HealthBot: Intelligent Healthcare Assistant using Large Language Models

Ashwin Sateesh Kumar
sateeshkumar.a@northeastern.edu

Bindu Latha Banisetti
banisetti.b@northeastern.edu

Nikhil Gudipally
gudipally.n@northeastern.edu

Sriram Hariharan Neelakantan
neelakantan.s@northeastern.edu

1 Introduction

The HealthBot initiative addresses the pressing need for accessible and dependable healthcare information. Given the increasing volume of online health-related queries, there is a clear demand for a sophisticated healthcare assistant proficient in accurately interpreting user questions about diseases, symptoms, and medications. Our approach revolves around the development of an intelligent bot, empowered by advanced Large Language Models (LLMs), that excels in offering users comprehensive and reliable advice. This HealthBot will emerge as a valuable resource, providing insights into associated symptoms, recommended treatments, and relevant medical conditions. We intend to achieve this by harnessing a diverse and meticulously curated dataset sourced from reputable medical references. To further enhance its capabilities, we will fine-tune the LLM using state-of-the-art transfer learning techniques.

2 Project Lifecycle

The process outlined in Figure 1 reflects a structured approach to developing an advanced healthcare assistant, like the HealthBot initiative, using Large Language Models (LLMs). Here's how each step in the figure relates to the context provided.

2.1 Scope: Define the Problem

The initiative begins by defining the specific problem: creating a bot capable of accurately responding to health-related inquiries, providing insights into symptoms, recommended treatments, and medical conditions.

2.2 Select: Choose Model

The selection of an appropriate LLM is crucial, and the model must be adept at handling the complexities of medical queries to ensure the HealthBot's effectiveness.

2.3 Adapt and Align Model

a. **Prompt Engineering:** Crafting prompts to guide the LLM towards generating the desired accurate responses.

b. **Fine-tuning:** Using a meticulously curated dataset from reputable medical references to fine-tune the LLM.

c. **Evaluate:** Assessing the HealthBot's accuracy and reliability in offering healthcare advice.

d. **Align with Human Feedback:** Incorporating feedback to refine the HealthBot's performance and ensure its advice aligns with real-world medical knowledge.

2.4 Application Integration

a. **Optimize and Deploy Model for Inference:** After fine-tuning and evaluation, the model is optimized for quick and accurate real-world operation.

b. **Augment Model and Build LLM-powered Applications:** The LLM is augmented and integrated into user-facing applications, establishing the HealthBot as a valuable resource.

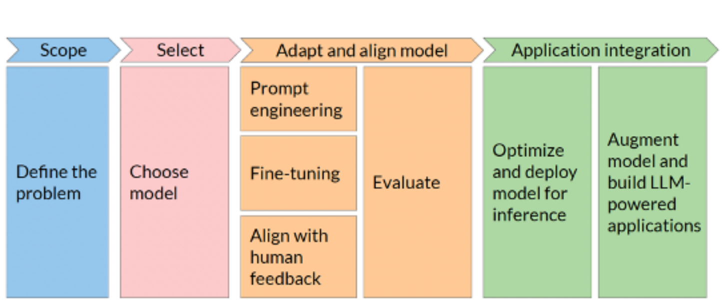

Figure 1: General Architecture of LLM based Applications (Credit: DeepLearning.AI)

3 Project Design

The project's architecture, as depicted in Figure 2, is meticulously designed to harness the synergy between these components, considering the Project Lifecycle. It showcases a holistic approach to building a conversational AI system that not only identifies user intents and extracts relevant entities but also fine-tunes a language model to generate contextually appropriate responses.

In the architecture of this project, the first step involves identifying the disease intent through an intent classifier. Simultaneously, a named entity recognizer detects relevant entities in the user input. These two processes run in parallel, with the input and labeled data forming the foundation for prompt engineering. Through prompt engineering, prompt-completion pairs are generated, creating a structured input-output framework. This framework is then employed for fine-tuning a Large Language Model (LLM), which learns to generate appropriate responses based on the input. The resulting responses undergo validation and feedback from a reward model through reinforcement learning, ensuring that the model evolves to provide valid and meaningful responses. This intricate interplay between intent classification, entity recognition, prompt engineering, fine-tuning, and reinforcement learning contributes to the robustness and effectiveness of the conversational system.

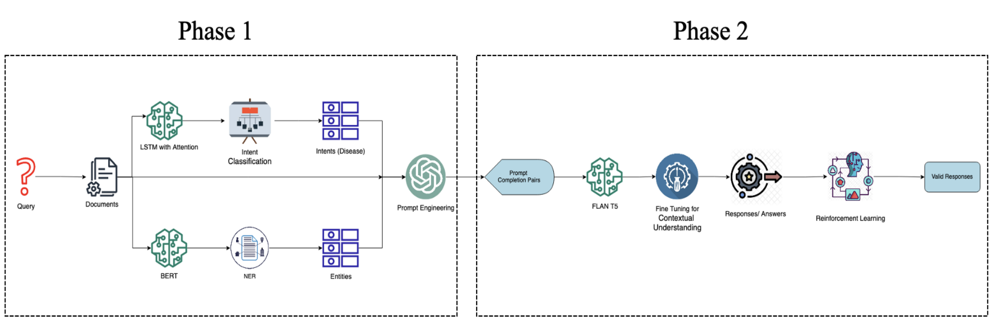

Figure 2: Healthbot's Architecture

4 Data Preparation and Processing

The backbone of the HealthBot project is a robust dataset, integral for the training of the Large Language Models (LLMs) that are central to the chatbot's operation. This dataset is constructed from a detailed compendium of disease-related data, sourced from reputable medical databases and repositories.

