

# **NEWSIN - A NEWS SUMMARIZER** **AND ANALYZER**

Submitted in partial fulfillment of the requirements of the degree of

## **BACHELOR OF COMPUTER ENGINEERING**

by

Anushree Salunke (20102179)

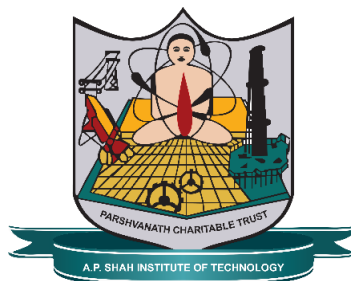
Eisha Saini (20102025)

Pooja Tumma (20102126)

Sanskriti Shinde (21202018)

Guide:

**PROF. SUCHITA DANGE**



Department of Computer Engineering

A. P. SHAH INSTITUTE OF TECHNOLOGY, THANE

(2022-2023)



A. P. SHAH INSTITUTE OF TECHNOLOGY, THANE

## CERTIFICATE

This is to certify that the Mini Project 2B entitled “**NEWSIN-A News Summarizer and Analyzer**” is a bonafide work of “**Anushree Salunke (20102179)**”, “**Eisha Saini (20102025)**” , “**Pooja Tumma (20102126)**”, “**Sanskriti Shinde (21202018)**” submitted to the University of Mumbai in partial fulfillment of the requirement for the award of the degree of **Bachelor of Engineering in Computer Engineering**

---

**Guide Name**  
**Prof. Suchita Dange**

---

**Project Coordinator**  
**Prof. Deepak Khachane**

---

**Head of Department**  
**Prof. Sachin H.Malave**

Date :



## A. P. SHAH INSTITUTE OF TECHNOLOGY, THANE

### Project Report Approval for Mini Project-2A

This project report entitled ***NEWS SUMMARIZER AND ANALYZER*** is “Anushree Salunke (20102179)”, “Eisha Saini (20102025)”, “Pooja Tumma (20102126)”, “Sanskriti Shinde (21202018)” approved for the degree of ***Bachelor of Engineering in Computer Engineering, 2022-23.***

Examiner Name

1. \_\_\_\_\_

2. \_\_\_\_\_

Signature

\_\_\_\_\_

\_\_\_\_\_

Date:

Place:

# Declaration

We declare that this written submission represents my ideas where others ideas or words have been included, I have adequately cited and referenced the original sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. I understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

-----

Anushree Salunke (20102179)

-----

Eisha Saini (20102025)

-----

Pooja Tumma (20102126)

-----

Sanskriti Shinde (21202018)

Date:

# Abstract

A summary condenses a lengthy document by highlighting salient features. It helps the reader to understand completely just by reading a summary so that the reader can save time and also can decide whether to go through the entire document. Summaries should be shorter than the original article so make sure to select only pertinent information to include the article. The main goal of a newspaper article summary is, the readers to walk away with knowledge on what the newspaper article is all about without the need to read the entire article. This work proposes a news article summarization system which access information from various local online newspapers automatically and summarizes information using heterogeneous articles. To make ad-hoc keyword based extraction of news articles, the system uses a tailor-made web crawler which crawls the websites for searching relevant articles. Computational Linguistic techniques mainly Triplet Extraction, Semantic Similarity calculation and OPTICS clustering with DBSCAN is used alongside a sentence selection heuristic to generate coherent and cogent summaries irrespective of the number of articles supplied to the engine. The performance evaluation is one using the ROUGE metric. The rapid progresses in digital data acquisition techniques have led to a huge volume of news data available in the news websites. Most of such digital news collections lack summaries. Due to that, online newspaper readers are overloaded with lengthy text documents. Also, it is a tedious task for human beings to generate an abstract for a news event manually since it requires a rigorous analysis of the news documents. An achievable solution to this problem is condensing the digital news collections and taking out only the essence in the form of an automatically generated summary which allows readers to make effective decisions in less time. The graph based algorithms for text summarization have been proven to be very successful over other methods for producing multi document summaries. The summary generated from knowledge graphs is more in line with human reading habits and possesses the logic of human reasoning. Due to the fast growing need of retrieving information in abstract form, we are proposing a novel approach for abstractive news summarization using the knowledge graphs to fulfill the need of having more accurate automatic abstractive news summarization and analyzer.

# Table of Contents

SR.NO	CONTENTS	PAGE.NO
1)	INTRODUCTION	7
2)	LITERATURE SURVEY	10
3)	PROBLEM STATEMENT AND OBJECTIVE SCOPE	16
4)	EXPERIMENTAL SETUP	17
5)	TECHNOLOGY STACK	18
6)	PROPOSED SYSTEM	20
7)	UML DIAGRAMS	20
8)	ALGORITHM	27
9)	METHODOLOGY	28
10)	CODE AND OUTPUT SCREENSHOTS	29
11)	RESULTS	38
12)	REFERENCES	43
13)	ANNEXURE	44
14)	ACKNOWLEDGEMENT	45

## List of Figures

1. Use Case Diagram	21
2. Data Flow Diagram (level 0)	22
3. Data Flow Diagram (level1)	23
4. Sequence Diagram	24
5. Activity Diagram	25

## List of Graphs

1. Gantt Chart	44
----------------	----

# Chapter 1

## Introduction

News is one of the foremost critical channels for obtaining data. In any case, it is more troublesome to extricate comparisons in news articles than in audits. The viewpoints are exceptionally different in the news. They can be the time of the occasions, the individual included, the states of mind of members, etc. These angles can be communicated expressly or verifiably in numerous ways. For illustration, “storm” and “rain” both talk around “weather”, and hence they can shape a potential comparison. All these issues raise awesome challenges to comparative summarization within the news domain. The errand of news summarization is to briefly entirety up the commonalities and contrasts between two comparable news points by utilizing human discernable sentences. The summarization framework is given two collections of news articles, each of which is related to a subject. The framework ought to discover idle comparative perspectives, and produce depictions of those viewpoints in a pairwise way, i.e. counting portrayals of two themes at the same time in each viewpoint. For illustration, when comparing the seismic tremor in Haiti with the one in Chile, the rundown ought to contain the escalation of each temblor, the harms in each fiasco zone, the responses of each government, etc. These days the huge volume of data in electronic frame is expanding quickly. It can be organized information like databases, company bequest information; or unstructured information like content, pictures etc. Around 85 and 90% of information is held in unstructured frames. Subsequently, content mining is essential for extricating and overseeing valuable data from unstructured sets of information, such as news reports, emails and web pages, utilizing different content mining procedures. Hence, text mining has become an imperative and dynamic inquiry about the field.

It is well known that content mining strategies have for the most part been created for the English dialect since most electronic information is in English. Utilizing this to our advantage, it is an self-evident another step to utilize these procedures for filtering through the large number of accessible online information to mine truths and figures from different sources and after that summarize them proficiently to utilize in following different occasions in and around



an range beneath Police law. In this paper, the data extraction of news articles is based on computational etymological methods to summarize the content. The summarization handle includes sifting, highlighting and organizing data which is concise, coherent and steadfast to the initial record. With the quick development of the broadcast frameworks, the web and online data administrations, increasingly data is accessible and available. Blast of data has caused a well recognized data over-burden issue. There's no time to examine everything and yet we ought to make basic choices based on anything data is accessible. Programmed summarization, done by machine, postures a or maybe challenging issue to computer researchers, due to the abstractness and complexity of human dialects. This issue was recognized and handled early within the 1950s. Since then a few well-known calculations have been created, but the accomplishment was bound by the confinement of the current Normal Dialect Processing technologies, and until nowadays it still remains an dynamic inquiry about the theme. The objective of this venture isn't to move forward on the existing calculations, but to consider and apply these calculations, combining with other valuable methods to deliver common sense. It recovers broadcast tv news from a record, analyzes the substance to recognize news stories by subtopic dialog. Substance of each story is summarized and imperative catchphrases are extricated. This data is to be put away in a central database, and a Data Recovery framework is to be actualized which lets clients hunt for any piece of news within the database. Such a framework has an advantage over other search engines, as we utilize summarization methods to find the story he/she is searching for in a shorter time, when compared to a standard content based look engine.

The key assignments in summarization are as follows:

1. Consequently extricate online articles from news websites based on a keyword.
2. Partition whole articles as a gathering of sentences, which acts as the dataset for advance processing.
3. Speaking to sentences in a machine discernable and justifiable format.
4. Identifying semantic likeness between sentences so as to dispense with real excess in summary.
5. Clustering comparative sentences to recognize between semantically distinctive sentences.

6. Picking sentences among clusters which speak to the data displayed by the comparing cluster.
7. Orchestrating the sentences chronologically to show the advancements as they happened.

News is one of the foremost critical channels for obtaining data. Be that as it may, it is more troublesome to extricate comparisons in news articles than in surveys. The viewpoints are exceptionally different in the news. They can be the time of the occasions, the individual included, the states of mind of members, etc. These perspectives can be communicated unequivocally or certainly in numerous ways. For example, “storm” and “rain” both talk about “weather”, and in this way they can frame a potential comparison. All these issues raise extraordinary challenges to comparative summarization within the news domain. The assignment of news summarization is to briefly whole up the commonalities and contrasts between two comparable news themes by utilizing human lucid sentences. The summarization framework is given two collections of news articles, each of which is related to a point. The framework ought to discover idle comparative viewpoints, and create portrayals of those angles in a pairwise way, i.e. counting portrayals of two points at the same time in each perspective. For illustration, when comparing the seismic tremor in Haiti with the one in Chile, the rundown ought to contain the concentration of each temblor, the harms in each calamity zone, the responses of each government, etc.

