Case Study Solution

Text Preprocessing:-

Cleaning of Text:-

- Lowercase
- Remove text in square brackets
- Remove URLs
- Remove HTML tags
- Remove non-alphabetic characters
- Remove extra whitespaces

Stopwords:-

Stop words are common words that may not add significant meaning to the text.

Lemmatization and Stemming:-

Stemming:- Reducing words to their base or root form by removing suffixes.
Example:
Input: "running", "runner", "ran"
Output: "run"

Lemmatization:- Converting words to their base or dictionary form using linguistic rules.
Example:
Input: "better"
Output: "good"

Choosing Between Lemmatization and Stemming:-

- Use stemming if you need a faster, less resource-intensive process and can tolerate some inaccuracies.
- Use lemmatization if you need more accurate results and can afford the extra computational cost.

Reasons for not applying Lemmatization and stemming together:-
Applying both lemmatization and stemming is generally unnecessary because they serve the same fundamental purpose: reducing words to a more fundamental form.
Stemming cuts off word endings in a crude way, often resulting in non-words. Since lemmatization already transforms words to their meaningful base form, additional stemming does not add significant value and may even detract from the accuracy.

Part-of-Speech (POS) Tagging:-

Assigning parts of speech to each word in a sentence (e.g., noun, verb, adjective).
Example:
Input: "The quick brown fox jumps over the lazy dog."
Output: [("The", "DT"), ("quick", "JJ"), ("brown", "JJ"), ("fox", "NN"), ("jumps", "VBZ"), ("over", "IN"), ("the", "DT"), ("lazy", "JJ"), ("dog", "NN")]

Named Entity Recognition (NER):-

Identifying and classifying entities in the text (e.g., names of people, organizations, dates).
Example:
Input: "Barack Obama was the 44th President of the United States."
Output: [("Barack Obama", "PERSON"), ("44th", "ORDINAL"), ("President", "TITLE"), ("United States", "LOCATION")]

Tokenization:-

Splitting text into smaller units called tokens (words, phrases, symbols).
Example:
Input: "Natural language processing is fun."
Output: ["Natural", "language", "processing", "is", "fun", "."]

Text Normalization:-

Converting text to a canonical form, such as expanding contractions (e.g., "isn't" to "is not") and standardizing spelling variations.
Example:
Input: "I've got to go."
Output: "I have got to go."

Bag of Words (BoW):

Represents text as a collection of word frequencies.

TF-IDF (Term Frequency-Inverse Document Frequency):

Measures the importance of a word in a document relative to a collection of documents.

Word Embeddings:

Dense vector representations of words (e.g., Word2Vec, GloVe, BERT).

Visualization:-

Size of Dataset:-

56402

Correlation Coefficient:-

```
                 question_length  answer_length
question_length         1.000000      -0.031733
answer_length          -0.031733       1.000000
```

The correlation coefficient ranges from -1 to 1.

- **1** indicates a perfect positive correlation.
- **-1** indicates a perfect negative correlation.
- **0** indicates no correlation.

question_length to question_length: The value is **1.000000**, which indicates a perfect positive correlation (as expected, since any variable is perfectly correlated with itself).

answer_length to answer_length: Similarly, the value is **1.000000**, indicating a perfect positive correlation with itself.

question_length to answer_length: The value is **-0.031733**, which indicates a very weak negative correlation between `question_length` and `answer_length`. This means that as the length of the question increases, the length of the answer tends to decrease slightly, but the relationship is very weak.

In summary, the correlation matrix shows that there is almost no linear relationship between the lengths of the questions and the answers in your dataset. The lengths of questions and answers do not seem to be significantly related based on this data.

LDA:-

LDA is a topic modeling technique used to discover the abstract topics that occur in a collection of documents.

```
Topic #0:
['people', 'world', 'facts', 'true', 'best', 'look', 'india', 'life',
'does', 'like']
```

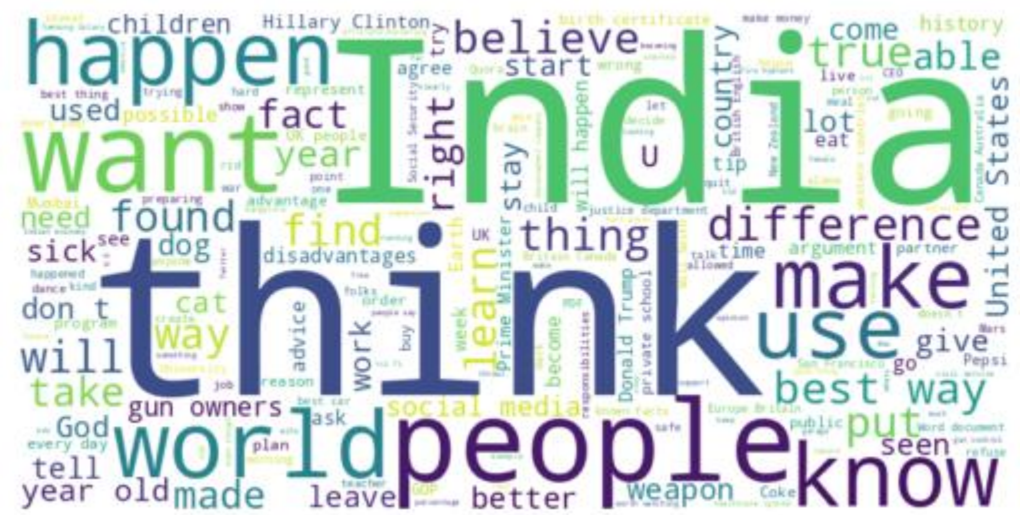
This topic seems to revolve around general concepts about people, the world, and facts, with a focus on India and life.

```
Topic #2:
['big', 'think', 'learn', 'trump', 'use', 'world', 'people', 'want',
'indian', 'does']
```

