iJRASET
International Journal For Research in Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 10 **Issue:** IV **Month of publication:** April 2022

DOI: https://doi.org/10.22214/ijraset.2022.41243

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue IV Apr 2022- Available at www.ijraset.com

Writing Companion

Sai Sri Harsha Pemmasani[1], V Padmana Nageswara Rao[2], Rupali Jindal[3], Tejish Kandukuri[4], Hemanth Seela[5]
[1, 3, 4, 5]*Undergraduate Student,* [2]*Professor, Computer Science and Engineering, GITAM [Deemed to be University], Visakhapatnam, Andhra Pradesh, India*

Abstract: ***This project aims to develop a writing assistant. It has been designed to check for spelling and grammatical errors, analyze sentiment, detect plagiarism, and translate text from one language to another. This tool, using Natural Language Processing (NLP), analyzes text and provides appropriate error-free recommendations.***
Working as a writing assistant, this project assists users with their day-to-day activities, whether an informal letter to a friend, an important project report, or an online blog post. Social networks and community forums have become one of the most used platforms to communicate with people who may be from a different region speaking a foreign language. It consists of a massive amount of data where positive content from one's point of view may not be the same for others. In such situations, a writing assistant may act as an extra set of eyes to catch common errors and improve writing.
Keywords: Sentiment Analysis, Language Translator, Plagiarism Checker, Grammar and Spelling

I. INTRODUCTION

Natural Language Processing (NLP) is often implemented while trying to categorize textual data. Sentimental analysis is the system that identifies and represents subjective emotions of textual data by categorizing them into different numbers or classes. As natural language is full of uncertainty, it is one of the toughest tasks in NLP. Written text and spoken language both provide a lot of information and are the most prevalent types of unstructured data. NLP can assist in the analysis of data and tasks such as sentiment analysis, cognitive assistant, and real-time language translation. Natural language processing-based writing companion systems have always included three features to help users improve their writing: spell check, grammar check, and style check. Spell checking ensures that the characters you write for the intended term adhere to the syntax rules of the language that you're typing in. Style checking ensures that the manner in which you express yourself is appropriate for the intended target audience. Grammar checking ensures that the checking tools can correct spellings, grammar, and propose better synonyms while also assisting in the delivery of content with greater clarity and engagement. They also aid in the readability of content, helping you to deliver your message in the most effective manner possible. In the last two decades, plagiarizing content has become much easier since all the information is readily available on the internet. Using the information without giving credit to the source is not just wrong on moral or ethical grounds but it may also cause economic loss to the original writer. NLP consists of modules that help check the similarity between texts. Using the plagiarism checking tool students, scholars or researchers can check the originality of their content.

II. REVIEW OF LITERATURE

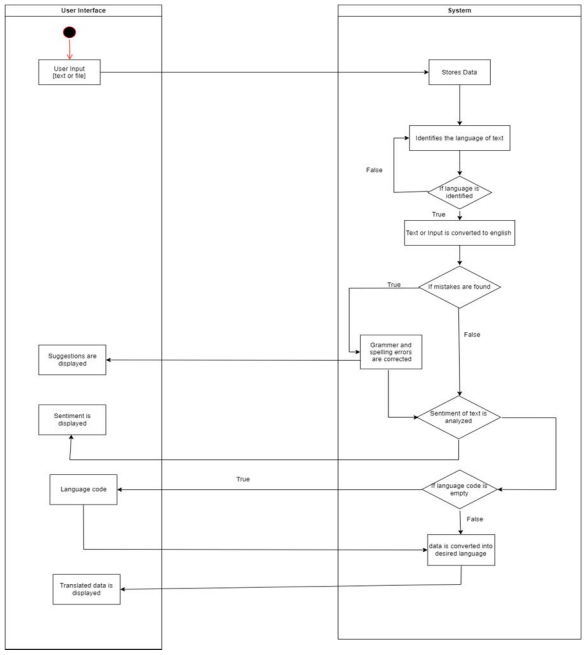
1) NLTK was first introduced in the year 2001 by Steven Bird, Edward Loper, Ewan Klein as a Python-based set of libraries and tools for symbolic and statistical natural language processing (NLP) for the English language. Steven Bird and Edward Loper of the University of Pennsylvania's Department of Computer and Information Science created it. Graphical demos and sample data are included in NLTK. It comes with a cookbook and a book that describes the basic ideas behind the language processing jobs that the toolkit supports.
2) NLTK was first introduced in the year 2001 by over 1800+ open source developers as a Python-based[majorly] set of libraries and tools. It may be used to compare files and generate file differences in a variety of formats, including HTML and context and unified diffs. For comparing directories and files, see the filecmp module.
3) DeepL Translator is a neural machine translation service launched in August 2017 by DeepL SE, a Cologne-based company. Major contributor for this repository during its development was Jaroslaw Kutylowski.
4) Streamlit is a San Francisco-based software company that provides an open-source platform for machine learning and data science teams to build Python-based data applications. The business was founded in 2018 by Adrien Treuille, Amanda Kelly, and Thiago Teixeira.
5) A method for detecting, underlining, and correcting grammatical errors in natural language text. It was designed by Prithiviraj Damodaran.

