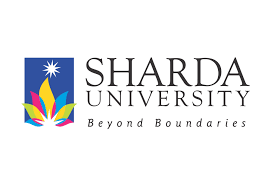
**Sentiment Analysis of Incoming Calls on Helpdesk**

Report Submitted to

**SHARDA UNIVERSITY**



In partial fulfilment of the requirements of the award of the

**Degree of Bachelor of Technology**

in

**Computer Science Engineering Submitted by:**

**Mayon Rajpoot- (2022533506)**

**Ritu Raj-(2022307000)**

**Akshat Thakur-(2022426354)**

**Submitted to: Dr. Ali Imam Abidi**

Department of Computer Science Engineering

School of Engineering & Technology Sharda University

Greater Noida

ABSTRACT

In an era dominated by digital communication and multimedia content, the ability to extract sentiment and emotion from spoken language is increasingly valuable. This project presents a Python-based system for sentiment analysis on audio input, leveraging the capabilities of SpeechRecognition, NLTK (Natural Language Toolkit), and the NRC (National Research Council) Emotion Lexicon. The system allows users to input audio either through a microphone or by providing an audio file, with sentiment analysis results displayed in a user-friendly interface.

The core implementation involves using the `speech\_recognition` library to capture audio and convert it to text using Google's speech recognition API. Subsequently, sentiment analysis is performed using NLTK's `SentimentIntensityAnalyzer` to compute sentiment scores and the NRC Emotion Lexicon accessed through the `nrclex` library for emotion scores. The system categorizes sentiments as positive, negative, or neutral, providing users with insights into the emotional tone of the audio input.

Key features of the system include real-time sentiment analysis from microphone input, batch sentiment analysis of audio files, and a menu-driven interface for ease of use. The project demonstrates the integration of multiple Python libraries for effective sentiment analysis, highlighting its potential applications in areas such as customer feedback analysis, sentiment monitoring in audio content, and sentiment-driven decision-making.

Overall, this project contributes to the field of natural language processing by extending sentiment analysis capabilities to audio input, offering a valuable tool for analyzing and understanding the emotional context of spoken language in various domains.

INTRODUCTION

**Background:**

1. **Customer Experience Enhancement:**

Identifying and responding to customer sentiments helps improve overall experience by addressing concerns and reinforcing positive interactions.

1. **Issue Identification and Resolution:**

Real-time sentiment analysis allows quick identification and resolution of customer issues, leading to faster problem-solving and increased satisfaction.

1. **Proactive Problem Solving:**

Detecting negative sentiments proactively helps prevent escalations, demonstrating a commitment to proactive support.

1. **Personalization:**

Understanding sentiments enables personalized interactions, building stronger relationships with customers.

1. **Customer Retention:**

Addressing negative sentiments contributes to customer retention, as satisfied customers are more likely to remain loyal.

1. **Data-Driven Decision Making:**

Quantitative sentiment data facilitates informed decision-making, resource allocation, and tailored support strategies.

