INFOCLIPS: NLP-based summarization of YouTube Videos and Text

A MINI PROJECT REPORT

Submitted in partial fulfilment of the requirements for the award of the degree of

Bachelor of Technology

in

COMPUTER SCIENCE AND ENGINEERING

by **BATCH-7B**

K.S.V Ravi Sastry	**N.V.S.R Srujan**
(20331A0588)	(20331A05C6)
K. Sarath Kumar	**I.Deena**
(20331A0584)	(20331A0573)

Under the supervision of

Mrs. B. Sujatha

(Assistant Professor)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

MVGR COLLEGE OF ENGINEERING (A)

VIZIANAGARAM-535005, ANDHRA PRADESH (INDIA)

(Accredited by NBA, NAAC, and permanently Affiliated to Jawaharlal Nehru Technological University Kakinada)

2022 - 2023

CERTIFICATE

This is to certify that the project report entitled " **INFOCLIPS: NLP-based summarization of YouTube videos and text**" being submitted by

K.S.V RAVI SASTRY, N.V.S.R SRUJAN, K.SARATH KUMAR, I.DEENA bearing registered numbers **20331A0588, 20331A05C6, 20331A0584, 20331A0573** respectively, in partial fulfilment for the award of the degree of **"Bachelor of Technology" in Computer Science and Engineering** is a record of bonafide work done by them under my supervision during the academic year 2022- 2023.

Mrs. B. Sujatha	**Dr P Ravi Kiran Varma**
Assistant Professor	**Head of the Department**
Dept. of CSE	**Dept. of CSE**

DECLARATION

We hereby declare that the work done on the dissertation entitled "**INFOCLIPS: NLP-based summarization of YouTube Videos and Text**" has been carried out by us and submitted in partial fulfilment for the award of credits in Bachelor of Technology in Computer Science and Engineering of MVGR College of Engineering (Autonomous) and affiliated to Jawaharlal Nehru Technological University (Kakinada). The various contents incorporated in the dissertation have not been submitted for the award of any degree of any other institution or university.

ACKNOWLEDGEMENTS

We respect and thank our project mentor **Mrs. B. Sujatha**, who took keen interest on our project work and guided us all along, till the completion of our project work by providing all the necessary information for developing a good system. We are extremely thankful to him for providing such a nice support and guidance.

We also thank **Dr. K. V. L. Raju**, Principal, and **Dr. P. Ravi Kiran Varma**, Head of the Department, for extending their utmost support and cooperation in providing all the provisions for the successful completion of the project. We sincerely thank all the members of the staff in the Department of Computer Science & Engineering for their sustained help in our pursuits. We thank all those who contributed directly or indirectly in successfully carrying out this work.

N.S.S. Harshith (20331A05C8)

K. Devaraju (20331A0592)

K. Yeswanth Teja (20331A0581)

G. Vasu Devaraju (20331A0567)

ABSTRACT

INFOCLIPS: NLP-based summarization of YouTube videos and text is a web-based application that offers a time-efficient and effective way of summarizing lengthy YouTube videos and text. Our application uses advanced natural language processing (NLP) techniques to extract key information from the video or text and generate a concise summary.

Our web application features sentimental analysis for YouTube videos, reflecting the overall tone of the video, which is particularly helpful for users who want to quickly determine whether the video is positive, negative, or neutral. Additionally, users can input a keyword, and our web application will automatically search for the keyword and skip the video to the relevant section. This makes it more convenient to navigate through the material and saves time for users who need specific information.

INFOCLIPS is a valuable tool for those who want to optimize their learning and productivity by efficiently summarizing large volumes of information. With our application, users can revisit key ideas without having to re-watch or re-read the entire material, making it an essential tool for students, professionals, and anyone seeking to streamline their workflow.

In summary, INFOCLIPS offers a powerful solution to the challenge of navigating through lengthy YouTube videos and text. With its advanced NLP techniques, sentimental analysis, and keyword search capabilities, it offers an efficient and effective way to extract key information and summarize it in a concise and user-friendly format.