©IJRASET: All Rights are Reserved

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue IV Apr 2022- Available at www.ijraset.com

III. METHODOLOGY

Below figure shows the implementation or work flow of the process in the web application that is based on machine learning using python.



A lot of the data that we want to analyze could be containing human-readable text in the form of unstructured data. Before analyzing that data, we need to first preprocess it for which NLTK can be used. We can preprocess the data using the following steps-
Example sentences[input]:
'Hello Ms. Sara, how are you doing?'
'The weather is great today.'

1) Step:1 Tokenization

The given text or sentence input will be tokenized ,i.e., broken down into sequence of words, sentences to help in better interpretation of text in step:1 as shown below:
'Hello', 'Ms.', 'Sara', ',', 'how', 'are', 'you', 'doing', '?'
'The', 'weather', 'is', 'great', 'today', '.'

2) Step:2 Stop word removal

Stop words are a list of words that are predefined in NLTK, also known as noise, which are usually repeated a lot in the English language and carry very little meaning. Words such as 'is', 'am', 'in', 'there', 'the', etc., are considered noise in the text and are removed automatically.
'Hello', 'Ms.', 'Sara', 'doing', '?'
'The', 'weather', 'great', '.'

3) Step:3 Stemming and lemmatization

Stemming is a process of normalization that reduces words to their root word or removes their derivational affixes. For example, words experimentation, experimented, and experimenting reduce to a common root word "experiment."
Lemmatization is an advanced form of stemming. Stemming works on an expression without any knowledge of the context whereas lemmatization works by using vocabulary and morphological analysis. For example, The word "worse" has "bad" as its lemma. This expression will get missed by stemming because it requires a look-up in the dictionary..

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

4) Bayes Theorem

$$P(h/D) = \frac{P(D/h) * P(h)}{P(D)}$$

a) P(h|D) increases with P(h) and P(D|h) according to Bayes theorem.

b) P(h|D) decreases as P(D) increases because the more probable it is that D will be viewed independently of h, the less evidence D provides in favor of h.

After finding the most probable or suitable data from the dataset it shows the resultant data and if not found from the data set solution is predicted using algorithms such as KNN algorithm, e.t.c.

IV. OVERVIEW OF TECHNOLOGIES

A. NLTK

NLTK stands for Natural Language Toolkit, developed by Steven Bird and Edward Cooper, is a python module used for performing linguistic courseware such as tokenization, lemmatization, stemming, e.t.c.

Using the NLTK module, we can remove stop words, analyze and find the polarity of keywords, e.t.c, which are the main objectives for analyzing a sentence.

B. Difflib

Difflib is a python module that is mainly used for finding the sequences of strings and matching them regardless of file format such as HTML, pdf, XML, e.t.c

This module is going to be used in this project for finding similar sequences across the sources.

C. Deep_translator

Deep_translator is a python module that is used for translating a sentence or a paragraph from one language to another, supporting different language translator modules such as Google translator, Microsoft Translator, Yandex, e.t.c supporting a total of 111 other languages.