# Chapter 2

## Literature Survey

[1]An outline condenses a long record by highlighting notable highlights. It makes a difference for the peruser to get it totally fair by perusing rundown so that the peruser can spare time and can choose whether to go through the complete report. Rundowns ought to be shorter than the first article so make beyond any doubt that to choose as it were relevant data to incorporate the article. The most objective of newspaper article summary is, the readers to walk absent with information on what the daily paper article is all approximately without the have to peruse the complete article. This work proposes a news article summarization framework which gets to data from different nearby on-line daily papers consequently and summarizes data utilizing heterogeneous articles. To form ad-hoc watchword based extraction of news articles, the framework employs a tailor-made web crawler which slithers the websites for looking at significant articles. Computational Phonetic procedures primarily Triplet Extraction, Semantic. Closeness calculation and OPTICS clustering with DBSCAN is utilized nearby a sentence determination heuristic to create coherent and persuasive rundowns independent of the number of articles provided to the motor. The execution assessment is done utilizing ROUGE metric.Extraction of a single rundown from numerous reports has intrigued since the mid-1990s, most applications being within the space of news articles. A few Web based news clustering frameworks were propelled by inquiry about multi-document summarization, for illustration Columbia NewsBlaster, or News In Substance. Typically distinctive from single-document summarization since the issue includes different sources of data that cover and supplement each other, and evacuation of repetitive realities which are displayed in a semantically comparative but syntactically diverse structure. Content Summarization has continuously been a region of dynamic intrigue within the scholarly community. In later times, indeed in spite of the fact that a few strategies have been created for programmed content summarization, proficiency is still a concern. Given the increment in estimate and number of reports accessible online, an productive programmed news summarizer is the require of the hour. In this paper, we propose a strategy of content summarization which centers on the problem of distinguishing the foremost imperative parcels of the content and creating coherent rundowns. In our strategy, we do not require full semantic translation of the content, instead we make a summary employing a show of theme movement within the content determined from lexical chains. We

show an optimized and proficient calculation to create content rundown utilizing lexical chains and utilizing the WordNet thesaurus. Within the time of present day Information science and Big Data, it is not a ponder to empower machine learning to get its human dialect and know what individuals are feeling and considering with their surroundings. The term is called Opinion Examination or Supposition Mining which combines the power of normal dialect preparation, content examination and computational etymology to classify subjective information or the passionate state of the writer/subject/topic. Rather than just identifying a positive/negative/neutral estimation, being able to extricate catchphrases that intensifies different feelings such as bliss, fervor, dissatisfaction, fear etc. from the substance. Progresses in individual computing and data innovations have in a general sense changed how maps are created and expanded, as numerous maps nowadays are exceedingly intelligent and conveyed online or through portable gadgets. Appropriately, we ought to consider interaction as an essential complement to representation in cartography and visualization. UI (client interface) / UX (client encounter) portrays a set of concepts, rules, and workflows for fundamentally considering the plan and utilization of an intuitive item, map-based or something else. To see what the world is looking for, there's a Trending Looks page expansion to Google Trends that distributes the foremost frequently searched terms alongside their look volume and related news stories of the past 24-hour across different nations. The URL of every day search trend list is accessible at <https://trends.google.com/trends/trendingsearches/daily?geo=US>. For this project, the first trending look theme is chosen and analyzed to form a content analytics visualization report. To get the foremost later information about the look subject, different news articles related to the look topic and Twitter information source is used. Advance, we moreover overcome the restrictions of the lexical chain approach to produce a great rundown by executing pronoun determination and by recommending unused scoring strategies to use the structure of news articles. The prior approaches in content summarization centered on deriving content from lexical chains produced amid the topic progression of the article. These approaches were preferred since it did not require full semantic elucidation of the article. The approaches moreover blended a few strong knowledge sources like a part-of-speech tagger, shallow parser for the identification of ostensible bunches, a division algorithm and the WordNet thesaurus. According to Wikipedia, WordNet could be a lexical database for the English dialect. It bunches English words into sets of synonyms called synsets, gives brief definitions and usage examples, and records a number of relations among these synonym sets or their individuals. For illustration two faculties of “bike” are spoken as: cruiser, bicycle and bicycle, bike, wheel, cycle.

[2] Words of the same category are linked through semantic relations like synonymy, which is the study of words with the same or comparative meaning, or the quality of being comparative, and hyponymy, which relates to words of more particular meaning than a common or superordinate term applicable to it. A commonly used example to demonstrate hyponym is: daffodil, which may be a hyponym for blossom. [4] All these information sources are freely accessible substances. This project speaks to a framework that allots opinion scores and extricates key feelings associated with the conclusion communicated in these news stories and Twitter posts on a certain trending search topic. In spite of the fact that full comprehension of characteristic dialect content remains well past the control of machines, the actualized measurable investigation of tolerably straightforward opinion signs can provide an important quantitative rundown of these expansive sums of subjective information. The venture is essentially executed on Microsoft Control BI, Python and R programming platforms. Power BI may be an information visualization device that underpins a huge run of information sources (for all intents and purposes any data source) to stack, change and clean the information into an information model. An awesome version of Control BI is ready to interface to a web page and convert its information into a dataset. In Control BI, the dataset is usually alluded to as an Inquiry or Table. Morris, Jane and Hirst to begin with presented the concept of lexical chains. In any given article, the linkage among related words can be utilized to create lexical chains. A lexical chain may be a consistent gathering of semantically related words which delineate a thought within the archive. The connection between the words can be in terms of equivalent words, characters and hypernyms/hyponyms. These relations can be utilized to bunch things occurrences in a lexical chain given the condition that each thing is allocated to only one chain. The challenging errand here is deciding the chain to which a specific thing will be doled out since it may have different faculties or settings. Too, indeed in spite of the fact that there's a single setting for the noun usage, it may well be still ambiguous to decide the lexical chain. This passage presents center concepts from UI/UX plan critical to cartography and visualization, centering on issues related to visual plan. To begin with, a principal qualification is made between the utilization of an interface as an instrument and the broader encounter of an interaction, a refinement that isolates UI plan and UX plan. Norman's stages of the interaction system at that point is summarized as a direct demonstration for understanding the client encounter with intuitive maps, noticing how diverse UX plan arrangements can be connected to breakdowns at diverse stages of the interaction. At last, three measurements of UI plan are depicted: the basic interaction administrators that frame the essential building pieces of an interface, interface styles that execute these administrator primitives, and suggestions for visual plan of an interface. Summarization has been seen as a two-step process.

[3]The first step is the extraction of vital concepts from the source content by building a halfway representation of some sort. The moment step employs this middle of the road representation to create a rundown. To analyze and get a huge amount of unstructured information like client conclusion, user feedback, item audits, Content Analytics is utilized to determine meaning out of content and written communications. There are a few diverse strategies utilized to analyze content and unstructured data. To rapidly distinguish common themes and issues that arise among clients, recurrence of these themes can be numbered. Some of the time a bunch of words can provide more understanding than a fair one word alone. Interior GIScience interaction most commonly is treated by the exploration push of geographic visualization. Interactivity reinforces visual thinking, engaging clients to externalize their thinking by inquiring a wide run of one of a kind diagram representations, thus overcoming the hindrances of any single diagram arrangement. Geovisualization engages this instinctively considering the reason for examination rather than communication, with the objective of making present day hypotheses and unconstrained encounters around darkened geographic wonders and shapes. As a result, much of the early work on interaction in cartography and visualization is specific to coherent disclosure, considering ace experts as the target client bunch. UI and UX are not the same, isolated in their center on interfacing versus intelligence. An interface may be an apparatus, and for advanced mapping this instrument empowers the client to control maps and their fundamental geographic data. An interaction is broader than the interface, depicting the two-way question-answer or request-result discourse between a human client and a computerized question interceded through a computing gadget. Subsequently, an interaction is both contingent—as the reaction is based on the task, making loops of interactivity—and empowering—giving the client organization within the mapping prepare with changes unexpected on his or her interface and needs. Our overview into the work wiped out the field of summarization examined and made a difference in the issues mentioned and the challenges within the field. We implemented the basic lexical chain demonstrated as examined by Silber and McCoy and after that included our upgrades to resolve the issue of anaphora determination and time complexity of lexical chain era. Since we are summarizing news articles, a few extraction on the basis of legitimate things is essential. We earlier described our scoring procedure for legitimate nouns. Nowadays, UI/UX plan requires thought to utilize cases past exploratory geovisualization and clients past master analysts. Interaction permits clients to see numerous areas and outline scales as well as customize the representation to their interface and needs. Interaction moreover engages clients within the cartographic plan handle, progressing availability to geographic data and dissolving conventional boundaries between mapmaker and outline client.