CONTENTS

List of Abbreviations

AI	-	Artificial Intelligence
ML	-	Machine Learning
CNN	-	Convolutional Neural Networks
DNN	-	Deep Neural Networks
NLP	-	Natural Language Processing
LDA	-	Latent Dirichlet Allocation
LSA	-	Latent Semantic Analysis

List of Figures

CHAPTER 1
INTRODUCTION

InfoClips: A Machine learning-based video and PDF summary generator project aims to develop a tool that can automatically generate a summary of a YouTube video's transcript. The tool would be helpful for content creators who want to provide a quick summary of their video's content or for viewers who don't have the time to watch the entire video but still want to know what it's about and pdf summary generator is another tool that gives the whole summary of the pdf precisely.

The project would involve developing a machine learning model that can accurately summarize the transcript of a YouTube video. This would require training the model on a large dataset of video transcripts and their corresponding summaries. Once the model is trained, it could be integrated into a web application or browser extension that users could use to generate summaries of any YouTube video.

Text Summarization is an important Natural Language Processing (NLP) task. Text summarization involves condensing a larger text into smaller sizes by preserving the meaning to convey the core message of the text. Basically, it wraps up long text into main points so that we can read any long text in short form. There are two main approaches to summarizing texts. They are – 1) Extraction- based Summarization 2) Abstraction-based Summarization. In this project, we are used the Extraction-based summarization.

The goal is to provide a brief overview of the content, saving time and improving efficiency for the user. And the main objective of this is to make it user-friendly by providing the content available to the user in their own comfortable way. It can be applicable in various applications like online learning platforms, news broadcasting etc.

Overall, the InfoClips: A Machine learning-based video and PDF summary generator project has the potential to be a useful tool for both content creators and viewers, making it easier to access and digest video content on the platform.

OBJECTIVES

- Develop a web-based platform that creates an accurate and reliable summarization system that cangenerate concise and informative summaries for YouTube videos and Texts.
- To develop machine learning algorithms and natural language processing (NLP) techniques that can effectively identify and extract the most important information from video and text content.
- Make the website user-friendly and easy to navigate, allowing users to easily input their YouTubeURLs and Text and access the summary of the video and text content.
- To enable users to quickly and efficiently review and analyze large amounts of content without having to spend time watching or reading through the entire video or document.
- Optimize the website for scalability, ensuring that it can handle a large volume of requests fromusers without experiencing performance issues.

CHAPTER 2

LITERATURE SURVEY

1. "CartoonGAN: Generative Adversarial Networks for Photo Cartoonization" proposed a GAN-based technique for photo cartoonization, achieving state-of-the-art results on benchmark datasets.
2. "Image Cartoonization Using Convolutional Neural Networks" introduced a CNN-based approach for cartoonizing images, achieving promising results and computational efficiency.
3. "Fast Image-to-Style Transfer: Exploring Color Spaces" presented a technique for fast image style transfer using a novel color space transformation.
4. "Real-Time Neural Style Transfer for Videos" introduced a technique for transferring the artistic style of an image to a video in real-time using deep learning.
5. "Towards Real-Time Cartoon Animation of Facial Expressions" presented a real-time cartoon animation system that used a GAN to generate expressive cartoon faces.

Summary of Literature Review:

These existing works demonstrate the effectiveness of deep learning techniques for image cartoonization and image style transfer. A chatbot that converts images into cartoons could be developed using a combination of these techniques. The chatbot could use a GAN or a CNN to generate the cartoon image and a style transfer technique to apply a specific style to the cartoon image. The chatbot could also be designed to generate expressive cartoon avatars from images of faces in real-time using the facial landmark detection algorithm and the GAN-based cartoon animation technique.

