PRESTIGE
INSTITUTE OF MANAGEMENT & RESEARCH, GWALIOR

MINOR PROJECT REPORT

ON

RELAX – A LEARNING CURVE

Towards Partial Fulfillment of Requirement

Of

Bachelor of Computer Application

Batch 2021-2024

SUBMITTED TO

Information Technology Department,

PIMRG

SUBMITTED BY

Ω Ravi Kishan
Ω Shivani Yadav
Ω Vishnu Tomar
BCA – IV SEM

DECLARATION

We, **Ravi Kishan, Shivani Yadav, and Vishnu Tomar** students of BCA IV Sem. of Prestige Institute of Management & Research, Gwalior, hereby declare that the Major Project Report entitled "**RELAX – A LEARNING CURVE**" submitted by us in the line of partial fulfillment of course objectives for the Bachelor of Computer Application.

We assure you that this project report is the result of our efforts and that any other institute for the award of any degree or diploma has not submitted it.

Date: 23-06-2023

Place: Gwalior

Name:

Ravi Kishan

Shivani Yadav

Vishnu Tomar

CERTIFICATE

This is to certify **that Ravi Kishan, Shivani Yadav, and Vishnu Tomar** of BCA IV Semester of Prestige Institute of Management & Research Gwalior, have completed their Minor Project. They have prepared this report entitled "**RELAX-A LEARNING CURVE**" under my direct supervision and guidance.

Date: 23-06-2023

Place: Gwalior

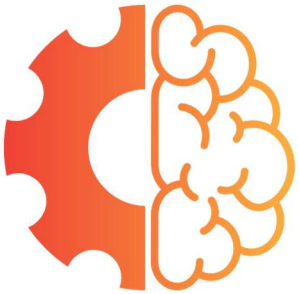
Asst.Prof. Mahendra Singh Yadav

(Faculty Guide)

TABLE OF CONTENT

UNIT – 1: INTRODUCTION

We ***Ravi Kishan***, ***Shivani Yadav,*** and ***Vishnu Tomar*** are the students of the BCA IV semester were assigned the task to make an application where we need to use GENERATIVE AI to build an application to solve daily life problems using AI. Which helps people to work faster and more efficiently.



This application should also help us learn the GENERATIVE AI concept and work on practice.

Thus, we made a trilogy of applications to demonstrate the application of GENERATIVE AI. Let's talk briefly about these applications. We are going to know more about it in upcoming chapters.

1 The first application is about how natural language models learns the content of the different type of file and text to learn and generate output to make life easier. In this application, we are trying to teach the content of a pdf file of the program/machine learning model to understand and give information according to our questionnaire.

In this application, we upload the pdf file and the model going to understand the content and open a chat section to talk to pdf. In the chat section, we are going to ask the question which answers consist in pdf files. You can just ask the question and the application is going to find the answer in the pdf files and give the answers in natural language. This is a fully web-based application that is easy to use and works faster.

This application has an easy-to-use UI and navigation for more details like

- Δ You have a beautiful chat section.
- Δ You can see the currently selected pdf in the viewer section.

- You can get the direct page number, where the answer is written.
- You can upload a new pdf to proceed.
- You can select between multiple books/pdfs.

2 The second application is an extended version of the previous application. In this application, we are trying to teach the context of ***Shrimad Bhagavat Gita*** to a machine learning model. This model trains the slokas and their meaning of Shrimad Bhagavat Gita. In this application, you can ask any question to Shrimad Bhagavat Gita in natural language and the application tries to find the solution to your daily life problem in the meaning of Gita slokas.

Shrimad Bhagavad Gita is a holy religious book of Hinduism. This book is said by ***Shri Krishna (God in Hinduism)***. In the middle of the war of Mahabharata, Arjun (one of the main characters of Mahabharata) was confused to fight with his family members. Then Shri Krishna told him the meaning of life and the way to live life. This is the story of Shrimad Bhagavat Gita.

In this application, you ask the question about your problem and the machine learning model finds the answer in the meaning of the slokas of Shrimad Bhagavat Gita. This application is very useful for those people who are interested in Shrimad Bhagavat Gita and want to know the meaning of the slokas of Shrimad Bhagavat Gita. This application is also useful for those people who are facing problems in their life and want to know the solution to their problems in the meaning of Shrimad Bhagavat Gita.

In this application, you can get the slokas and their translation which is related to your answer and you can get more details about the slokas by clicking on the slokes.

3 The third application is about the ability of GENERATIVE AI to generate images from text.

The model can generate images from your prompt text and you can also control the image style by adding style images. The machine learning model read the prompt and finds similar images and the basis of learning/understanding creates the image.

This application is not about finding the image on the internet and showing you the image. It is about generating the image from the text. The model is trained on the 14 million images and it can generate the image from the text.

In this application, you need to write a prompt according to the image you want to generate. For example, if you want to generate the image of a cat, you can write "A cat sitting on the table". The model will generate the image of a cat sitting on the table. You can also control the style of the image by adding the style image.

This application uses the DALLE API for generating images.

You also can share the images with the community and also can see the images generated by other users. You just need to generate an image and write your name and click on the share button. The image is uploaded.

The project will require the following skills:

1. Better understanding and practice of the Python programming language.
2. Better understanding and practice of the ***pandas'*** library and other data science libraries.
3. A better understanding of the machine learning (Natural Language Processing) workflow.
4. You will learn how to build a machine-learning model to predict the sentiment of a text.
5. You will learn how to build and deploy a ***Flask*** web app.

6. You will learn basic web development concepts such as ***HTML, CSS, and Tailwind***.
7. You will have a better understanding of tools such as
    - ∑ Git and GitHub
    - ∑ Google Colab
    - ∑ Jupyter Notebook
    - ∑ Visual Studio Code
    - ∑ Pycharm
    - ∑ Heroku
    - ∑ etc.
8. In some applications, you need to have a good understanding of web scraping using tools such as ***Beautiful Soup and Selenium***.

We have divided these works among our teammates, each has to prepare his/her work and compile all the work in a single unit. After the compilation of the project, every teammate has to teach others about their contribution to the project.

We have faced some challenges while working on this project and resolved some by participating in discussions with our teammates. Everyone worked hard for this project by giving their hundred percent.

This project, there can be many future enhancements that can be made in this project such as making the model more accurate with more training and scalability of data. The project can be faster to use a more effective algorithm for training and processing models. It can be easier to use to change in UI. For more future enhancements, we talk at the last.

Thank you, Prestige Institute of Management & Research, Gwalior for giving us this opportunity to

work as a team and showcase our learning. It taught us a lot about machine learning technology and how to work as a team.

UNIT – 2: PROBLEM STATEMENT

In this project, we have three different applications of **NLP** which consist of different kinds of problem statements. The first two applications are based on text generation and the last one is based on image generation.

Let's talk about the first two application's problem statements. and after that, we will talk about the last application's problem statement.

Q. What is the purpose of making Doc Chat?

In this project, we came up with a problem statement that a computer can understand the human natural text and its meaning. We know we already know the answer to this problem statement. We all know that computers can understand the human natural text and its meaning. But how? How can a computer understand the human natural text and its meaning? The answer is NLP. NLP is a field of computer science, artificial intelligence, and computational linguistics concerned with the interactions between computers and human (natural) languages. As such, NLP is related to the area of human–computer interaction. Many challenges in NLP involve natural language understanding, that is, enabling computers to derive meaning from human or natural language input, and others involve natural language generation.

NLP leads us to an interesting journey about how the NLP models know many things. Surely, the answer is data. The data is the key to the success of NLP models. The more data you have, the more accurate your model will be. But the data is not the only thing that makes the model accurate. Many things make the model accurate.

The question is going to more depth that what type of data we need to make the model accurate. The answer is labeled data. Labeled data is the data that has been assigned a label. For example, a piece of data can be an image, a text, or an audio file. The label is a description of the data. For example, an image can be labeled as “cat” or “dog”, a text can be labeled as “positive” or “negative”, and an audio file can be labeled as “happy” or “sad”. The labeled data is used to train machine learning models. The more labeled data you have, the more accurate your model will be.

Then I am thinking about how we get knowledge to survive in this world. The answer is we learn. we learn from parents, teachers, and many more. We study to get the necessary knowledge to survive in this world. So, the question is how do we learn? The answer is we learn from books. a lot of books, we study a lot of books to get knowledge.

I just want to know how a machine learning model (NPL model) learns from books to find the answer to the question. So, I am going to make a model that can learn from books and answer the question.

For the first application, we are going to create an application that takes and pdf file-based book/document and learns the content of the book/document, and answers the question based on the content of the book/document.

Q. What is the purpose of making Gita Saar?

After creating the first application, we thought that this model can learn the content in a structured way or not. So, we are going to create a second application more advanced than the first application. So, we decide to train a model with the context of the book Shrimad Bhagavat Gita.

The problem is the first application is learned based on pages in the book but in this case, we want to train the model based on slokas and their context which makes search across the slokas, not the pages and we can get the answer of the question more accurately. we also get the slokas number in which the answer is present. This makes things more accurate and understanding the context of the slokas.