[4]Progressively, interaction empowers geographic examination, connecting computing to cognition in order to scale the human intellect to the complexity of the mapped wonder or prepare. In like manner, interaction has been recommended as an essential complement to representation in cartography, together organizing modern cartographic grant and hone. For dialog of extra impacts on UI/UX plan in cartography and visualization, see Geocollab. Content analytics offer tremendous openings for companies in any case of industry. Companies and individuals need to settle on superior educated commerce choices based on identifiable and quantifiable knowledge. With movements in Content Examination, companies can presently be able to mine text to get experiences and make strides to their benefit to thrive within competitive advertising. There are countless businesses that can be benefitted by applying content analytics to collate and act on company's information. Maybe, the application is most suited to showcase information about the media space that a company lives in and how it is gotten by its gathering of people. We were able to auto-summarize news articles and compare outlines created by them to analyze what scoring parameters would lead to way better results. Within the handle, we tweaked strategies we had investigated on to use the fact that we were managing with news articles as it were. We found that journalists take after a settled design to type in a news article. They start with what happened and when it happened within the first paragraph and proceed with an elaboration of what happened and why it happened within the talking after sections. We wanted to use this information while scoring the sentences by giving the things showing up within the to begin with sentence the next score. But after checking on the preparatory comes about our scoring method as depicted in Barzilay and Elhadad we realized that the first sentence always got a high score since it had things that were rehashed a few times within the article. This can be intuitively consistent since the primary sentence of the article continuously has nouns that the article talks about almost. A number of disciplines, callings, and information ranges contribute to UI/UX plan, counting ergonomics, realistic plan, human-computer interaction, data visualization, brain research, convenience building, and web plan. Extra systems for understanding UX plans have been advertised as UX gets to be formalized conceptually and professionally. For occurrence, Fitts' law giving an early understanding of indicating intuitive was based on brain research thinks about almost human substantial development, Encourage, Foley et al.'s three plan levels were determined from inquire about on human-computer interaction whereas Garrett's five planes of plan are advertised from web plan involvement. At last, most proposals depict UI/UX experience. At long last, most proposals portray UI/UX as a plan prepare that incorporates numerous, user-centered assessments, making use of strategies and measures built up in Ease of use Designing (see Convenience

Building). In later years, the blast of social media has made accessible an exceptional sum of real-time information to a degree of open conclusion. Agreeing to the *Unused York Times*, more than one billion election-related tweets were posted on Twitter amid the final presidential decision, from the to begin with presidential talk about until decision day . As a social media stage, Twitter has emerged as a well known communication channel between pioneers of challenging parties and voters. Amid decision campaigns, the challenging parties and voters express their conclusions on Twitter generating a tremendous sum of unstructured information. Interested parties can utilize this information to monitor election campaigns, gauge political polarization and negative campaigning, and indeed forecast election comes about combined with conventional off-line race surveying at any point in time. The Text Analytics module of this framework can offer assistance to reply to a few basic questions like which issues are getting more consideration from the open, what candidates are gathering more positive / negative sentiment, people's response to occasions amid the decision campaign in genuine time etc. As with representation plans and the visual factors , an interaction can be deconstructed into its essential building pieces . Interaction primitives portray the basic components of interaction that can be combined to make an interaction technique. Researchers in cartography and related areas identify advancement of a scientific categorization of interaction primitives as the foremost squeezing required for the understanding of interaction, as such a scientific categorization verbalizes the total solution space for UI/UX plan. In like manner, there are presently a run of scientific categorizations advertised within the UI/UX writing, counting scientific categorizations particular to cartography and visualization .



# Chapter 3

## Problem Statement

In the current world situation when there is a rapid increase in technology, the data on the World Wide Web is increasing at a tremendous rate. However due to the hectic schedule of the people and an intense amount of news available on various different websites it becomes difficult for people to be daily updated with the knowledge of the surroundings. Also as the web is getting developed on a daily basis the news which might get surfaced on the web may not provide an overview of the news . We are not able to analyze what news should be read.

## Objectives and Scope

1. To extract data from a News Website.
2. To categorize the news as per the domains.
3. To provide summarize content of the news
4. To provide the analyzed news in the form of graphs, charts along with categories.
5. To extract data through news Website
6. Segregating that news as per the Domain
7. Extracting the domain data count for summarization and analyzation
8. To store that count of the domain data in the database
9. Summarizing the news and displaying them
10. After summarization is done then Extracting the domain data count from the database for analyzing.
11. Then as per category displaying that data in graphical representations

# Chapter 4

## Experimental Setup

### Hardware Requirements

Basic 64 bit windows 10 Laptop with i3 core processor.

Windows 10 (8u51 and above)

RAM: 128 MB

Disk space: 124 MB for JRE;

Processor: Minimum Pentium 2 266 MHz processor

### Software Requirements

HTML 5/CSS 3 : FRONT-END

Python 3.10 : BACKEND

NLP

Plotly

Application Connected via : News Website

Database : MySQL

# Technology Stack

## HTML:

The Hyper Text Markup Language, or HTML is the standard markup language for documents designed to be displayed in a web browser. It can be assisted by technologies such as Cascading Style Sheets (CSS) and scripting languages such as JavaScript.

## CSS:

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation document written in a markup language such as HTML. CSS is a cornerstone technology of the World Wide Web, alongside html and JavaScript.

## MYSQL:

MySQL is a fast, easy-to-use RDBMS being used for many small and big businesses. MySQL is a very powerful program in its own right. It handles a large subset of the functionality of the most expensive and powerful database packages. MySQL uses a standard form of the well-known SQL data language. MySQL works on many operating systems and with many languages including PHP, PERL, C, C++, JAVA, etc.

## **Plotly Library:**

Plotly's Python graphing library makes interactive, publication-quality graphs. Examples of how to make line plots, scatter plots, area charts, bar charts, error bars, box plots, histograms, heatmaps, subplots, multiple-axes, polar charts, and bubble charts.

## **NLTK :**

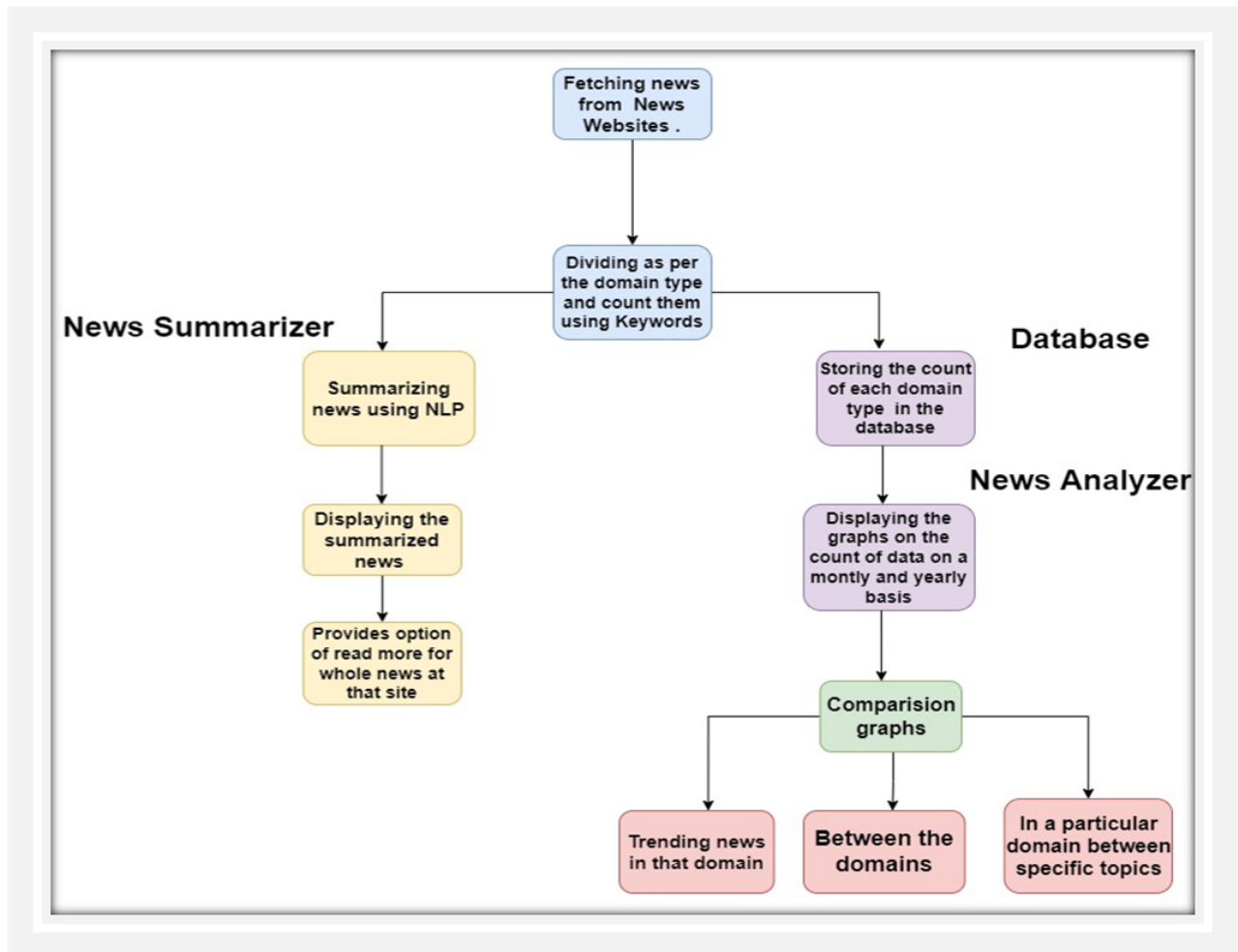
NLTK -- the Natural Language Toolkit -- is a suite of open source Python modules, data sets, and tutorials supporting research and development in Natural Language Processing. NLTK requires Python version 3.7, 3.8, 3.9 or 3.10.

