

Documentation: Chatbot System

1. Introduction:

This documentation provides an overview of a conversational AI chatbot that offers real-time assistance and engages with users in various contexts. The chatbot leverages **Natural Language Processing (NLP)** models and is deployed using **FastAPI** to ensure real-time interactions with users.

2. Data Collection and Preprocessing:

- **Data Sources:** Conversation datasets, FAQs, or domain-specific data.
- **Preprocessing Tasks:**
 - Tokenization, removal of stopwords, and text normalization using **NLTK** or **spaCy**.
 - Sentence embedding generation using **BERT** or **word2vec**.
 - Handling out-of-vocabulary (OOV) tokens and ensuring efficient intent classification.

Example Pipeline:

- Sklearn pipelines used to streamline the preprocessing and classification tasks.

Based on the structure of the documentation and proposal you provided for the *Personalized Movie Recommendation System*, here is a matching outline for a new chatbot project.

Documentation: Chatbot System

1. Introduction:

This documentation provides an overview of a conversational AI chatbot that offers real-time assistance and engages with users in various contexts. The chatbot leverages **Natural Language Processing (NLP)** models and is deployed using **FastAPI** to ensure real-time interactions with users.

2. Data Collection and Preprocessing:

- **Data Sources:** Conversation datasets, FAQs, or domain-specific data.
- **Preprocessing Tasks:**
 - Tokenization, removal of stopwords, and text normalization using **NLTK** or **spaCy**.
 - Sentence embedding generation using **BERT** or **word2vec**.
 - Handling out-of-vocabulary (OOV) tokens and ensuring efficient intent classification.

Example Pipeline:

- Sklearn pipelines used to streamline the preprocessing and classification tasks.

3. Chatbot Models:

- **Intent Recognition:** Uses deep learning models (e.g., RNN, Transformer models) to identify the user's intent.
- **Response Generation:** Combines **pre-trained models** (e.g., GPT-based architectures) and **retrieval-based responses** for accurate conversations.

4. Model Evaluation and Optimization:

- **Metrics for Evaluation:**
 - Intent classification: **Accuracy, Precision, Recall**.
 - Response generation: **BLEU** and **ROUGE** scores.
- **Optimization Techniques:**
 - Hyperparameter tuning and iterative model improvement based on test conversations.

5. API Integration:

- **FastAPI Endpoints:**
 - **/health-check:** Confirms the server is operational.
 - **/chat:** Accepts user input and returns a chatbot response in real-time.

6. Test Samples:

Include sample interactions showcasing the chatbot's ability to understand various intents and generate accurate responses.

7. Conclusion:

This chatbot system applies advanced NLP techniques to engage with users effectively. By integrating with **FastAPI**, the system can provide seamless and interactive conversations, meeting user needs in real-time.