IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

Twitter Sentiment Analysis On Chatgpt Tweets Using SVM And BERT Model

Guide -Prof.S.P.Gunjal

Sanika Pawar[1], Sakshi Kshirsagar[2] ,Sayali Shilimkar[3], Sejal Kasare[4] 1(Informqtion Technology, SKNSITS SPPU)

Abstract: With the rapid adoption of AI technologies like OpenAI's ChatGPT, understanding public sentiment becomes critical for developers and researchers to gauge user experiences and expectations. This paper presents a sentiment analysis of tweets mentioning ChatGPT, employing two models: Support Vector Machine (SVM) and Bidirectional Encoder Representations from Transformers (BERT). Tweets were collected using the Twitter API and preprocessed to remove noise and convert the text into suitable formats for analysis. The SVM model was used as a baseline, relying on TF-IDF for feature extraction, while BERT was fine-tuned to classify tweets into positive, negative, or neutral sentiments. The results demonstrate that while SVM provides acceptable accuracy for straightforward text, BERT outperforms it by effectively capturing nuanced sentiment and complex contextual relationships within the tweets. This analysis provides insights into public opinion on ChatGPT and highlights the effectiveness of transformer-based models in sentiment analysis on social media platforms.

Key Word: SVM, BERT, Sentiment Analysis ,Twitter ChatGPT ,Natural Language Processing (NLP)

I. Introduction

The field of Artificial Intelligence (AI) has experienced significant advancements in recent years, particularly in the domain of Natural Language Processing (NLP). One of the most notable developments in NLP is OpenAI's ChatGPT, a conversational AI model that has gained widespread popularity for its ability to generate human-like text responses across various contexts. ChatGPT is widely discussed on social media platforms like Twitter, where users express diverse opinions about its functionality, potential applications, limitations, and ethical implications.

With the increasing influence of social media on public opinion, **sentiment analysis** has emerged as a powerful tool for interpreting user-generated content. Sentiment analysis allows researchers to classify textual data into categories such as positive, negative, or neutral sentiment, providing valuable insights into how a product or technology is perceived. Understanding these sentiments is essential for improving AI models like ChatGPT and addressing user concerns.

This paper focuses on **Twitter sentiment analysis** related to ChatGPT, utilizing two prominent machine learning models: **Support Vector Machine (SVM)** and **Bidirectional Encoder Representations from Transformers (BERT)**. While SVM is a traditional machine learning model known for its efficiency in text classification tasks, BERT represents a more advanced deep learning approach designed to capture the context and semantic meaning of words. The comparison between these two models will help determine their effectiveness in analyzing short, informal, and context-dependent social media data like tweets.

By analyzing the sentiment of tweets mentioning "ChatGPT," this study aims to provide insights into the public's perception of the model, identifying trends in positive, negative, and neutral feedback. The findings can inform future improvements to ChatGPT and other conversational AI technologies, as well as contribute to the broader field of sentiment analysis on social media platforms.

II. Material And Methods

1. Data Collection:

The dataset for this study consists of tweets mentioning **"ChatGPT"** collected from **Twitter** using the **Tweepy API**, a Python library for accessing the Twitter API. The dataset contains tweets spanning over a period of six months to capture diverse opinions and sentiments about ChatGPT. To ensure a balanced dataset, the following filters were applied:

- **Language**: Only English-language tweets were considered.
- **Retweets and Replies**: Retweets were excluded to avoid duplication, while replies were included to capture conversational contexts.
- **Hashtags and Mentions**: Tweets containing the hashtag #ChatGPT or the mention @ChatGPT were prioritized.

After applying these filters, a total of **10,000 tweets** were collected, with each tweet containing the following fields: tweet text, user ID, timestamp, and retweet count.

2. Data Preprocessing:

Preprocessing is a crucial step in sentiment analysis to ensure that the data is clean and consistent. The following preprocessing techniques were applied:

- **Tokenization**: Each tweet was broken down into individual words (tokens) using the NLTK library.
- **Lowercasing**: All text was converted to lowercase to ensure uniformity.
- **Stop Word Removal**: Common English stop words (e.g., "is", "the", "and") were removed using the NLTK stopword list.
- **Punctuation Removal**: All punctuation marks, special characters, and emojis were removed to avoid noise in the analysis.
- **Lemmatization**: Words were reduced to their base or root form using WordNetLemmatizer to handle variations (e.g., "running" becomes "run").

3. Sentiment Labeling:

Sentiment labeling was performed using **VADER Sentiment Analysis**, a lexicon and rule-based tool designed for social media text. Each tweet was assigned a sentiment score categorized into one of three classes: •

Positive: Score > 0.05

- **Negative**: Score < -0.05
- **Neutral**: -0.05 ≤ Score ≤ 0.05

This process resulted in a labeled dataset with **5,000 positive**, **3,000 negative**, and **[1],000 neutral** tweets.

4. Feature Extraction:

To convert the text data into a format suitable for machine learning models, the following feature extraction methods were used:

- **TF-IDF (Term Frequency-Inverse Document Frequency)**: This method was applied to transform the tweets into numerical vectors. It measures how important a word is in the context of the entire dataset, giving higher weight to rare but significant words.

[1] | Page

- **BERT Embeddings**: For the BERT model, pre-trained embeddings from the BERT-base-uncased model were used to capture the contextual relationships between words in the tweets.

5. **Model Implementation:**

a. **Support Vector Machine (SVM):**
The **SVM** algorithm was chosen due to its efficiency in binary and multiclass classification tasks. The implementation process included the following steps:

- **Training and Testing Split**: The dataset was split into **80% training** and **20% testing** data.
- **Kernel Selection**: A **linear kernel** was selected, as it performed well in initial trials.
- **Hyperparameter Tuning**: A grid search method was used to optimize key parameters, including the penalty parameter C and the gamma value for better classification performance.

b. **Bidirectional Encoder Representations from Transformers (BERT):**
The **BERT model**, a state-of-the-art deep learning model for NLP tasks, was implemented using the **Hugging Face Transformers library**. The key steps involved:

- **Pre-trained BERT Model**: The BERT-base-uncased model was fine-tuned on the labeled dataset.
- **Fine-tuning**: BERT was fine-tuned using the training set for **5 epochs** with a **learning rate of 2e-5**.
- **Batch Size and Optimizer**: The batch size was set to 16, and the **Adam optimizer** was used for optimization.
- **Text Classification Layer**: A softmax layer was added on top of the BERT model to classify tweets into the positive, negative, or neutral sentiment categories.

6. Model Evaluation:

Both models were evaluated using standard classification metrics, including:

- **Accuracy**: The percentage of correctly classified tweets.
- **Precision**: The ratio of true positive predictions to the total predicted positives.
- **Recall**: The ratio of true positive predictions to the total actual positives.
- **F1-Score**: The harmonic mean of precision and recall, providing a balanced evaluation of the models.
Additionally, **confusion matrices** were generated for both SVM and BERT to visualize the performance on each sentiment class. The **Receiver Operating Characteristic (ROC) curve** was plotted to compare the models' ability to distinguish between positive and negative sentiments.