References of Research Papers:

- https://openaccess.thecvf.com/content_cvpr_2018/papers/Chen_CartoonGAN_Generative_Adversarial_CVPR_2018_paper.pdf
- https://www.researchgate.net/publication/362056064_CartoonStyle_Image_Rendering_Transfer_Based_on_Neural_Networks
- https://arxiv.org/pdf/2204.13339
- https://openaccess.thecvf.com/content_cvpr_2017/papers/Huang_RealTime_Neural_Style_CVPR_2017_paper.pdf

- http://ivizlab.sfu.ca/arya/Papers/IEEE/Proceedings/C%20A%20-%2098/Real-time%20Facial%20Animation.pdf

.

CHAPTER 3

TOOLS AND TECHNOLOGIES

This project requires web technologies for building user-interface and establish the backend server connection. This project involves human-language interpretation which involves a wide range of mechanisms involved to build the right summary for the user. Summarizing the requirements and necessary technologies, we can classify them into two phases – interface development, summary development.

I. **Interface Development:**
   i) HTML (Hyper Text Markup Language)
   ii) CSS (Cascading Style Sheets)
   iii) JAVASCRIPT
   iv) AJAX
   v) PYTHON
   vi) Django

II. **Summary Development**
   i) NLTK (Natural Language Tool Kit)
   ii) GENSIM
   iii) NUMPY
   iv) YOUTUBE_TRANSCRIPT_API
   v) JSON (Java Script Object Notation)

The usage of each module is described below:

1) **HTML (Hyper Text Markup Language):** HTML is the standard markup language used to create web pages. It provides a standardized way to structure and present content on the web. HTML elements are the building blocks of a web page and include headings, paragraphs, images, videos, links, and more. HTML is a static language, meaning it cannot change without a page refresh.

2) **CSS (Cascading Style Sheets):** CSS is used to add style and formatting to HTML documents. It allows web designers to control the look and feel of a web page, including layout, typography, colors, and more. CSS uses selectors to target specific HTML elements

and apply styles to them. CSS is a dynamic language, meaning it can be changed without a page refresh.

3) **JavaScript:** JavaScript is a high-level, dynamic programming language that is used to add interactivity to web pages. It is commonly used for client-side scripting, where scripts run in the user's web browser, and can be used for tasks such as form validation, dynamic content updates, and more. JavaScript can also be used for server-side scripting, using technologies such as Node.js.

4) **AJAX (Asynchronous JavaScript and XML):** AJAX is a web development technique that allows web pages to be updated asynchronously without reloading the entire page. It uses JavaScript to make requests to the server and update the content of a web page dynamically. AJAX can be used for tasks such as form submission, search suggestions, and more.

5) **Python:** Python is a high-level, general-purpose programming language that is easy to learn and use. It is widely used in web development, scientific computing, data analysis, and more. Python is known for its readability and simplicity, making it a popular choice for beginners and experts alike. It has a large and active community of developers, making it easy to find help and support.

6) **Django:** Django is a high-level Python web framework that follows the Model-View-Controller (MVC) architectural pattern. It is designed to make web development fast and easy, with built-in tools for URL routing, template rendering, database modeling, and more. Django is known for its security features and scalability, making it a popular choice for large-scale web applications.

7) **NLTK (Natural Language Tool Kit):** NLTK is a Python library for natural language processing tasks such as tokenization, stemming, tagging, and sentiment analysis. It provides a set of tools and algorithms for working with human language data, and is widely used in research and industry for tasks such as text classification, machine translation, and more.

8) **Gensim:** Gensim is a Python library for topic modeling and document similarity analysis. It provides an easy-to-use interface for working with large text datasets and includes algorithms such as Latent Dirichlet Allocation (LDA) and Latent Semantic Analysis (LSA).

Gensim is widely used in research and industry for tasks such as document clustering, topic modeling, and more.

9) **NumPy:** NumPy is a Python library for scientific computing and data analysis. It provides an efficient array manipulation interface and supports a wide range of mathematical operations, making it useful for tasks such as linear algebra, Fourier analysis, and more. NumPy is widely used in research and industry for tasks such as data analysis, machine learning, and more.