D. Streamlit

Streamlit is a python module developed to host machine learning-based applications to the web from either a local server or a remote server. This python module is used to integrate our multiple python modules into the internet using Heroku.

V. RELATED WORK

A. NLTK

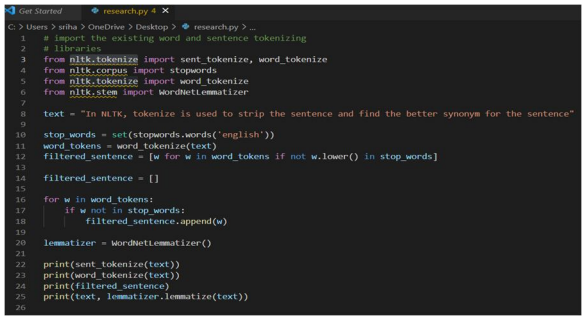
In the nltk module, sent_tokenize() and word_tokenize are used to tokenize sentences and words according to the user's desire.

A list of stop words is present in the nltk library and can be removed from the sentence or paragraph using the corpus.

Sentences can be lemmatized using wordnetlemmatizer.

The following code explains the working of the NLTK module in our project.

1) Code



```
# import the existing word and sentence tokenizing
# libraries
from nltk.tokenize import sent_tokenize, word_tokenize
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize
from nltk.stem import WordNetLemmatizer

text = "In NLTK, tokenize is used to strip the sentence and find the better synonym for the sentence"

stop_words = set(stopwords.words('english'))
word_tokens = word_tokenize(text)
filtered_sentence = [w for w in word_tokens if not w.lower() in stop_words]

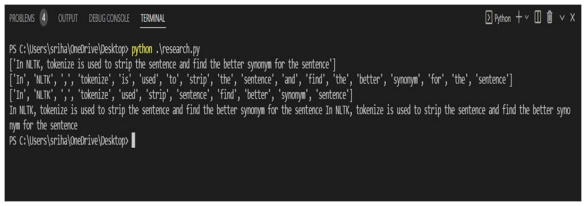
filtered_sentence = []

for w in word_tokens:
    if w not in stop_words:
        filtered_sentence.append(w)

lemmatizer = WordNetLemmatizer()

print(sent_tokenize(text))
print(word_tokenize(text))
print(filtered_sentence)
print(text, lemmatizer.lemmatize(text))
```

2) Output



More than just words, human communication is also more than just words. Sentiments are made up of a combination of words, tone, and writing style.

One of two methods can be used to accomplish this:

Count the amount of positive and negative terms in a given text, and the bigger the count, the more positive the content is.

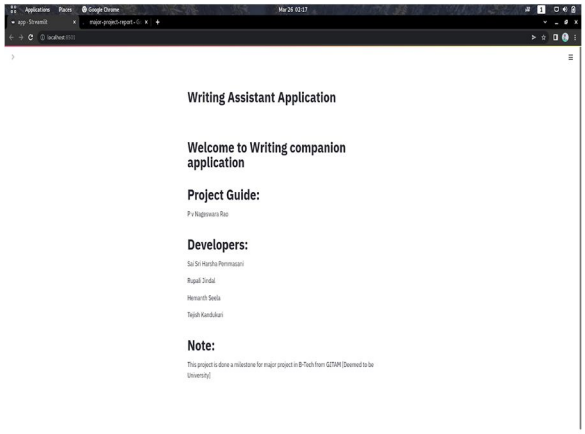
A method based on machine learning: Create a classification model based on the pre-labeled dataset of positive, negative, and neutral values.

