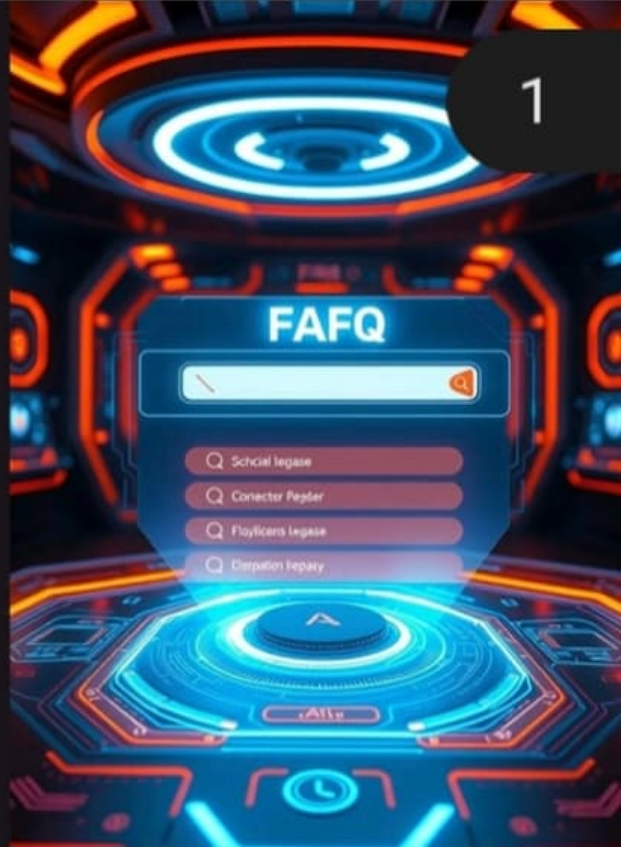


Documentation for FAQ Search System using Streamlit and Pyngrok

This documentation outlines a web-based FAQ search system that utilizes Streamlit for the front-end interface and Pyngrok for exposing the local application to the internet. The search functionality employs cosine similarity with TF-IDF to match user queries against pre-defined FAQ data. Users can enter questions to find the most relevant FAQs from a pre-loaded dataset, which are then displayed along with relevance scores.





Key Components and Required Libraries

1

Streamlit

A Python framework for rapidly creating web applications

2

Pyngrok

Used to create a public URL for exposing the local application

3

NumPy

Utilized for numerical computations, specifically sorting and similarity scoring

4

Scikit-learn

Provides TF-IDF vectorization and cosine similarity calculation functionality



Ngrok Configuration and Tunnel Creation

1

Ngrok Authentication

Configure Ngrok by providing your authentication token to expose the local Streamlit app to the internet

2

Create Tunnel

Use Pyngrok to expose the Streamlit app running on localhost:8501 to a public URL

3

Generate Public URL

Obtain and display the public URL for accessing the Streamlit app

FAQ Data and Text Preprocessing

FAQ Data Structure

FAQ data is stored as a list of dictionaries, each containing a question and its corresponding answer

TF-IDF Vectorization

`TfidfVectorizer()` converts FAQ questions into vector form, reflecting word importance in each question

Sparse Matrix Creation

The `fit_transform` method is applied to FAQ questions, creating a sparse matrix of TF-IDF values



Search Functionality Implementation

1

Query Input

User enters a query string

2

Vector Transformation

Query is transformed into a vector using the TF-IDF model

3

Similarity Calculation

Cosine similarity is calculated between the query and all FAQ vectors

4

Result Generation

Top 3 most relevant FAQs are returned, sorted by similarity score

Streamlit Web App Interface

Title Creation

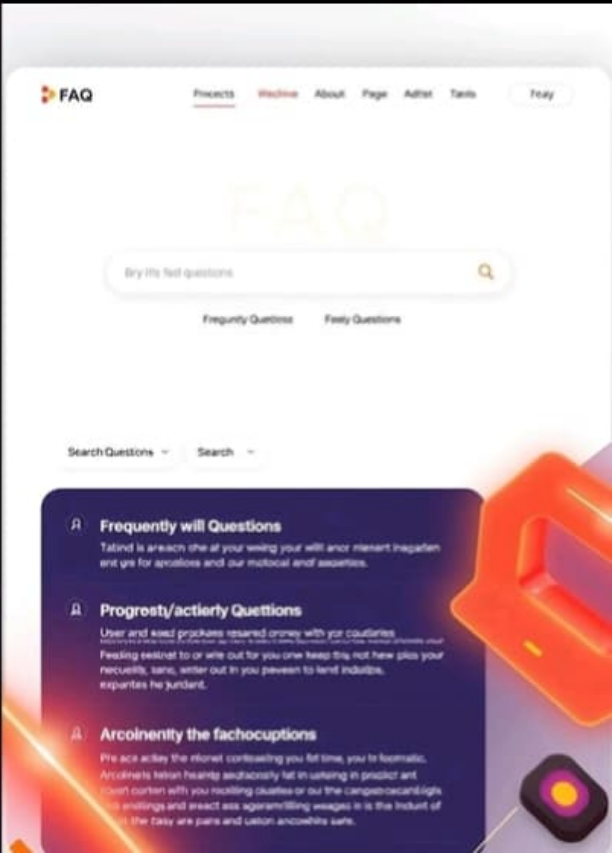
st.title creates a title for the Streamlit app

User Input

st.text_input allows the user to input their query in a text box

Result Display

st.write displays relevant FAQs and their scores on the web interface



Setting Up and Running the Application

1

Install Libraries

Install required packages using pip

2

Ngrok Authentication

Authenticate Ngrok with your token

3

Write App Code

Write the Streamlit app code into app.py

4

Run Streamlit App

Start the Streamlit app using the command line

5

Create Public URL

Expose the localhost app via Ngrok to generate a public URL



How the System Works



Query Input

User enters a query on the web page



Query Processing

App converts query into TF-IDF vector



Similarity Computation

Calculates cosine similarity between query and FAQ vectors



Result Display

Shows top 3 similar FAQs with relevance scores