Shrimad Bhagavat Gita is a book which is written in the Sanskrit language. So, we are going to use the English translation of the book. We are going to use the English translation of the book which is available on the internet. In this application, the user can type the question and the model find the accurate answer to the question in Slokes and also give the Slokes number in which the answer is present. You can also get more details about the slokas by clicking on the sloke.

Q. What is the purpose of making an Image Generator?

After creating the first two applications, we thought that this model can learn the content of the image or not. So, we are going to create a third application more advanced than the first two applications. So, we decide to train the model in the context of an image. But this is more difficult than the first two applications. So, we decide to use the pre-train model for this application. We are going to use the pre-train model DALLE. DALLE is a GPT-3 like transformer model that is trained to generate images from text. You can just type and get the image generated.

The application should not be about searching the image across the internet, it is about the generation of the image based on the text.

Q. How the application is going to build?

We have a big problem to give the shape of the application means we need a web-based application. So, we are going to use the Flask framework to create the web-based application. Flask is a micro web framework written in Python. It is classified as a microframework because it does not require particular tools or libraries. Flask supports extensions that can add application features as if they were implemented in Flask itself. Extensions exist for object-relational mappers, form validation, upload handling, various open authentication technologies, and several common framework-related tools. Extensions are updated far more regularly than the core Flask program.

UNIT – 3: PROBLEM ON PRACTICE

This project is all about learning not about the final product. I have been wanting to learn how machine learning works. what are generative ai and natural language processing? I have been reading about it for a while and I wanted to try it out.

Let's start with the basics. We are going to understand the basics of machine learning and how it works. what are the terminologies used in this project like generative ai and natural language processing? what are they and how they work. we understand all in a sequence.

In the way of describing the structure and functions of this project, I would like to use an approach of Jinja's explainer technique. In this technique, I break things into sequences and explain one after another. We break the concept into some chapters and explain it.

These are the chapters:

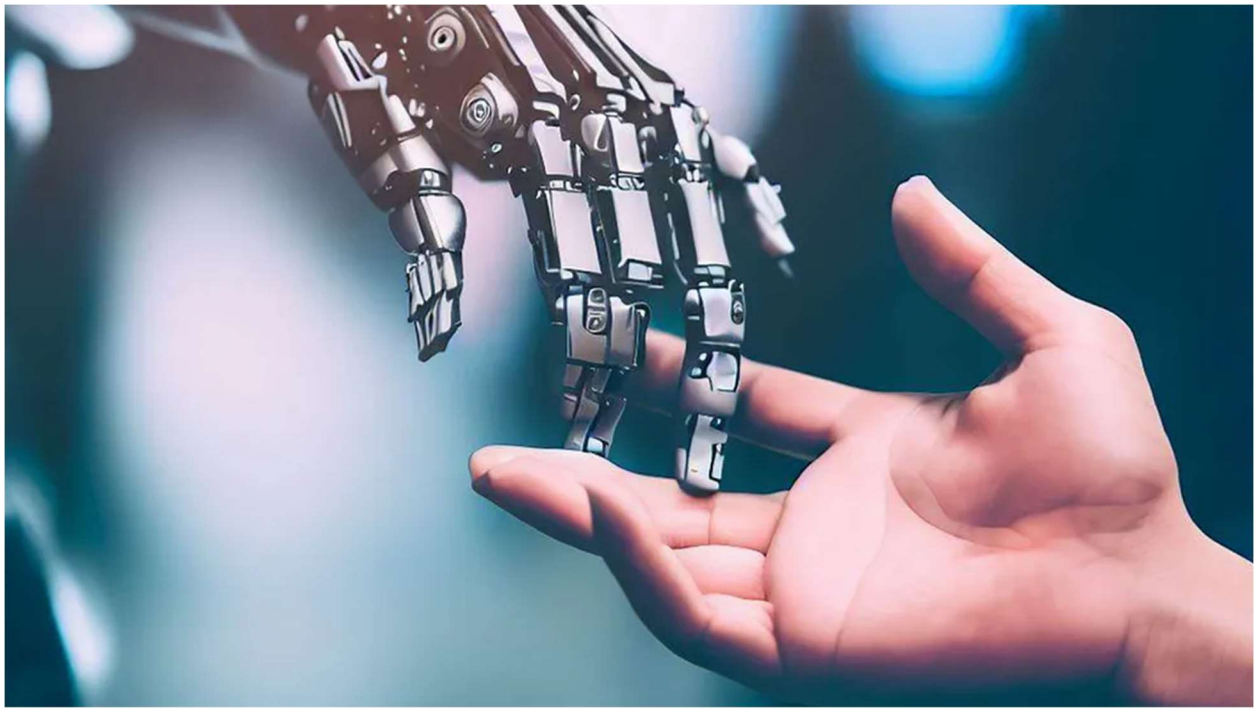
1. TERMINOLOGY
    a. ARTIFICIAL INTELLIGENCE
    b. GENERATIVE AI
    c. NATURAL LANGUAGE PROCESSING
2. TOOLS
    a. PYTHON
    b. FLASK
    c. REACT
    d. HTML, CSS, AND JAVASCRIPT
    e. VISUAL STUDIO CODE
    f. JUPYTER NOTEBOOK
3. DOC CHAT
4. GITA SAAR
5. IMAGE GENERATOR

CHAPTER 1: TERMINOLOGY

Artificial Intelligence: Empowering the Future of Technology and Society

Introduction – Artificial Intelligence (AI) has emerged as a groundbreaking field of study that aims to develop intelligent systems capable of performing tasks that typically require human intelligence. It encompasses a range of techniques and algorithms designed to enable computers to learn, reason, perceive, and interact with the world in ways that mimic human cognitive abilities. This essay explores the concept of artificial intelligence, its applications, challenges, and its impact on various aspects of society.

Defining Artificial Intelligence– Artificial Intelligence refers to the development of intelligent systems that can autonomously analyze, interpret, and respond to data or tasks, simulating human-like intelligence. AI systems employ techniques such as machine learning, natural language processing, computer vision, and robotics to process and understand complex information, learn from experience, and make informed decisions.



Application of Artificial Intelligence–

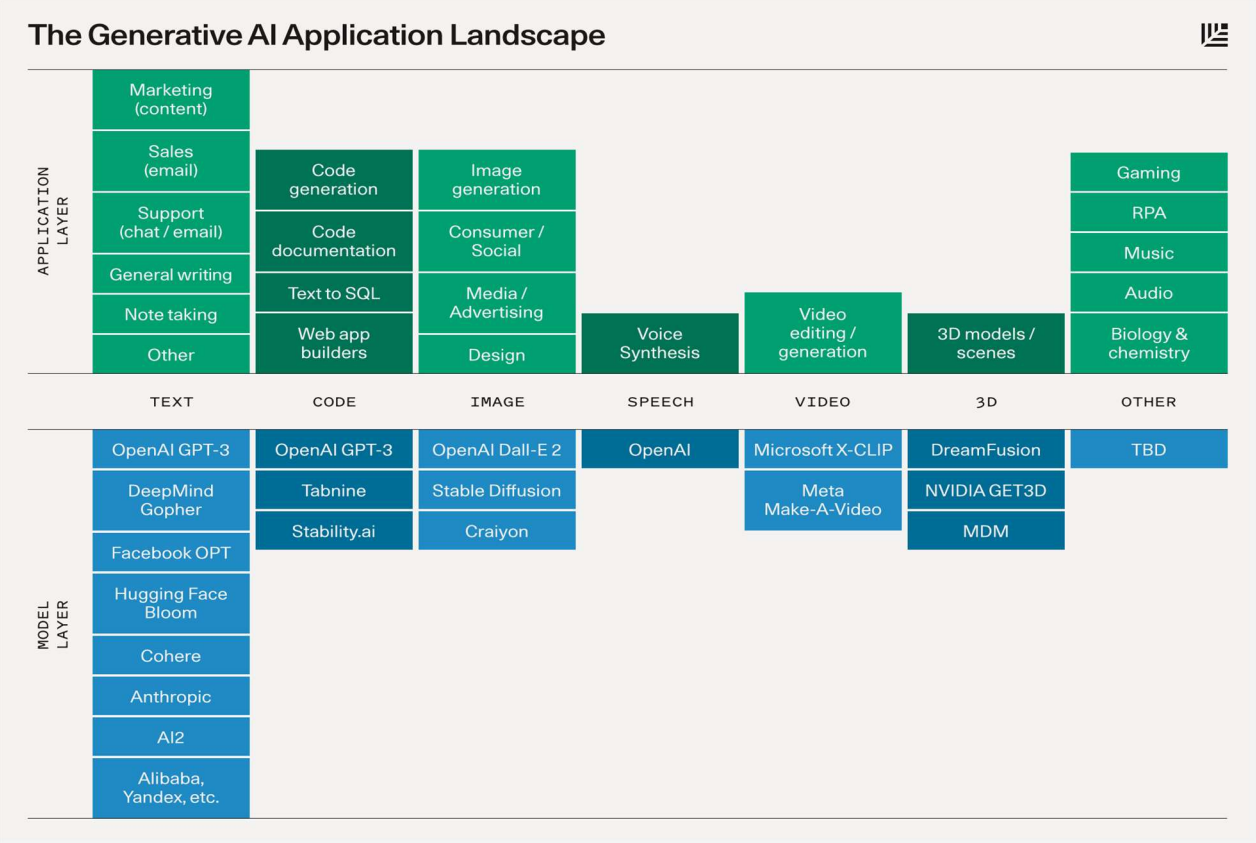
1. Image and Object Recognition: Machine learning algorithms, particularly convolutional neural networks (CNNs), can analyze and classify images, enabling applications such as facial recognition, object detection, and autonomous vehicle perception.
2. Natural Language Processing (NLP): Machine learning techniques are used to analyze and understand human language. NLP applications include machine translation, sentiment analysis, chatbots, and speech recognition.
3. Recommendation Systems: Machine learning algorithms power recommendation systems, which suggest personalized content or products based on user preferences and behavior. These systems are commonly found in e-commerce, streaming services, and social media platforms.
4. Fraud Detection: Machine learning models can identify patterns and anomalies in data, making them valuable in fraud detection and prevention. They can flag suspicious transactions, detect fraudulent activities, and protect against cyber threats.
5. Predictive Analytics: Machine learning enables organizations to predict future outcomes based on historical data. Applications include sales forecasting, demand prediction, risk assessment, and predictive maintenance.
6. Healthcare: Machine learning is revolutionizing healthcare by aiding in disease diagnosis, personalized treatment recommendations, drug discovery, and medical image analysis.
7. Financial Analysis: Machine learning algorithms are used in finance for credit scoring, algorithmic trading, fraud detection, and risk assessment. They analyze large datasets, identify patterns, and make predictions to support decision-making.