VI. RESULTS AND DISCUSSIONS

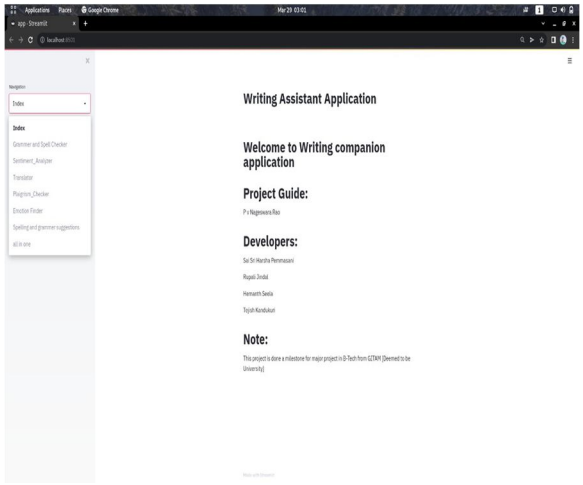
The proposed system developed in python and hosted in streamlit is designed so that it is user-friendly, efficient, and supports every platform.

A. Home Page

This is the home page of our project application.

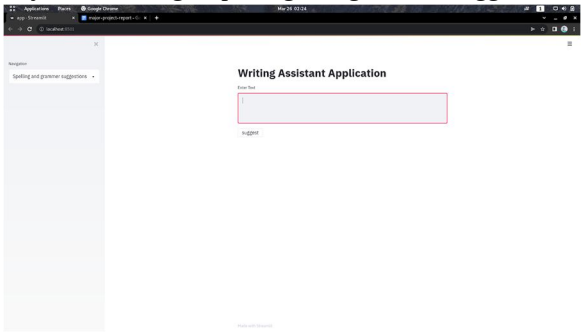


The below picture shows a dropdown menu on the left-hand side displaying all the different components of our project.

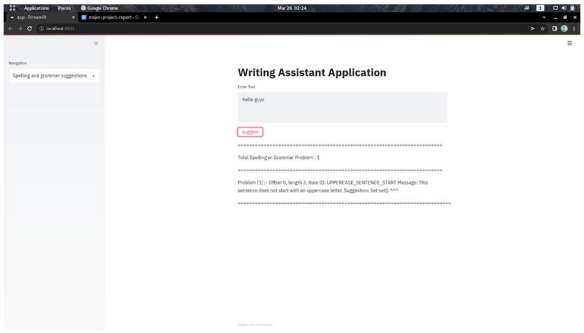


B. Spelling and Grammar Suggestions

This feature allows the user to enter a body of text and get spelling and grammar suggestions.

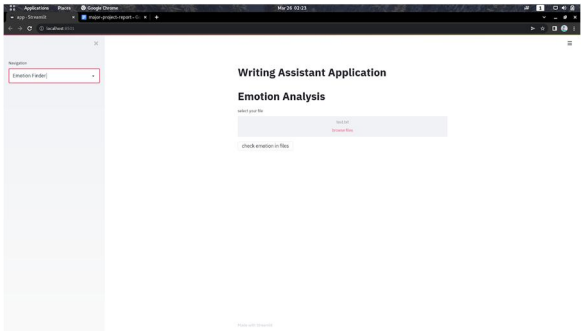


The below picture shows the output displaying the suggestion.

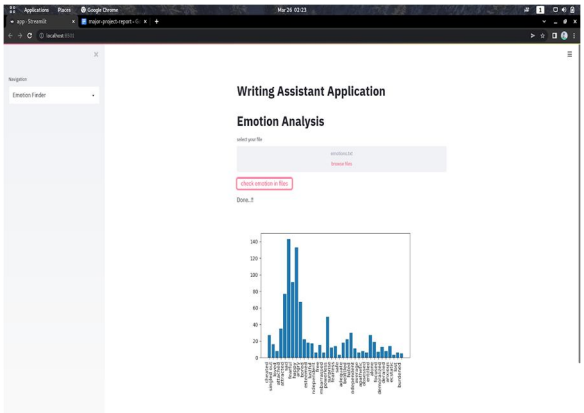


C. Emotion Finder

This feature allows the user to input a text file and get the various emotions present in the text file in the form of a bar graph.

