Smart Research Guide for Efficient Study of a Topic Related to the Computer Science Domain

Ganesh Shingre[1], Abhinav Shivalkar[2], Sahil Shaikh[3], Ruchita Yadav[4]

[1] *2/6 Jai Ambika Niwas, Umesh Nagar, Dombivli, Maharashtra- 421202*

[2] *A/001, Bldg no. 6, Shree Laxmi Park, Lokmanya Nagar, Thane, Maharashtra- 400606*

[3] *A/401, Gundecha Heights, L.B.S Road, Kanjurmarg (w), Maharashtra- 400078*

[4] *D/303, KapilaVastu, Uthalsar, Kolbad naka, Thane, Maharashtra- 400601*

Abstract

Research plays a very important role while we are studying or working on an important topic. In this technological generation where the smallest of small things are now searched with the help of technology. A lot of information is available on the web which makes it difficult for a person to find some key precise points on a specific topic which is like finding a needle in the ocean. Also reading big documents or articles is time consuming. Hence for overcoming this difficulty, 'smart research guide' comes into picture where if information or URL about a specific topic is uploaded on our web application then it makes it easier for the user to learn about the topic. Question answering system[1] is a task of natural language processing. The model named Bidirectional Encoder Representation from Transformer[2][3] was proposed to solve the limits of NLP jobs and can be used for closed domain question answering system. Different features in our app like dataset exploration, semantic search and intent identification will further help to simplify the research process.

1. Introduction

Search is one of the tasks which can be performed on daily basis. Generating relevant and accurate information matters in such tasks is challenging. People are searching the results by asking questions rather than putting the search words. People working in specific domains need to get the relevant documents for much information. Question Answering is a type of information retrieval[6] task which ensures that the questions asked in natural language are answered correctly. To find information about a specific topic research is done, it basically is an in-depth

search to gain knowledge and nowadays as there is a lot of technological enhancement there are a lot of topics that are available and which people want to explore, hence a lot of research papers are published which help people in analysing data and also these papers can be used for further research and this is how data is maintained and more information is added into these data bases. There are many famous research paper publishing sites such as IEEE, research gate, etc. Instead of reading the whole document we can upload the document on our system and get answers to our queries. We can perform this for different types of file formats as well as for videos. Dataset exploration will help to better understand the datasets mentioned in the research articles. Understanding the data becomes easier with the help of data visualization. In our app we can visualize the data using different libraries which will help the user to find any patterns. We can upload multiple documents or the path to the folder containing the documents and input our query. The app will return the topmost relevant documents. Identifying the intent of the given subject or paragraph is very important to know the purpose of the text and can help us to find out if it contains information relevant to the topic of research. It will help in simplifying the research of the user.

2. Proposed Idea

To make the study of research papers more convenient and easier we have proposed four features in our application titled ‘Smart Research Guide’. The ‘context-based question answering system’ feature will help the user to quickly analyse the context by asking relevant questions. ‘Dataset exploration’ allows user to explore the various features and characteristics of the datasets mentioned in the research documents. To find the documents from the set of research documents that are having the related content to the user’s query, ‘semantic search’ will be useful. ‘Intent identification’ helps user to find domain area of the suggestions provided for the future enhancement in the research document.

2.1. Context-based Question Answering System

In this feature the application allows users to enter text as a context and ask any question based on that context. It tries to answer the query with best possible answer by referring the context provided. The application uses BERT (Bidirectional Encoder Representations from Transformers) model which is a pre-trained language model that can be fine-tuned[4][5] for various natural language processing tasks, including question answering. For question answering task we have fine-tuned[14] the model against smaller version of SQuAD (Stanford Question Answering Dataset).

The user can provide the context in three ways:
a) text: Context will be in a plain text format.

b) PDF file: A PDF file or document will be given as a context. This is done using the PdfReader object from the PyPDF2 module of Python programming language.

c) YouTube video URL: User will provide YouTube video URL as an input. Application generates the transcript for the video by using 'YouTubeTranscriptApi' and use it as a context.

The context then will be used to find the answer for the query raised by the user.