This Table 1 below breaks down the dataset preparation process into distinct steps, providing a clear description of each step's purpose and its role in the overall dataset preparation for the HealthBot project.

Process	**Description**
Data Cleansing	- Normalize various attributes. - Remove anomalies to ensure data quality and consistency.
Categorization	- Classify data into medical terminologies such as symptoms, diagnoses, and treatment options. - Enables precise entity extraction (NER).
Dataset Augmentation	- Programmatic synthesis using techniques like paraphrasing and back-translation. - Inclusion of diverse linguistic patterns. - Expands the dataset and enhances model generalization.
Response Template Development	- Craft templates using heuristic rules and medical knowledge. - Dictate the format and substance of the HealthBot's responses. - Parameterize templates for entity injection to personalize responses.
Semantic Structuring	- Organize data using ontological frameworks and medical taxonomies. - Reflect relationships and dependencies among various medical concepts. - Enhance the chatbot's understanding of medical topics.
Validation and Quality Checks	- Use automated scripts to verify the format and consistency of the data. - Conduct manual reviews to ensure medical accuracy and relevance of the content.
Finalization	- Result: High-fidelity dataset for the HealthBot. - Provides a foundation for delivering medically sound and contextually relevant advice. - Demonstrates a commitment to user-centric healthcare chatbot.

Table 1: Dataset Preparation Process for HealthBot

Below are the three main steps of the pipeline with descriptions and hypothetical examples:

4.1 Intent Classification

For the disease classification task, we provide the model with a collection of queries specifically related to various diseases. Each query serves as input, and the corresponding label is the name of the disease it pertains to. For example:

- **Input**: "What are the symptoms of Influenza?"
  - **Label**: "Influenza"

4.2 Named Entity Recognition

For Named Entity Recognition (NER), we have chosen a set of specific entity categories to enhance the precision of our system. These entities are categorized as shown in Table 2. To prepare the text data for NER, we divide each query into individual words and assign a numerical label based on the entity category to which each word belongs, using the provided dictionary. For example,

- **Input**: "What are the physical symptoms associated with persistent cough?"
  - **Entities**:["symptoms", "persistent", "cough"]
  - **Label**: [0, 0, 0, 0, 2, 0, 0, 4, 3]

Once this entity-labeling step is completed, we tokenize the input texts and feed them, along with the corresponding label arrays, into the BERT model. This comprehensive approach allows our system to recognize and categorize entities accurately, enhancing its ability to provide precise and contextually relevant responses to user queries.

Entity Category	Numerical Label
Miscellaneous (MISC)	0
medicine	1
symptom	2
disease	3
severity	4
sensation	5
body	6

Table 2: Entity Categories and Numerical Labels

4.3 Prompt Completion pairs for Contextual Understanding

To prepare the data for fine-tuning, we meticulously craft each prompt by concatenating the start prompt and end prompts to the actual input query which will be discussed in detail in section 5.3. The prompt and completion pairs are shown in below example. This combined input is carefully tokenized using a specialized tokenizer, designed to handle the intricacies of the FLANt5/GPT2 model. Once tokenized, these input sequences are then fed into the FLANt5/GPT2 model for fine-tuning.

- **Prompt**: **start prompt** + "I'm experiencing abdominal pain. What should I do?" + **end prompt**
  - **Completion**: "I'm sorry to hear that you're experiencing abdominal pain. In the meantime, you may consider taking pain relievers like ibuprofen to alleviate discomfort. Remember, it's crucial to seek professional advice for a comprehensive assessment of your condition."

With all these meticulous steps and processes completed, the HealthBot project has successfully curated a comprehensive dataset comprising approximately 5000 input queries. The distribution of these queries, categorized by their respective diseases, is thoughtfully visualized in Figure 3. Notably, this dataset is designed to be balanced, with a consistent number of input queries allocated to each disease category. This balance ensures that the chatbot's training data is equally representative across various medical conditions, promoting fairness and accuracy in its responses.