Interpretation: This topic might be centered around significant issues or events, learning, and opinions, with a mention of 'Trump' indicating political discussions.

Wordcloud:-

A word cloud is a visual representation of text data where the size of each word indicates its frequency or importance in the dataset. It's a useful tool for quickly understanding the most prominent terms and themes in your text data.



Implementation of various ML Model:-

Bert:-

Stands for Bidirectional Encoder Representations from Transformers. It leverages the transformer architecture, particularly focusing on the bidirectional aspect, meaning it reads the text in both directions (left-to-right and right-to-left) to better understand the context of a word in relation to its surroundings. For question-answer tasks, BERT is fine-tuned on specific datasets where it learns to predict the start and end positions of the answer span within a passage. This fine-tuning allows BERT to excel in extracting precise answers from a given context.

Bert Models specific for Question and answer task:-

1. Bert-Base
2. Bert-Large
3. Roberta
4. DistilBert

Transformers:-

Forms the backbone of both BERT and GPT, providing a flexible and powerful architecture for a range of NLP tasks.
Transformers rely on a mechanism called self-attention, which allows the model to weigh the importance of different words in a sentence regardless of their position. This ability to attend to all parts of the input sequence simultaneously, rather than sequentially as in traditional RNNs, enables transformers to capture long-range dependencies and context more effectively

Transformer models for Question and Answer task:-

1. T5 (Text to text T5 Transformer)
2. Alberta

GPT:-

stands for Generative Pre-trained Transformer. Unlike BERT, which is designed primarily for understanding and extracting information, GPT focuses on text generation. GPT models are trained using a large corpus of text in an unsupervised manner and then fine-tuned for specific tasks. In the context of question-answer tasks, GPT can be fine-tuned to generate coherent and contextually relevant answers based on the input question.

Improvements:-

Datasets:-

We can format the dataset in certain way by providing context and put in JSON format.

Each such dictionary contains two attributes, the "context" and "qas".

- context: The paragraph or text from which the question is asked.
- qas: A list of questions and answers (format below).

Questions and answers are represented as dictionaries. Each dictionary in qas has the following format.

- id: (string) A unique ID for the question. Should be unique across the entire dataset.
- question: (string) A question.
- is_impossible: (bool) Indicates whether the question can be answered correctly from the context.
- answers: (list) The list of correct answers to the question

A single answer is represented by a dictionary with the following attributes.

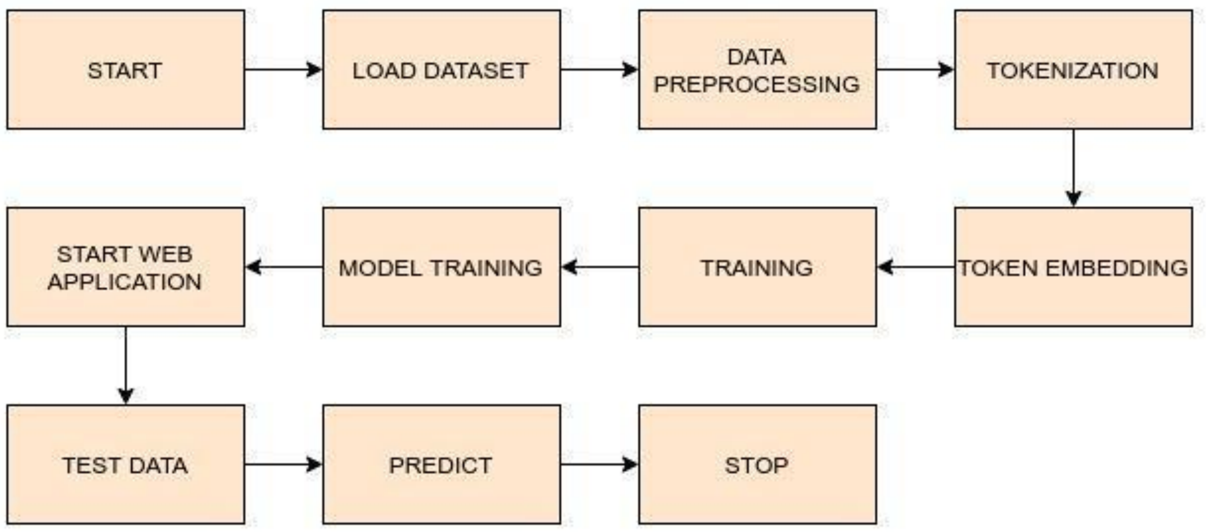
- text: (string) The answer to the question. Must be a substring of the context.
- answer_start: (int) Starting index of the answer in the context.