2.2. Dataset Exploration

The application allows users to upload a dataset, explore its columns, visualize its features[7], and generate customizable plots. The application defines a function 'file_selector()' that takes a folder path as input and returns a file path selected by the user. The application prompts the user to enter the path of the directory where the dataset is stored. After the user selects the dataset, the application loads it into a 'Pandas DataFrame' and provides several options for exploring the data. The app uses 'Streamlit' widgets such as 'st.checkbox()', 'st.button()', 'st.multiselect()', 'st.selectbox()', and 'st.radio()' to enable users to interact with the data.

The app allows users to display the dataset, show the column names, show the shape of the dataset, select columns to show, display data types, show value counts by target/class, and display summary statistics. The application also allows users to visualize the data using correlation plots, count plots, pie plots, and bar graphs. The application also provides a customizable plot feature that enables users to choose the type of plot, X and Y columns, and plot dimensions. The application uses the 'Matplotlib' and 'Seaborn' libraries to generate plots.

2.3. Semantic Search

A semantic search[10][11] on a collection of PDF documents is performed to search for the PDF documents that contain the required topic. Semantic search is a technique that uses natural language processing (NLP) and machine learning algorithms to understand the meaning of a query and return relevant results. Users can input a query and the path to a directory containing PDF documents to search. The script then extracts the text from each PDF file in the directory and stores it in a list.

Next, the script uses the 'Scikit-learn' library to create a pipeline that performs TF-IDF vectorization[12], dimensionality reduction using latent semantic

analysis (LSA), and normalization of the resulting matrix. TF-IDF is a common method for representing text as a numerical vector, LSA is a technique for reducing the dimensionality of the vector space, and normalization is a process that scales the vectors to have unit length. After the pipeline is created, the script defines a function called ‘semantic_search’ that takes a query and the processed data as input and returns the top most similar documents to the query based on cosine similarity.

2.4. Intent Identification

The intent detection process is carried out by calling the OpenAI API[8] with the 'openai.Completion.create' function. The model parameter specifies the language model[9] to use for the task, which is 'davinci:ft-personal-2023-03-13-15-40-35' which is fine-tuned[13] for intent identification from the text related to the computer science domain. The prompt parameter is the user's text, followed by a newline character and the 'Intent:' string, which tells the model to predict the intent of the text.

```
!openai api fine_tunes.follow -i ft-WaYhG3u0SaZZ4WdxA1IpK9VO

     [2023-03-13 15:16:45] Created fine-tune: ft-WaYhG3u0SaZZ4WdxA1IpK9VO
     [2023-03-13 15:20:47] Fine-tune costs $0.34
     [2023-03-13 15:20:47] Fine-tune enqueued. Queue number: 11
     [2023-03-13 15:21:57] Fine-tune is in the queue. Queue number: 10
     [2023-03-13 15:21:58] Fine-tune is in the queue. Queue number: 9
     [2023-03-13 15:23:01] Fine-tune is in the queue. Queue number: 8
     [2023-03-13 15:24:02] Fine-tune is in the queue. Queue number: 7
     [2023-03-13 15:26:06] Fine-tune is in the queue. Queue number: 6
     [2023-03-13 15:26:07] Fine-tune is in the queue. Queue number: 5
     [2023-03-13 15:28:01] Fine-tune is in the queue. Queue number: 4
     [2023-03-13 15:31:50] Fine-tune is in the queue. Queue number: 3
     [2023-03-13 15:33:07] Fine-tune is in the queue. Queue number: 2
     [2023-03-13 15:33:21] Fine-tune is in the queue. Queue number: 1
     [2023-03-13 15:33:26] Fine-tune is in the queue. Queue number: 0
     [2023-03-13 15:34:39] Fine-tune started
     [2023-03-13 15:37:42] Completed epoch 1/4
     [2023-03-13 15:38:25] Completed epoch 2/4
     [2023-03-13 15:39:09] Completed epoch 3/4
     [2023-03-13 15:39:52] Completed epoch 4/4
     [2023-03-13 15:40:35] Uploaded model: davinci:ft-personal-2023-03-13-15-40-35
     [2023-03-13 15:40:36] Uploaded result file: file-d6xUgTf4e6GT5GGL1or4TEAF
     [2023-03-13 15:40:37] Fine-tune succeeded
```

