Tuijin Jishu/Journal of Propulsion Technology
ISSN: 1001-4055
Vol. 44 No. 6 (2023)

Automating Pdf Interaction Using Langchain

Karan R.[1], M. Rahul Kumar[2], Rubanshiju A.[3] T. Kirubadevi[4]

[4]Asst.Professor , [1,2,3] Final Year B.Tech,

[1234]Dept of Computer science and Engineering

[1,2,3,4] Dr MGR Educational and Research Institute, Chennai, India

Abstract

With the world becoming more and more advanced in AI, there is a significant increase in demand. With more and more adoptive usage, the value of time and money has become the most prominent in every field. We handle documents, study materials, guidelines, theses, theories, research papers, etc. in the form of PDF. We all prefer it as the most convenient and efficient way of sharing documents over the internet. We came up with the idea of developing an interactive chatbot that responds to the query raised by the user. This saves time and provides the user with relevant information from within the document they want. Our motive is to provide users with a simple interface that consists of a chatbot that provides relevant information with respect to the document they want in a normal chat-like experience.

1. Introduction

In the fast-evolving and fast paced landscape of research and academia, the efficient interaction with PDF document plays an important role. For both research professionals and students, the ability to automate PDF interactions has become not the just a convenience, but a necessity in today's digitally-driven world. PDFs have become the de facto format for sharing scholarly articles, research paper, textbooks, and other critical documents. However, manual manipulation of PDFs can be time-consuming and error- prone, often hindering the pace of research and learning.

Automating PDF Interaction, has emerged as a transformative solution, offering a streamlined and effective way to navigate, extract data, annotate, and collaborate with PDF documents. This technology is bridging the gap between the vast reservoir of digital knowledge and the researchers and students seeking to access, analyze and contribute to it. In this exploration of Automating PDF Interaction, we will delve into the simplistic way, in which this technology is revolutionizing the research industry and enhancing the academic journey for students. We are intending to come up with a solution that involves the use of Generative AI tools, python libraries, and OpenAI APIs to overcome this problem to a certain extent. This environment comprises a chat screen, in which the user enters the query in the normal English way. This creates a kind of interaction with the system, and as a

result, the relevant information will be generated below the query and will not be taken from outside the document. The generated response below will be related to the relevantuery being asked, and it does not depend on the nature or essence of the query being used.

The evolving landscape of research and academia is increasingly reliant on efficient interaction with PDF documents. PDFs have become the standard format for sharing scholarly articles, research papers, textbooks, and critical documents. However, the manual-handling of PDFs can be time-consuming and error-prone, impeding the pace of research and learning. As a result, automating PDF interactions has transitioned from being a convenience to a necessity in today's digital world. Automating PDF interactions is a transformative solution that offers a streamlined and effective way to navigate, extract data, annotate, and collaborate with PDF documents. This technology is bridging the gap between the vast reservoir of digital knowledge and the researchers and students

seeking to access, analyze, and contribute to it. Let's explore how this technology is revolutionizing the research industry and enhancing the academic journey for student.

2. Literature Study

The literature study explores the evolving landscape of Automating PDF Interaction novel language-based technology. The research delves into the intersection of language processing and PDF manipulation, aiming to provide an in-depth understanding of existing approaches, challenges and potential advancements in the field. The introduction sets the stage by outlining the significance of automating PDF interaction and introduces LangChain as a promising solution. It highlights the growing need for handling of PDF documents in research and business contexts. This section reviews existing language-based technologies, including natural language processing (NLP) and machine learning, emphasizing their applications in document processing. Special attention is given to their relevance in automating interactions and with PDF files. Examining traditional PDF manipulation techniques provides a baseline for understanding the challenges associated with automating interactions. This section reviews existing tools and methods, outlining their strengths and limitations. This versatility in usage, coupled with applications's ability to run on consumer's laptop and open-source Python [3] software positions Automating PDF Interaction Application as a significant aid to researchers, assisting in conducting effective and efficient literature reviews.

"Knowledge Gathering", "Knowledge Extraction", and "Knowledge Synthesis", each of the module encapsulates a fundamental aspect of the Automating Application process,and together, they present a holistic approach to conducting literature reviews [2].

A detailed exploration of LangChain is presented, elucidating its architecture, capabilities, and potential applications in automating PDF interactions. This section aims to establish LangChain’s unique features that distinguish it from other technologies. Vector database is a collection of data stored as mathematical representations. They make it easier for machine learning models to remember previous inputs, allowing machine learning to be used to power search, recommendations and text generation use-cases. We have used FAISS (Facebook AI Search Similarity) vector database.

