toolkit configure each tool based on your needs for example inside audit set the cross settings and in position tracking set the keywords you want to track	-29.1133	1348.675	705.6	
4	38:24.4	go to site audit under your project click start audit to analyse your website rave the audit report to find technical seo issues like broken links missing meta tags and crawl errors	-30.1711	1449.567	705.6	
5						

Audio Details stored in Excel along with other metrices



Predictive Text Frequency first set of samples



Predictive Text Frequency second set of samples

CONCLUSION

- In the context of machine learning, both mobile voice-to-text and predictive text systems leverage advanced algorithms and models to enhance user interaction and efficiency.
- Machine learning models, particularly deep learning techniques, are used to process audio inputs and accurately transcribe speech into text that enables these systems to understand complex speech patterns and adapt to various accents and language.
- Complex algorithms analyze text patterns to predict the next word in a sentence. These models learn from vast amounts of text data to understand language structure and context, allowing them to suggest relevant word choices.
- The ability of machine learning to adapt and process large datasets makes it highly effective in developing and enhancing voice-to-text and predictive text technologies. They significantly improve user convenience and communication and continuously evolve by incorporating user feedback and real-world data.