# Chapter 5

## Proposed System

### 5.1 Flow Chart :-

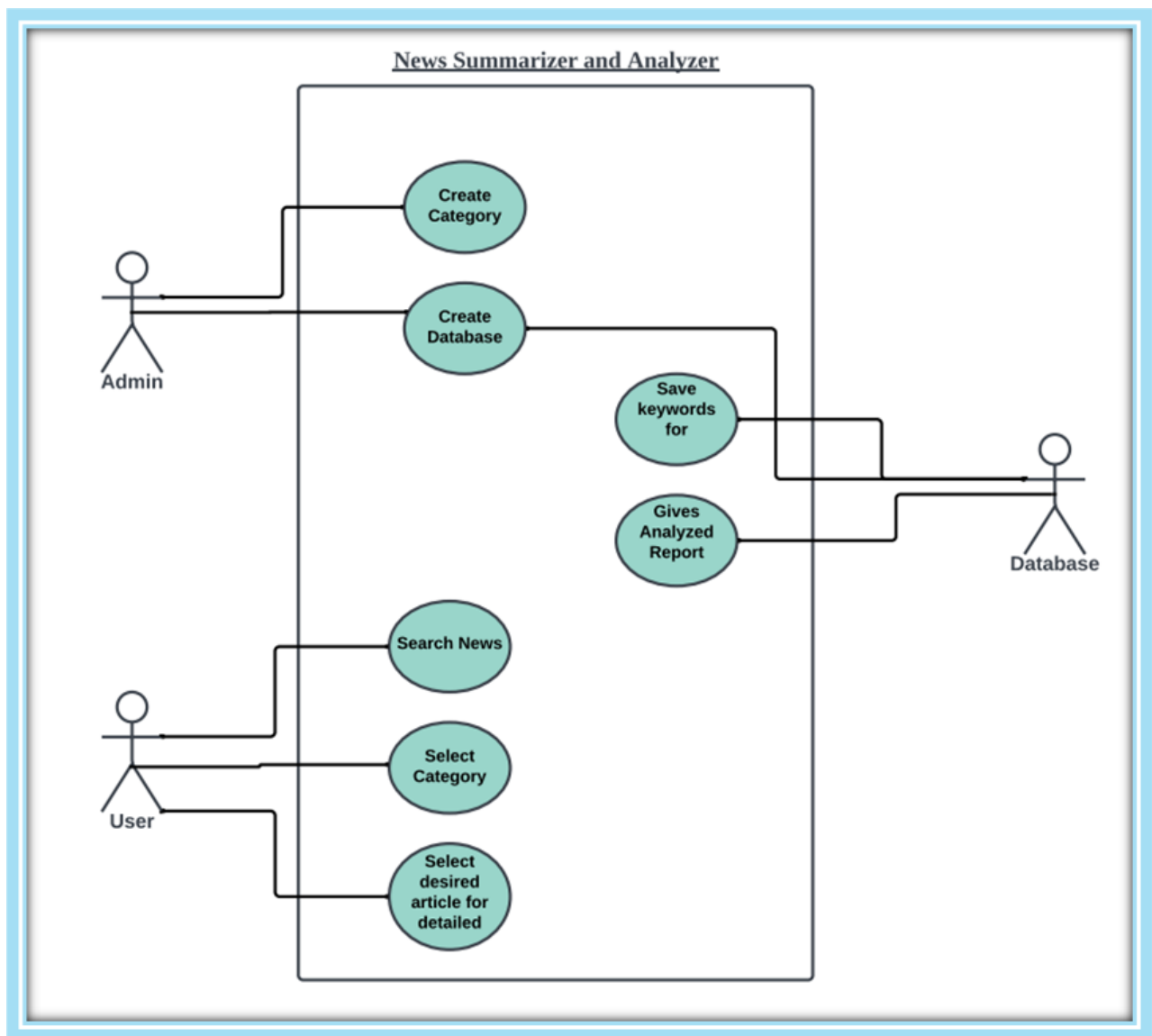
A flowchart is a picture of the separate steps of a process in sequential order.



## 5.2 UML diagram

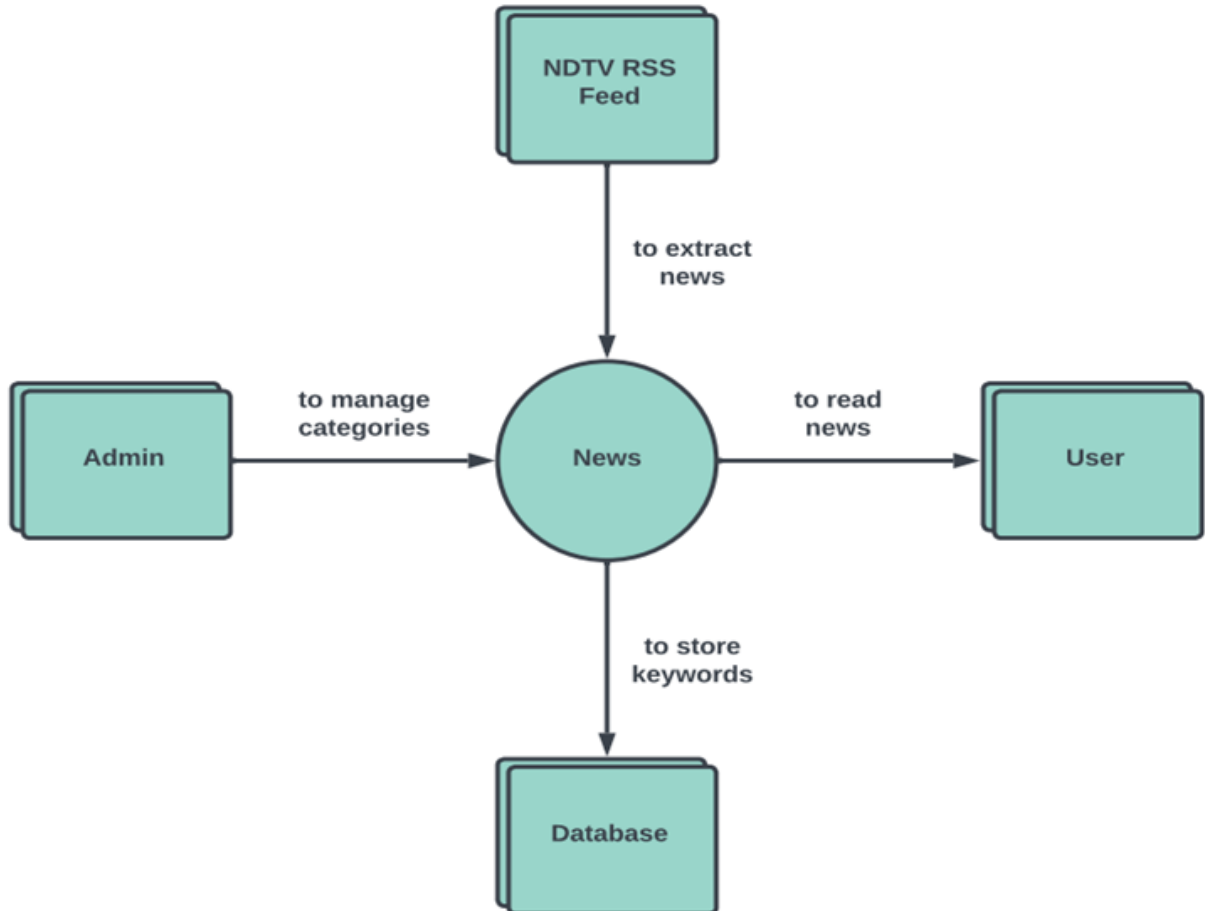
### 5.2.1 Use Case :-

Use case diagram is a graphic depiction of the interactions among the elements of a system. Use cases will specify the expected behavior, and the exact method of making it happen.