Here is the sample format:-

```
train_data = [
  {
    "context": "Mistborn is a series of epic fantasy novels written by American author Brandon
Sanderson.",
    "qas": [
      {
        "id": "00001",
        "is_impossible": False,
        "question": "Who is the author of the Mistborn series?",
        "answers": [
          {
            "text": "Brandon Sanderson",
            "answer_start": 71,
          }
        ],
      }
    ],
  },
  {
    "context": "The first series, published between 2006 and 2008, consists of The Final
Empire,"
          "The Well of Ascension, and The Hero of Ages.",
    "qas": [
      {
        "id": "00002",
        "is_impossible": False,
                "question": "When was the series published?",
                "answers": [
                    {
                        "text": "between 2006 and 2008",
                        "answer_start": 28,
                    }
                ],
            },
            {
                "id": "00003",
                "is_impossible": False,
                "question": "What are the three books in the series?",
                "answers": [
                    {
                        "text": "The Final Empire, The Well of Ascension, and The Hero of Ages",
                        "answer_start": 63,
                    }
                ],
            },
            {
                "id": "00004",
                "is_impossible": True,
                "question": "Who is the main character in the series?",
                "answers": [],
            },
        ],
    },
]
```

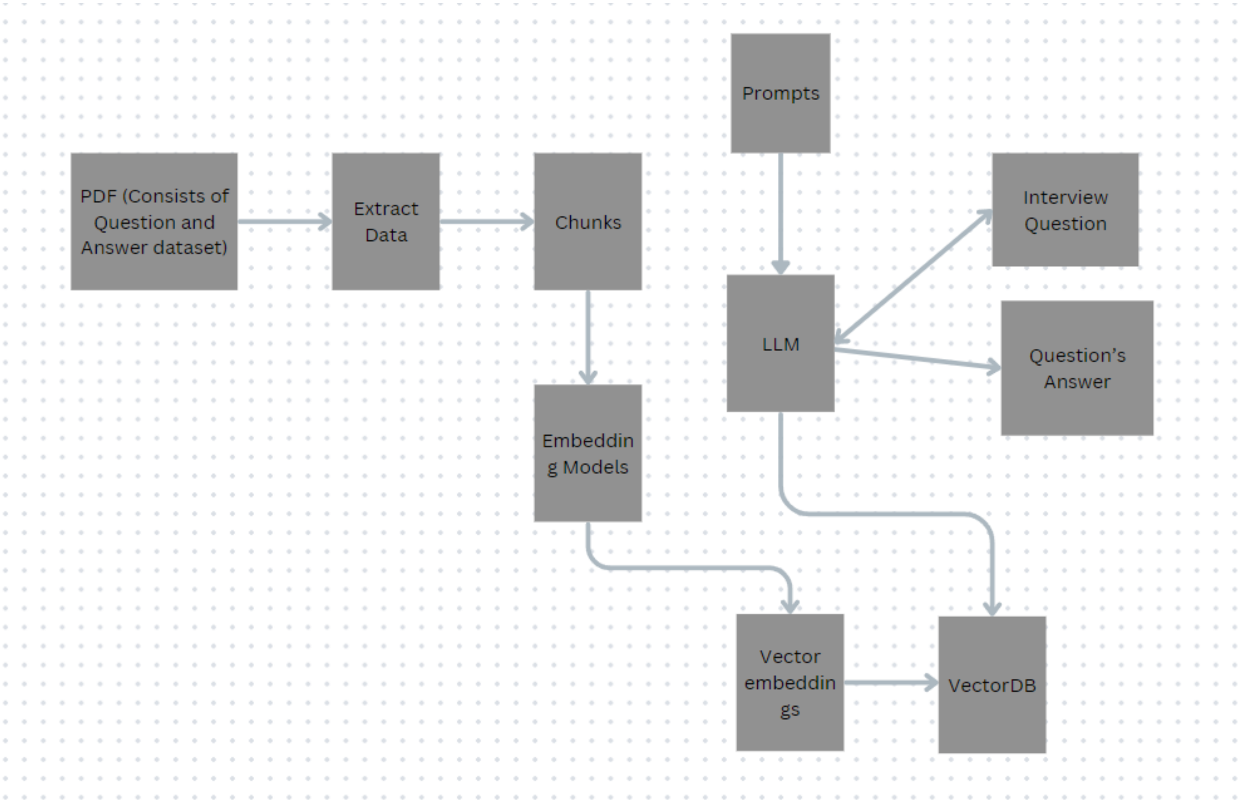
Web Application:-

We can use it for Question and Answer application. Here is the Work flow diagram of that.



GenAI Application

We can also use it for chatbot task using GenAI. Here is the workflow diagram of that


Prompts
PDF (Consists of Question and Answer dataset)
Extract Data
Chunks
Interview Question
LLM
Question's Answer
Embedding Models
Vector embeddings
VectorDB