Figure 1: Fine-tuning of GPT-3 model for intent identification

The remaining parameters control the behaviour of the API. The parameter 'max_tokens' specifies the maximum number of tokens to generate, 'temperature' controls the randomness of the output, and 'stop' specifies the sequence of tokens at which to stop generating text. The result of the API call is stored in the 'response' variable, which contains a list of predicted intents. The script extracts the top intent from the list and displays it to the user as output. If there is no intent detected, the output will be empty.

3. Results

3.1. Generating answers from input data or link

From figure 2, we can see that we are able to extract answers from the input paragraph, in which the input is in plain text format. In figure 3, the input is given in the format of a pdf file which we scan using PdfReader module in python and extract answers for the input questions. From figure 4, we can see that we can extract answers from videos as well. For that we need to upload the URL of the YouTube video, using which we will generate the transcript of the video which will be used to answer the user's questions.

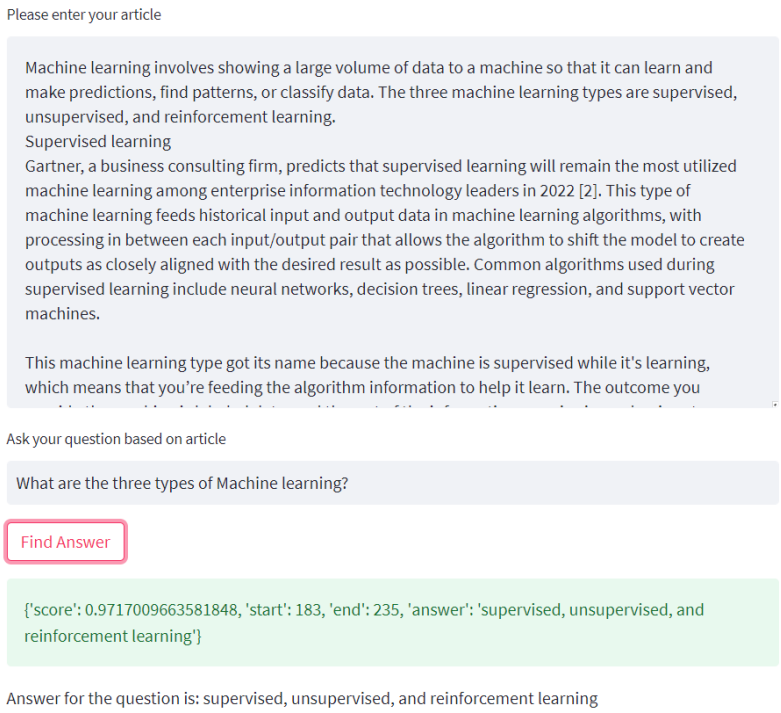


Figure 2: Text based question answering with paragraph as input

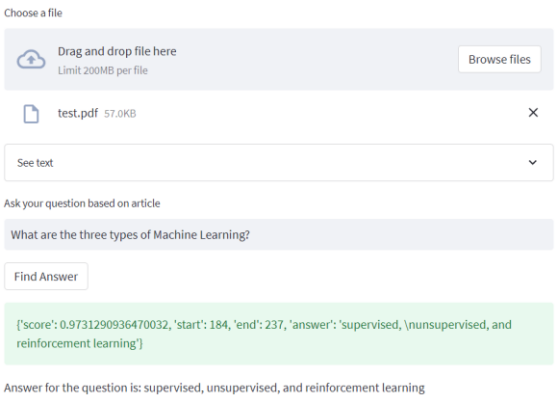


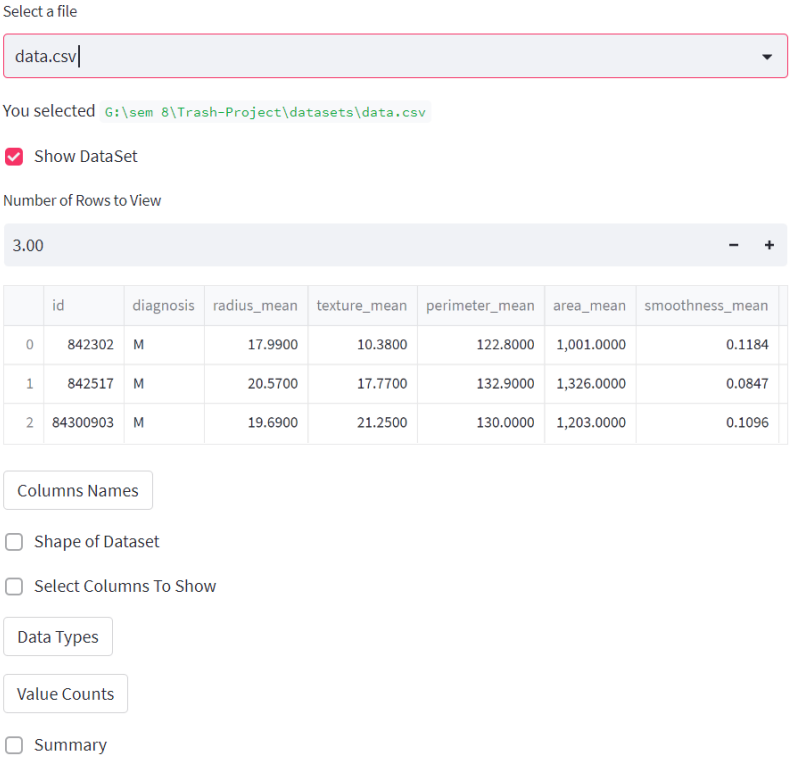
Figure 3: Text based question answering with

Figure 4: Text based question answering with video URL as input

PDF document as input