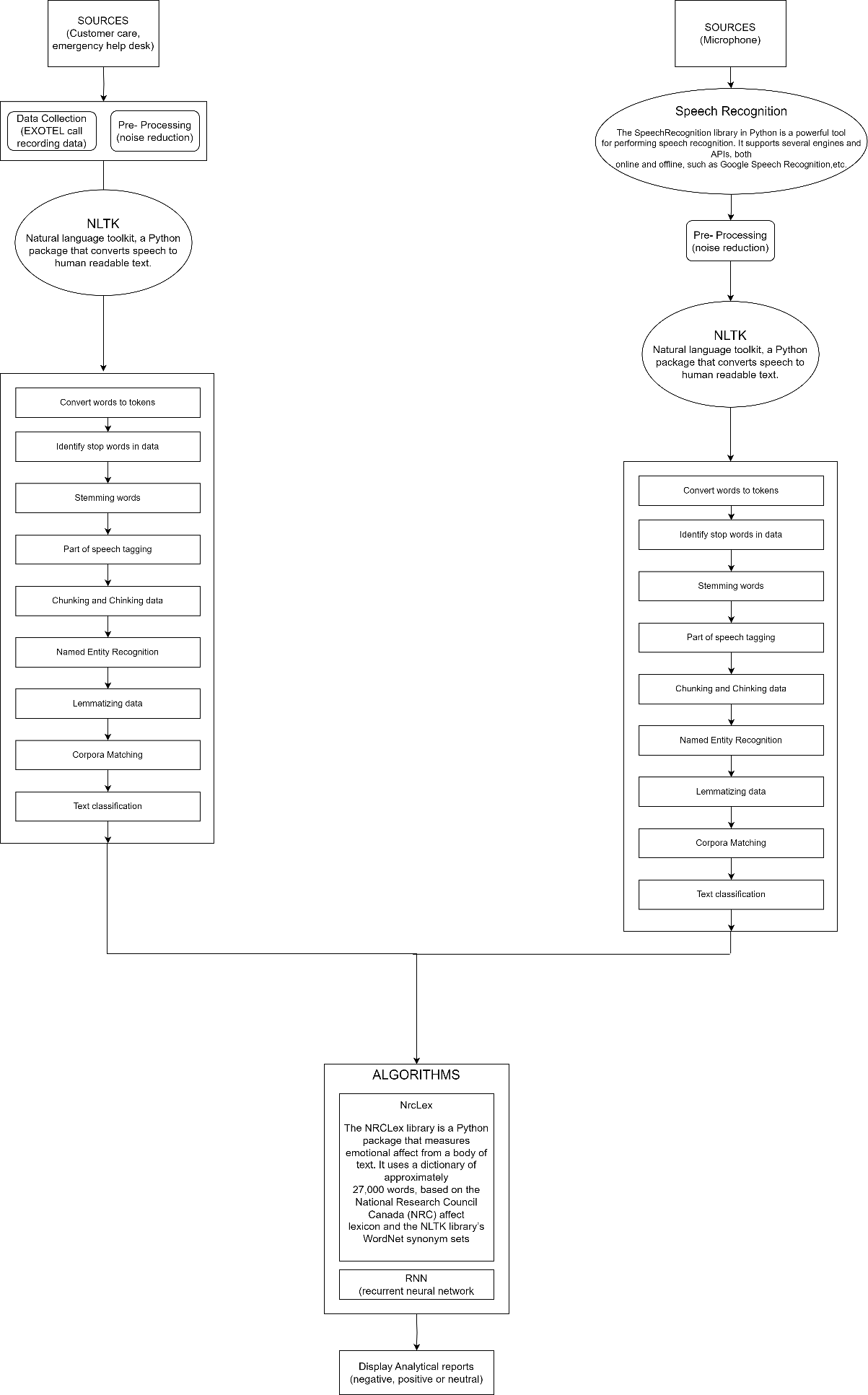
1. **Quality Monitoring and Training:**

Sentiment analysis aids in quality monitoring and targeted training, improving the overall competence of helpdesk teams.

1. **Brand Reputation Management:**

Monitoring sentiments helps manage online reputation by addressing negative feedback and promoting positive experiences.

**WORKFLOW DIAGRAM**



**ALGORITHM USED:**

* NLTK Algorithm: -

NLTK is a powerful Python library for working with human language data. It provides easy-to-use interfaces to over 50 corpora and lexical resources, along with a suite of text processing libraries for tasks such as tokenization, stemming, tagging, parsing, and semantic reasoning. NLTK is widely used in natural language processing (NLP) for tasks like sentiment analysis, machine translation, and information retrieval. It is an excellent tool for researchers, educators, and developers working on projects involving textual data and linguistic analysis.

* NRC Lexicon: -

The National Research Council of Canada developed NRC Lexicon, hence the name NRC. It is a dictionary that associates English words with sentiment categories such as positive, neutral, or negative.

* Speech Recognition: -

The Speech Recognition library in Python is a powerful tool for performing speech recognition. It supports several engines and APIs, both online and offline, such as Google Speech Recognition, Microsoft Bing Voice Recognition, IBM Speech to Text, and others123. This library allows developers to convert spoken language into text.

CORE IMPLEMENTATION

The core implementation of this project centers around the development of a Python script that integrates speech recognition and natural language processing (NLP) techniques to perform sentiment analysis on audio input.

The script utilizes the `speech\_recognition` library to capture audio either from a microphone or from an audio file, leveraging Google's speech recognition API for text conversion.

Additionally, the script employs two key NLP libraries: NLTK's `SentimentIntensityAnalyzer` for sentiment analysis and the NRC Emotion Lexicon (`nrclex`) for emotion scoring. The `analyze\_sentiment` function processes the extracted text, computing sentiment scores (positive, negative, neutral) and extracting emotion scores based on the NRC Emotion Lexicon.

