***Project Mid Report***

**SENTIMENT ANALYSIS ON ARTICLES**

**Submitted for the requirement of**

**Project course**

**COMPUTER SCIENCE & ENGINEERING**

****

**Submitted By:**

**Tushar Lohani (19BCS1559)**

**Ankit Kumar (19BCS1561)**

**Sakshi Kumari (19BCS1570)**

**Submitted To:**

**Charanpreet Kaur**

**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING**

**CHANDIGARH UNIVERSITY, GHARUAN**

**June 2021**

**1. Introduction**

This project is based on NLP that means Natural Language Processing. Natural language processing (NLP) is the ability of a computer program to understand human language as it is spoken and written -- referred to as natural language. It is a component of artificial intelligence.

Sentiment analysis is the process of using natural language processing, text analysis, and statistics to analyze customer sentiment. The best businesses understand the sentiment of their customers -- what people are saying, how they’re saying it, and what they mean. Customer sentiment can be found in tweets, comments, reviews, or other places where people mention your brand. Sentiment Analysis is the domain of understanding these emotions with software, and it’s a must-understand for developers and business leaders in a modern workplace.

As with many other fields, advances in deep learning have brought sentiment analysis into the foreground of cutting-edge algorithms. Today we use natural language processing, statistics, and text analysis to extract, and identify the sentiment of words into positive, negative.

To address the context issue, a lot of research surrounding sentiment analysis has focused on feature engineering. Creating inputs to a model that recognize context, tone, and previous indications of sentiment can help increase accuracy and get a better overall sense of what the author is trying to say. For an interesting example, check out this paper in Knowledge-Based Systems that explores a framework for this kind of contextual focus. Search engines also use a similar technique called semantic search that determines the intent and contextual meaning of users’ search terms.

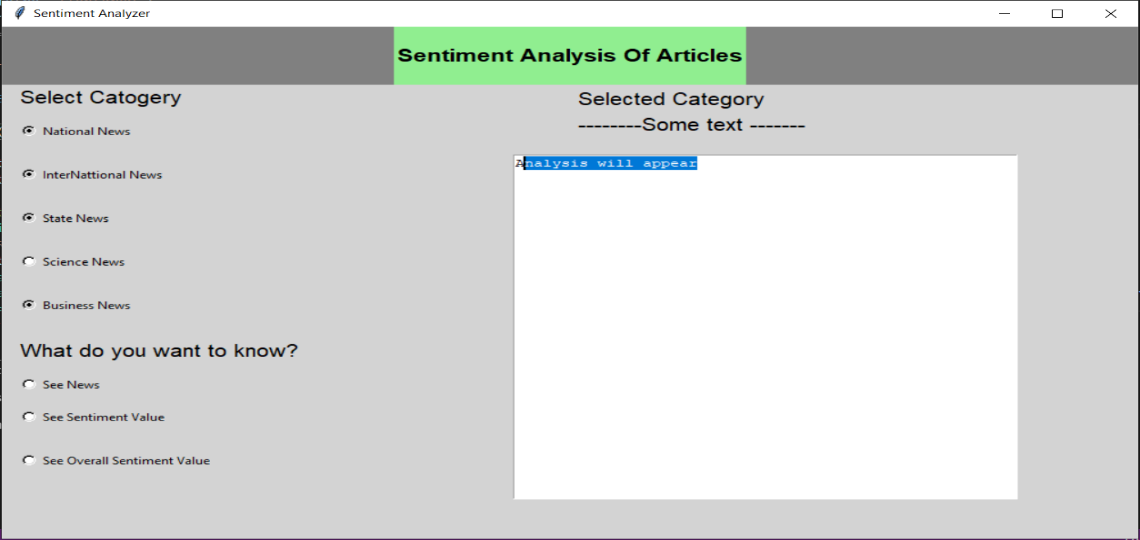


Fig:- GUI of sentiment analysis

**2. Background**

**2.1 Objectives:**

The principle targets of this task are:

1. To build up a sentiment analysis that can be used by everyone.
2. To make it cheap and easy to use.
   1. **Scope:**

The extent of this project is:

1. To provide the use of technology.
2. To make an application free or cheap to use.
3. Easily understand able.
   1. **Level of Access:**

The application will have two levels of access:

1. The Administrator

2. The Customer

**2.4 Summary:**

Sentiment Analysis is the process of determining whether a piece of writing is positive, negative or neutral. A sentiment analysis system for text analysis combines natural language processing (NLP) and machine learning techniques to assign weighted sentiment scores to the entities, topics, themes and categories within a sentence or phrase.

Sentiment analysis helps data analysts within large enterprises gauge public opinion, conduct nuanced market research, monitor brand and product reputation, and understand customer experiences. In addition, data analytics companies often integrate third-party Sentiment Analysis APIs into their own customer experience management, social media monitoring, or workforce analytics platform, in order to deliver useful insights to their own customers.

This article will explain how basic sentiment analysis works, evaluate the advantages and drawbacks of rules-based sentiment analysis, and outline the role of machine learning in sentiment analysis. Finally, we’ll explore the top applications of sentiment analysis before concluding with some helpful resources for further learning.