Conclusion – Artificial Intelligence has become a driving force behind technological advancements, transforming industries, and revolutionizing the way we live and work. Its applications in healthcare, finance, transportation, education, and customer service are just the tip of the iceberg. However, addressing challenges related to bias, privacy, ethics, and workforce displacement is crucial to ensuring the responsible and beneficial use of AI. With continued research, ethical considerations, and collaboration, artificial intelligence has the potential to shape a future that leverages intelligent systems to enhance human capabilities, drive innovation, and solve complex societal challenges.

Generative AI: Unleashing the Creative Power of Machines

Introduction – Artificial Intelligence (AI) has made significant strides in recent years, and one of its most intriguing and innovative branches is generative AI. This subset of AI focuses on enabling machines to create original content, such as images, music, and even text, without explicit human guidance. Generative AI is revolutionizing various fields, from art and design to entertainment and scientific research. This essay explores the concept of generative AI, its applications, and its implications for society.

Understanding Generative AI – Generative AI utilizes machine learning algorithms to generate new and original content that resembles human-created content. It leverages a variety of techniques such as deep learning, neural networks, and probabilistic models to learn from existing data and produce novel outputs. The primary goal of generative models is to capture the underlying patterns and structures in the training data to create new content that is coherent, visually appealing, and aligned with human preferences.



Applications of Generative AI –

1. *Applications in Creative Industries* – Generative AI has the potential to revolutionize the creative landscape across various industries. Visual arts, can generate stunning and realistic images, allowing artists to explore new styles and techniques effortlessly. For instance, the use of generative adversarial networks (GANs) enables the creation of lifelike paintings, digital sculptures, and even realistic landscapes. Similarly, in the field of music, generative models can compose original melodies and harmonies, creating endless possibilities for musicians, composers, and sound designers.
2. *Enhancing Design and Fashion* – Generative AI also finds its application in design and fashion. By analyzing vast amounts of data and style patterns, AI algorithms can generate innovative designs, textures, and patterns for clothing, accessories, and interior decor. Designers can leverage generative models to explore a wide range of design possibilities quickly and efficiently, ultimately pushing the boundaries of creativity and fashion.
3. *Advancements in Natural Language Processing* – Generative AI has made significant strides in the field of natural language processing (NLP) as well. Language models like OpenAI's GPT have demonstrated impressive capabilities in generating coherent and contextually relevant text. This technology has implications for content creation, automated translation, chatbots, and even virtual storytelling. By generating realistic and engaging narratives, generative AI models can captivate audiences and redefine the way we consume and interact with written content.
4. *Ethical Considerations and Challenges* – While Generative AI holds immense promise, it also raises ethical considerations and challenges. One significant concern is the potential misuse of generative models to create deepfake content or propagate misinformation. The ability to generate realistic images, videos, and text can be exploited to deceive, manipulate, or defame individuals and institutions. As Generative AI progresses, it becomes crucial to develop robust mechanisms for authentication and verification to address these concerns effectively.

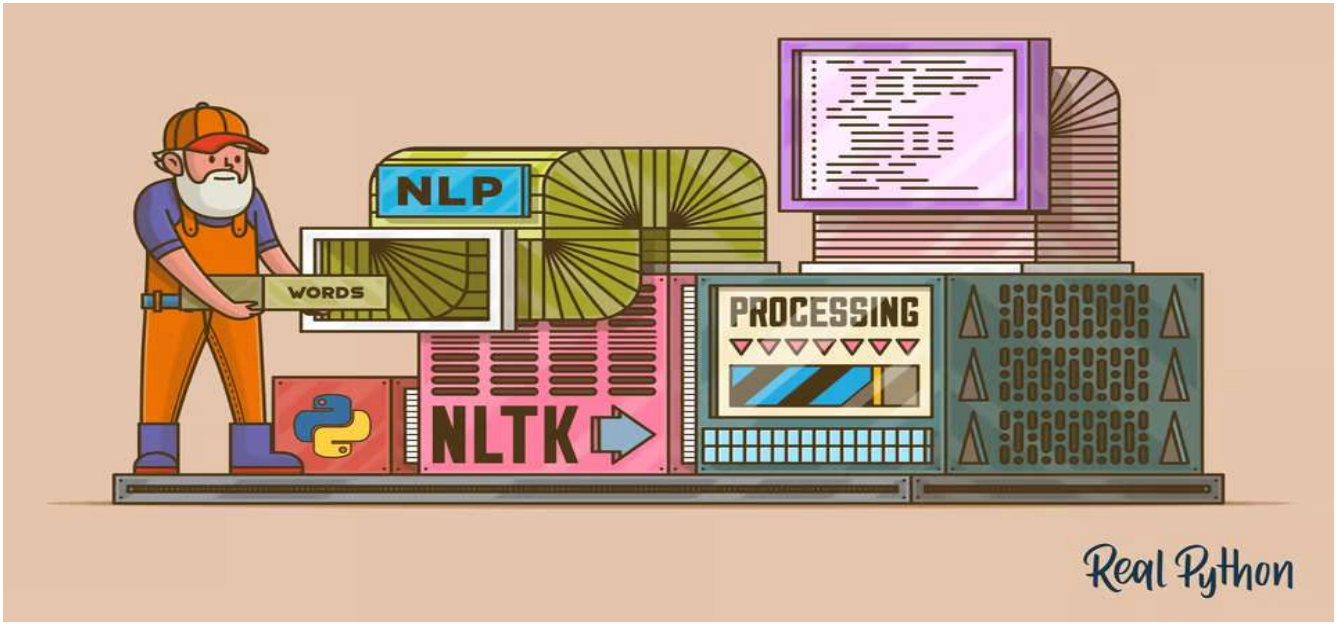
Conclusion – Generative AI has emerged as a groundbreaking technology that has the potential to revolutionize the creative industries and transform the way we perceive and interact with various forms of content. Its ability to generate realistic visuals, music, and text opens up new

avenues for artistic expression, design innovation, and storytelling. While ethical concerns remain, with responsible development and regulation, generative AI can unlock unprecedented opportunities for human creativity and imagination, ushering in a new era of human-machine collaboration.

Natural Language Processing: Bridging the Gap between Humans and Computers

Introduction – Natural Language Processing (NLP) is a subfield of artificial intelligence (AI) that focuses on the interaction between humans and computers using human language. It involves the development of algorithms and models to enable computers to understand, interpret, and generate natural language. NLP has made significant progress in recent years, transforming the way we communicate and interact with technology. This essay explores the concept of NLP, its applications, and its impact on various domains.

Understanding and Processing Language – Language is a complex and dynamic system, characterized by its structure, grammar, and meaning. NLP aims to equip computers with the ability to comprehend and process language similar to humans. Machine learning algorithms, such as recurrent neural networks (RNNs) and transformers, have been instrumental in achieving this goal. These algorithms learn from vast amounts of labeled text data, allowing computers to understand the semantic meaning, syntax, and context of human language.