3. System Methodology

Login Page

A basic login page developed using HTML, CSS and JavaScript. An existing user can proceed with the username and password they have used at the time of registration. After successful login, user will be redirected to the webpage.

Database

The login credentials gets validated with the database. PhpMyAdmin is the database used in the backend which contains the credentials stored. Using PHP, it checks those credentials and directs the user to the application page, if the credentials match otherwise throws an error not found.

Application page

The application page comprises of a sidebar and an upload button at the top. To begin querying with the PDF, the user needs to upload the PDF file, using upload button. The max file size the user can upload is 1GB.

Chat Window

Once the user uploads the file, it takes few seconds for the chat box to load. The user then can enter the query within the chatbox. The entered query will be revised below the chatbox, so that the user can check, what query they typed into the chatbox.

Result Window

Results related to query will be displayed below the chatbox. User can enter the next query within the Chatbox itself.

The entered query will be displayed below the chatbox. Information related to query will be displayed below the entered query.

4. Experimental System

The current system handles 3000+ tokens at a time. However, in the proposed system, we are using the existing model that works on 4096 tokens. The model implemented is the "Da Vinci Model," which belongs to OpenAI and takes in the query and classifies text on the basis of tokens. The user then asks a query related to the file they have uploaded. The relevant information will then be displayed below the query.

Users can ask the model about a particular context, and the model will reply with relevant information below the entered query chatbox.

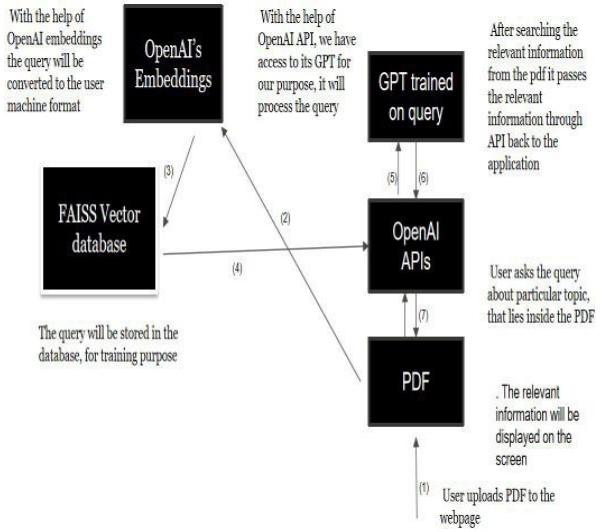

Fig.4.1.Work-flow

Our system comprises a simplistic approach in dealing with PDF files. First, the users have to enter credentials on the login page. Once the user has entered the credentials, and if the credentials match the database, then only they will be authorized to use the PDF Interaction interface. The web page will automatically be redirected to the Interaction Interface.Each character entered into the chatbox by the user transforms into embeddings. Once it transforms into embeddings, it is stored in the FAISS vector database. It is this vector database that is used for training the GPT model that will eventually provide accurate results. How it works, depends on the query as well as the language in which the query is embedded. This helps in determining the metrics of the model that enhance the stability and results of the application.

5. Results and Discussions

The application of Automating PDF Interaction using LangChain provides outcomes that are relevant to the query raised by the user. The model used is the existingmodel of OpenAI DA VINCI, that returns a maximum of 4096 tokens. The interface is simplistic that helps individuals to raise the query relevant to the PDF file they are uploading.