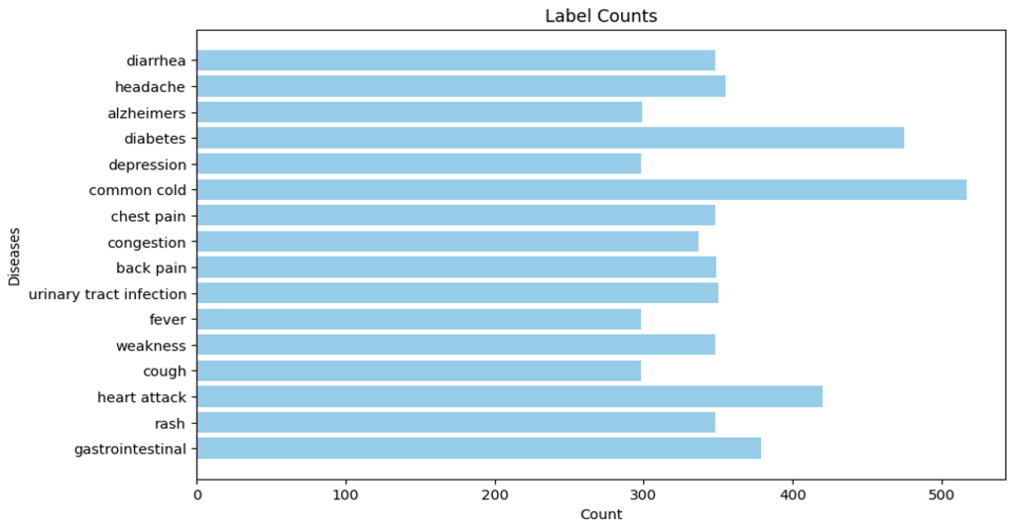

Figure 3: Data Distribution across all Diseases

5 Methods

5.1 Disease and Entity Recognition

Our primary method for disease and entity recognition begins with intent classification, where the overarching goal is to identify the specific disease or medical condition addressed in a user's query. To achieve this, we employ an LSTM model enhanced with an attention mechanism, allowing for accurate disease detection. Simultaneously, from the same user query, we aim to extract relevant entities as outlined in Table 2. This entity extraction is accomplished using BERT. By combining these two approaches, we not only discern the disease in question but also extract essential entities, ensuring that HealthBot comprehensively grasps and addresses user inquiries.

5.2 Mapping Entities to Knowledge Graph

In our approach to mapping entities to the knowledge graph, we initiated by meticulously collecting data on the ten most common medicines and symptoms associated with various diseases from authoritative medical sources. This valuable information is structured as hierarchical dictionaries as shown in Figure 4, with disease names at the top level, symptoms and medicines within the second level, and their respective names at the lowest level. This hierarchical structure allows us to efficiently organize and navigate the data. We then match the identified disease entity with our knowledge graph, to extract relevant information which is then used for prompt engineering.

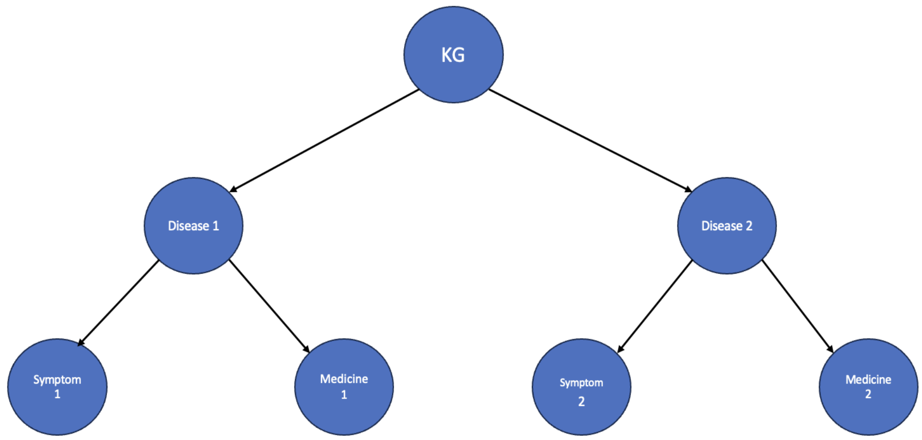

Figure 4: Structure of Knowledge Graph

5.3 Prompt Engineering

In the next step, which is prompt engineering, we focus on creating well-structured and concise instructions in natural language to guide language models in generating the desired responses or actions. Several key factors are taken into account during prompt engineering:

- Effective prompts should be characterized by clarity, providing precise context and instructions, ensuring the model understands the intended task.
- They should also offer the necessary context, maintain specificity, and often include examples to convey the desired response format.
- **Start Prompt:** The start prompt is the initial instruction that provides context or sets the stage for the model. It serves as a clear introduction to the task at hand, ensuring that the model understands the user's query and the expected response.
- **End Prompt:** The end prompt defines what the model is supposed to accomplish or conclude in its response. It provides the model with a specific direction on how to generate relevant and informative outputs.

With all the information gathered from previous steps, including the user's query, disease recognition, entity extraction, and knowledge graph mapping, the prompts are thoughtfully designed as follows:

- **Input**: "What are the physical symptoms associated with persistent cough?"
  - **Start Prompt**:For the context of {disease}, please consider the symptom and medicine information below:
  - **End Prompt**: Please provide an answer to the following question:{Input}

Combining all of them So the whole prompt looks like

For the context of {disease}, please consider the symptom and medicine information below:

Symptoms for {disease}: {symptom_info} Medicines for {disease}: {medicine_info}

Please provide an answer to the following question:
{input}

5.4 Fine-Tuning GPT2 / FLAN-T5 for Response Generation

During this phase, we will incorporate explicit instructions as prompts into the fine-tuning process of GPT2 / FLAN-T5. This strategic approach ensures that the model generates valid and contextually relevant responses when presented with user queries. The provided completions offer the model a clear understanding of the expected response format and help it grasp the context of the queries. Fine-tuning, within our context, entails customizing the pre-trained GPT2 / FLAN-T5 model by training select parameters in the final layers. This adaptation aims to deepen the model's comprehension of nuances in healthcare-related queries and enable it to generate meaningful responses. These instructions play a crucial role in guiding the model to prioritize accuracy and relevance when formulating responses. The completions derived from this fine-tuning process will serve as the model's responses to user queries. By fine-tuning GPT2 / FLAN-T5 with healthcare-specific prompts and responses, our objective is to significantly enhance the model's capacity to offer informative and precise healthcare guidance, aligning closely with user expectations. This process is illustrated in Figure 5.

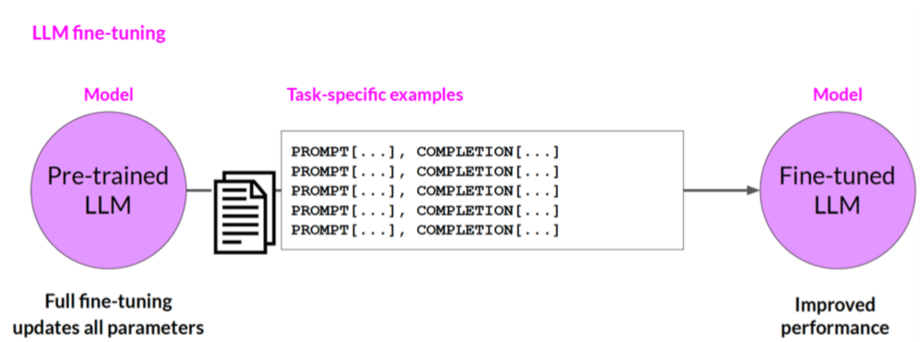

Figure 5: Instruction Fine Tuning (Credit: DeepLearning.AI)

The following Table 3 outlines essential parameters for response generation in LLMs. Each parameter plays a crucial role in fine-tuning the model's behavior during the generation of text-based responses. These parameters offer control over response length, vocabulary selection, diversity, and creativity, allowing for tailored and contextually relevant outputs in various applications.

Parameter	**Description**	**Effect on Response**
k (Top-k Sampling)	Top-k sampling limits the vocabulary to the top-k most likely words.	Controls diversity and narrows down word choices.
p (Top-p Sampling)	Top-p sampling selects from the smallest set of words whose combined probability exceeds p.	Adapts the vocabulary size dynamically based on probabilities.
max_length	Maximum number of output tokens in the response.	Constrains response length to a specified limit.
num_beams	Number of beams for beam search.	Enhances the diversity of generated responses by considering multiple possibilities.
temperature	Temperature affects token selection, with higher values making the distribution flatter and lower values sharpening it.	Alters the randomness of token selection, influencing response creativity.

Table 3: Parameters for Response Generation

5.5 Human Feedback and Reinforcement Learning

To ensure the quality and validity of our responses, we incorporate human feedback into our refinement process, serving as a vital mechanism for continuous improvement of our HealthBot's performance. Additionally, when time permits, we explore reinforcement learning methods to further enhance the responses generated by our HealthBot.

In this context, we treat our fine-tuned GPT2 / FLAN T5 model as our agent, response generation as its action, the input context window as the environment, and the current response as the current state. The details are as follows:

- **Current State as Input Context:** The current state is defined by the user's input and the conversational context. This forms the basis for the HealthBot's response.
- **Action as Response Generation:** The action here is the HealthBot generating a response. This action is based on the model's learning and the current conversational context.
- **Evaluating Simulated Conversations:** The HealthBot simulates potential future conversations based on its response, evaluating them for effectiveness using criteria such as accuracy, relevance, and user satisfaction.
- **Iterative Improvement:** The HealthBot continuously improves its response strategies through real interactions and these simulated conversation rollouts, enhancing its ability to provide helpful and contextually appropriate advice.

This process is visualized in Figure 6, and it demonstrates our commitment to refining the HealthBot's responses through a combination of human feedback and reinforcement learning techniques.

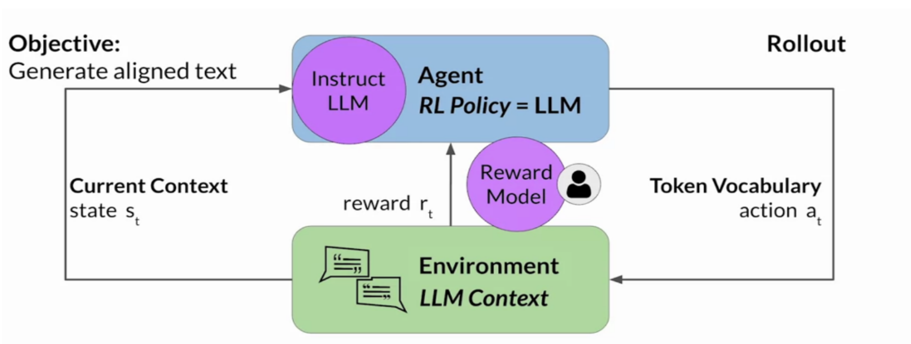

Figure 6: Human Feedback and Reinforcement Learning (Credit: DeepLearning.AI)

In our approach, we implement a simple policy network as a reward model, utilizing cosine similarity as the cost function. This involves generating embeddings for both the instruction-tuned LLM response and the actual response and measuring the cosine similarity between them. If the similarity exceeds a predefined threshold, the response is deemed valid; otherwise, the LLM's weights are updated. This iterative cycle continues until we obtain relevant and valid responses, thereby optimizing the overall user experience.

Figure 7 below illustrates the iterative process of refining HealthBot responses through human feedback and reinforcement learning, with an emphasis on updating model weights for improved responses.