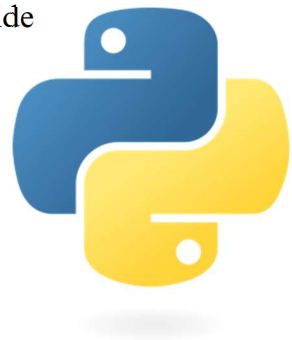
Application of Natural Language Processing –

1. *Text Classification and Sentiment Analysis*: NLP techniques enable computers to classify text into predefined categories or sentiments. Supervised learning algorithms, such as support vector machines (SVMs) and deep learning models, are commonly used for text classification tasks. Sentiment analysis, a specific application of text classification, involves determining the sentiment expressed in text, whether it is positive, negative, or neutral. This capability has proven valuable in areas such as social media monitoring, customer feedback analysis, and brand reputation management.
2. *Information Extraction and Named Entity Recognition*: Extracting relevant information from unstructured text is a fundamental task in NLP. Named Entity Recognition (NER) algorithms identify and classify named entities such as people, organizations, locations, and dates within the text. This enables efficient information retrieval, entity linking, and knowledge extraction from large volumes of textual data. NER has applications in fields such as information retrieval, legal document analysis, news analysis, and biomedical research.
3. *Question Answering and Dialogue Systems*: NLP techniques have paved the way for question-answering systems and dialogue systems that can engage in natural language conversations with humans. Question-answering models analyze questions and provide relevant answers by understanding the intent and extracting information from text sources. Dialogue systems, also known as chatbots or virtual assistants, use NLP to understand and respond to user queries or instructions, offering personalized assistance and support in various domains.

Conclusion – Natural Language Processing has transformed the way humans interact with computers and information. By leveraging machine learning and linguistic analysis, NLP has enabled applications such as text classification, sentiment analysis, machine translation, dialogue systems, and information extraction. However, challenges remain in achieving a deeper understanding of language and ensuring fairness and ethical use. With ongoing research and development, NLP holds the potential to further enhance communication, information retrieval, and human-computer interaction, fostering a world where language is no longer a barrier between humans and technology.

CHAPTER 2: TOOLS

1. ***Python*** **–** Python is a versatile and popular programming language that has gained significant traction in recent years. Known for its simplicity, readability, and extensive range of libraries, Python has become the go-to language for a wide variety of applications, ranging from web development to data analysis and artificial intelligence. In this essay, we will explore the key features and advantages of Python, its use cases in different domains, and its impact on the world of programming.



Advantages of Python:

- λ Presence of third-party modules
- λ Extensive support libraries (NumPy for numerical calculations, Pandas for data analytics, etc.)
- λ Open source and large active community base
- λ Versatile, Easy to read, learn and write
- λ User-friendly data structures
- λ High-level language
- λ Dynamically typed language (No need to mention data type based on the value assigned, it takes data type)
- λ Object-Oriented and Procedural Programming language
- λ Portable and Interactive

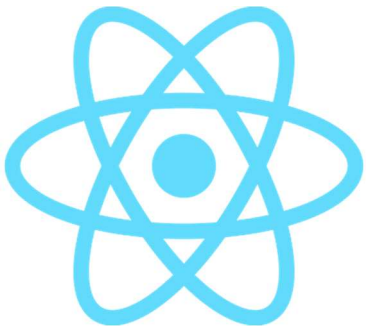
2. ***Flask*** **–** Flask's simplicity, flexibility, and extensibility have made it a popular choice among developers for building web applications. Its minimalistic design, powerful routing and request handling, template rendering capabilities, and seamless integration with other libraries and tools have contributed to its success. Flask's lightweight nature and scalability make it suitable for a wide range of projects, from simple prototypes to complex web

applications. With an active community and ongoing development, Flask continues to evolve and solidify its position as a reliable and efficient web framework in the world of Python and web development.

Features of Flask:

- λ ***Lightweight***: Flask is a lightweight framework because it is independent of external libraries and it gives a quick start for web development having complex applications.
- λ ***Compatible***: Flask is compatible with the latest technologies such as machine learning, agile development, cloud technologies, etc.
- λ ***Independent***: Flask allows full control to the developers for creating web applications. A developer can experiment with the libraries and architecture of the framework.
- λ ***Integrated Unit Testing***: Flask offers an integrated unit testing feature that helps in faster debugging, robust development, and independence to do experiments.
- λ ***Flexible and Scalable***: Flask supports WSGI templates that help in flexibility and scalability in the web development process.
- λ ***Secure Cookies***: Secure cookie is an attribute of an HTTP request that enables the security of channels and ensures no unauthorized person has access to the text. Flask supports the feature of secure cookies.

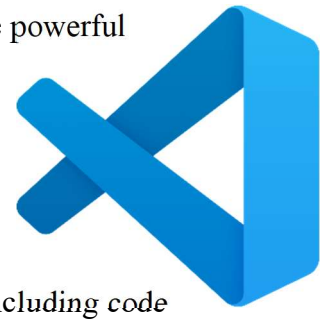
3. ***React –*** React has revolutionized web development by introducing a powerful component-based architecture, efficient rendering with the virtual DOM, and a focus on code reusability and efficiency. Its unidirectional data flow pattern and extensive ecosystem further enhance its capabilities. With the advent of React Native, React has extended its influence to mobile development, making it a versatile choice for creating cross-platform applications. React's impact on the web development landscape continues to grow, empowering developers to build dynamic, interactive, and



performant user interfaces. As React evolves and the community expands, its role in shaping the future of web and mobile development remains undeniable.

4. ***HTML, CSS & JavaScript*** – HTML, CSS, and JavaScript form the foundation of modern web development. HTML provides structure and content presentation, CSS adds visual styling and layout, and JavaScript brings interactivity and dynamic behavior. These three languages work together to create dynamic and visually appealing websites and web applications that deliver an engaging user experience. As technology continues to advance, HTML, CSS, and JavaScript will continue to evolve, ensuring that web developers have the tools and capabilities to create innovative and immersive online experiences.

5. ***Visual Studio Code*** –Visual Studio Code (famously known as VS Code) is a free open-source text editor by Microsoft. VS Code is available for Windows, Linux, and macOS. Although the editor is relatively lightweight, it includes some powerful features that have made VS Code one of the most popular development environment tools in recent times. VS Code supports a wide array of programming languages from Java, C++, and Python to CSS, Go, and Dockerfile. Moreover, VS Code allows you to add on and even create new extensions including code linkers, debuggers, and cloud and web development support.



6. ***Jupyter Notebook*** – Jupyter Notebook is an open-source web application that allows users to create and share documents that contain live code, equations, visualizations, and narrative text. It provides an interactive computational environment, commonly used for data science, data analysis, and machine learning tasks. Jupyter Notebook supports various programming languages, including Python, R, Julia, and Scala, making it a versatile tool for different domains.

the key feature of Jupyter Notebook is its ability to combine code, visualizations, and text in a single document called a notebook. Notebooks are organized into cells, which can contain different types of content. Code cells allow users to write and execute code interactively, displaying the output directly below the cell. This interactive workflow allows for quick prototyping, experimentation, and data exploration.

CHAPTER 3: DOC CHAT

Introduction – Doc Chat is a web-based application that allows you to chat with your document. It is a one-to-one chat with your document. This application is based on natural language processing and machine learning. In this application, you can teach the pdf document to a computer and make a question-answer chatbot. You can also chat with your document. This application is useful for students, teachers, researchers, and professionals.

Advantages of Doc Chat:

1. You can chat with your document.
2. You can teach the pdf document to a computer and make a question-answer chatbot.
3. You can upload multiple documents.
4. You can switch between documents.
5. You can see the entire document.
6. You can see the page of the document which is currently being used to find the answer.
7. You can browse the large number of documents and their models which are created by other users.

Process of the building –

Step 1 - First, we create a Flask application.

Step 2 - We create a file uploading system where the user can upload the pdf file.

Step 3 - We write a Python script for converting the pdf to a vector store.

Step 4 - After that, we write a Python script for creating the model from the vector store.

Step 5 - We create a chat section where the user can chat with the document.

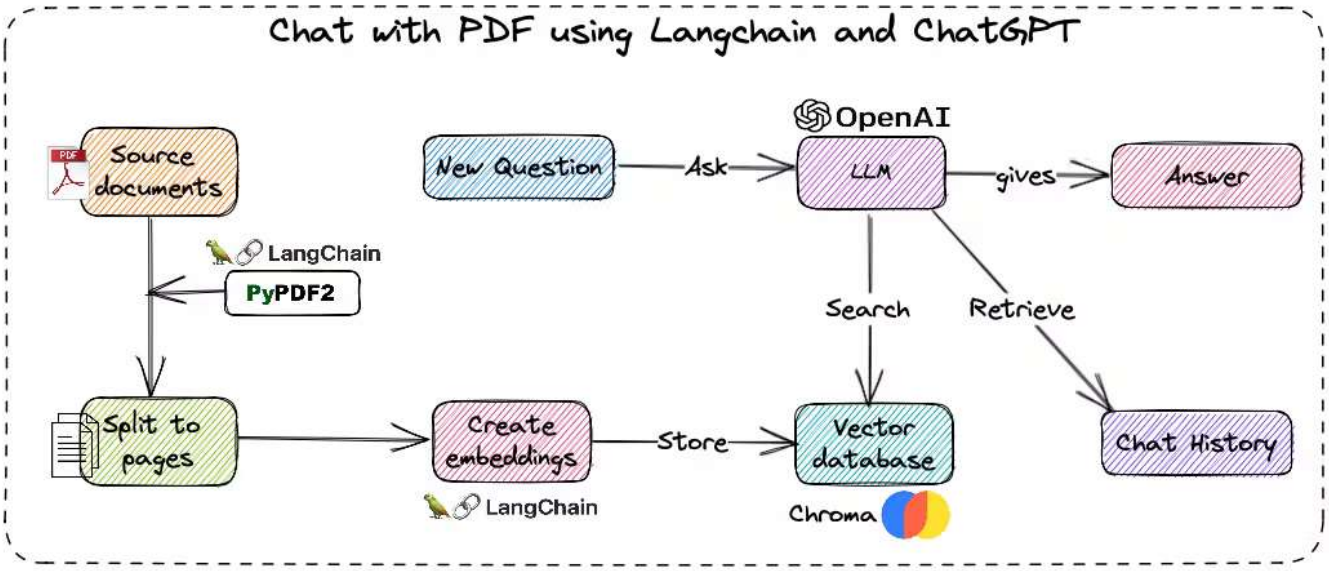
Step 6 - We create the sidebar for the options like upload, chat, and browse.