10) **YouTube Transcript API:** The YouTube Transcript API allows developers to retrieve the transcript of a YouTube video. It can be used for various applications such as content analysis, language learning, and more. The API returns a JSON object containing the transcript, which can be processed using various programming languages such as Python, JavaScript, and more.

11) **JSON (Java Script Object Notation):** JSON is a lightweight data interchange format that is easy for humans to read and write and easy for machines to parse and generate. It is based on a subset of JavaScript and is often used to transmit data between a server and a web application. JSON supports a variety of data types, including strings, numbers, booleans, arrays, and objects, making it a flexible format for data exchange. JSON has become a popular alternative to XML for data interchange due to its simplicity and ease of use.

12) **Recurrent Neural Networks (RNNs):** RNNs are another type of deep learning neural network that are commonly used for text summarization tasks.

13) **Latent Semantic Analysis (LSA):** LSA is a mathematical method that can be used to analyze and summarize text data. it can identify important topics in a document and extract key phrases to generate a summary.

CHAPTER 4

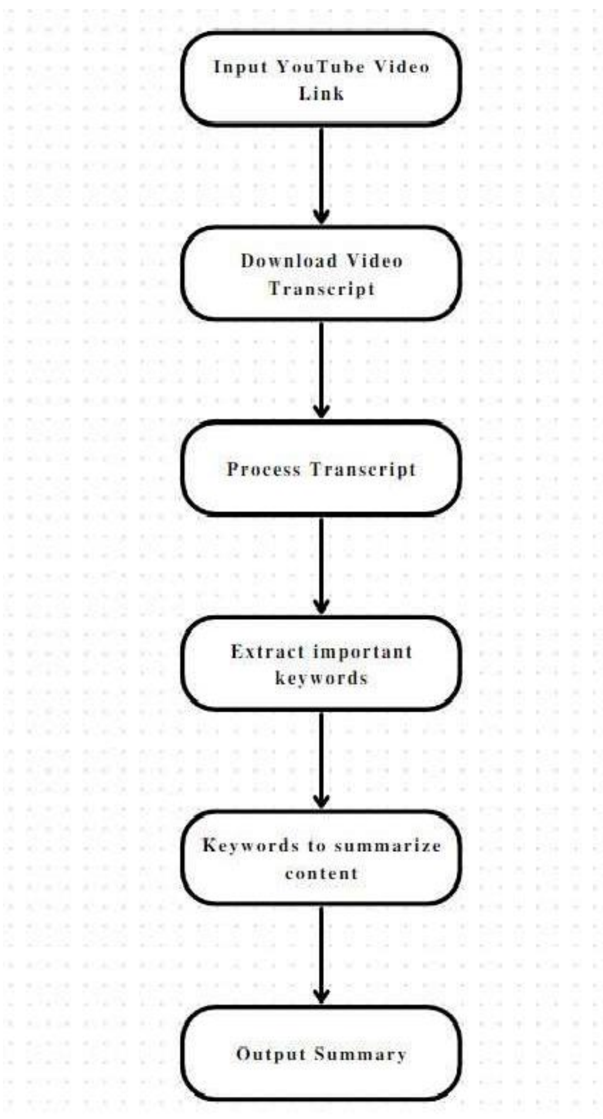
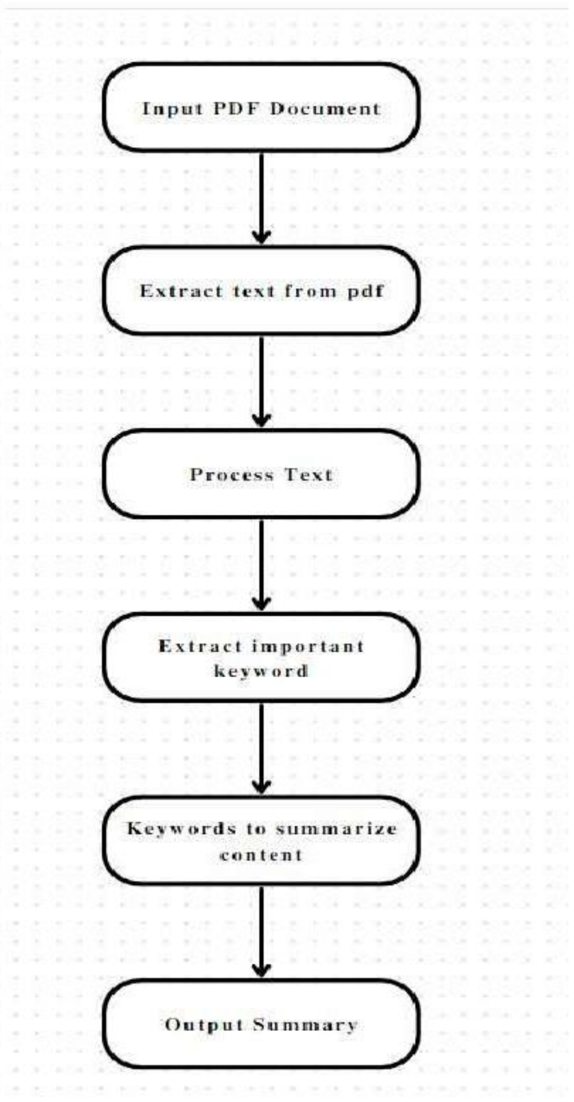
DESIGN AND IMPLEMENTATION

