

YouTube Video Summary

Full Transcript:

hey everyone welcome back to my Channel today I'm going to show you how to build a YouTube video summarizer using streamlit and some powerful NLP tools this tool will automatically generate summaries extract keywords perform topic modeling and even analyze the sentiment of YouTube video transcripts let's dive in first let's talk about the tools we'll be using we'll Leverage The YouTube transcript API to fetch transcripts nltk for natural language processing CET learn for topic modeling hugging faces Transformers for summarization and text blob for sentiment analysis we'll wrap all this functionality in a streamlit web application sounds fun right let's get started we will now see how the application works enter the YouTube URL select length of summary and press summarize button the application will provide you a text summary with top keywords topics discussed in the video and a sentiment analysis let us take a look at the GitHub repository we have two files in the repository which supports the application main.py which includes the application code and requirements.txt file which dictates all required packages in the requirements.txt file there are nine packages mentioned streamlet a framework for building interactive web apps for data science and machine learning projects YouTube transcript API a library to fetch transcripts or subtitles from YouTube videos programmatically nltk natural language toolkit it's used for text tokenization stop words removal and lemmatization which are essential pre-processing steps for text analysis CET learn it's used for keyword extraction and topic modeling Transformers a library used to create a summarization pipeline to generate summaries of the video transcripts text blob a library for sentiment analysis to determine the polarity positive negative and subjectivity of the text numpy a fundamental package for scientific Computing in Python tensorflow it's required by the Transformers library to perform deep learning tasks for text summarization TF kis this resolves compatibility issues between kis 3 and the current tensorflow version now let's take a look at the code this streamlit application is designed to provide comprehensive text analysis for YouTube video transcripts by leveraging various natural language processing NLP techniques the app can summarize transcripts extract

keywords perform topic modeling and conduct sentiment analysis the main aim is to help users gain insights from video content quickly and effectively the app application starts by importing necessary libraries including streamlit for building the web interface YouTube transcript for fetching video transcripts nltk for text processing CET learn for machine learning models Transformers for advanced NLP tasks and text blob for sentiment analysis Additionally the app ensures that essential nltk data sets are downloaded which are required for tokenizing text removing stop words and lemmatizing words text summarization is a core feature of this app the summarized text function utilizes the HuggingFace Transformers library to create a text summarization pipeline this function takes the video transcript as input and generates a concise summary making it easier for users to grasp the main points of lengthy video content without watching the entire video keyword extraction is handled by the extract keywords function this function tokenizes the transcript text removes common stop words and lemmatizes the remaining words to their base forms using count vectorizer from CET learn the function identifies and ranks the most frequent keywords in the text these keywords provide a quick overview of the main topics discussed in the video topic modeling is performed using the topic modeling function which employs latent Dirichlet allocation LDA from scikit learn this function vectorizes the text and identifies the top words associated with each topic by extracting these topics the app can present users with the main themes and subject areas covered in the video offering deeper insights into the content the app also includes a helper function extract video ID which uses regular Expressions to extract the video ID from various YouTube url formats this ensures that users can input different types of YouTube URLs and the app will correctly identify and process the video the main function of the streamlit app creates the web interface users can enter a YouTube video URL and specify the maximum length for the summary using a slider when the summarize button is clicked the app extracts the video ID fetches the transcript and performs summarization keyword extraction topic modeling and sentiment analysis the results including the summary keywords topics and sentiment analysis are then displayed on the streamlit interface to ensure a smooth user experience the app includes error handling for various scenarios such as invalid YouTube URL unavailable transcripts and other exceptions appropriate error messages are displayed to guide

users in correcting their input or understanding the issue overall this streamlit application provides a powerful tool for analyzing YouTube video content by combining summarization keyword extraction topic modeling and sentiment analysis it offers users a comprehensive overview of video transcripts helping them save time and gain insights quickly to create this application you have to log to your streamlit account and press create app button on your dashboard streamlit will then provide you a form which is linked to your GitHub account you will have to select the appropriate name name of repository and file which has the application code make sure to edit and check whether the domain is available or not when you press deploy button streamlit will create the app for you if you find this video helpful and informative please hit the like share and subscribe button please comment if that is necessary for you thanks for watching see you in the next next video

Summary:

hey everyone welcome back to my Channel today I'm going to show you how to build a YouTube video summarizer using streamlit and some powerful NLP tools this tool will automatically generate summaries extract keywords perform topic modeling and even analyze the sentiment of YouTube video transcripts let's dive in first let's talk about the tools we'll be using we'll Leverage The YouTube transcript API to fetch transcripts nltk for natural language processing CET learn for topic modeling hugging faces Transformers for summarization and text blob for sentiment analysis we'll wrap all this functionality in a streamlit web application sounds fun right let's get started we will now see how the application works enter the YouTube URL select length of summary and press summarize button the application will provide you a text summary with top keywords topics discussed in the video and a sentiment analysis let us take a look at the GitHub repository we have two files in the repository which supports the application main.py which includes the application code and requirements.txt file which dictates all required packages in the requirements.txt file there are nine packages mentioned streamlet a framework for building interactive web apps for data science and machine learning projects YouTube transcript API a library to fetch transcripts or subtitles from YouTube videos programmatically nltk natural language toolkit it's used for text tokenization stop words removal and

lemmatization which are essential pre-processing steps for text analysis CET learn it's used for keyword extraction and topic modeling Transformers a library used to create a summarization pipeline to generate summaries of the video transcripts text blob a library for sentiment analysis to determine the polarity positive negative and subjectivity of the text numpy a fundamental package for scientific Computing in Python tensorflow it's required by the Transformers library to perform deep learning tasks for text summarization TF kis this resolves compatibility issues between kis 3 and the current tensorflow version now let's take a look at the code this streamlit application is designed to provide comprehensive text analysis for YouTube video transcripts by leveraging various natural language processing NLP techniques the app can summarize transcripts extract keywords perform topic modeling and conduct sentiment analysis the main aim is to help users gain insights from video content quickly and effectively the app application starts by importing necessary libraries including streamlit for building the web interface YouTube transcript for fetching video transcripts nltk for text processing CET learn for machine learning models Transformers for advanced NLP tasks and text blob for sentiment analysis Additionally the app ensures that essential nltk data sets are downloaded which are required for tokenizing text removing stop words and litiz words text summarization is a core feature of this app the summarized text function utilizes the hugging phas Transformers library to create a text summarization pipeline this function takes the video transcript as input and generates a concise summary making it easier for users to grasp the main points of lengthy video content without watching the entire video keyword extraction is handled by the extract keywords function this function tokenizes the transcript text removes common stop words and litiz the remaining words to their base forms using count vorer from CET learn the function identifies and rans the most frequent keywords in the text these keywords provide a quick overview of the main topics discussed in the video topic modeling is performed using the topic modeling function which employs latent derit allocation LDA from sikit learn this function vectorizes the text and identifies the top words associated with each topic by extracting these topics the app can present users with the main themes and subject areas covered in the video offering deeper insights into the content the app also includes a helper function extract video ID which uses regular

Expressions to extract the video ID from various YouTube url formats this ensures that users can input different types of YouTube URLs and the app will correctly identify and process the video the main function of the streamlit app creates the web interface users can enter a YouTube video URL and specify the maximum length for the summary using a slider when the summarize button is clicked the app extracts the video ID fetches the transcript and performs summarization keyword extraction topic modeling and sentiment analysis the results including the summary keywords topics and sentiment analysis are then displayed on the streamlit interface to ensure a smooth user experience the app includes error handling for various scenarios such as invalid YouTube URS unavailable transcripts and other exceptions appropriate error messages are displayed to guide users in correcting

Keywords: video, text, transcript, youtube, topic

Topics: the, and, for, to, a