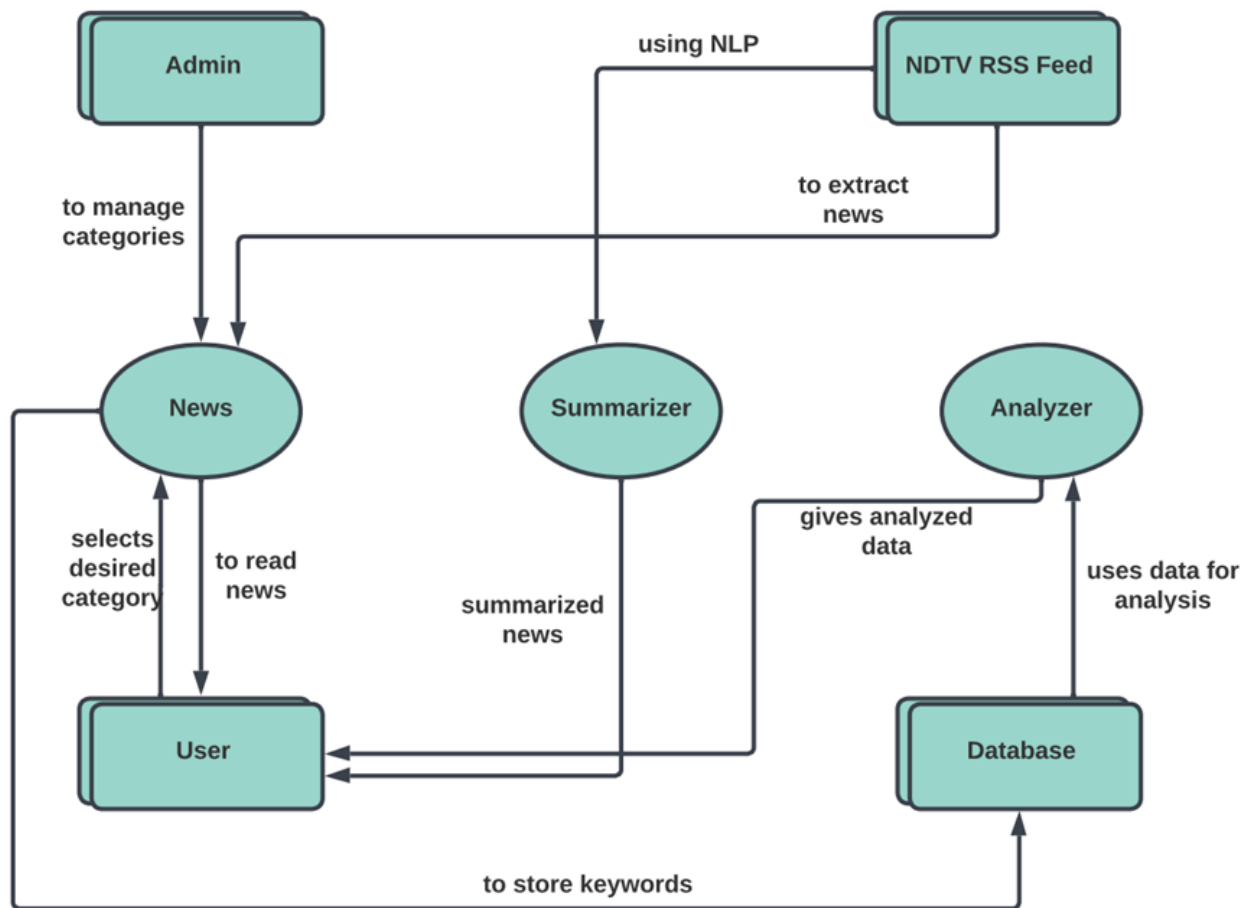


### 5.2.2 Data Flow Diagram (Level 0) :-

A data flow Diagram is a graphical representation of the “flow” of data through an information system, modeling its process aspects. A DFD is often used as a preliminary step to create an overview of the system without going into great detail, which can later be elaborated.

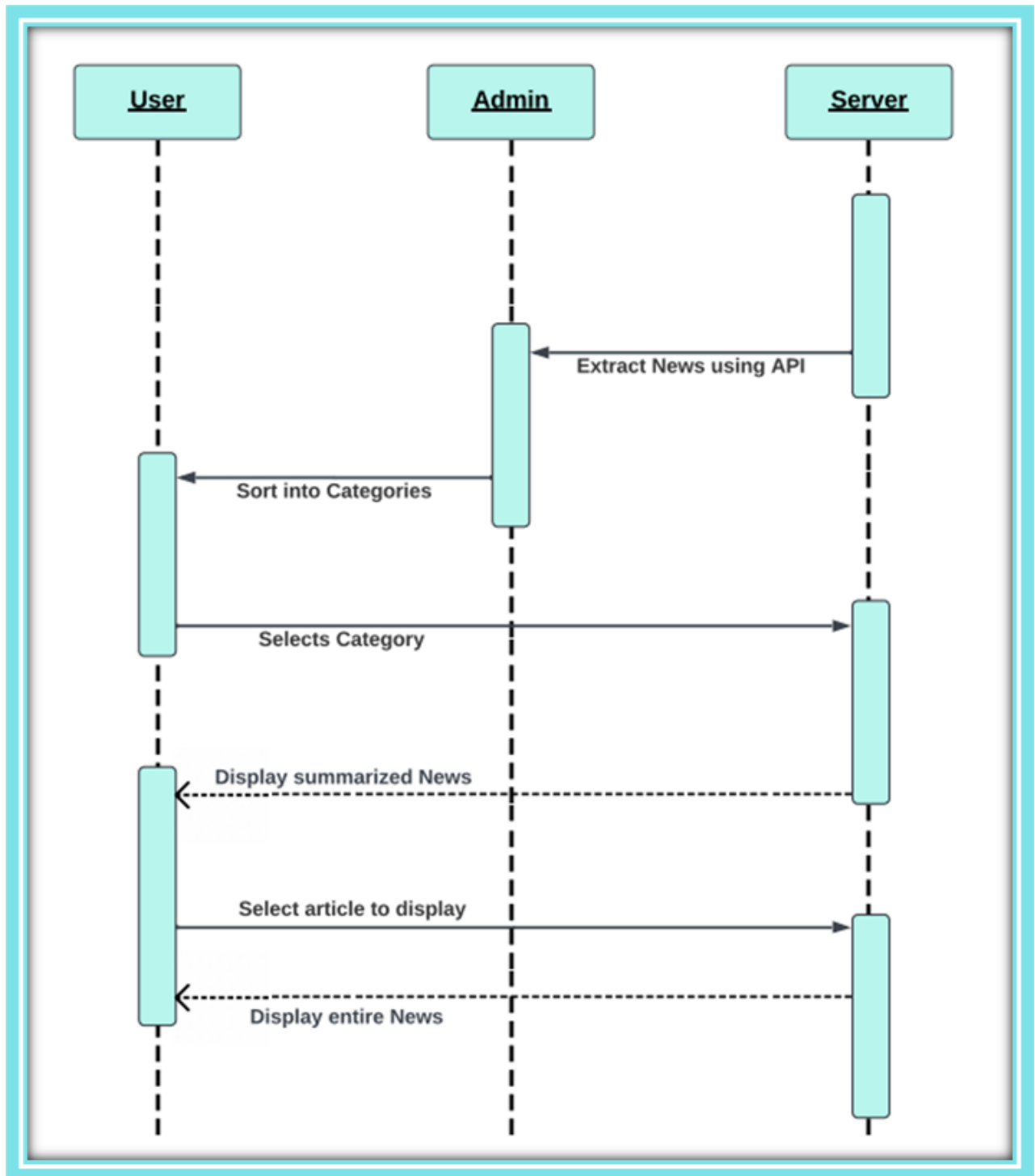


### 5.2.3 Data Flow Diagram (Level 1) :-



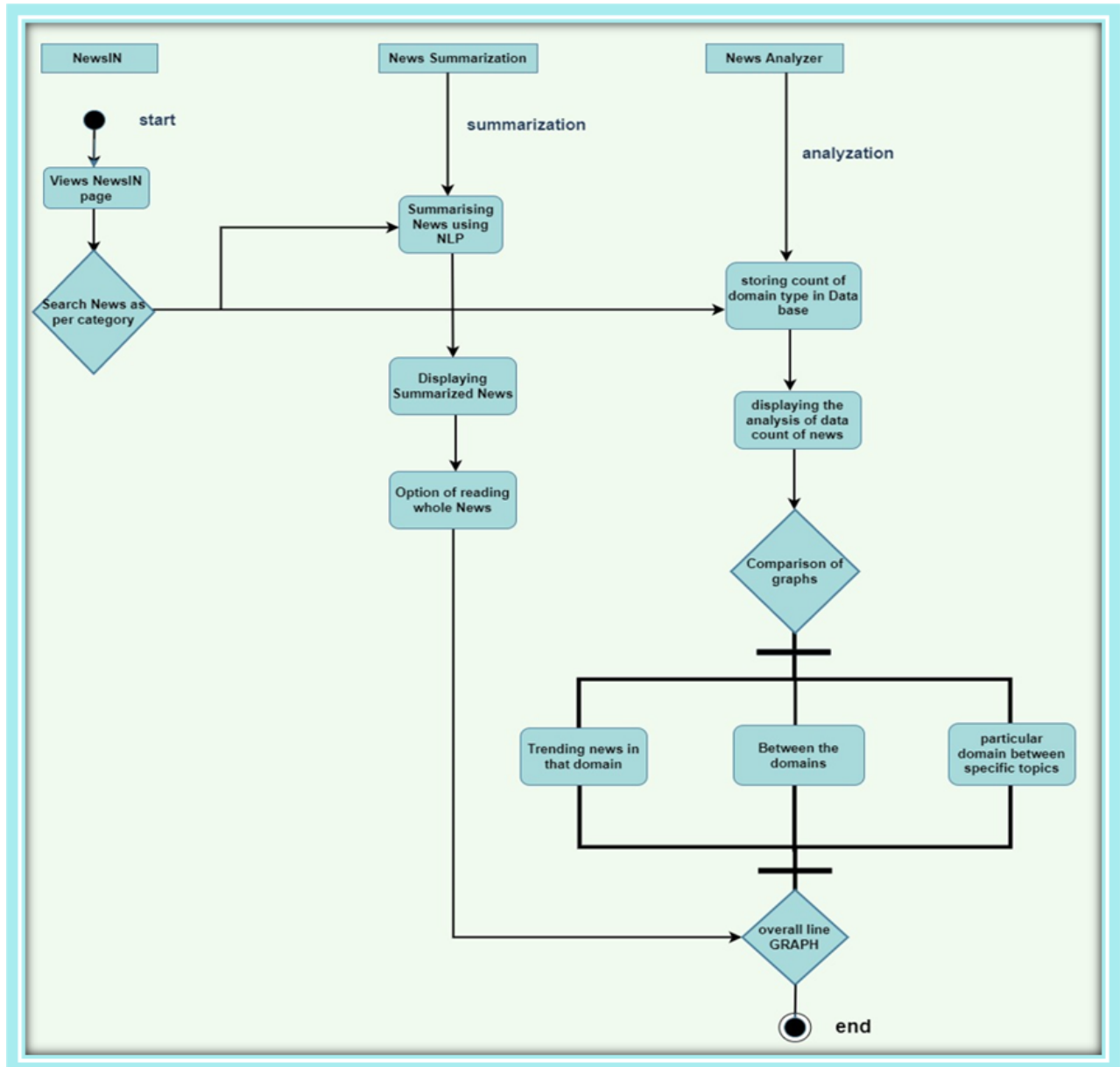


## 5.2.4 Sequence Diagram :-



## 5.2.5 Activity Diagram :-

Activity diagram is another important diagram in UML to describe dynamic aspects of the system. Activity diagram is basically a flowchart to represent the flow from one activity to another activity.