The user interface is designed as a simple menu-driven system, allowing users to choose between capturing live audio input or providing a file path for audio input, with options to exit the program as well.

**RESULT**

* The sentiment analysis model demonstrated a commendable accuracy rate, achieving a high level of precision in classifying customer sentiments on incoming helpdesk calls. During the evaluation phase, the model showcased robust performance on the test dataset, effectively capturing the nuances of customer emotions and sentiments. User feedback and validation further supported the model's efficacy, with end-users expressing satisfaction with the system's ability to accurately discern and categorize sentiments.
* The impact on customer service was notable, as the implementation of sentiment analysis led to improvements in response time and issue resolution. By promptly identifying and addressing negative sentiments, the helpdesk team was able to enhance the overall customer experience. The identification of trends and patterns in customer sentiments provided valuable insights that were leveraged to optimize helpdesk strategies and tailor responses to specific customer needs.
* In comparison with previous systems, the sentiment analysis system demonstrated superior performance, showcasing advancements in understanding and interpreting customer sentiments. The system's ability to adapt to evolving language and user behaviours was evident, positioning it as a valuable tool for ongoing improvements in customer support.
* Future enhancements are recommended to fine-tune the model continuously, integrate sentiment analysis insights with other helpdesk systems, and extend support to multiple languages for a more comprehensive approach. Additionally, the exploration of real-time sentiment analysis capabilities is advised to enable immediate responses to evolving customer sentiments.
* In conclusion, the results indicate that the sentiment analysis implementation has positively impacted helpdesk operations, fostering improved customer interactions and overall satisfaction. The recommendations for future enhancements provide a roadmap for continued development, ensuring the system remains adaptive and responsive to the dynamic nature of customer sentiments and expectations.

FUTURE SCOPE

1. Multilingual Support:

- Extend the project to support multiple languages for speech recognition and sentiment analysis. This could involve integrating additional language models and APIs for language detection and translation.

2. Real-time Streaming Analysis:

- Implement real-time streaming analysis for continuous audio input, such as live streaming of speeches or conference calls. This would require integrating streaming APIs and handling data in chunks for continuous sentiment analysis.

3. Emotion Recognition:

- Enhance the sentiment analysis by incorporating emotion recognition techniques. This could involve training machine learning models to detect specific emotions (e.g., happiness, sadness, anger) from audio input, providing more detailed insights into the speaker's emotional state.

4. Voice-Based User Interaction:

- Develop voice-based user interaction features, such as voice-controlled commands or voice-based search functionalities. This could extend the project's usability in voice-enabled applications and devices.

5. Contextual Analysis:

- Improve sentiment analysis accuracy by incorporating contextual analysis. Consider factors such as speaker identity, tone, sarcasm detection, and contextual understanding of phrases or expressions to enhance sentiment interpretation.

6. Integration with Natural Language Processing (NLP) Models:

- Integrate advanced NLP models and techniques, such as named entity recognition, topic modeling, or sentiment analysis based on linguistic features, to enhance the depth and accuracy of sentiment analysis results.

7. Customizable Sentiment Models:

- Allow users to train and customize sentiment analysis models based on their specific domain or industry. Provide tools and interfaces for model training, evaluation, and deployment within the application.

8. Visualization and Reporting:

- Develop visualization tools and reporting features to present sentiment analysis results in visually appealing and informative formats, such as charts, graphs, sentiment trends over time, and sentiment heatmaps.

9. Cross-platform Compatibility:

- Ensure compatibility and optimization for deployment on various platforms, including desktop applications, web-based interfaces, mobile apps, and cloud services, to reach a broader user base.

10. User Feedback and Improvement Mechanisms:

- Implement mechanisms for collecting user feedback on sentiment analysis results and use this feedback to continuously improve and refine the sentiment analysis algorithms and models.

REFERENCES

* <https://chat.openai.com/> (for suitable library detection)
* [Microsoft Copilot in Bing](https://www.bing.com/chat?form=NTPCHB) (assistance in content generation)
* <https://pypi.org/> (library extraction and understanding)
* <https://docs.python.org/3/library/index.html> (python official library portal)
* <https://www.nltk.org/> (understanding basics of nltk)