1. **SUMMARY**:
    - Used youtube_transcript_api library to get the transcript of the video.
    - Save the transcript as a JSON file.
    - Used regular expressions to search for specific keywords or timestamps in the transcript.
    - Used library NLTK to generate an abstractive summary of the video.
2. **Website**:
    - Created a Django template that displays a form with a text input for the YouTube URL and a submit button.
    - When the form is submitted, used a Django view to get the URL from the form data and send it to the backend.
    - Other sections are provided to promote the website and make it ease for the user.

So generic implementation is as follows:

1. **Receive user input**: User enters a YouTube URL into the text input on the homepage and clicks submit.
    a. When the user submits the form, the URL is sent to a Django view for processing.
2. **Verify Url**: The Django view receives the YouTube URL and verifies the url and sends it to the backend.
    a. The view sends a POST request to the backend API with the URL as the payload.
3. **Transcript of the Video**: The youtube_transcript_api library is used to retrieve the transcript of the video.
4. **Saving as JSON file**: The generated transcript is saved in the Json file.
5. **Receive user input** : The user enters a search term into the search box.
    a. When the user submits the form, the search term is sent to a Django view for processing.
6. **Searching for the key**: The key is searched in the Json file using regular expressions and returns the timestamp.

7. **Clicking the timestamp**: when the user clicks on the time stamp the video player seeks to that timestamp.
8. **Provide Summary**: Once the user clicks the summary button it displays the summary to the user.
9. **Provide Sentiment Analysis**: Once the user clicks the sentiment Analysis button it displays whether the video is positive, negative, or neutral.


Input YouTube Video Link
Download Video Transcript
Process Transcript
Extract important keywords
Keywords to summarize content
Output Summary


FLOWCHART

ALGORITHM

Input: Long-form content (e.g., research paper or article)

1.Preprocess the text and images in the content by removing stop words, punctuation, and other noise

2.Use natural language processing and computer vision techniques to analyze the content.

3.Identify the most important information and key points from the content, using techniques such as text summarization and image analysis.

4.If the user has selected a video summary: a. Use animation and visuals to create an engaging summary of the content.

5.If the user has selected a PDF summary: a. Format the summary to be concise and easy to read.

6.Output the summary for the user to review and provide feedback.

7.If necessary, refine the summary based on user feedback and repeat the process. Output the final summary for the user to use and share.