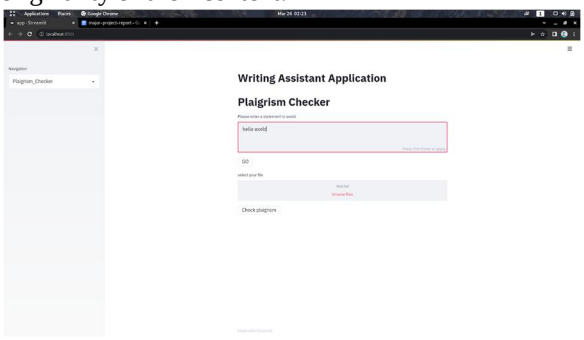


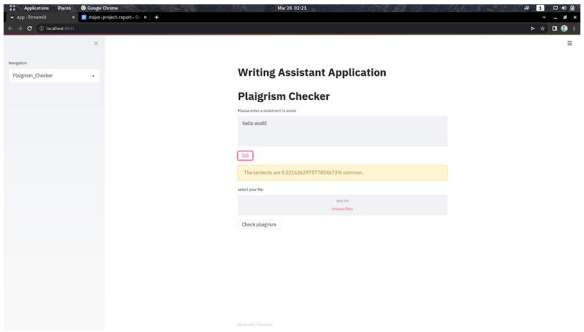
The below picture shows the output in the form of a bar graph where the X-axis represents different emotions and the Y-axis represents the number of words in that file



D. Plagiarism Checker

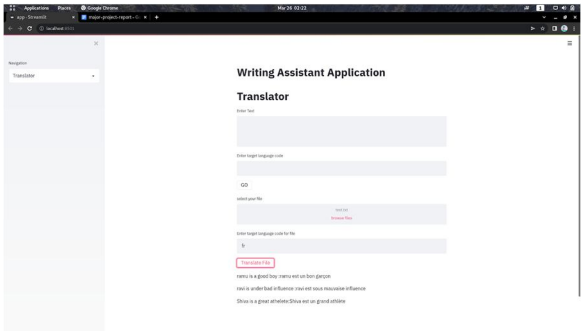
This feature allows a user to check the originality of their content.





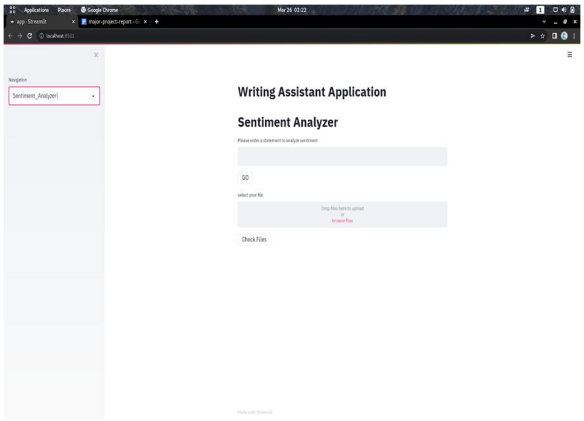
E. Language Translator

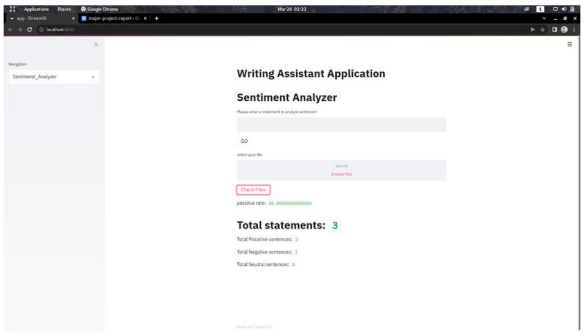
This feature allows the user to enter text or a text file and translate it to the desired language by entering the language code. The output displays the original sentence and the translated sentence separated with the help of a colon.



F. Sentiment Analyzer

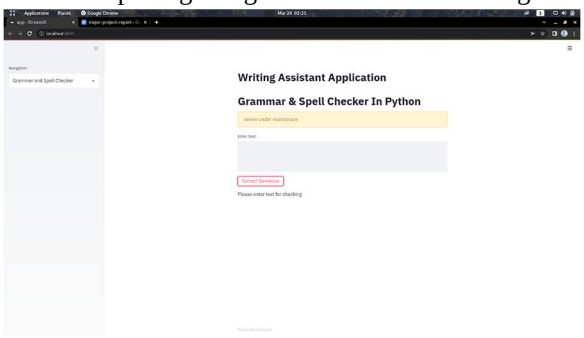
This feature allows the user to analyze the sentiment of the given text and classify it as positive, negative, or neutral.