Step 7 - We create the browse section where the user can browse a large number of documents and their models which are created by other users.

Step 8 - On the browse section, we create the phase where the user can see the current entire document and the page of the document which is currently being used to find the answer.

Step 9 - We made the selection system of a model in the browse section Where the user can select the model and chat with the document in the collection.

Step 10 - We use socket.io for the chat system. Which is a real-time chat system. Socket io is a library that is used for the real-time chat system. When the user sends the message to the server, the server sends the message to the client and the client shows the message to the user. The utility of socket io is that it is a two-way communication system. It is an event-based system. It is a bidirectional system. The user doesn't need to refresh the page for the new message.



Step 11 - We create a minimalistic but interactive UI using HTML, CSS, and JavaScript.

Step 12 - We create a database for storing the user information and the document information.

How to use –

Step 1 - First, you need to open the application.

Step 2 - Then you need to upload the pdf file. The upload Option is available on the sidebar.

Step 3 - You can also select any document from the browse section. The browse option is available on the sidebar.

Step 4 - After uploading the pdf file, you need to wait for the model creation. It will take some time.

Step 5 - After the model creation, you can chat with your document. Chat option is available on the sidebar.

Step 6 - Your uploaded document will be available in the browse section. You can also chat with your document from the browse section. The current document will be selected by default.

Step 7 - You can also browse the current pdf document and the page of the document which is currently being used to find the answer. The option is available on the sidebar.

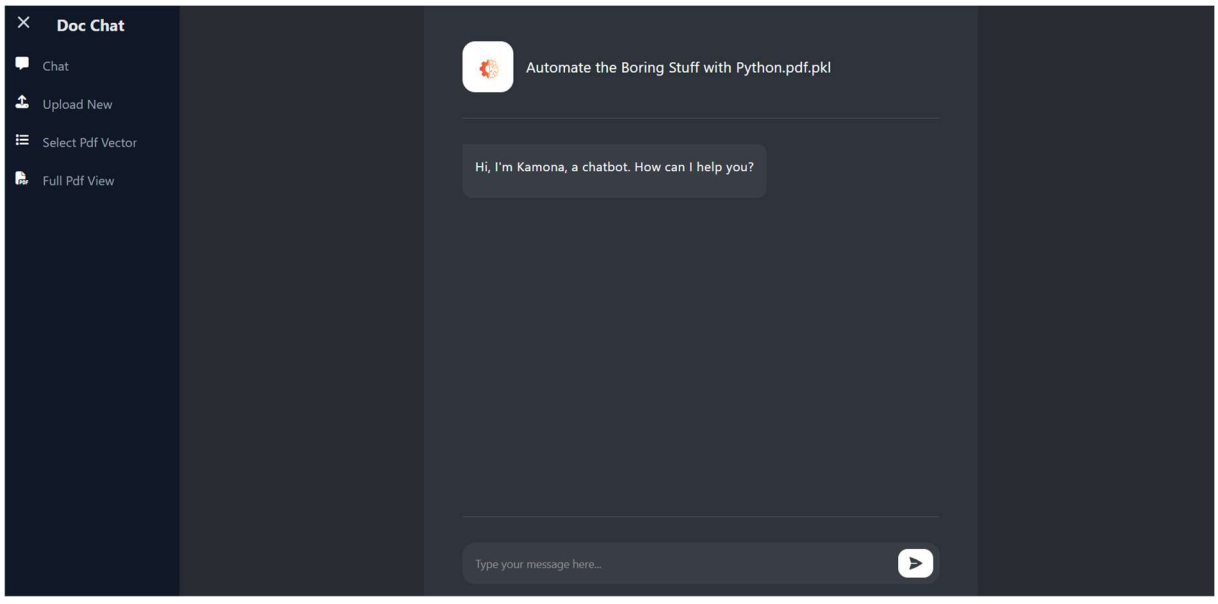
Step 8 - When you search for the answer and the answer is found, the result is shown in the chat section.

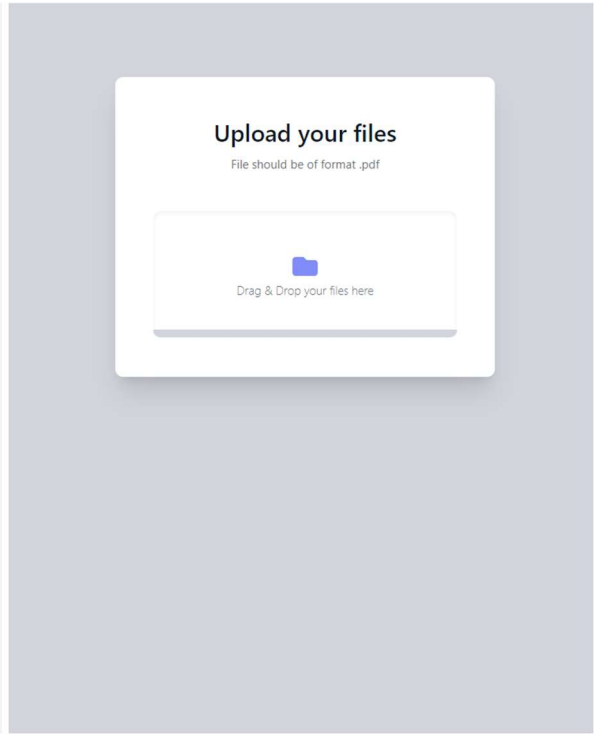
Step 9 - After the result is shown, there is a multiple-page number written at the end of the result. You can click on the page number to see the page of the document which is currently being used to find the answer.

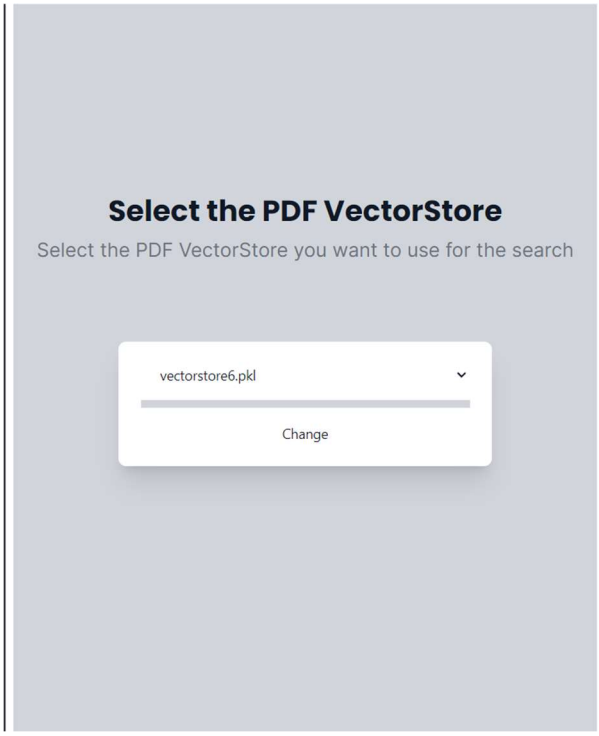
Step 10 - You can chat multiple times with your document.