INPUTS&OUTPUTS

Inputs:

1.Youtube video url

2.length of summary(in words, sentences, or paragraphs)

3.Pdf document

Outputs:

1.Summary of the Youtube video transcript(text)

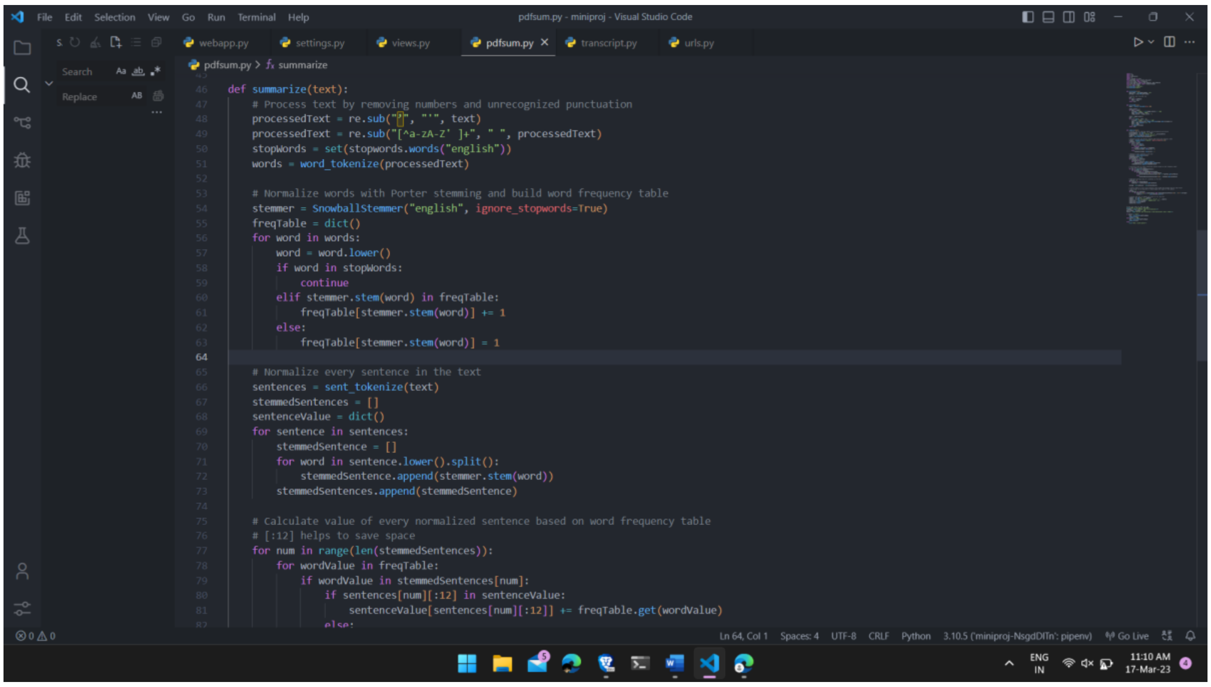
2.Summary of Pdf document

SUMMARY

InfoClips is a project that aims to create a machine learning-based tool that generates videoand PDF summaries of long-form content such as research papers and articles. The tool is designed to help users quickly and easily understand the main points of the content without having to read the entire document.

The algorithm used in InfoClips is based on deep learning techniques such as natural language processing and computer vision. The tool analyzes the text and images in the document and identifies the most important information, which is then used to generate a summary in video or PDF format.

CODING:



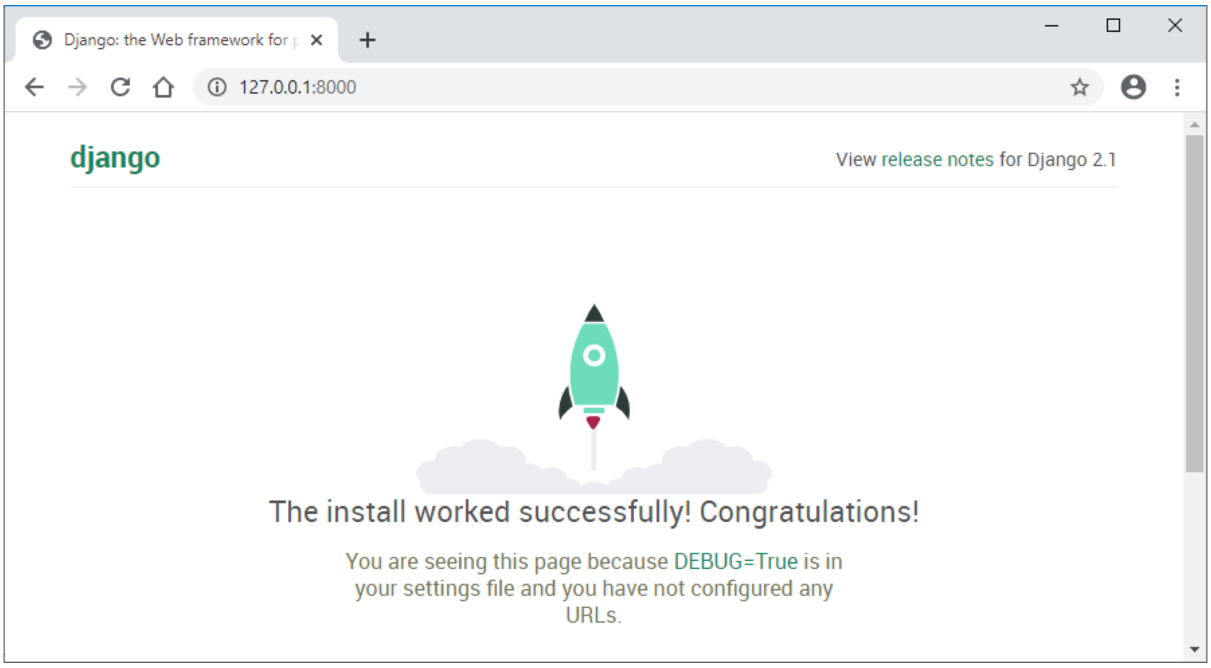
```python
def summarize(text):
    # Process text by removing numbers and unrecognized punctuation
    processedText = re.sub("'", "'", text)
    processedText = re.sub("[^a-zA-Z' ]+", " ", processedText)
    stopWords = set(stopwords.words("english"))
    words = word_tokenize(processedText)

    # Normalize words with Porter stemming and build word frequency table
    stemmer = SnowballStemmer("english", ignore_stopwords=True)
    freqTable = dict()
    for word in words:
        word = word.lower()
        if word in stopWords:
            continue
        elif stemmer.stem(word) in freqTable:
            freqTable[stemmer.stem(word)] += 1
        else:
            freqTable[stemmer.stem(word)] = 1

    # Normalize every sentence in the text
    sentences = sent_tokenize(text)
    stemmedSentences = []
    sentenceValue = dict()
    for sentence in sentences:
        stemmedSentence = []
        for word in sentence.lower().split():
            stemmedSentence.append(stemmer.stem(word))
        stemmedSentences.append(stemmedSentence)

    # Calculate value of every normalized sentence based on word frequency table
    # [:12] helps to save space
    for num in range(len(stemmedSentences)):
        for wordValue in freqTable:
            if wordValue in stemmedSentences[num]:
                if sentences[num][:12] in sentenceValue:
                    sentenceValue[sentences[num][:12]] += freqTable.get(wordValue)
                else:
```