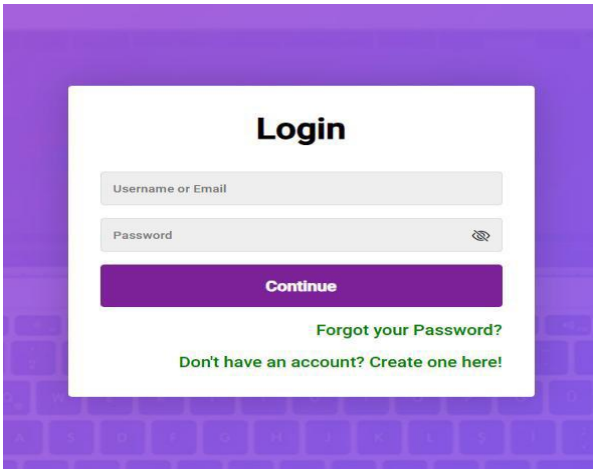

Fig.5.1.Login Page

Fig.5.3.Result Screen

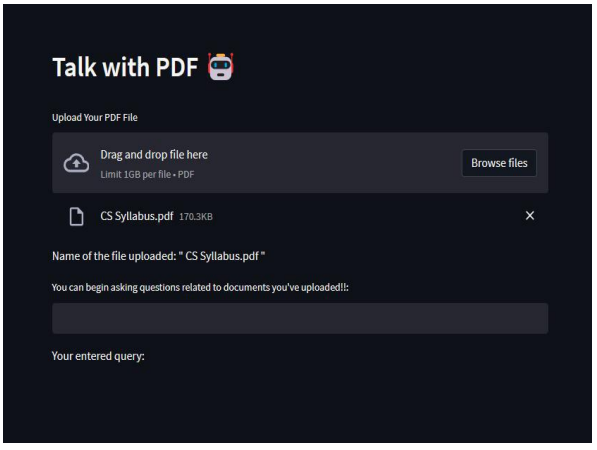

Fig.5.2.Chat Window

Conclusion

This system is useful for those domains, like medicine, engineering, and science, and for keeping up with the latest research findings. This system is designed to adapt to and evolve with the fast-paced growth of artificial intelligence. It reduces the amount of time and effort required for conducting literature reviews.Also, not only does it help the model understand the interaction, but it also helps the model produce relevant queries in the upcoming queries. In conclusion, the automating PDF interaction application holds immense promise for both researchers and students. For researchers, it offers the means to streamline the retrieval, organization and analysis of vast volumes of scholarly content. It accelerates the pace of discovery. Meanwhile for students, it presents a valuable tool for efficient document management, aiding in research projects, and collaborative endeavours. Embracing PDF automation is not just a technological leap. It's a game-changer that empowers researchers and students to excel in their field. This makes knowledge more accessible and research more efficient than ever before. It promotes scalability, and saves time for the users. The model that is used for processing and producing the result, later can be used for different applications as well. As this is a reinforcement learning, it is very useful for students from computer science background to understand how the model is working.

References

[1] OpenAI. GPT-4 technical report, (arXiv:2303.08774), 2023

[2] David A. Tovar: AI Literature Suite(arXiv:2308.02443)

[3] ,2023

[4] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier

[5] Raparthi, M., Dodda, S. B., & Maruthi, S. (2023). Predictive Maintenance in IoT Devices using Time Series Analysis and Deep Learning. Dandao Xuebao/Journal of Ballistics, 35(3). https://doi.org/10.52783/dxjb.v35.113

[6] Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. Llama 2: Open foundation and fine-tuned chat models. arXiv, (arXiv:2307.09288), 2023.

[7] Guido Van Rossum and Fred L. Drake. Python 3.9.5 Documentation. Python Software Foundation, https://docs.python.org/3/, 2021.

[8] Charles R. Harris, K. Jarrod Millman, Stefan J. van der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, Julian Taylor, Sebastian Berg, Nathaniel J. Smith, et al. Array programming with NumPy. Nature, 585(7825):357–362, 2020.

[9] Jeff Reback, Wes McKinney, jbrockmendel, Joris Van den Bossche, Tom Augspurger, Phillip Cloud, Simon Hawkins, gfyoung, Sinhrks, Adam Klein, Brock Petersen, Matthew Roeschke, Jeremy Tratner, Chang She, William Ayd, Shlomi Naveh, Mark Garcia, Vytautas Jancauskas, Kai Dong, Jason Schendel, Andrew Hayden, Ben Pardee, Faisal Aish, Tom Horrocks, et al. pandas-dev/pandas: Pandas, 2020.

[10] Yinfei Yang, Daniel Cer, Amin Ahmad, Mandy Guo, Jax Law, Noah Constant, Gustavo Hernandez Abrego,

Steve Yuan, Chris Tar, Yun-Hsuan Sung, Brian Strope, and Ray Kurzweil. Multilingual universal sentence encoder for semantic retrieval. arXiv, (arXiv:1907.04307), 2019.

[11] Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, Yun-Hsuan Sung, Brian Strope, and Ray Kurzweil. Universal sentence encoder. arXiv, (arXiv:1803.11175), 2018.

[12] Thomas Donoghue. Lisc: A python package for scientific literature collection and analysis. Journal of Open Source Software, 4(41):1674, 2018.

[13] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel,

[14] B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. Journal of Machine Learning Research, 12:2825–2830, 2011.

[15] pdfgpt. https://github.com/bhaskatripathi/pdfGPT.

[16] Crossref.https://www.crossref.org/documentation/retri eve-metadata/rest-api/.

[17] CORE API. https://core.ac.uk/services/api