3.2. Exploring datasets mentioned in research documents

As shown in figure 5, we can explore any dataset mention in the research document. User provides the path from the local machine where the datasets are present. Then application will provide an option to select any dataset from the specified location. User can explore the datasets, can find the characteristics and features of selected dataset. User can also find the correlation between the different parameters and can visualize the results.



	id	diagnosis	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness_mean
0	842302	M	17.9900	10.3800	122.8000	1,001.0000	0.1184
1	842517	M	20.5700	17.7700	132.9000	1,326.0000	0.0847
2	84300903	M	19.6900	21.2500	130.0000	1,203.0000	0.1096

Figure 5: Dataset exploration

3.3. Extracting documents related to given topic from multiple documents

Using 'semantic search' feature user can search for the particular documents from multiple documents contain required topic. As shown in figure 6, user provides the path of all the files in the local machine. Application will search for the files having requested topic in it and display them on the screen along with similarity score.

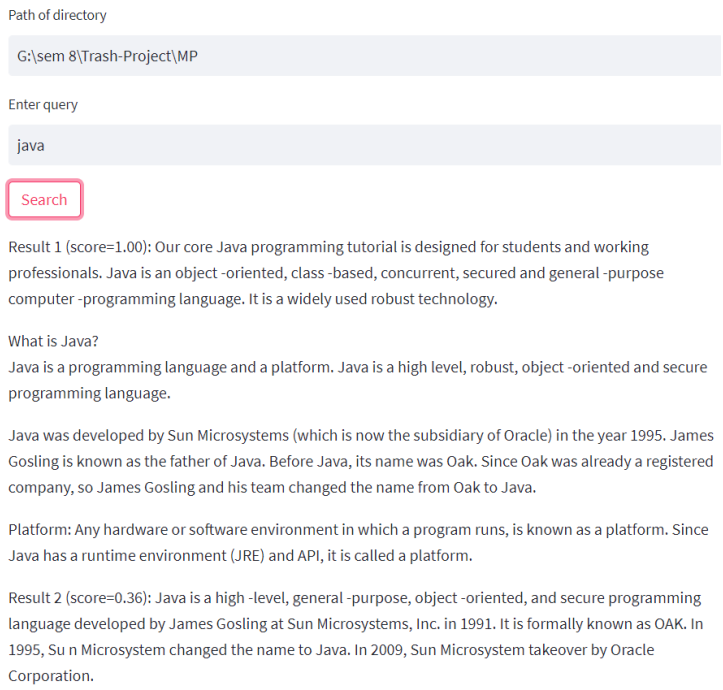


Figure 6: Semantic search

3.4. Intent Identification

As shown in Figure 7, user will insert the text into the text-area and click on 'Find Intent'. Since the intent for the given text which is related to computer science domain is 'Database Design', hence it is returned as answer.

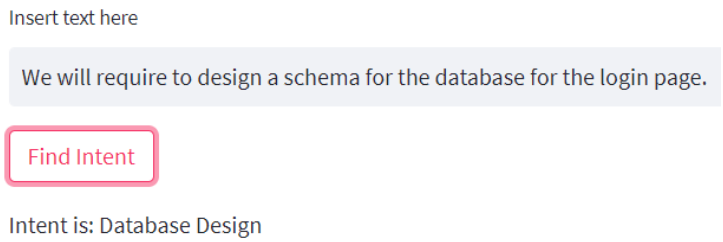


Figure 7: Intent identification

4. Conclusion

The future search engine will be evolved with question answer in technology because many users want to have exact matched answers to his/her question instead of browsing a set of returned web articles. In this project, we have created a question answering system which is equipped with many different informative features that can be very useful in any research process. The dataset exploration feature makes it easier to understand the data and find any patterns. We can find the documents from the set of research documents that are having the related content to the user's query by using semantic search. Intent identification helps

user to find domain area of the suggestions provided for the future enhancement in the research document. The proposed application will help the user to study the topic related to computer science domain more efficiently and can save majority of their time.

References

[01] Min-Yuh Day, Yu-Ling Kuo, “A Study of Deep Learning for Factoid Question Answering System.” 2020

[02] Nguyen Thi Mai Trang, Maxim Shcherbakov, “Vietnamese Question Answering System from Multilingual BERT Model” 2020

[03] Hongchao Jiang, Baoqi Yang, Haowen Wang, Li Jin, “A BERT-Bi-LSTM based knowledge graph question answering method.” 2020

[04] Quoc Thai Nguyen, Thoai Linh Nguyen, Ngoc Hoang Luong, Quoc Hung Ngo, "Fine-Tuning BERT for Sentiment Analysis of Vietnamese Reviews" 2020

[05] Ferhat Demirkiran, Aykut Cayir, Ugur Unal, Hasan Dag, "Website category classification using fin-tuned BERT language model" 2020

[06] Bineet Kumar Jha, Chandra Mouli Venkata Srinivas Akana, Anand R, "Question Answering System with Indic multilingual-BERT" 2021

[07] T. Liu, D. Bangash Ahmed, F. Bouali, G. Venturini, "Visual and interactive exploration of a large collection of Open Datasets" 2013

[08] Yuanbin Qu, Peihan Liu, Wei Song, Lizhen Liu, Miaomiao Cheng, "A Text Generation and Prediction System: Pre-training on New Corpora Using BERT and GPT-2" 2020

[09] Sahil Papalkar, Arati Nagmal, Shreya Karve, "A review of dialogue intent identification methods for closed domain conversational agents" 2018

[10] Duygu Tümer, Mohammad Ahmed Shah, Yiltan Bitirim, "An Empirical Evaluation on Semantic Search Performance of Keyword-Based and Semantic Search Engines: Google, Yahoo, Msn and Hakia" 2009

[11] Vajenti Mala, D.K.Lobiyal, "Semantic and Keyword Based Web Techniques

in Information Retrieval" 2016

[12] Hai Dong, Farookh Khadeer Hussain, Elizabeth Chang, "A Survey in Semantic Search Technologies" 2008

[13] “Fine-tuning OpenAI API”, OpenAI, 2022. [Online] available https://platform.openai.com/docs/guides/fine-tuning

[14] “Finetune a model”, huggingface.co, 2021. [Online] available https://huggingface.co/docs/transformers/training