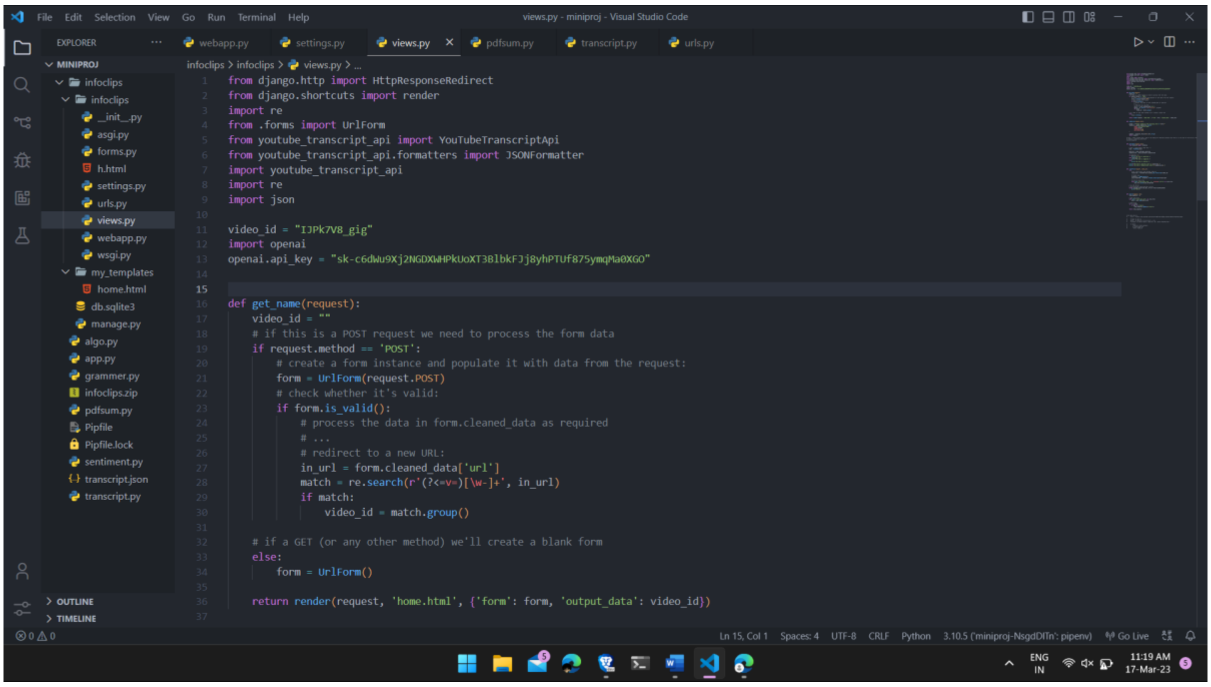
Fig(i): snippet for pdf summarizer


Django: the Web framework for p
127.0.0.1:8000
django
View release notes for Django 2.1

The install worked successfully! Congratulations!

You are seeing this page because DEBUG=True is in your settings file and you have not configured any URLs.

Fig(ii): Django installation



```python
from django.http import HttpResponseRedirect
from django.shortcuts import render
import re
from .forms import UrlForm
from youtube_transcript_api import YouTubeTranscriptApi
from youtube_transcript_api.formatters import JSONFormatter
import youtube_transcript_api
import re
import json

video_id = "IJPk7V8_gig"
import openai
openai.api_key = "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"


def get_name(request):
    video_id = ""
    # if this is a POST request we need to process the form data
    if request.method == 'POST':
        # create a form instance and populate it with data from the request:
        form = UrlForm(request.POST)
        # check whether it's valid:
        if form.is_valid():
            # process the data in form.cleaned_data as required
            # ...
            # redirect to a new URL:
            in_url = form.cleaned_data['url']
            match = re.search(r'(?<=v=)[\w-]+', in_url)
            if match:
                video_id = match.group()

    # if a GET (or any other method) we'll create a blank form
    else:
        form = UrlForm()

    return render(request, 'home.html', {'form': form, 'output_data': video_id})

```

Fig(iii): code snippet for Youtube Summarizer

CONCLUSION:

The output of the tool is a summary that includes key points and highlights from the original document. The video summary includes animations and visuals to help explain the content in an engaging way, while the PDF summary is designed to be a concise and easy-to-read document.

Overall, InfoClips has the potential to be a valuable tool for researchers, students, and anyone who needs to quickly understand the main points of long-form content. It could also be useful for content creators who want to provide a summary of their work for their audience.

REFERENCES: