Executive Summary: Implementation of a Chatbot using Jupyter Python

Babu Pallam (P2849288)

November 14, 2024

Instructor: Dr. Aboozar Taherkhani
MSc. Artificial Intelligence
CSIP5103 – Neural Systems and NLP

Executive Summary

Background & Purpose:

This project aimed to build a chatbot by exploring different AI and non-AI methods. Chatbots are useful in applications like customer service and virtual assistants. The project used RNN(Recurrent Neural Network)s with Luong attention, CNN(Convolutional Neural Network)s, MLP(Multilayer Perceptron)s, transformer models, and rule-based bots. Each model was evaluated with BLEU(Bilingual Evaluation Understudy) scores and cosine similarity to check how well they matched reference responses. This study helps identify effective chatbot designs that are scalable and adaptable.

Data Processing:

Data preprocessing was essential for consistency and clarity. Text was converted to lowercase, and special characters were removed. Common stopwords like "is" and "the" were filtered out to reduce noise. Words were lemmatized to their base forms (e.g., "running" to "run") and stemmed to their root forms, reducing vocabulary size. Punctuation was removed to reduce distractions, and sentences were split into individual words (tokens) for word-level processing. Data augmentation was applied by shuffling words in sentences to add variation and improve model performance.

Model Implementation Approaches:

Eight primary approaches were developed and tested using the Cornell Movie-Dialogs Corpus. The base model used an RNN sequence-to-sequence with Luong attention mechanism. The best-tuned model used Bayesian optimization to tune parameters like batch size, hidden size, learning rate, dropout, layers, teacher forcing ratio, and gradient clipping. CNN and MLP models, without attention, provided alternative architectures for performance checking. Non-AI approaches included rule-based and menu-based bots, which were structured but limited with functionalities. An RNN-based model with Luong attention was also tested with the Persona-Chat dataset. Transformer methods included Microsoft's DialoGPT-large for high-quality conversations and a transfer learning model (DialoGPT-small) integrated with an SQL database for smart home equipment. In addition to that, integrating a UI(user interface) through Gradio made an interactive, user-friendly experience. The observations have been noted.

Results and Analysis:

The results showed differences between non-AI and AI methods. Non-AI bots gave structured but rigid responses, while AI models varied in performance. The RNN models had high structural alignment (BLEU score of 0.9529) but moderate semantic relevance. Fine-tuning done with Bayesian Optimization gave good result w.r.t. limited computational power. The CNN and MLP models reached perfect semantic similarity (cosine score of 1.0), but their low BLEU scores indicated poor structural alignment. Transformer encode-decoder model, and model with transfer learning have been implemented, though computational resources were a big challenge.

Conclusion & Recommendations:

Transformer and transfer learning models are recommended for applications needing adaptable, high-quality responses. RNNs with attention provide a good balance but need more tuning for specific cases. Non-AI methods work well for simple, rule-based responses. Future work should focus on tuning encoder and decoder parameters individually, using larger datasets, and addressing computational constraints to enhance chatbot performance.