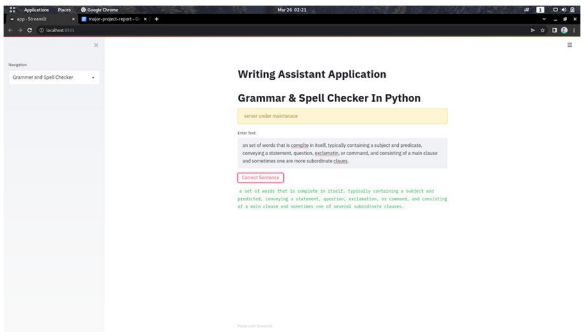




G. Grammar and spell-check

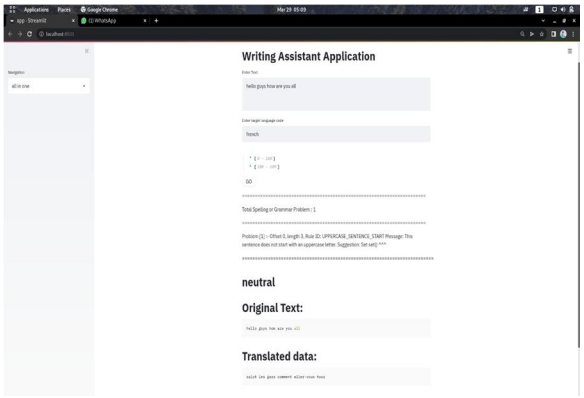
This feature displays the correct sentence free of spelling and grammatical errors when given a body of text as input.





H. All in one

This feature has all the components of our project integrated where users can check spelling and grammatical errors, find the sentiment of text, and also translate it into another language. The output below shows one grammatical error, it classifies the sentiment of the text as neutral and also translates it into the target language which is french.