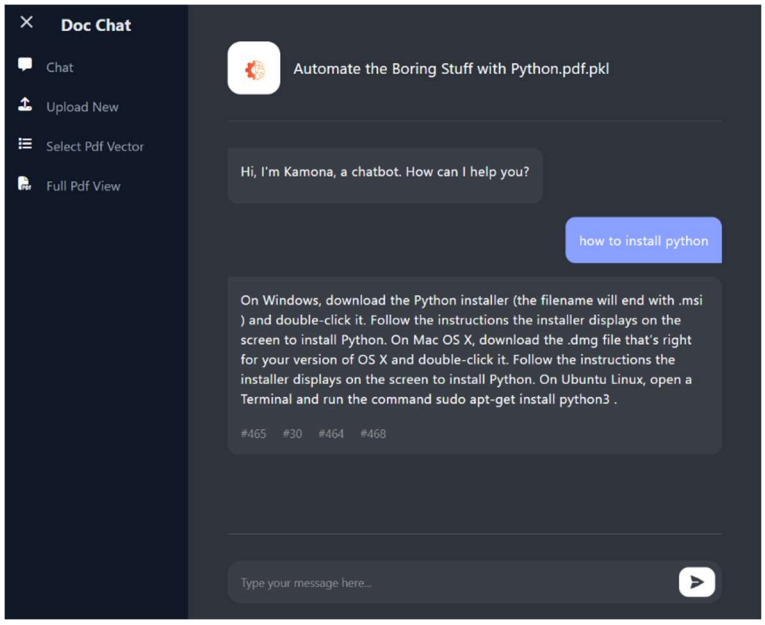
Step 11 - You can also chat with multiple documents by switching the models. The option is available in the browse section.

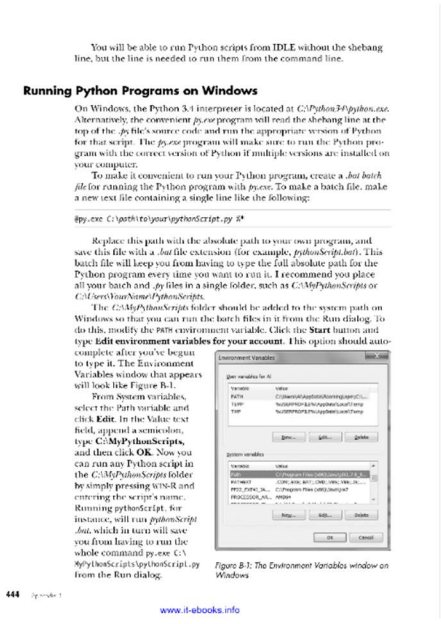
Screenshots –











You will be able to run Python scripts from IDLE without the shebang line, but the line is needed to run them from the command line.

Running Python Programs on Windows

On Windows, the Python 3.4 interpreter is located at *C:\Python34\python.exe*. Alternatively, the convenient *py.exe* program will read the shebang line at the top of the *.py* file's source code and run the appropriate version of Python for that script. The *py.exe* program will make sure to run the Python program with the correct version of Python if multiple versions are installed on your computer.

To make it convenient to run your Python program, create a *.bat batch file* for running the Python program with *py.exe*. To make a batch file, make a new text file containing a single line like the following:

```
@py.exe C:\path\to\your\pythonScript.py %*
```

Replace this path with the absolute path to your own program, and save this file with a *.bat* file extension (for example, *pythonScript.bat*). This batch file will keep you from having to type the full absolute path for the Python program every time you want to run it. I recommend you place all your batch and *.py* files in a single folder, such as *C:\MyPythonScripts* or *C:\Users\YourName\PythonScripts*.

The *C:\MyPythonScripts* folder should be added to the system path on Windows so that you can run the batch files in it from the Run dialog. To do this, modify the PATH environment variable. Click the **Start** button and type **Edit environment variables for your account**. This option should auto-complete after you've begun to type it. The Environment Variables window that appears will look like Figure B-1.

From System variables, select the Path variable and click **Edit**. In the Value text field, append a semicolon, type **C:\MyPythonScripts**, and then click **OK**. Now you can run any Python script in the *C:\MyPythonScripts* folder by simply pressing WIN-R and entering the script's name. Running *pythonScript*, for instance, will run *pythonScript.bat*, which in turn will save you from having to run the whole command *py.exe C:\MyPythonScripts\pythonScript.py* from the Run dialog.

Figure B-1: The Environment Variables window on Windows

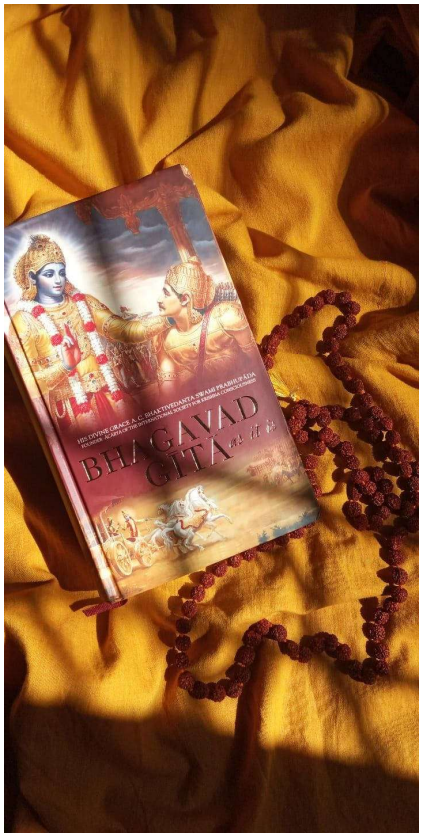
444

www.it-ebooks.info

CHAPTER 4: GITA SAAR

Introduction – Gita Saar is also a web-based application that allows us to ask questions to Shrimad Bhagavat Gita. This application is based on natural language processing and machine learning. In this application, we try to teach the context of Shrimad Bhagavat Gita to a machine learning (natural language processing) model. In this application, you can ask your life problems to Shrimad Bhagavat Gita. The application finds the context of your question and gives you the answer from Shrimad Bhagavat Gita. The model first understands the context of your question and according to slokas and their meaning finds the solution to the problem. This application has a chat section where you can ask the question and answer generated by Shrimad Bhagavat Gita (Krishna Ji).

Q. What is Shrimad Bhagavat Gita?



Ans – The Shrimad Bhagavad Gita, often referred to simply as the Bhagavad Gita or Gita, is a sacred Hindu scripture that is considered one of the most important texts in Indian philosophy and spirituality. It is a 700-verse dialogue between Prince Arjuna and Lord Krishna, who serves as his charioteer and spiritual guide.

The Bhagavad Gita is part of the Indian epic, the Mahabharata, specifically within the Bhishma Parva (Book of Bhishma), and is set in the context of the Kurukshetra War, a great battle between two factions of a royal family. Arjuna, a skilled warrior and one of the Pandava brothers, finds himself overwhelmed and conflicted about fighting against his relatives, teachers, and loved ones on the battlefield.

Arjuna's moral and philosophical dilemma sets the stage for a profound discourse on duty, righteousness, and the nature of life

and death. Krishna imparts spiritual wisdom and guidance to Arjuna, addressing his doubts and helping him understand his role and responsibilities as a warrior and as an individual seeking self-realization.

The Bhagavad Gita has had a profound influence on Hindu philosophy, spirituality, and Indian culture. It is highly regarded as a practical guide for leading a righteous and purposeful life while facing the challenges and complexities of the world. Its teachings have also had a significant impact on other philosophical and religious traditions worldwide.

Process of Building –

Step 1 - First, we create a Flask application

Step 2 - Find the English Translation of the Shrimad Bhagavat Gita

We have a big problem collecting data from the book (Shrimad Bhagavat Gita). The previous model train of the pdf data which divided into the form of pages. But in this case, we don't want to divide the data on the base of pages, we want to divide the content on the base of slokas for that we can train more effectively and find the source of the answer. We need to divide the data into slokas which consist of some details –

- ∇ Slokas (Sanskrit)
- ∇ Chapter
- ∇ Verse
- ∇ Pronunciation
- ∇ Synonyms
- ∇ Translation
- ∇ Meaning 1
- ∇ Meaning 2
- ∇ Meaning 3
- ∇ Meaning 4
- ∇ Meaning 5

We find the solution to the problem by Web Scrapping. For that, we write the Python script with Selenium and BeautifulSoup to scrap the data from the internet (Websites). This makes the

problem almost solved. After scraping the data, we use data analysis tools for Data Cleaning, Data Visualization, and training the model. We use some tools –

- Pandas
- Numpy
- BeautifulSoup
- Lxtm
- Selenium
- Matplotlib
- Sqlite3

Step 3 - We collect the data from the internet using Web Scraping

Step 4 - We clean the data and make it ready for the model

Step 5 - We train the model using the data

Step 6 - We deploy the model on the Flask application

Step 7 - We create a chat section where you can ask the question and answer generated by Shrimad Bhagavat Gita (Krishna Ji).