# Chapter 6

## Modules:

### 6.1 Summarization Module:

1. Extracting news from NDTV News website.
2. Fetching news using RSS Feed.
3. Using request url to view the xml file.
4. Scrapping the data using beautiful soup.
5. Performing summarization with the help of punkt package.

### 6.2 Analyzing Module:

1. After summarization, we perform an analysis of the news.
2. For performing analysis, we store the news count of all the domains.
3. The news count of domains is stored in the database with the help of keywords.
4. Once the count got stored in the database, we converted it into a CSV file for performing analysis.
5. The analysis part will be implemented by using Plotly.
6. By using plotly we generated pie charts for all the domains and also a line graph that will represent the weekly analysis of news.

# Chapter 7

## Project Implementation

### 7.1 Algorithm

Step 1: Import all the Modules.

Step 2: Initialize a variable with 'Analyzer.xlsx'.

Step 3: Initialize function search topic.

Step 4: Open site.

Step 5: Read

Step 6: Initialize news\_list=sp\_page.findall('item') #finding news

Step 7: return news\_list

Step 8: Repeat from steps 3 for various functions of various categories.

Step 9: Initialize function news\_poster.

Step 10: Open poster link

Step 11: Read data

Step 12: Open Image

Step13: Close function

Step 14: Initialize run() function

Step 15: Select category=[' --select-- ', 'Trending News', 'Favorite Topics', 'Search Topic']

    If category[0] warning("Please select type !")

    elif category[1] show trending news

    elif category[2] select favorite topic  
        show favorite topic

    Elif category[3] enter your topic  
        show entered topic


Step 16: End

## 7.2 Methodology :-

1. Extracting news from NDTV News website.
2. Fetching news using RSS Feed.
3. Segregating the news based on different domains i.e., sports, technology, etc.
4. Summarizing the news using NLP.
5. After summarization we perform analysis on the news.
6. For performing analysis we store the news count of all the domains.
7. The news count of domains is stored in the database with the help of keywords.
8. Once the count got stored in the database, we converted it into a csv file for performing analysis.
9. Analysis part was implemented by using Plotly.
10. By using plotly we generated pie charts for all the domains and also a line graph that will represent the weekly analysis of news.

## 7.3 Code

### App.py

```
from ctypes import alignment
import streamlit as st
from PIL import Image
from bs4 import BeautifulSoup as soup
from urllib.request import urlopen
from newspaper import Article
import io
import nltk
import pandas as pd
import plotly.express as px
nltk.download('punkt')
st.set_page_config(page_title='NewsIn: A News  Summarizer and Analyzer',
page_icon='./Meta/newspaper.ico')

hide_st_style = """
            <style>
            #MainMenu {visibility: hidden;}
            footer {visibility: hidden;}
            header {visibility: hidden;}
            </style>
            """

st.markdown(hide_st_style, unsafe_allow_html=True)

excel_file='analyzer1.xlsx'
excel_files='analyzer.xlsx'

#sheet_name='DATA'

def fetch_news_search_topic(topic):
    site = 'https://news.google.com/rss/search?q={}'.format(topic)
    op = urlopen(site) # Open that site
    rd = op.read() # read data from site
    op.close() # close the object
    sp_page = soup(rd, 'xml') # scrapping data from site
    news_list = sp_page.find_all('item') # finding news
```

```

    return news_list

def fetch_top_news():
    site = 'https://feeds.feedburner.com/ndtvnews-trending-news'
    op = urlopen(site) # Open that site
    rd = op.read() # read data from site
    op.close() # close the object
    sp_page = soup(rd, 'xml') # scrapping data from site
    news_list = sp_page.find_all('item') # finding news
    return news_list

def world_news():
    site = 'https://feeds.feedburner.com/ndtvnews-world-news'
    op = urlopen(site) # Open that site
    rd = op.read() # read data from site
    op.close() # close the object
    sp_page = soup(rd, 'xml') # scrapping data from site
    news_list = sp_page.find_all('item') # finding news
    df=pd.read_excel(excel_file,
                     usecols='U:W',
                     header=0)

    pie_chart=px.pie(df,
                     title='World',
                     values='world_count',
                     names='world_date')

    st.plotly_chart(pie_chart)
    df=pd.read_excel(excel_file,
                     usecols='BH:BM',
                     skiprows=32,
                     nrows=1,
                     header=0)

    pie_chart=px.pie(df,
                     title='World',
                     #values=['Russia','USA','UK','Asia'],
                     values= [152, 62, 47, 60],
                     names=['Russia','USA','UK','Asia'])

    st.plotly_chart(pie_chart)
    return news_list

def nation_news():
    site = 'https://feeds.feedburner.com/ndtvnews-india-news'

```

```

op = urlopen(site) # Open that site
rd = op.read() # read data from site
op.close() # close the object
sp_page = soup(rd, 'xml') # scrapping data from site
news_list = sp_page.find_all('item') # finding news
df=pd.read_excel(excel_file,
                 usecols='I:K',
                 header=0)

pie_chart=px.pie(df,
                 title='Nation',
                 values='nation_count',
                 names='nation_date')

st.plotly_chart(pie_chart)
df=pd.read_excel(excel_file,
                 usecols='AO:AR',
                 skiprows=32,
                 nrows=1,
                 header=0)

pie_chart=px.pie(df,
                 title='Nation',
                 values=[55,133,57,64],
                 names=['covid','government','crime','court'])

st.plotly_chart(pie_chart)
return news_list

def business_news():
    site = 'https://feeds.feedburner.com/ndtvprofit-latest'
    op = urlopen(site) # Open that site
    rd = op.read() # read data from site
    op.close() # close the object
    sp_page = soup(rd, 'xml') # scrapping data from site
    news_list = sp_page.find_all('item') # finding news
    df=pd.read_excel(excel_file,
                    usecols='A:C',
                    header=0)

    pie_chart=px.pie(df,
                    title='Business',
                    values='business_count',
                    names='business_date')

    st.plotly_chart(pie_chart)
    df=pd.read_excel(excel_file,
                    usecols='Z:AD',
                    skiprows=32,

```



```

        nrows=1,
        header=0)

pie_chart=px.pie(df,
                 title='Business',
                 values=[62,61,39,63,61],
                 names=['Crypto','Market','Money','Economy','Industry'])
st.plotly_chart(pie_chart)
return news_list

def technology_news():
    site = 'https://feeds.feedburner.com/gadgets360-latest'
    op = urlopen(site) # Open that site
    rd = op.read() # read data from site
    op.close() # close the object
    sp_page = soup(rd, 'xml') # scrapping data from site
    news_list = sp_page.find_all('item') # finding news
    df=pd.read_excel(excel_file,
                    usecols='Q:S',
                    header=0)

    pie_chart=px.pie(df,
                     title='Technology',
                     values='technology_count',
                     names='technology_date')

    st.plotly_chart(pie_chart)
    df=pd.read_excel(excel_file,
                    usecols='BB:BE',
                    skiprows=32,
                    nrows=1,
                    header=0)

    pie_chart=px.pie(df,
                     title='Technology',
                     values=[76,132,126,72],
                     names=['5G','Mobile','New_gadgets','Crypto'])

    st.plotly_chart(pie_chart)
    return news_list

def entertainment_news():
    site = 'https://feeds.feedburner.com/ndtvmovies-latest'
    op = urlopen(site) # Open that site
    rd = op.read() # read data from site
    op.close() # close the object
    sp_page = soup(rd, 'xml') # scrapping data from site
    news_list = sp_page.find_all('item') # finding news

```

```

df=pd.read_excel(excel_file,
                  usecols='E:G',
                  header=0)

pie_chart=px.pie(df,
                 title='Entertainment',
                 values='entertainment_count',
                 names='entertainment_date')

st.plotly_chart(pie_chart)

df=pd.read_excel(excel_file,
                  usecols='AG:AL',
                  skiprows=33,
                  nrows=1,
                  header=0)

pie_chart=px.pie(df,
                 title='Entertainment',
                 values=[137,61,50,60],
                 names=['bollywood','hollywood','tv','web_series'])

st.plotly_chart(pie_chart)

return news_list

def sports_news():
    site = 'https://feeds.feedburner.com/ndtvsports-latest'
    op = urlopen(site) # Open that site
    rd = op.read() # read data from site
    op.close() # close the object
    sp_page = soup(rd, 'xml') # scrapping data from site
    news_list = sp_page.find_all('item') # finding news
    df=pd.read_excel(excel_file,
                     usecols='M:O',
                     header=0)

    pie_chart=px.pie(df,
                     title='Sports',
                     values='sports_count',
                     names='sports_date')

    st.plotly_chart(pie_chart)

    df=pd.read_excel(excel_file,
                     usecols='AV:AX',
                     skiprows=32,
                     nrows=1,
                     header=0)

    pie_chart=px.pie(df,
                     title='Sports',
                     values=[209,59,34],