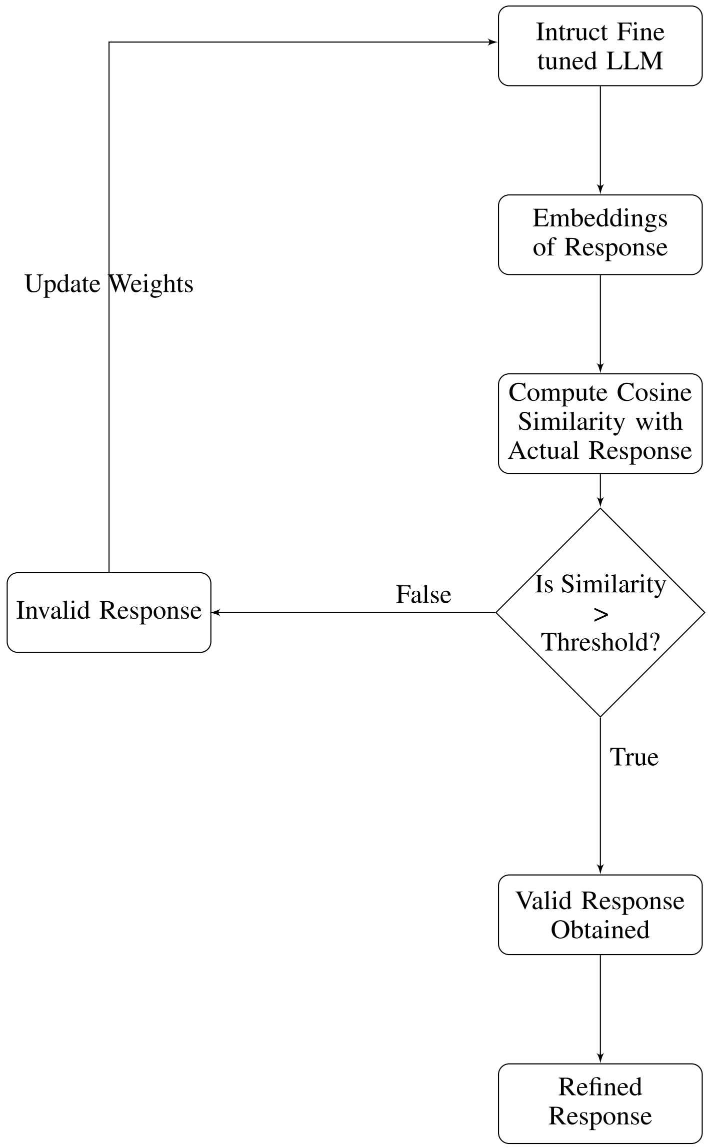

Figure 7: Refining HealthBot Responses with Reinforcement Learning

5.6 Model Selection

In the HealthBot project, model selection plays a pivotal role due to the diverse nature of the tasks involved. The project encompasses three primary tasks: Intent Classification, Named Entity Recognition (NER), and Response Generation. Each of these tasks necessitates distinct modeling approaches, underlining the importance of choosing the right models to achieve optimal results.

For Intent Classification, the primary goal is to classify input queries into specific health-related intents or diseases. This task demands a sequence model capable of understanding the context and nuances of user queries. After careful consideration, we opted for Long Short-Term Memory (LSTM) networks augmented with attention mechanisms. LSTMs excel at handling sequences and have the ability to capture dependencies in sequential data effectively. The addition of attention mechanisms allows the model to focus on relevant parts of the input, making it well-suited for intent classification.

Named Entity Recognition (NER) is another critical task in the project, involving the identification of specific entities like diseases, symptoms, and medications within the text. For this task, BERT

(Bidirectional Encoder Representations from Transformers) emerged as the preferred choice. BERT is renowned for its contextual understanding of text and its ability to capture relationships between words. This makes it exceptionally apt for NER, where context is paramount in identifying entities accurately.

For response generation and question answering, models that excel in natural language understanding and generation are essential. Here, we leveraged models like GPT-2 and FLAN (a variant of T5). GPT-2, available in various sizes ranging from small to xlarge, is known for its capabilities in generating coherent and contextually relevant text. Its large-scale language model can be fine-tuned for specific tasks, making it ideal for generating informative responses. Similarly, FLAN T5, a variant of T5, is well-suited for the task due to its ability to understand and generate human-like responses, making it an excellent choice for our project's conversational aspect.

In summary, model selection in the HealthBot project is driven by the specific requirements of each task. We have chosen LSTM with attention for Intent Classification, BERT for NER, and GPT-2 along with FLAN T5 for response generation. These models have been selected based on their strengths in handling various aspects of natural language understanding and generation, ensuring the project's effectiveness in delivering reliable and informative healthcare information.

Below Table 4 gives detailed information about the models considered for each task

Table 4: Model Details

Model	**Hyperparameters**	**Trainable Parameters**	**Evaluation Metric**
LSTM with Attention	Loss: CrossEntropyLoss Optimizer: Adam Learning Rate: 0.001 Batch Size: 16 Embedding Dimension: 128 Epochs: 10 Hidden Size: 128 Number of Layers: 2	4571155	Accuracy
BERT Base Uncased	Optimizer: AdamW Learning Rate: 2e-5 Batch Size: 4 Embedding Dimension: 768 Epochs: 5	108897031	Accuracy, F1 Score
GPT-2 Base	Loss: CrossEntropyLoss Optimizer: AdamW Learning Rate: 2e-5 Batch Size: 8 Embedding Dimension: 768 Epochs: 10 Eps: 1e-8	124440576	Semantic Similarity
FLAN - T5 - Base	Loss: CrossEntropyLoss Optimizer: AdamW Learning Rate: 1e-5 Batch Size: 8 Embedding Dimension: 768 Epochs: 10	247577856	Semantic Similarity

6 Results and Discussion

6.1 Intent Classification

The results for intent classification using three different models are presented in Table 5 below. Among these models, LSTM with Attention exhibited the best performance and is also relatively lightweight, making it the preferred choice.

Table 5: Model Performance for Intent Classification

Model	Train Accuracy	Validation Accuracy
LSTM with Attention	0.98	0.96
Bi-RNN	0.95	0.92
Transformer	0.96	0.95

The choice of accuracy as our primary evaluation metric was deliberate, as our overarching objective was to precisely classify each query into its corresponding disease category. Here for this task especially we have accurately determined the disease from the context of the query. So our goal was to classify each query to its correct disease category. Hence we choose accuracy as a metric.

Below Figure 8, provides a comprehensive visual representation of the training and validation accuracy along with average loss progress for the LSTM model over the course of 10 epochs.

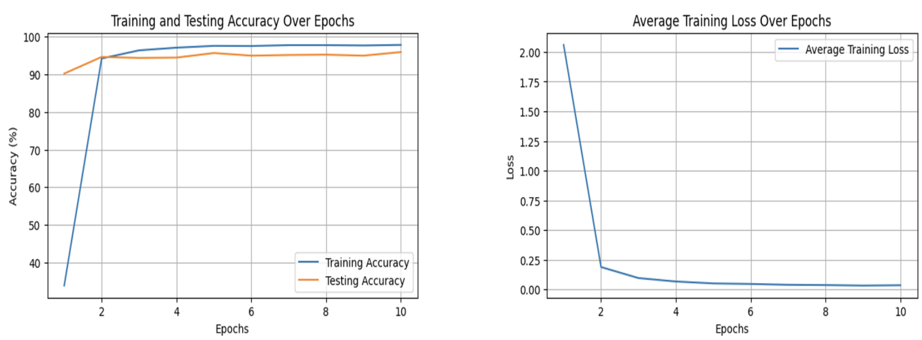

Figure 8: Accuracy and Loss plots for Intent Classification

With the hyperparmeters defined in Table 4 and training the LSTM for 10 epochs, the performance metrics for each class and accuracy are summarized in Table 6.

The high accuracies achieved across various diseases in the results Table 5 signify that the model has demonstrated a remarkable ability to understand and correctly classify the queries based on the mentioned diseases. This strong performance underscores the model's proficiency in disease detection and highlights its potential for practical healthcare applications.

Table 6: Performance Metrics for Intent Classification

Class	Precision	Recall	F1-Score	Accuracy
Alzheimers	1.00	1.00	1.00	100.00%
Back Pain	0.98	0.98	0.98	98.31%
Chest Pain	0.95	1.00	0.97	100.00%
Common Cold	0.97	0.72	0.83	72.00%
Congestion	0.90	1.00	0.95	100.00%
Cough	0.97	0.95	0.96	95.24%
Depression	1.00	0.98	0.99	98.39%
Diabetes	0.82	0.97	0.89	97.39%
Diarrhea	1.00	0.99	0.99	98.84%
Fever	0.97	1.00	0.98	100.00%
Gastrointestinal	0.97	0.95	0.96	95.08%
Headache	1.00	1.00	1.00	100.00%
Heart Attack	1.00	0.94	0.97	93.75%
Rash	1.00	0.99	0.99	98.57%
Urinary Tract Infection	0.98	1.00	0.99	100.00%
Weakness	0.97	0.99	0.98	98.53%

6.2 Named Entity Recognition

Our approach involved leveraging BERT Uncased, a state-of-the-art transformer model, to accurately recognize and categorize entities within user queries. Prior to hyperparameter tuning as indicated in Table 4, the model was trained over five epochs, resulting in a remarkable training accuracy of 96% and a testing accuracy of 96%, as indicated in Figure 9.

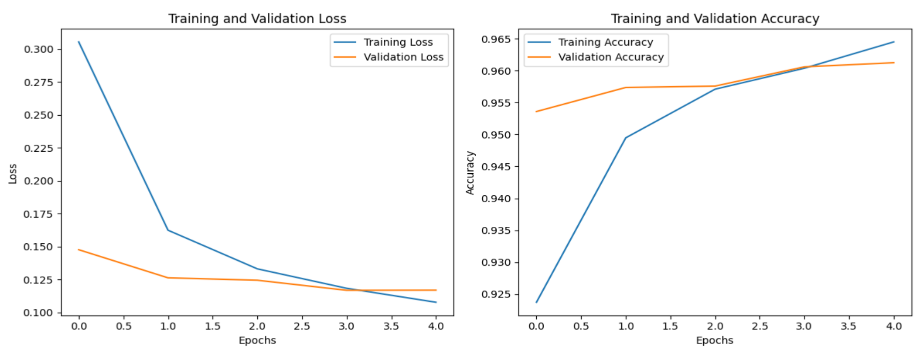

Figure 9: Accuracy and Loss for BERT NER

While accuracy provides valuable insights into the model's ability to classify words into their respective entity categories, it may not comprehensively gauge whether all words in a query precisely correspond to the corresponding entities. Therefore, we adopted the F1 score as our primary evaluation metric. Currently we were able to achieve recall of 0.8, a precision of 0.88, and an F1 score of 0.84, which in-turn are decent results for our usecase. The F1 score considers both precision and recall, providing a more holistic assessment of our NER model's performance. This choice aligns with our goal of ensuring that every word in a query is accurately and contextually matched to its corresponding entity.

6.3 Contextual Understanding and Response Generation

In the context of contextual understanding and response generation, we successfully fine-tuned the GPT-2 model for 10 epochs, as depicted in Figure 10, showcasing the loss plot during training. Furthermore, we extended our model's capabilities by refining it using a Policy Network as a reward model and employing cosine similarity as the loss function in subsequent fine-tuning steps.

In our evaluation process, we employed semantic similarity as a crucial metric to assess the quality of responses generated by our HealthBot. Semantic similarity measures the likeness of meaning between two pieces of text and is highly relevant to our use case. This choice allowed us to ensure that the responses provided by our model were not only contextually accurate but also semantically aligned with the user's queries. We achieved an impressive **average semantic similarity score of 0.7**, indicating a high degree of relevance between our responses and the user's intent. This score underscores the effectiveness of our approach, as it accounts for the meaningfulness of responses beyond their length, offering a robust evaluation of our model's performance.

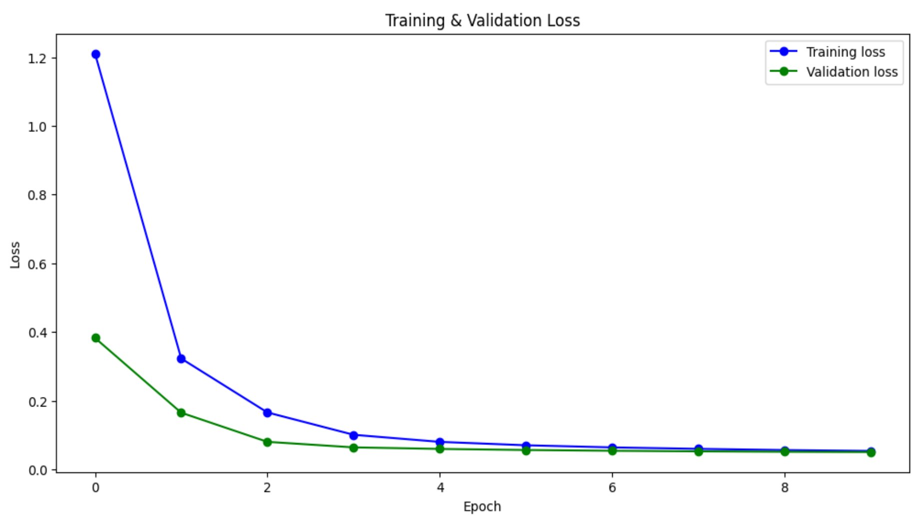

Figure 10: Loss Plot of GPT2

Before introducing reinforcement learning into our model, we conducted fine-tuning on the GPT-2 using prompt-completion pairs. The examples provided here represent the output generated by the model at this stage. While the model demonstrates an understanding of the prompts and provides some relevant information, there is room for improvement in terms of coherence and specificity. The generated responses may include extraneous information and exhibit some inconsistencies. The subsequent application of reinforcement learning techniques aims to refine these responses further and enhance their relevance and accuracy.

Example 1:
Prompt: Answer the following question:
What is diabetic neuropathy, and how does it manifest?
Actual Answer:
Diabetic neuropathy is nerve damage caused by high blood sugar, leading to numbness or pain in extremities.
Generated Answer:
Diabetes neuropathic pain is a common cause of nerve damage and can lead to nerve degeneration. symptoms of diabetic nerve pain include: Nausea, Numbness, Muscle spasms, Fatigue. diabetic nerve conditions spinal cord damage. The neuropathological conditions include diabetic retinopathy, Sore throat, Tingling or numbness in the neck, Headache

Example 2:
Prompt: Answer the following question:
What are the top three treatments available for chest pain?
Actual Answer:
Three medicines for chest pain include: Nitroglycerin, Aspirin, and Beta-blockers.
Generated Answer:
Three medicines for a heart attack include: Aspirin, Nitroglycerin, and Thrombolytics. Symptoms of chest discomfort include chest pain or discomfort, pain radiating to the arm, jaw, or back, and shortness of breath. Three medicines are: Nitrofurantoin (Tylenol), Aspergine, Rivastigmine, and Beta-blockers.

The depicted interface in Example 3 showcases the real-time interaction between users and the chatbot system following the implementation of reinforcement learning techniques. Notably, the responses provided by the chatbot exhibit a significant improvement in terms of meaningfulness and relevance compared to earlier versions of the system. Users can engage in more productive and informative conversations, highlighting the effectiveness of reinforcement learning in enhancing the bot's conversational abilities.

Example 3:

```
HealthBot
---------
Hello! I'm HealthBot. I can help you with general health questions.
Type 'quit' to leave the chat.

You: What are the medicines of diabetes?
HealthBot: Here's what I found for you:
The medicines for diabetes include: Oral antidiabetic drugs, Insulin,
TENS therapy, Fiber supplement, Vitamin C supplements, Thrombolytics,
Beta-blockers (TENS inhibitors).

You: What are the symptoms of UTI?
HealthBot: Here's what I found for you:
The symptoms associated with UTIs include: Gas, Chills, Constipation,
Blood or mucus-filled sores, Heart attack, Diplopia, Hunger, and Urgent
need to have a bowel movement evaluation. Medicines for UTI include H.
pylori, Antiviral medications, Antiplatelet medications. This may
suggest preventive options to help alleviate symptoms like heartburn,
ulcers or other gastrointestinal symptoms. It may be necessary to seek
medical attention for serious underlying causes, such as multiple
sclerosis or severe cases of COPD-like conditions.

You: quit
HealthBot: Goodbye!
```

In Example 4, we observe the chatbot's responses under a unique condition where the temperature parameter is set to its maximum value of 2. This elevated temperature setting unleashes a higher level of creativity within the model. If we compare the responses to the same question, the chatbot's responses in Example 4 become notably more expansive and diverse, often presenting longer and more elaborate sentences. This creative mode showcases the model's capability to generate imaginative and varied responses, when the shape of the probability distribution widens, offering a glimpse into its potential for generating engaging and contextually rich dialogues.

Example 4:

```
HealthBot
---------
Hello! I'm HealthBot. I can help you with general health questions.
Type 'quit' to leave the chat.

You: What are the medicines of diabetes?
HealthBot: Here's what I found for you:
The medicines for diabetes include: Fiber-containing foods,
Antiplatelet medications, Lifestyle changes, Dietary modifications, and
Dietary changes. Medicines include Fiber, Antioxidants, Vitamin C
supplements, Sulfonylureas, Calcium channel blockers, Bismuth
subsalicylate supplements.

You: What are the symptoms of UTI?
HealthBot: Here's what I found for you:
The symptoms associated with UTIs include: Abruption of bowel movement,
Gas, Vibration, Heart attack, Abrupt bowel movements. Medicines for
urinary tract infection include H. pylori, Antipyretic medications,
Antiplatelet medications. The medicines include antihistamines,
Prolonged urination, Pain relievers for underlying causes, and
Antiinflammatory medications for specific causes. The medicines are
safe for use in adults and children. It's important to consult a
healthcare provider for proper evaluation and diagnosis.

You: quit
HealthBot: Goodbye!
```

7 Conclusion

The development of the HealthBot project has been a significant endeavor, aimed at addressing the critical need for accessible and reliable healthcare information. Throughout the project, we encountered several challenges that highlighted the complexity of the task at hand. Capturing diverse user intents and enabling the model to discern subtle differences in queries proved to be a formidable challenge. Creating structured data templates for tasks like Named Entity Recognition (NER) and Intent Classification was essential to ensure consistent and informative training data. We also had to incorporate natural language variations and colloquial expressions to make the model's responses more user-friendly. Managing the integration of multiple models for NER, Intent Classification, and large-scale language generation added another layer of complexity. Additionally, the lack of comprehensive and specialized health-related datasets from the internet and limited access to high-performance computing resources posed challenges during training and fine-tuning of large language models like GPT-2.

Looking ahead, the future scope of the project is promising. We intend to gather more data on a wider variety of diseases, exploring additional research papers to enhance the depth and breadth of our knowledge base. To improve scalability and resource optimization, we plan to explore deploying the health chatbot on cloud platforms such as AWS. Extending the chatbot's capabilities to support multiple languages, with the inclusion of an in-built translator, will make healthcare information accessible to a more diverse user base. Furthermore, we aim to include more disease types in our repertoire and expand the chatbot's capabilities to collaborate with AI solutions in other domains, such as nutrition, fitness, mental health, and lifestyle management, providing users with holistic well-being support. Additionally, we plan to suggest help links and contact information to ensure users have access to the necessary resources for their healthcare needs.

In conclusion, the HealthBot project represents a significant step towards democratizing healthcare information and support. Despite the challenges faced, the project has laid a strong foundation for future enhancements and expansions, making it a valuable resource for users seeking reliable and comprehensive health-related information.

References

Zihan Liu1, Feijun Jiang2, Yuxiang Hu2, Chen Shi2, Pascale Fung1, *NER-BERT: A Pre-trained Model for Low-Resource Entity Tagging*. `https://arxiv.org/abs/2112.00405`

Jacob Devlin, Ming-Wei, Chang Kenton, Lee Kristina Toutanova, *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*. `https://arxiv.org/pdf/1810.04805.pdfs`

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, *Scaling Instruction-Finetuned Language Models*. `https://arxiv.org/pdf/2210.11416.pdf`

Jannik Kossen1, Tom Rainforth, Yarin Gal, *In-Context Learning in Large Language Models: Understanding Label Relationships Beyond Conventional Learning*. `https://arxiv.org/pdf/2307.12375.pdf`

LasseRegin, *Medical Question Answer Data*, GitHub repository. `https://github.com/LasseRegin/medical-question-answer-data/tree/master`

PuneetShivaay, *Deep Learning Based Chatbot For Medical Assistance*, GitHub repository. `https://github.com/PuneetShivaay/Deep-Learning-Based-Chatbot-For-Medical-Assistance/tree/master`

ronak-07, *Medical Expert System Knowledge Base*, GitHub repository. `https://github.com/ronak-07/Medical-Expert-System-Knowledge-Base/tree/master`

gargmegham, *Medical GPT*, GitHub repository. `https://github.com/gargmegham/MedicalGPT/tree/main`

gargmegham, *Hugging Face Datasets*, Dataset Repository. `https://huggingface.co/datasets?p=1&sort=trending&search=health`

Chung et al., *"Scaling Instruction-Finetuned Language Models"*, `https://arxiv.org/abs/2210.11416`