

**Title: Sentiment Analysis Chatbot**

Anushka Maulik Parikh, UG Btech CSE in Nirma University

Date of Submission: August 2, 2024

Teachnook Educational Institution

**Project Overview**

This chatbot is designed to understand how users are feeling by identifying their sentiment. It can tell if a user’s mood is positive, negative, or neutral. For example, if someone says, “I love this!” the chatbot recognizes that as positive.

What makes this chatbot special is that it doesn’t just label feelings; it also shows how strong those feelings are. For instance, it can indicate that a user’s sentiment is 80% positive or 60% negative. This percentage helps users see not only what they’re feeling but also how strongly they feel about it.

By using advanced language processing techniques, the chatbot can understand different ways people express themselves, whether they’re being straightforward or a bit more complicated. This means it can handle a variety of responses, making conversations feel more natural and engaging. Overall, this chatbot aims to provide valuable insights into emotions, helping users better understand their feelings and how they express them.

**Objectives -**

1. To develop a chatbot capable of identifying user sentiment (positive, negative, neutral).
2. To enhance user interaction through natural language processing (NLP) techniques.

**Programming Language -** Python

**Files -**

Internship Project.ipynb

Internship Project.pdf

**Methodology**

1. **Data Collection**

Word Lists: Two lists of words are created—positive\_words and negative\_words. These lists are curated from common positive and negative sentiments to serve as the basis for sentiment analysis.

1. **Preprocessing**

User Input Handling: The chatbot processes user input by converting it to lowercase and stripping any leading or trailing spaces. It also splits the input into individual words for analysis.

Punctuation Removal: Any punctuation (e.g., periods, commas, quotes) is removed from the words to ensure accurate matching with the sentiment lists.

1. **Sentiment Analysis Logic**

Word Counting: The program iterates through each word in the user input, checking if it exists in either the positive or negative word lists.

A counter (p) is incremented for each positive word found.

A counter (n) is incremented for each negative word found.

Sentiment Percentage Calculation: After counting the words, the program calculates the percentage of positive and negative sentiments based on the total number of words:

* If the input contains no words, both percentages are set to 0.
* The percentage of positive sentiment (p\_percentage) is calculated as (p/l)×100(p / l) \times 100(p/l)×100, where lll is the total number of words.
* Similarly, the negative percentage (n\_percentage) is calculated as (n/l)×100(n / l) \times 100(n/l)×100.

1. **Sentiment Classification**

The chatbot classifies the overall sentiment based on the counts of positive and negative words:

If the count of positive words (p) is greater than negative words (n), the sentiment is classified as "Positive."

If the count of negative words (n) is greater than positive words (p), the sentiment is classified as "Negative."

If the counts are equal, the sentiment is classified as "Neutral."

1. **User Interaction**

The chatbot interacts with users in a loop, prompting them to enter a word, sentence, or paragraph for analysis.

Users can exit the interaction by typing "exit," and a farewell message is displayed.

1. **Output Presentation**

After analyzing the input, the chatbot prints the positive and negative percentages, followed by the overall sentiment classification.

1. **Testing and Evaluation**

The chatbot is tested with various user inputs to evaluate its performance and accuracy in sentiment detection.

Common phrases and variations in language are used to ensure the robustness of the sentiment analysis.

**Conclusion**

The methodology outlined above details the steps taken to develop the sentiment analysis chatbot, from defining the problem to implementing the logic for sentiment classification. This systematic approach ensures that the chatbot can accurately analyze user input and provide meaningful insights into sentiment, enhancing user engagement and understanding.

This methodology can be included in your project documentation to give a clear overview of the development process and the thought behind your design choices.