```

```

        names=['cricket','football','tennis'])

    st.plotly_chart(pie_chart)
    return news_list

def fetch_category_news(topic):
    site =
'https://news.google.com/news/rss/headlines/section/topic/{}'.format(topic)
    op = urlopen(site) # Open that site
    rd = op.read() # read data from site
    op.close() # close the object
    sp_page = soup(rd, 'xml') # scrapping data from site
    news_list = sp_page.find_all('item') # finding news
    return news_list

def fetch_news_poster(posters_link):
    try:
        u = urlopen(posters_link)
        raw_data = u.read()
        image = Image.open(io.BytesIO(raw_data))
        st.image(image, use_column_width=True)
    except:
        image = Image.open('./Meta/no_image.jpg')
        st.image(image, use_column_width=True)

def display_news(list_of_news, news_quantity):
    c = 0
    for news in list_of_news:
        c += 1
        st.write('**({}) {}**'.format(c, news.title.text))
        news_data = Article(news.link.text)
        try:
            news_data.download()
            news_data.parse()
            news_data.nlp()
        except Exception as e:
            st.error(e)
        fetch_news_poster(news_data.top_image)
        with st.expander(news.title.text):
            st.markdown(
                '''<h6 style='text-align:
justify;'>{}</h6>'''.format(news_data.summary),

```

```

        unsafe_allow_html=True)
        st.markdown("[Read more at {}...]({})".format(news.source.text,
news.link.text))
        st.success("Published Date: " + news.pubDate.text)
        if c >= news_quantity:
            break

def run():
    st.title("NewsIn: A News 📰 Summarizer and Analyzer")
    image = Image.open('./Meta/newspaper.png')

    col1, col2, col3 = st.columns([3, 5, 3])

    with col1:
        st.write("")

    with col2:
        st.image(image, use_column_width=False)

    df=pd.read_excel(excel_files,
                      usecols = 'C,B,E,H,K,N,Q'
                      )
    fig=px.line(df,
                x='Date',
                y=['business_count', 'entertainment_count', 'nation_count', 'sports_count', 'tec
hnology_count', 'world_count'],
                title='Line Graph')
    st.plotly_chart(fig)

    with col3:
        st.write("")
        category = ['--Select--', 'Trending 🔥 News', 'Favourite 💙 Topics',
'Search 🔍 Topic']
        cat_op = st.selectbox('Select your Category', category)
        if cat_op == category[0]:
            st.warning('Please select Type!!')

        elif cat_op == category[1]:
            st.subheader("✅ Here is the Trending 🔥 news for you")
            no_of_news = st.slider('Number of News:', min_value=5, max_value=15,
step=1)
            news_list = fetch_top_news()

```

```

display_news(news_list, no_of_news)

elif cat_op == category[2]:
    av_topics = ['Choose Topic', 'WORLD', 'NATION', 'BUSINESS',
'TECHNOLOGY', 'ENTERTAINMENT', 'SPORTS']
    st.subheader("Choose your favourite Topic")
    chosen_topic = st.selectbox("Choose your favourite Topic",
av_topics)
    if chosen_topic == av_topics[0]:
        st.warning("Please Choose the Topic")

    elif chosen_topic == av_topics[1]:
        no_of_news = st.slider('Number of News:', min_value=5,
max_value=15, step=1)
        news_list = world_news()
        display_news(news_list, no_of_news)

    elif chosen_topic == av_topics[2]:
        no_of_news = st.slider('Number of News:', min_value=5,
max_value=15, step=1)
        news_list = nation_news()
        display_news(news_list, no_of_news)

    elif chosen_topic == av_topics[3]:
        no_of_news = st.slider('Number of News:', min_value=5,
max_value=15, step=1)
        news_list = business_news()
        display_news(news_list, no_of_news)

    elif chosen_topic == av_topics[4]:
        no_of_news = st.slider('Number of News:', min_value=5,
max_value=15, step=1)
        news_list = technology_news()
        display_news(news_list, no_of_news)

    elif chosen_topic == av_topics[5]:
        no_of_news = st.slider('Number of News:', min_value=5,
max_value=15, step=1)
        news_list = entertainment_news()
        display_news(news_list, no_of_news)

```

```

        elif chosen_topic == av_topics[6]:
            no_of_news = st.slider('Number of News:', min_value=5,
max_value=15, step=1)
            news_list = sports_news()
            display_news(news_list, no_of_news)

        else:
            no_of_news = st.slider('Number of News:', min_value=5,
max_value=15, step=1)
            news_list = fetch_category_news(chosen_topic)
            if news_list:
                st.subheader("✅ Here are the some {} News for
you".format(chosen_topic))
                display_news(news_list, no_of_news)
            else:
                st.error("No News found for {}".format(chosen_topic))

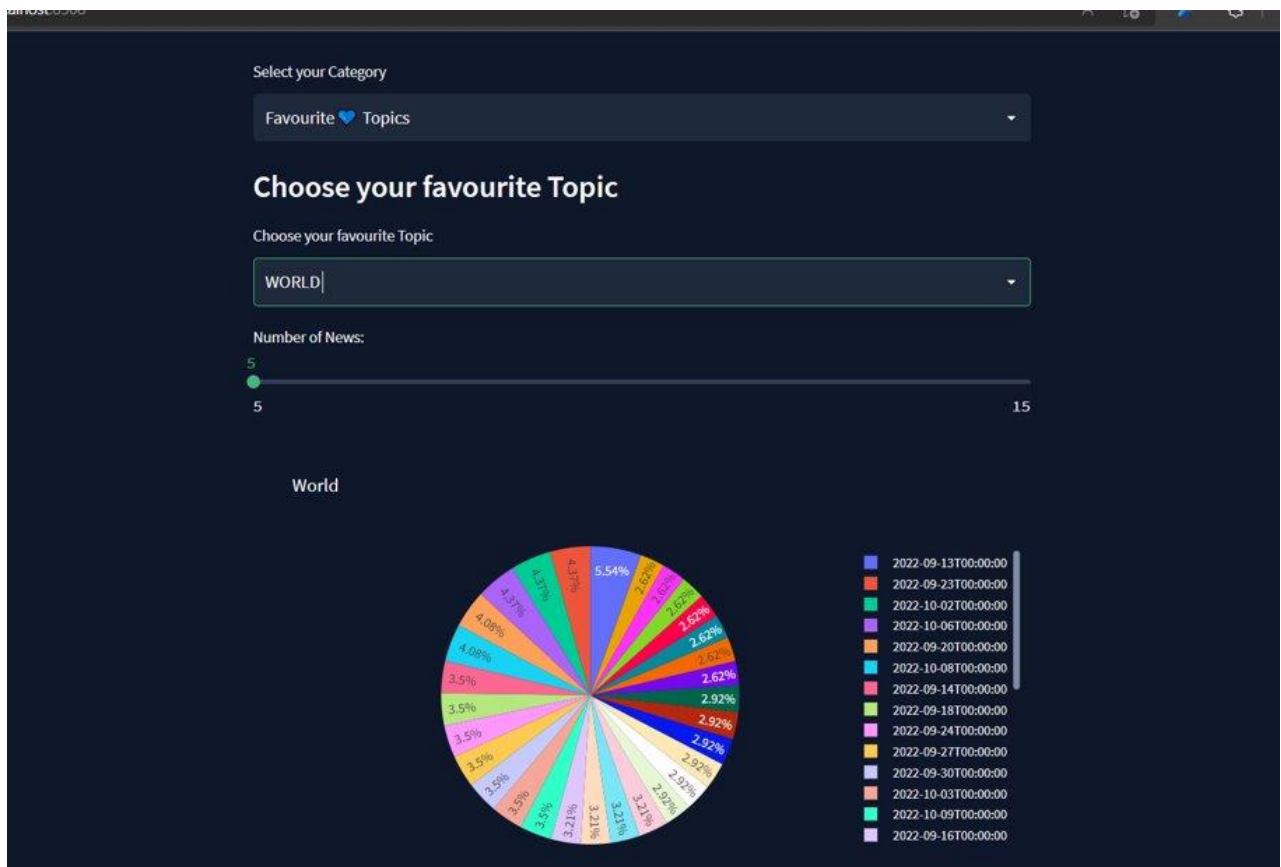
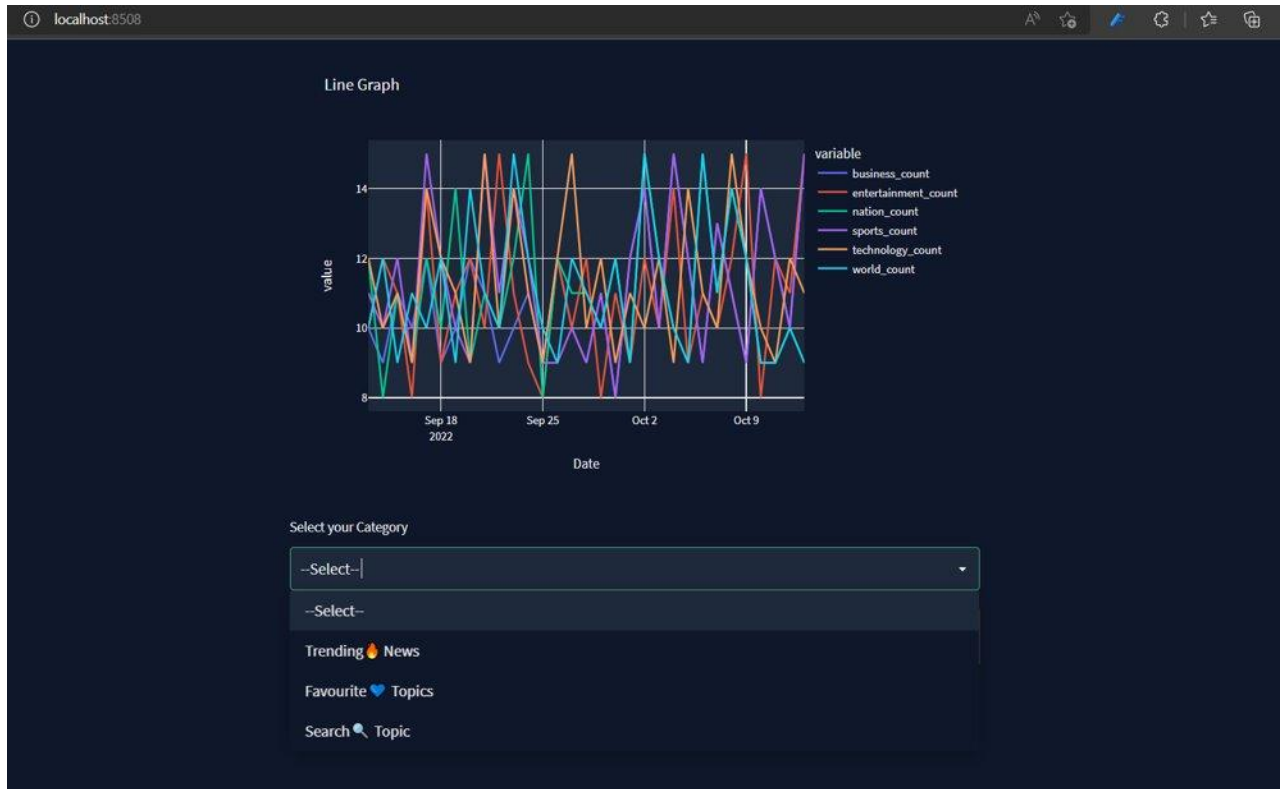
    elif cat_op == category[3]:
        user_topic = st.text_input("Enter your Topic🔍")
        no_of_news = st.slider('Number of News:', min_value=5, max_value=15,
step=1)

        if st.button("Search") and user_topic != '':
            user_topic_pr = user_topic.replace(' ', '')
            news_list = fetch_news_search_topic(topic=user_topic_pr)
            if news_list:
                st.subheader("✅ Here are the some {} News for
you".format(user_topic.capitalize()))
                display_news(news_list, no_of_news)
            else:
                st.error("No News found for {}".format(user_topic))
        else:
            st.warning("Please write Topic Name to Search🔍")

run()

```

## 7.4 Output Screenshots:





World



(1) Russia Says Its Goals In Ukraine May Be Achieved Through Talks: Report



Russia Says Its Goals In Ukraine May Be Achieved Through Talks: Report

The direction has not changed, the special military operation continues, Russia said. (File)The Kremlin was quoted as saying on Thursday that the goals of its "special military operation" in Ukraine are unchanged, but that they may be achieved through negotiations. "The direction has not changed, the special military operation continues, it continues in order for us to achieve our goals," Peskov was quoted as saying. While Russia has said before that it is prepared to negotiate, the repeated references this week to the possibility of dialogue are striking. Foreign Minister Sergei Lavrov said on Tuesday that Moscow was open to talks with the West, but the United States dismissed the statement as "posturing".

[Read more at Reuters...](#)

Published Date: Thu, 13 Oct 2022 20:08:22 +0530

(2) Pair Of Levi's Jeans From 1880s Sells For \$76,000 At An Auction In US






## Russia Says Its Goals In Ukraine May Be Achieved Through Talks: Report

The comments by Kremlin spokesman Dmitry Peskov to Russian newspaper Izvestia were the latest in a series of statements this week stressing Moscow is open to talks - a change of tone that follows a series of humiliating defeats.

World | Reuters | Updated: October 13, 2022 8:16 pm IST

### TRENDING

 "Is It Too Much To Ask In Democracy?" How 2 Judges Differed On Hijab Ban

 IND vs WA: India Lose To Western Australia In Second Practice Game

 Kerala Killer's Facebook Posts Just Days After Women's Torture, Murder

 Entertainment  
Richa-Ali Reception: What Guests Wore  
21 Slides



The direction has not changed, the special military operation continues, Russia said. (File)



**London:** The Kremlin was quoted as saying on Thursday that the goals of its

Days After Women's  
Torture, Murder

Entertainment  
Richa-Ali Reception:  
What Guests Wore  
21 Slides

Tech  
OnePlus 10T 5G: All  
You Need to Know  
12 Slides

Beauty  
Pooja Hegde's  
Favourite Hairstyle  
10 Slides

SHOPPING



The direction has not changed, the special military operation continues, Russia said. (File)



**London:** The Kremlin was quoted as saying on Thursday that the goals of its "special military operation" in Ukraine are unchanged, but that they may be achieved through negotiations.

The comments by Kremlin spokesman Dmitry Peskov to Russian newspaper Izvestia were the latest in a series of statements this week stressing Moscow is open to talks - a change of tone that follows a series of humiliating defeats for Russian forces as the war in Ukraine nears the end of its eighth month.

"The direction has not changed, the special military operation continues, it continues in order for us to achieve our goals," Peskov was quoted as saying.

"However we have repeatedly reiterated that we remain open to negotiations to achieve our objectives."

Flight Offers

# Chapter 8

## Result

In this project, we were able to accurately summarize the news and show the data on the User Interface. News was shown on the application according to the particular domain that was chosen, and many additional features were also available to assist the user acquire accurate information. The data has been shown in diagrammatic fashion using a variety of graphics.[1] The user will be able to analyze the global condition with the aid of these graphs. Utilizing the News API, we have gathered the most recent news articles from a variety of news websites, organized them, and then displayed them all in one location. The user gains a general idea of the story's subject matter and current global implications through the news summary. After analyzing, we discovered that the opening phrase consistently received a good grade since it had nouns that were repeated throughout the piece[2].

The project helped us understand the trends in the news on a daily basis i.e. which news is more popular than others. What keywords were used which made that news popular. What is the difference between the popularity of the news? The UI of this page was created using a python module named streamlit. It is a platform which helps display the page in a more dynamic way[4].

Most text analytics works have been performed using R programming languages[3]. But we implemented this using Python language and tried to increase the efficiency and the accuracy of the details we are providing.

## References

[1]Rananavare, Laxmi & Reddy, P.. (2018). Automatic News Article Summarization. International Journal of Computer Sciences and Engineering. 6. 230-237. doi: 10.26438/ijcse/v6i2.230237.

[HTTPS://WWW.RESEARCHGATE.NET/PUBLICATION/325775102\\_AUTOMATIC\\_NEWS\\_ARTICLE\\_SUMMARIZATION](https://www.researchgate.net/publication/325775102_AUTOMATIC_NEWS_ARTICLE_SUMMARIZATION)

[2]P. Sethi, S. Sonawane, S. Khanwalker and R. B. Keskar, "Automatic text summarization of news articles," 2017 International Conference on Big Data, IoT and Data Science (BIG DATA), 2017, pp. 23-29, doi: 10.1109/BIGDATA.2017.8336568.

[HTTPS://IEEEEXPLORE.IEEE.ORG/DOCUMENT/8336568](https://ieeexplore.ieee.org/document/8336568)

[3]Nahar, J., Kline, D, Layman, L., Modares Nezhad, M. (2019) Daily Text Analytics of News and Social Media with Power BI. Annals of the Master of Science in Computer Science and Information Systems at UNC Wilmington, 13(2) paper 2. <http://csbapp.uncw.edu/data/mscsis/full.aspx>

[HTTPS://UNCW.EDU/CSB/MSCSIS/COMPLETE/PDF/NAHAR\\_FALL2019.PDF](https://uncw.edu/csb/mscsis/complete/pdf/NAHAR_FALL2019.PDF)

[4] Roth, Robert. (2017). User Interface and User Experience (UI/UX) Design. Geographic Information Science & Technology Body of Knowledge. 2017. doi:10.22224/gistbok/2017.2.5.

[HTTPS://WWW.RESEARCHGATE.NET/PUBLICATION/317660257\\_USER\\_INTERFACE\\_AND\\_USER\\_EXPERIENCE\\_UIUX\\_DESIGN\\_NEWS\\_SUMMARIZATION\\_AND\\_ANALYZER.PPTX](https://www.researchgate.net/publication/317660257_USER_INTERFACE_AND_USER_EXPERIENCE_UIUX_DESIGN_NEWS_SUMMARIZATION_AND_ANALYZER.PPTX)

[5] Developer Documentation for Web Development

[HTTPS://DEVDOCS.IO](https://devdocs.io)



## Acknowledgement

We have great pleasure in presenting the mini project report on “**NEWSIN-A News Summarizer and Analyzer**”. We take this opportunity to express our sincere thanks towards our guide **Prof. Suchita Dange**, Department of Computer Engineering, APSIT Thane for providing the technical guidelines and suggestions regarding line of work. We would like to express our gratitude towards his constant encouragement, support and guidance through the development of the project.

We thank **Prof. Sachin Malave, Head of Department**, Computer Engineering, APSIT for his encouragement during the progress meeting and providing guidelines to write this report.

We also thank the entire staff of APSIT for their invaluable help rendered during the course of this work. We wish to express our deep gratitude towards all our colleagues of APSIT for their encouragement.

Sr.No	Student Name	Student Id
1	Anushree Salunke	20102179
2	Eisha Saini	20102025
3	Pooja Tumma	20102126
4	Sanskriti Shinde	21202018