VII. CONCLUSIONS

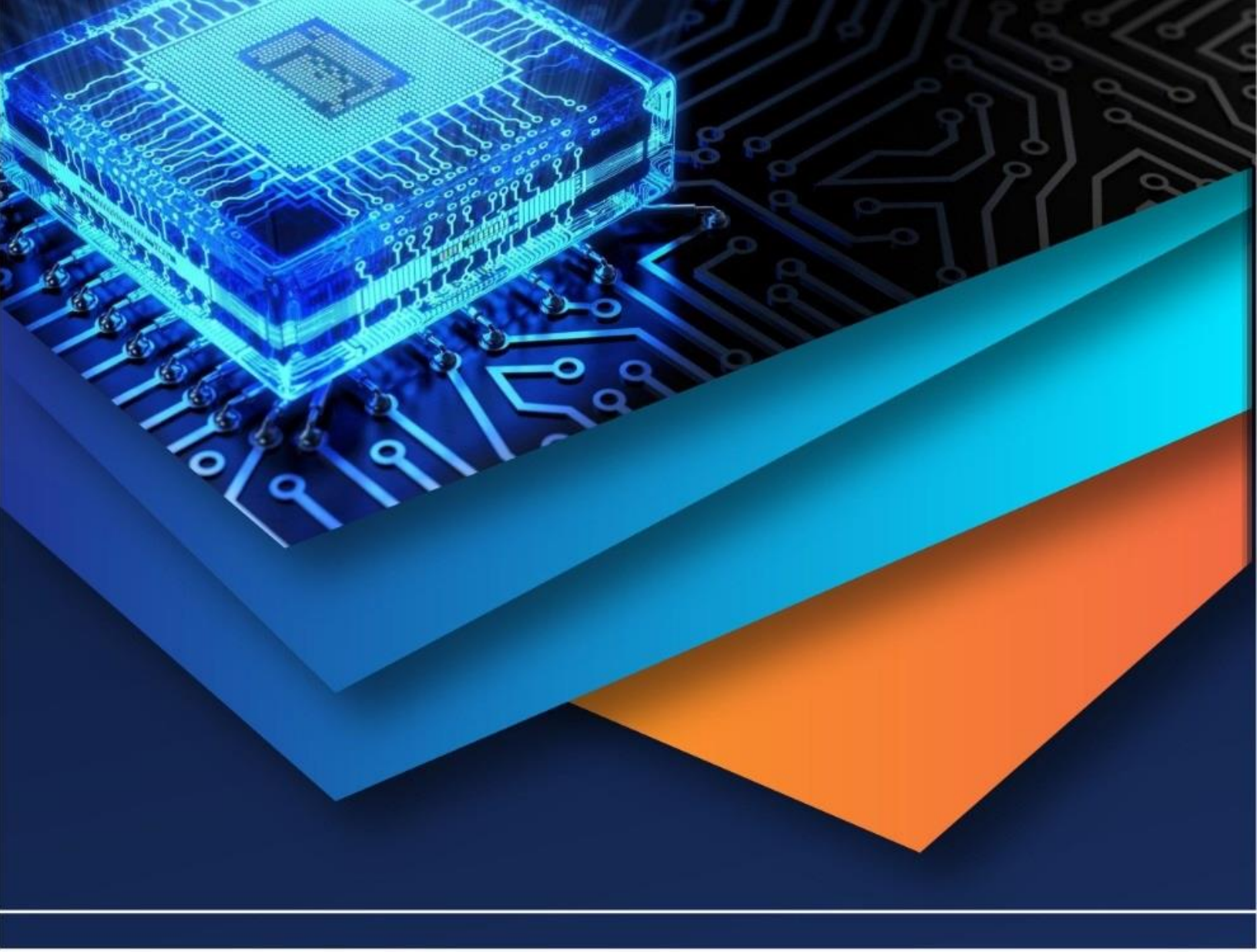
There are many technologies that assist users in writing their essays error-free and in an appropriate way. Our web application takes the input from users in two formats either text or file [then converts them to text]. The text is then processed such as tokenization, noise removal, word embedding, e.t.c. Then the process text is trained accordingly to the user-selected model and performs the task respectively.

VIII. ACKNOWLEDGMENTS

We are grateful for the assistance of our professor P. V. Nageswara Rao in the practical application of our knowledge in Machine Learning and Web Development. We sincerely consider it an honor to work and gain knowledge under the guidance of Bhramaramba Ravi and Gurpreet Singh Chhabra as our project reviewers. We sincerely offer our gratitude to the Department of Computer Science and Engineering and the respective faculty for providing this wonderful opportunity. We thank each and every one of those who participated and helped in the succession of our project, both directly and indirectly.

REFERENCES

[1] Bird, Steven. (2006). NLTK: The natural language toolkit. 10.3115/1225403.1225421.
[2] Damodaran,Prithiviraj. “PrithivirajDamodaran/Gramformer: A framework for detecting, highlighting and correcting grammatical errors on natural language text. Created by Prithiviraj Damodaran. Open to pull requests and other forms of collaboration.” GitHub, https://github.com/PrithivirajDamodaran/Gramformer. Accessed 22 March 2022.
[3] NLTK :: Natural Language Toolkit, https://www.nltk.org/. Accessed 22 March 2022
[4] Jaiswal, Nikhil. “SequenceMatcher in Python. A human-friendly longest contiguous &… | by Nikhil Jaiswal.” Towards Data Science, https://towardsdatascience.com/sequencematcher-in-python-6b1e6f3915fc. Accessed 5 April 2022.
[5] Welcome to deep_translator's documentation! — deep_translator documentation, https://deep-translator.readthedocs.io/en/latest/. Accessed 5 April 2022
[6] “streamlit/streamlit: Streamlit — The fastest way to build data apps in Python.” GitHub, https://github.com/streamlit/streamlit. Accessed 5 April 2022



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)