How to use –

Step 1 - First, Open the application

Step 2 - Then click on the chat section

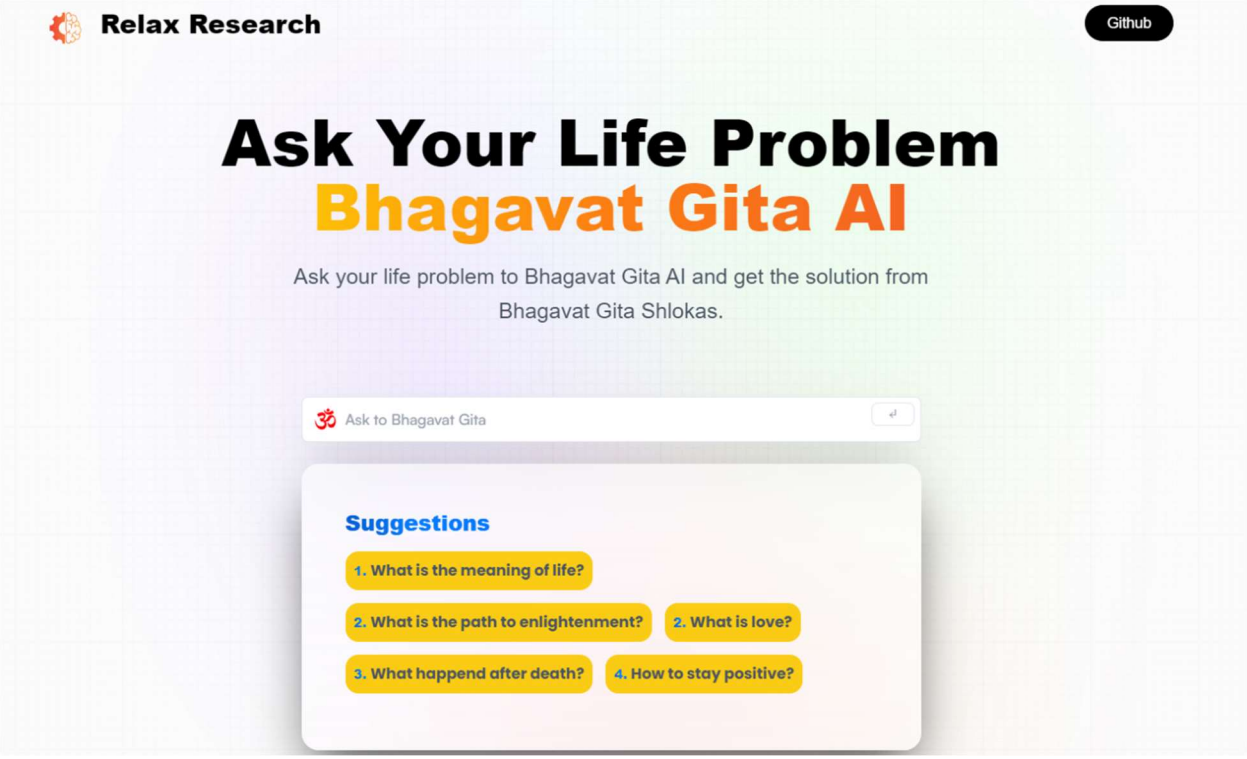
Step 3 - Then ask the question

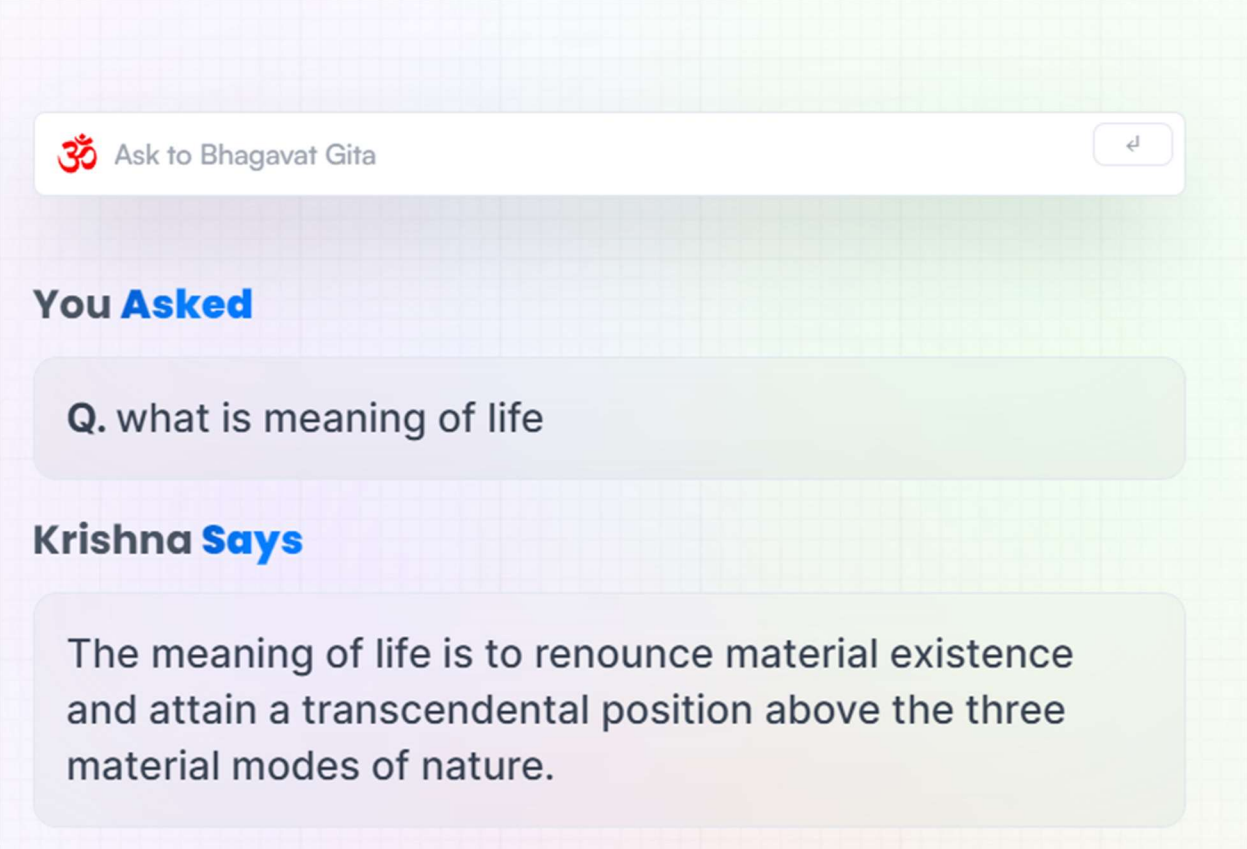
Step 4 - Then click on the submit button

Step 5 - Then you will get the answer from Shrimad Bhagavat Gita (Krishna Ji)

Step 6 - After that, you can ask another question

Screenshots –





Slokes

धूमेनाव्रियते वह्निर्यथादर्शो मलेन च ।यथोल्बेनावृतो गर्भस्तथा तेनेदमावृत्तम् ॥ ३८ ॥

As fire is covered by smoke, as a mirror is covered by dust, or as the embryo is covered by the womb, the living entity is similarly covered by different degrees of this lust.

- Gita 3.38

सन्न्यासस्य महाबाहो तत्त्वमिच्छामि वेदितुम् ।त्यागस्य च हृषीकेश पृथक्केशिनिषूदन ॥ १ ॥

Arjuna said: O mighty-armed one, I wish to understand the purpose of renunciation [tyāga] and of the renounced order of life [sannyāsa], O killer of the Keśī demon, master of the senses.

- Gita 18.1

CHAPTER 5: IMAGE GENERATOR

Introduction – Image Generator is also a web-based application that generates images based on user prompts. You need to just write a prompt and it will generate the image based on that prompt. The machine learning model generates the image according to your text. This application is not about searching images across the internet. The image which is generated by the model is new and unique. You can also download the image which is generated by the model. You can also upload the image on the community page where other users can see the generated images by you and you also can see images generated by other users and download them.

Our React application harnesses the incredible capabilities of DALLE to provide you with a user-friendly interface where you can unleash your imagination and bring your ideas to life. Whether you're an artist seeking inspiration, a designer in need of visual concepts, or simply someone looking to explore the boundaries of creativity, our Image Generator Flask Application is the perfect tool for you.

Advantage of Image Generator –

- You can generate the image based on your prompt.
- You can download the image which is generated by the model.
- You can upload the image to the community page.
- You can see the images uploaded by other users on the community page and can download them.
- You can regenerate the image by clicking on the generate button.
- You can generate the images for your work -
  - You can generate the images for your blog.
  - You can generate the images for your website.
  - You can generate the images for your YouTube channel.
  - You can generate the images for your social media posts.
  - You can generate the images for your projects.
  - You can generate the images for your presentation.
  - You can generate the images for your assignment.

- » You can generate the images for your research paper.
- » You can generate the images for your thesis.
- » You can generate the images for your book.

ℵ and many more.

Process of Building –

Step 1 - First, we create a Node Express js server to handle the requests from the React application.

Step 2 - Then, we create a React application to handle the user interface.

Step 3 - We use the DALLE model to generate images.

Step 4 - We use Rest API to communicate between the React application and the Node Express js server.

Step 5 - We use the React Router to navigate between the pages.

Step 6 - We create a form to take the user input and send it to the server.

Step 7 - The server then sends the request to the DALLE model and the model generates the image.

Step 8 - The server then sends the image to the cloudignary server and the cloudignary server stores the image and sends the URL to the server.

Step 9 - The server then sends the URL to the database and the database stores the URL and sends the request id to the server.

Step 10 - The server returns the API response to the React application and the React application displays the image to the user.

Step 11 - The user can download the image and can also upload the image to the community page.

