RULES AND REGULATIONS

Question 1: Building an AI Conversational Agent

Question 2: Designing a Corporate Query Ticketing System

Prizes:

- Opportunity for an Internship
- Cash Prize pool up to ₹10000
- Chance for Best Performance Team to Win Amazon Voucher

Participant Options:

- **Attend Question 1**

  Or

- **Attend Question 2**

Recommendation:

- Attend Question 1:
  - Best performance team may receive an Amazon voucher.
- If time permits, consider attending both questions for a broader experience.

Submission Guideline:

- **GitHub Repo:** Upload your code to Github. Create an account if you don't have one. Make sure your repository is **public**. Make the folder organization as follows:
  - Web Interface:(Q1 or Q2)
    - All the related files
  - Chatbot:(Q1)
    - All the related files
  - Report:(Q1)
    - Report file (in PDF/PPT)
- Also, provide a video demo of less than 50 MB.(Q1 or Q2)
- Share the GitHub repo link on UnStop.

PROBLEM STATEMENT 1

Building an AI Conversational Agent

Overview:

We're looking for talented developers to create a smart AI chatbot capable of understanding, summarizing, and answering questions based on the law (name of law) and cases (enclosed).

What You Need to Do:

Your task involves developing an AI-powered chatbot that can read through the given content and then respond to user queries about that context. The data to train the model can be found in the link given: https://www.kaggle.com/datasets/braindeadiiest/chatbot-data-for-legalcare. The dataset mainly consists of Partnership Act legal documentation and a few case files that deal with the Partnership Act. Since the data provided is in the form of unstructured text, your approach to measuring accuracy will require manual annotation and careful evaluation.

- **Manual Annotation:** Annotate the data with questions and correct answers based on the law and provided cases. To serve as ground truth.
- **Evaluation Dataset:** Train the model using the dataset. Prepare one test data, consisting of proper Acts and their outcomes based on the case files and the legal Partnership Act documentation, and check the model accuracy, i.e. how well the model answers the query from the users.
- **Evaluation Metrics:** Use appropriate qualitative assessments or semantic similarity measures to evaluate relevance and correctness of chatbot responses, as traditional metrics may not apply due to unstructured data.
- **Human Evaluation:** Human evaluators compare chatbot responses with expected answers to provide accurate feedback.
- **Confidence Scores:** Utilize provided confidence scores (if available) as an additional measure of accuracy, with higher scores indicating greater likelihood of accuracy.

N.B.: The model should not be generating summaries of the case files or any legal documentation that have been used to train the model. It should be able to provide crisp and correct outcome to help users in lawsuit.

Constraints:

- Use only Open-Source models (e.g., Llama2) with up to 7 billion parameters or create your own models from scratch.
- Avoid using expensive commercial AI models like GPT-4, Claude, or Gemini.
- You have the option to refine your model using Reinforcement Learning from Human Feedback (RLHF).
- Keep an eye on minimizing the time and resources needed like GPU to train and update your model.

What We Expect:
Your final submission should include three main components:

- **A User-Friendly Interface:** Design a simple web interface where users can type in questions and add relevant documents (PDFs, Word files, plain text, or HTML) that the chatbot can learn from.
- **An Informative Dashboard:** Create a dashboard that shows different metrics related to the chatbot's performance, like how accurate its responses are, how quickly it responds to queries, and any trends or patterns in user interactions, such as number of queries having certain keywords.
- **A Brief Report:** Create a report in either PPT/PDF format, that will consist of model pipeline, methodologies, results and analysis, along with the features and intractability of the web interface. Give the entire executable code for both web interface and chatbot application. Also attach with it the environment setup requirements for both web and chatbot.
- **GitHub Repo:** Upload your code to Github. Create an account if you don't have one. Make sure your repository is **public**. Make the folder organization as follows:
    - Web Interface:
        - All the related files
    - Chatbot:
        - All the related files
    - Report:
        - Report file (in PDF/PPT)
- Also, provide a video demo of less than 50 MB.
- Share the GitHub repo link on UnStop.

How We'll Evaluate Your Work:
We'll be looking at three key criteria:

Response Time: Aim for your chatbot to give its initial response in under 3 seconds.
Accuracy and Confidence: Aim for an accuracy or confidence score between 70% and 90%.
Summary Length: Keep your chatbot's responses brief, with summaries under 5,000 characters.
Ready to Take on the Challenge?
N.B.
If you're up for it, start by familiarizing yourself with the requirements and get ready to show off your coding skills! Remember, this challenge isn't just about building a chatbot – it's about creating an intelligent, efficient, and user-friendly tool that can help people find the information they need quickly and accurately. Good luck!