|  |  |
| --- | --- |
|  |  |
|  |
|  |

**High Impact Skills Development Program for Gilgit Baltistan**

**Natural Language Processing Module Project**

**Project Title:** Design and Development of Topical Chatbot

**Learning Objectives:**

* Understand the fundamentals of Transformer-based encoder-decoder architecture for conversational chatbot
* Learn how to use Tensorflow/Pytorch/Huggingface
* Learn how to use Google Colab
* Learn how to train and fine-tune a Transformer

**Overview:**

A topical chatbot is a specialized AI designed for conversations about a specific subject. Unlike general chatbots, these focus on one topic, like medicine or finance. They're trained to understand and discuss that area, using datasets of relevant conversations. Topical chatbots provide accurate and helpful information within their domain, enhancing user engagement and knowledge in that subject. However, they may struggle with topics beyond their training and lack broader conversational abilities.

**Dataset:**

The dataset used for this project is Topical Chat dataset from Amazon. It consists of over 8000 conversations and over 184000 messages. Within each message, there is: A conversation id, which is basically which conversation the message takes place in. Each message is either the start of a conversation or a reply from the previous message. There is also a sentiment, which represents the emotion that the person who sent the message is feeling. There are 8 sentiments: Angry, Curious to Dive Deeper, Disguised, Fearful, Happy, Sad, and Surprised.

This dataset can be used in machine learning to simulate a conversation or to make a chatbot. It can also be used for data visualization, for example you could visualize the word usage for the different emotions. The dataset can be downloaded from the link below.

<https://github.com/alexa/Topical-Chat>

Further detail of the dataset is also given on the link above.

The project can be divided into several stages:

**Preprocessing:** Apply necessary processing techniques on the text dataset suitable for this task.

**Training:** The dataset is already split into various splits. Use a deep learning framework such as TensorFlow, Pytorch, or Huggingface to train a Transformer on the annotated dataset. You can either create a customized model or employ a pre-trained network (either fine-tuned or as feature extractor).

**Evaluation:** Evaluate the performance of the model on the evaluation set. The performance of a chatbot is measured using both objective (quantitative) measures and subjective (qualitative) measures. You can pick relevant objective and subjective performance metrics for your project. Some of the performance measures used for chatbot performance analysis are given below.

1. Response Accuracy: It measures how accurate the chatbot's responses are in providing correct and relevant information to user queries.
2. Precision and Recall: Precision calculates the percentage of correct positive predictions among all positive predictions made by the chatbot. Recall calculates the percentage of correct positive predictions among all actual positive instances. These metrics help assess the chatbot's ability to correctly identify relevant information.
3. F1 Score: The F1 score is the harmonic mean of precision and recall. It provides a balanced measure of the chatbot's accuracy, especially when precision and recall have different levels of importance.
4. User Satisfaction: Collecting user feedback through surveys or ratings can help measure how satisfied users are with the chatbot's responses and interactions. Net Promoter Score (NPS) is one commonly used metric for this purpose.
5. Engagement: This metric assesses how effectively the chatbot keeps users engaged in the conversation. It includes metrics like session duration, average turn length, and the number of interactions per session.
6. Completion Rate: Measures the percentage of interactions or sessions that the chatbot successfully completes without user abandonment.
7. Fallback Rate: Fallbacks occur when the chatbot can't understand or respond appropriately to a user query and has to redirect the user to a human agent or other resource. Monitoring the frequency of fallbacks can help identify areas for improvement in the chatbot's capabilities.
8. Churn Rate: Churn rate indicates the percentage of users who stop engaging with the chatbot after an initial interaction. A high churn rate may suggest dissatisfaction or ineffective communication.
9. Human Handoff Rate: In cases where chatbots are integrated with human agents, this metric evaluates the percentage of interactions that require escalation to a human.
10. Sentiment Analysis: Assessing user sentiment in interactions with the chatbot can provide insights into whether users are having positive, neutral, or negative experiences.
11. Task Completion Rate: Measures the percentage of user tasks or goals that the chatbot successfully assists with or completes.

**Testing:** The model should be tested on a separate test set of conversations that was not used for training, to ensure that it can generalize to new data.

**Requirements:**

1. Create account on GitHub
2. Share your profile link in the project report (to be submitted at the end of your project)
3. Upload your code to GitHub
4. Prepare a short 3-4-page report about your project that should contain:
   1. Project Title
   2. Your name, email address, github profile link
   3. A 250-word abstract of your project
   4. Project Details

* Overview of the problem and potential application areas
* A brief literature review (refer at least 4 articles from 2022-23) - highlight the

work, data, accuracy reported, merits and demerits.

* Model used (architecture, diagram, main components, parameters)
* Dataset used in the project -stats, data division (training, validation, test)
* Hyperparameter tuning
* Results and Evaluations
* Analysis of results-What are good results? What are bad results and why? Example conversations.
* How can you further improve the results?

5. Prepare a 5-slide presentation of your project and the work you have done.

6. Upload your project report and slides to the respective folders in LMS.