Step 12 - The user can also see the images uploaded by other users on the community page and can download them.

Step 13 - The community page fetches all the images from the database and displays them to the user.

How to use –

Step 1 - First, you need to open the application.

Step 2 - You came to the home screen where you can see the posts of other users who use this application to generate the image.

Step 3 - On the image tab, You can see the prompt to generate the image and the name of the user who generates it.

Step 4 - You can download images to your local files.

Step 5 - You can navigate to Create a Page using the Create button on the right corner.

Step 6 - The page is a simple form. You just need to fill it out.

Step 7 - Add the prompt on the prompt section/input field and click on generate button.

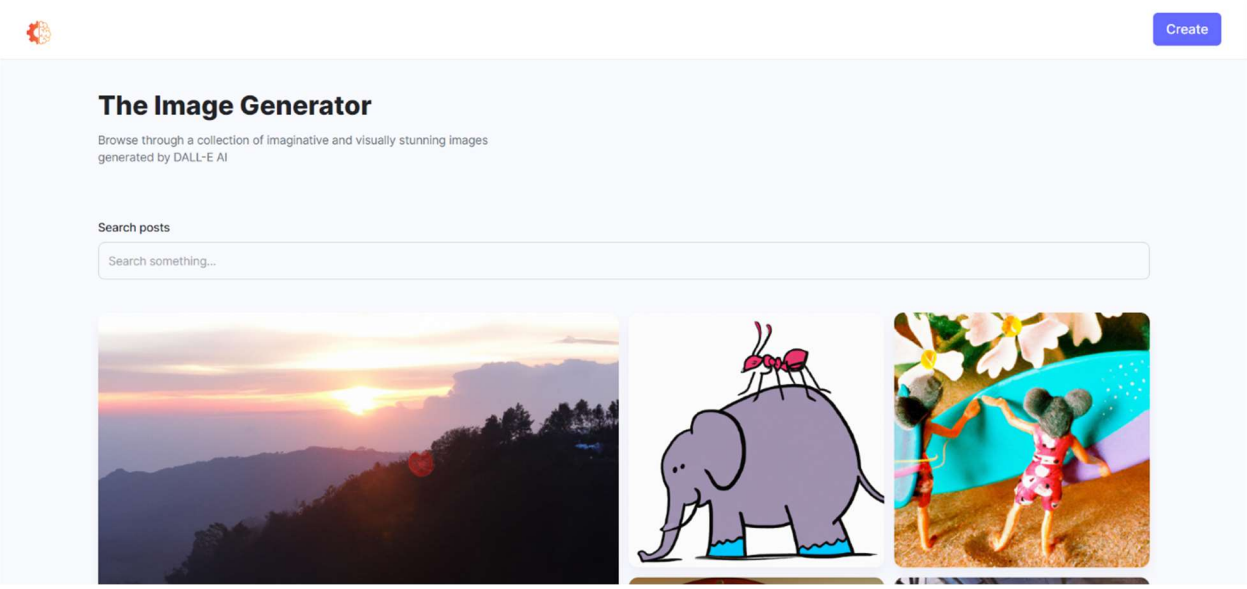
Step 8 - After some time, loading, the image is generated.

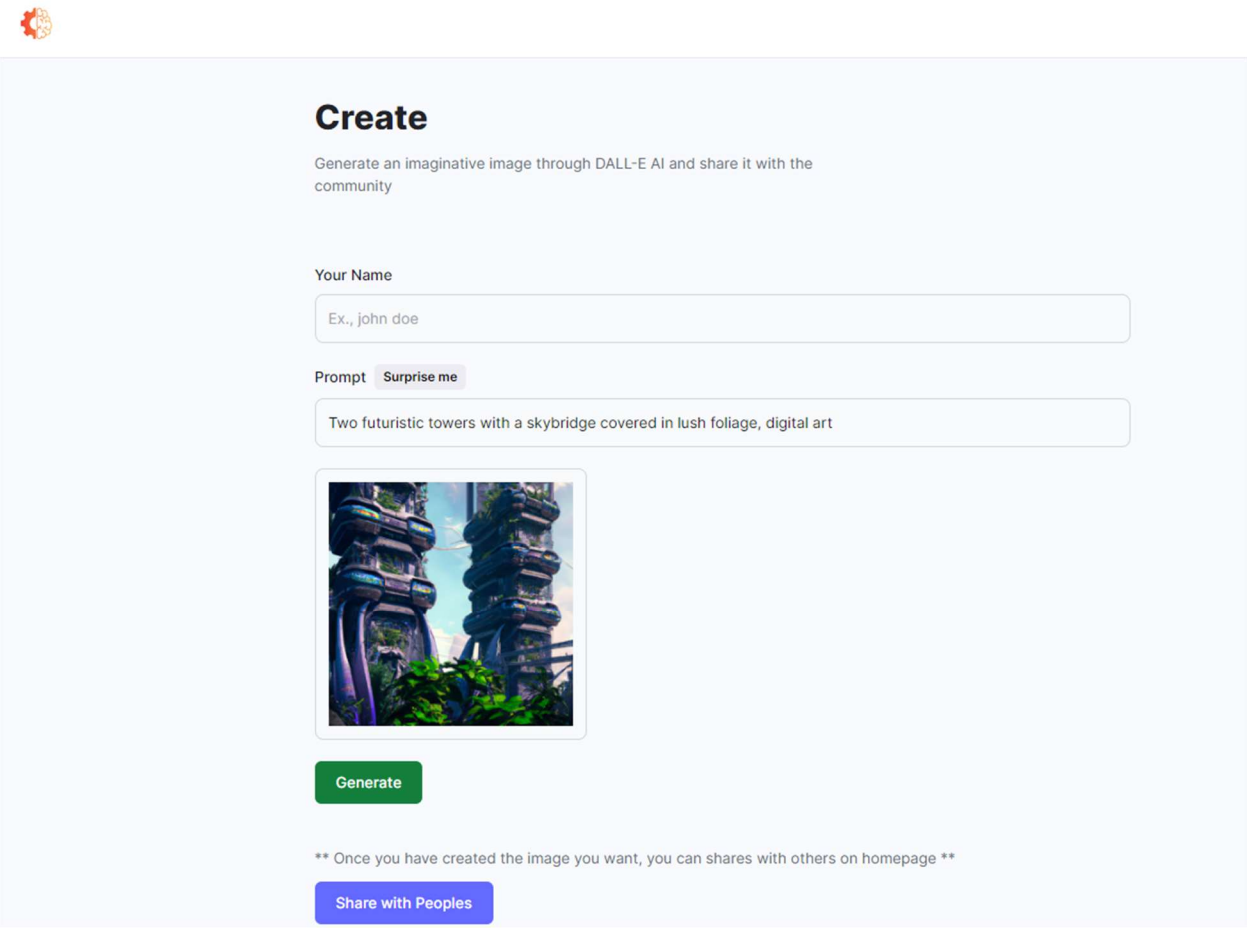
Step 9 - For uploading the image in the community, fill in your name in the upper input element and click the button share.

Step 10 - After clicking the button, the Page redirect to the home page, and your image is uploaded with the prompt and your name.

Step 11 - You can download that image and share the link of an image with your friends or family.

Screenshots –


Create
The Image Generator
Browse through a collection of imaginative and visually stunning images generated by DALL-E AI
Search posts
Search something...


Create
Generate an imaginative image through DALL-E AI and share it with the community
Your Name
Ex., john doe
Prompt
Surprise me
Two futuristic towers with a skybridge covered in lush foliage, digital art
Generate
** Once you have created the image you want, you can shares with others on homepage **
Share with Peoples

UNIT - 4: CONCLUSION

In conclusion, creating an application can be a challenging but rewarding endeavor. It requires a solid understanding of multiple skills such as –

- Python
- Flask
- Machine Learning
- Natural Language Processing
- Neural Network
- HTML, CSS, and JavaScript
- Socket IO
- Git
- Vector Store

By building this project, we gain hands-on experience in developing a complex artificial intelligence application. This project taught us how things work in natural language processing and how to implement model training. How you can implement models into web-based services. How the application layers work.

Additionally, the project can be a great opportunity to learn about new technologies and trends in

Machine learning, natural language processing, and Flask web development. It can also

be an opportunity for developers to explore different ways of building machine learning models and use the existing models.

Finally, it should be highlighted that building this trinity of projects is not only to show what can machine learning do but also to think about what new things can be done in further research and what is the possibility to solve real-life problems using machine learning and artificial intelligence. These types of things can make a major change in your life. It's also important to keep in mind that the development process is only half of the story and the

the project will also need to be properly tested, deployed, and maintained to be successful.

UNIT – 5: REFERENCES

Python Documentation - https://www.python.org/

LangChain Documentation - https://python.langchain.com/en/latest/

Flask Documentation - https://flask.palletsprojects.com/en/2.3.x/

OPENAI Documentation – https://platform.openai.com/docs/introduction

Gita DataSources – https://vedabase.io/en/library/bg/

Cloudinary Documentation - https://cloudinary.com/documentation

React Documentation - https://react.dev/learn