**3.Software and Hardware Requirements**

Using the requirement definition as a foundation, the requirements are divided into software and hardware. This is called system design.

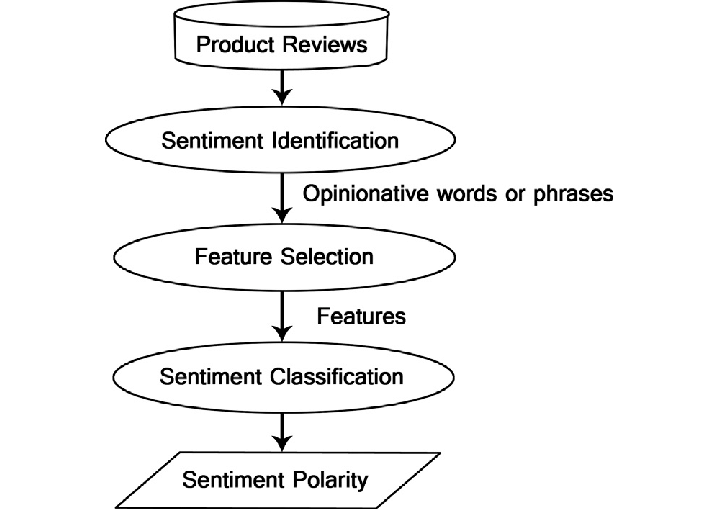
**3.1 Software requirements:**

The programming language to use will be Python.

The project will make use of Visual studio code.

**3.2 UML DIAGRAM:**

**UML DIAGRAM**



**3.3 PROJECT IMPLEMENTATION:**

from bs4 import BeautifulSoup

import requests

from urllib.request import urlopen

from urllib.error import HTTPError, URLError

mainURL = "https://asknkitkr.github.io/article/"

categories = [

    "categories/technology.html",

    "categories/book.html",

    "categories/entertainment.html",

    "categories/politics.html",

    "categories/product.html",

]

def checkURLConnection(url):

    try:

        urlopen(url)

    except HTTPError as e:

        print(e)

    except URLError as e:

        print(e)

    else:

        print("SUCCESS")

def webScrap(url):

    html = requests.get(url)

    soup = BeautifulSoup(html.text, "lxml")

    articles = soup.find\_all('div', class\_='card')

    for index, article in enumerate(articles):

        title = article.find('div', class\_='card-header').text.replace('  ', '')

        text\_articles = article.find('div', class\_='card-body').text.replace('  ', '')

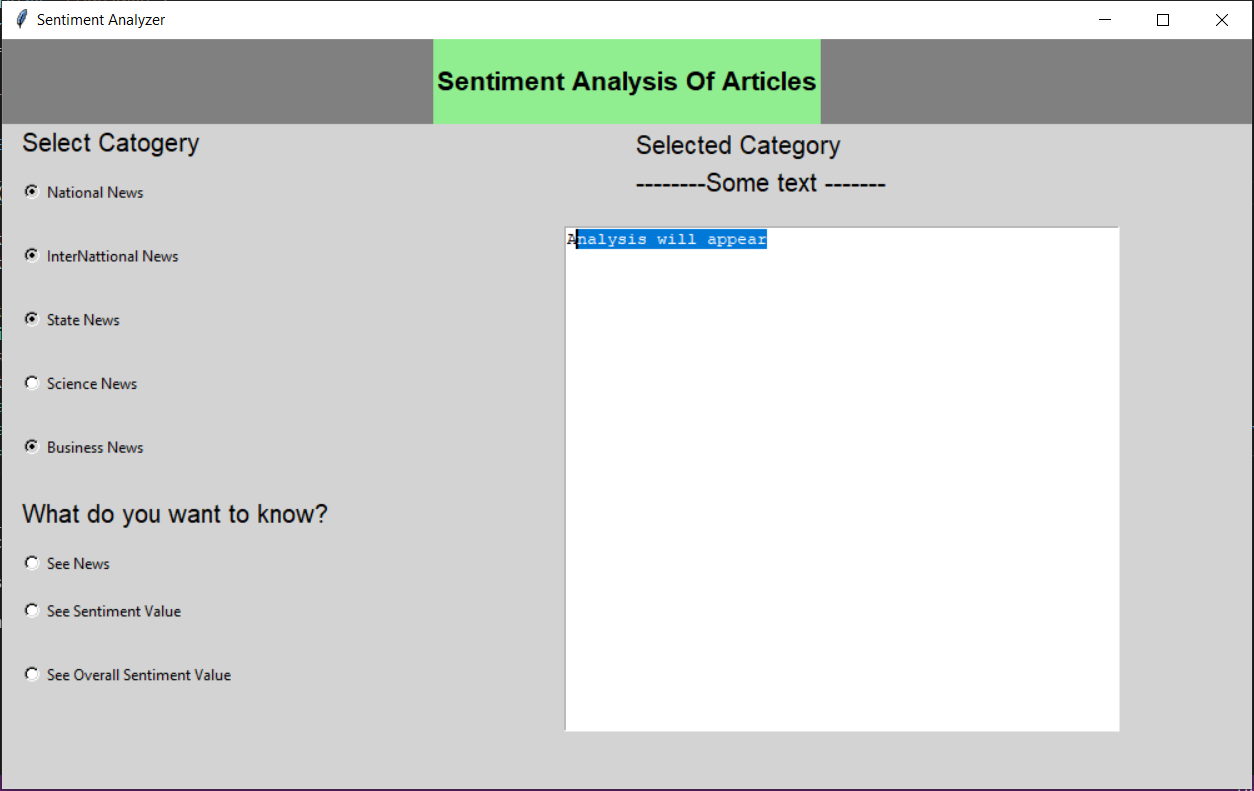
        print('{}. {}'.format(index+1, title.replace('\n', ' ')))

        print(text\_articles)

checkURLConnection(mainURL + categories[0])

webScrap(mainURL + categories[0])

**3.4 DESIGNING PHASE:**



**CODE:**

from tkinter import \*

root = Tk()

root.geometry ("1000x600")

root.title("Sentiment Analyzer")

f1 = Frame(root, bg="grey", relief=SUNKEN)

f1.pack(side=TOP, fill="x")

root.minsize(200,100)

root.maxsize(1200,900)

root.config(bg='lightgrey')

T = Text(root, height = 25, width = 55)

T.place(x=450,y=150)

Fact= ("Analysis will appear")

T.insert(END,Fact)

l = Label(f1,text="Sentiment Analysis Of Articles",font="Helvetica 16 bold",pady=20,bg='lightgreen')

l.pack()

Label(root, text = "Selected Category",bg='lightgrey',font= "lucida",padx=5).place(x = 500,y = 70)

Label(root, text = "--------Some text -------",font= "lucida",bg='lightgrey',padx=5,).place(x = 500,y = 100)

#var = IntVar()

var = StringVar()

var.set("Radio")

Label(root, text= "Select Catogery", font= "lucida",bg='lightgrey',justify=LEFT, padx=14).pack(anchor="w")

radio = Radiobutton(root, text= "National News", padx=14, bg='lightgrey', pady=14,variable=var,value="National News").pack(anchor="w")

radio = Radiobutton(root, text= "InterNattional News", bg='lightgrey',padx=14, pady=14, variable=var,value="InterNational News").pack(anchor="w")

radio = Radiobutton(root, text= "State News", padx=14, bg='lightgrey',pady=14, variable=var,value="State News").pack(anchor="w")

radio = Radiobutton(root, text= "Science News", padx=14,  bg='lightgrey',pady=14,variable=var,value="Science News").pack(anchor="w")

radio = Radiobutton(root, text= "Business News", padx=14, bg='lightgrey', pady=14,variable=var,value="Business News").pack(anchor="w")

var = StringVar()

var.set("Radio")

Label(root, text= "What do you want to know?", font= "lucida",justify=LEFT,bg='lightgrey', padx=14,pady=14).pack(anchor="w")

radio = Radiobutton(root, text= "See News", padx=14, variable=var,value="See News",bg='lightgrey').pack(anchor="w")

radio = Radiobutton(root, text= "See Sentiment Value", padx=14,pady=14, variable=var,value="See Sentiment Value",bg='lightgrey').pack(anchor="w")

radio = Radiobutton(root, text= "See Overall Sentiment Value", padx=14,pady=14, variable=var,value="See Overall Sentiment Value",bg='lightgrey').pack(anchor="w")

root.mainloop()

**3.5. Experimental Results and Discussion**

In this section of the document we will be discussing the verification and testing of each software component. All problems will be described in detail and the solutions we made to solve these problems. In this section we will also discuss our overall results of the project and what we could have done to improve upon our project. Future work for this project will also be mentioned in this section of the document.

**4.1 Thinking about Basic:**

Here we think about all basic things like how we design and create an application all the requirements and all the basic needs and think we requires for the project.

**4.2 Design Models and Mockups Designing:**

The models and mockups help to ensure clarity in view of the project as well as how it works. Stakeholders are to sit through this process as drawings are created.

**4.3 Application Creation:**

The application was created using python programming language. It was coded in Visual studio code and functions were created and all the library was imported using packages provided in python.

**4.4 Testing:**

Testing will be implemented on software. Test cases may be used to guide and understand the basic actions of both customers and employees. Any bugs or errors that occur will be identified and resolved.

**4.5 Finalization and Reports:**

All testing and function processes are finalized at this stage. Reports will be created to ensure all information and functionality is clear in order to make the user manual and to help ensure employees can use the software with ease.

**5.Conclusions:**

In conclusion, the system will be able to serve as an application when it is finally developed, where these small upcoming companies can make use of it to publish their services in a wide range and also help the company to manage their service more effectively. On the other hand, it will enable customers to freely make their desire choice more freely and interactively. In a nutshell, it can be summarized that the future scope of the project circles around maintaining information regarding:

* We can add home automation.
* We can give more advance software for virtual assistant including more facilities.
* We will host the platform on online servers to make it accessible worldwide.