INLP Mid Evaluation – Report

Los Pollos Hermanos

Ishaan Karan 2023114016
Atharv Johar 2023114010
Prajna Penmetsa 2023114017

We're developing an advanced NLP-driven system that harnesses deep learning models—BERT, LSTM, and CNN—to diagnose diseases based on user-reported symptoms. By leveraging sophisticated word embeddings from both GloVe and BioBERT, our platform captures the nuances of medical language and interprets complex symptom descriptions. Drawing on extensive historical case data, the system will also be developed to generate concise summaries of treatment options, empowering users with personalized, data-driven healthcare insights.

1. Introduction

The core task is primarily text classification—specifically, clinical text classification. The system analyzes user-provided symptom descriptions and maps them to disease categories. Additionally, we plan on incorporating an element of text summarization to condense treatment options.

General clinical text classifiers often struggle with a few key challenges that our system directly addresses:

1. Limited Domain-Specific Context: Traditional models frequently rely on generic word embeddings that miss the subtle nuances of medical language. By integrating BioBERT with GloVe, our system captures both general semantic and domain-specific contextual information.

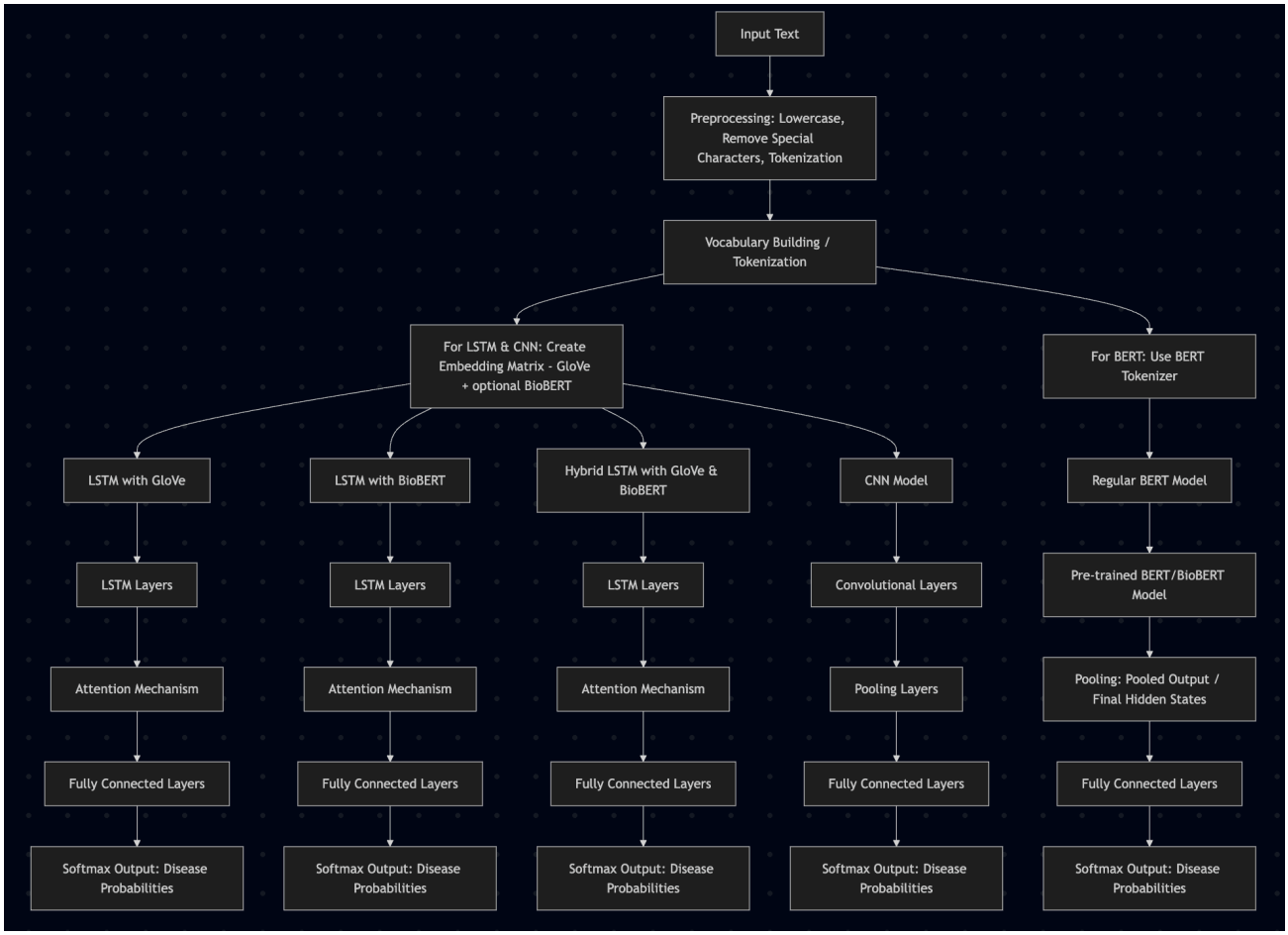
2. Handling Ambiguity in Symptom Descriptions: Many classifiers are built on methods that don't effectively disambiguate vague or overlapping symptom descriptions, our enhance the extraction of relevant features, ultimately leading to more precise disease predictions.

3. Lack of Integrated Treatment Insights: While many classifiers focus solely on identifying conditions, they fall short in providing actionable follow-up information. Our system goes a step further by summarizing treatment options, offering users not just a diagnosis but also guidance on potential interventions.

2. Architecture Overview

The system consists of several model variants:

- Regular BERT Model
- LSTM with GloVe
- LSTM with BioBERT
- LSTM with GloVe and BioBERT
- CNN Model



2.1 Pretraining Phase

Word Embeddings:

1. GloVe Embeddings: These are pretrained on large-scale general corpora (e.g., Common Crawl, Wikipedia) and serve as fixed word representations.
2. BioBERT Embeddings: BioBERT is pretrained on biomedical corpora. In our system, it's used to obtain domain-specific embeddings for clinical terms that GloVe might not capture adequately.

Transformer Models:

The BERT model we have used is a fine-tuned BERT model with compact variants, alongside bert-small and bert-medium, originally introduced in "Well-Read Students Learn Better: On the Importance of Pre-training Compact Models" and later adapted for downstream tasks in "Generalization in NLI: Ways (Not) To Go Beyond Simple Heuristics." This is a PyTorch version converted from

the original TensorFlow checkpoint available from the official Google BERT repository. We will be citing the use of this model as mentioned in its HuggingFace page.

2.2 Fine-Tuning Phase

Task-Specific Training:
The models are fine-tuned on our clinical dataset (detailed below) using supervised learning. During fine-tuning, model parameters are adjusted to minimize the loss on the disease classification task.

Training Pipeline:
The system tokenizes input text, builds a vocabulary, creates embedding matrices (combining GloVe and/or BioBERT), and splits the dataset into training, validation, and test sets. The fine-tuning happens on this clinical dataset using backpropagation through the chosen architecture (LSTM, CNN, or fine-tuned BERT).

2.3 Dataset

We are using the Kaggle Dataset ("kuc-hackathon-winter-2018").
Two CSV files are used:

- drugsComTrain_raw.csv
- drugsComTest_raw.csv

We have selected this dataset because it provides comprehensive real-world clinical data that includes detailed user-reported symptoms alongside their associated conditions and diseases. This dataset offers the necessary contextual information to train and fine-tune our models effectively. Its depth and variety ensure that our model can generalize well.

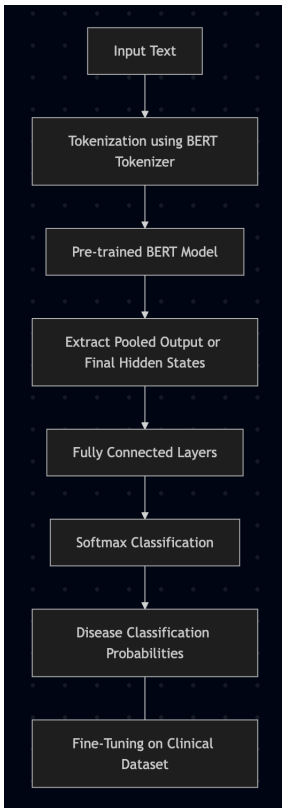
The system cleans the dataset (removing empty reviews or conditions) and combines the train and test data for vocabulary building and subsequent splitting into train, validation, and test sets.

2.4 Model Design

1. Regular BERT Model

This variant uses a pre-trained BERT (or BioBERT) model that is fine-tuned on the clinical data. The design is as follows:

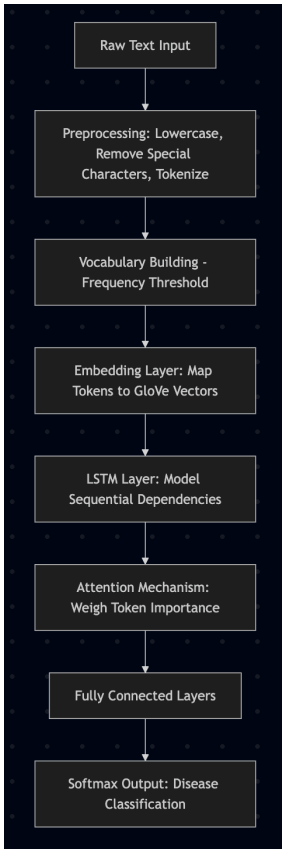
- Input Processing: Text is tokenized using the BERT tokenizer.
- Model Backbone: The tokenized input is fed directly into the pre-trained BERT model.
- Pooling: A pooled output (or the final hidden states) is extracted from BERT to summarize the input sequence.
- Classification Head: The pooled representation is passed through one or more fully connected layers followed by a softmax layer to output disease classification probabilities.
- Fine-Tuning: The entire model, including the BERT backbone, is fine-tuned on your clinical dataset, allowing the model to adapt its representations to the specific task.



2. LSTM with GloVe

This variant is built around an LSTM network using pre-trained GloVe embeddings.

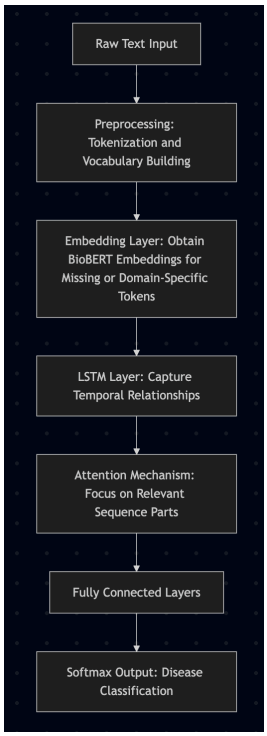
- Input Processing: The raw text is preprocessed (lowercased, special characters removed, tokenized) and a vocabulary is built using a frequency threshold.
- Embedding Layer: An embedding matrix is constructed using GloVe embeddings. Tokens are mapped to indices, and their corresponding GloVe vectors are used as input features.
- Sequential Modeling: The GloVe embeddings are passed to an LSTM layer that models the sequential dependencies of the text.



- Attention Mechanism: An attention layer is applied on the LSTM outputs to weigh the importance of each token dynamically.
- Output Layer: The attention-weighted features are fed into fully connected layers with a softmax output for disease classification.

3. LSTM with BioBERT

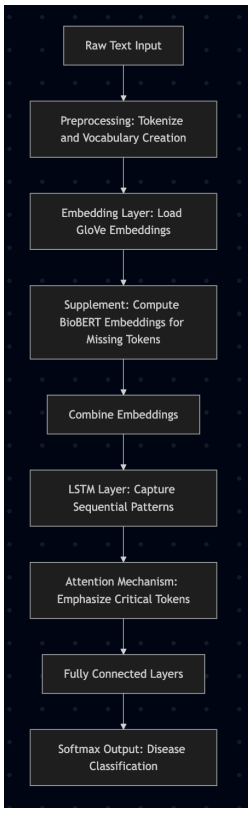
Here, the design leverages BioBERT to supplement domain-specific knowledge.



- Input Processing: Similar preprocessing (tokenization, vocabulary building) is performed.
- Embedding Layer: Instead of relying solely on GloVe, for tokens that are missing or require deeper biomedical context, embeddings are obtained from a BioBERT model.
- Sequential Modeling: The resulting embeddings are input to an LSTM network that captures temporal relationships.
- Attention Mechanism: An attention layer is used on top of the LSTM outputs to focus on the most relevant parts of the sequence.
- Output Layer: Fully connected layers map the final representations to disease categories with softmax providing probability estimates.

4. LSTM with GloVe and BioBERT (Hybrid Model)

This hybrid approach combines general-purpose and domain-specific embeddings.

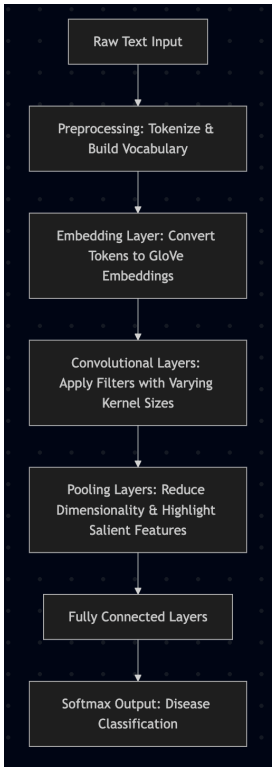


- Input Processing: The same text preprocessing and vocabulary creation are applied.
- Embedding Layer:
    - First, GloVe embeddings are loaded to initialize the embedding matrix.
    - For words not covered by GloVe or identified as domain-specific, BioBERT embeddings are computed and used to supplement the embedding matrix.

- Sequential Modeling: The combined embeddings are then passed through an LSTM layer that captures the sequential patterns.
- Attention Mechanism: An attention mechanism is applied over the LSTM outputs to emphasize critical tokens.
- Output Layer: The attention-weighted features are fed into fully connected layers to produce final disease classification outputs.

5. CNN Model

This variant uses convolutional neural networks to capture local patterns in the text.



- Input Processing: Text is preprocessed and tokenized, with a vocabulary built from the dataset.
- Embedding Layer: Tokens are converted into embeddings (typically using GloVe, though a hybrid approach could be applied here as well).
- Convolutional Layers: One or more convolutional filters with varying kernel sizes are applied to the embedding sequences. These layers capture local n-gram features that are indicative of symptom clusters.
- Pooling: Pooling layers (e.g., max pooling) reduce the dimensionality of the convolution outputs and help highlight the most salient features.
- Fully Connected Layers: The pooled feature maps are flattened and fed into fully connected layers, culminating in a softmax layer for disease classification.

3. Results

- BERT Model
    - Training and Validation:
      Train Loss: 0.4507
      Train Accuracy: 87.94%
      Train F1: 0.8719
      Validation Accuracy (Best after 10 epochs): 80.26%

Validation F1 (Best): 0.7936

- Testing:
  Test Loss: 1.0418
  Test Accuracy: 80.27%
  Test F1: 0.7952

  These results indicate that the fine-tuned BERT model is able to learn contextual representations from clinical text reasonably well, attaining a good balance between precision and recall (as reflected in its F1 score).

- Sample Prediction:
  For the input "I've had a fever for a few days, along with joint pain and a red rash on my skin," the model predicted a condition related to "Influenza Prophylaxis," which highlights that BERT captures the semantics of symptom descriptions.

- LSTM with GloVe Model
  - Testing:
    Test Loss: 1.4923
    Test Accuracy: 82.04%
    Test F1: 0.8122

    This slightly surpasses the BERT model in terms of raw accuracy (82.04% vs. 80.27%). It suggests that, for this dataset, the combination of GloVe embeddings and LSTM might generalize effectively to the clinical domain, at least on the specific conditions covered.

  - Sample Prediction:
    For the user input "I've been experiencing severe headaches and dizziness... along with sensitivity to light," the model predicted a form of anxiety disorder with over 92% confidence. While it does not necessarily reflect a real-world diagnosis, it indicates the model's

tendency to cluster certain symptoms (like headaches, dizziness, and sensitivity to light) with anxiety-related conditions in the dataset.

- LSTM with BioBert Model
    - Testing:
      Test Loss: 1.51
      Test Accuracy: 82.3%
      Test F1: 0.816

      This marginal improvement over the hybrid model indicates that relying solely on BioBERT embeddings for domain-specific vocabulary can be effective, especially if the dataset contains a high proportion of specialized clinical terms. As with the other LSTM variants, attention helps emphasize critical tokens within symptom descriptions.

- LSTM with GloVe and BioBERT (Hybrid Model)
    - Testing:
      Test Loss: 1.5407
      Test Accuracy: 82.04%
      Test F1: 0.8135

      Although the accuracy is the same as the LSTM with GloVe (82.04%), the F1 score is slightly higher (0.8135 vs. 0.8122), suggesting a marginal improvement in balancing precision and recall. Qualitative examples show that the hybrid approach can more confidently predict domain-specific conditions, as illustrated by the high-confidence predictions for “Obesity,” “High Blood Pressure,” and “Atopic Dermatitis.”

    - Sample Prediction:
        - “Not really diarrhea...” The model predicted “Irritable Bowel Syndrome” (24.09% confidence) when the actual condition was “Prevention of Thromboembolism in Atrial Fibrillation,” highlighting a misclassification.

- "I've been on Buspar for 7 days..." Correctly predicted "Anxiety" with 99.86% confidence.

- "I have been on Belviq for 9 days..." Correctly predicted "Obesity" with 99.94% confidence.

  These examples underscore the hybrid model's strong performance on many test samples but also reveal occasional misclassifications when symptom descriptions differ substantially from the typical patterns in the dataset.

- CNN Model
  The CNN-based approach is still under training and thus is not included in this comparison. Future iterations will evaluate how convolutional filters capture local n-gram patterns and how this compares to sequential modeling with LSTMs or contextual embeddings from BERT.

3.1 Command Line Interface (CLI)

We have developed a command-line interface (CLI) that integrates predictions from multiple models to determine the most confident disease classification for a given symptom description.

Current CLI results–

- For instance, when processing the input: *"I have dry, itchy, red patches on my skin that are scaly and sometimes develop small blisters that ooze."*, the CLI runs the prediction across four different model variants. The results were as follows:
  - BERT: Predicted Eczema with a confidence of 41.08%.
  - LSTM_BioBERT: Predicted Seasonal Allergic Conjunctivitis with a confidence of 35.29%.
  - LSTM_GloVe: Predicted Urticaria with a confidence of 51.67%.
  - LSTM_GloVe_BioBERT (Hybrid): Predicted Dermatitis with a confidence of 60.84%.

Based on these outputs, our CLI automatically selected the prediction from the LSTM_GloVe_BioBERT model, which had the highest confidence level (60.84%), and reported Dermatitis as the predicted condition. This integrated approach not only highlights the strengths and nuances of each model but also demonstrates how our system leverages ensemble insights to provide the most reliable diagnostic prediction.

```
PS C:\Users\johar\OneDrive\Desktop\inlp\inlproject4> python.exe .\predict_disease.py "I have dr
ometimes develop small blisters that ooze."

Running prediction for BERT...
Output from BERT:
Dataset already downloaded.

Based on the symptoms provided, the predicted condition is: Eczema

Top 5 possible conditions with confidence scores:
- Eczema: 0.4108 (41.08%)
- Atopic Dermatitis: 0.2330 (23.30%)
- Psoriasis: 0.0899 (8.99%)
- Dermatitis: 0.0589 (5.89%)
- Pruritus: 0.0554 (5.54%)



Running prediction for LSTM_BioBERT...
Output from LSTM_BioBERT:
Model loaded from ./best_lstm_model.pt

Based on the symptoms provided, the predicted condition is: Seasonal Allergic Conjunctivitis

Top 5 possible conditions with confidence scores:
- Seasonal Allergic Conjunctivitis: 0.3529 (35.29%)
- Urticaria: 0.1940 (19.40%)
- Eczema: 0.1572 (15.72%)
- Dermatitis: 0.1335 (13.35%)
- Urinary Incontinence: 0.0459 (4.59%)



Running prediction for LSTM_GloVe...
Output from LSTM_GloVe:
Model loaded from ./best_lstm_model.pt

Based on the symptoms provided, the predicted condition is: Urticaria

Top 5 possible conditions with confidence scores:
- Urticaria: 0.5167 (51.67%)
- High Blood Pressure: 0.2056 (20.56%)
- Muscle Pain: 0.0827 (8.27%)
```

```
Based on the symptoms provided, the predicted condition is: Urticaria

Top 5 possible conditions with confidence scores:
- Urticaria: 0.5167 (51.67%)
- High Blood Pressure: 0.2056 (20.56%)
- Muscle Pain: 0.0827 (8.27%)
- Allergic Rhinitis: 0.0427 (4.27%)
- Not Listed / Othe: 0.0279 (2.79%)



Running prediction for LSTM_GloVe_BioBERT...
Output from LSTM_GloVe_BioBERT:
Model loaded from ./best_lstm_model.pt

Based on the symptoms provided, the predicted condition is: Dermatitis

Top 5 possible conditions with confidence scores:
- Dermatitis: 0.6084 (60.84%)
- Cluster Headaches: 0.2995 (29.95%)
- Not Listed / Othe: 0.0137 (1.37%)
- Glaucoma: 0.0101 (1.01%)
- Undifferentiated Connective Tissue Disease: 0.0099 (0.99%)


Results from all models:
BERT: Prediction: Eczema, Confidence: 0.4108
LSTM_BioBERT: Prediction: Seasonal Allergic Conjunctivitis, Confidence: 0.3529
LSTM_GloVe: Prediction: Urticaria, Confidence: 0.5167
LSTM_GloVe_BioBERT: Prediction: Dermatitis, Confidence: 0.6084

Model with highest confidence:
LSTM_GloVe_BioBERT: Prediction: Dermatitis, Confidence: 0.6084
PS C:\Users\johar\OneDrive\Desktop\inlp\inlproject4>
```

3.2 Observations

1. The LSTM-based models achieved comparable or slightly higher accuracy in this dataset. This outcome may be due to the LSTM's capacity to leverage the domain-specific knowledge embedded in BioBERT (and partially in GloVe) and the nature of the dataset's symptom descriptions.

2. Integrating BioBERT provided a slight boost in handling specialized clinical terms that GloVe alone might not capture. It demonstrates the value of domain-specific embeddings in clinical NLP tasks.

3. The attention layer consistently improves interpretability by highlighting the most relevant tokens in a symptom description. This is particularly helpful in clinical settings where certain keywords (e.g., “headache,” “nausea,” “joint pain”) heavily influence the diagnosis.

4. Next Steps

1. Now that we have a CLI, we will try to monitor the outputs of this tool closely to understand which models consistently perform best and whether certain sentence types favor specific models. This analysis will help us refine model selection and improve overall predictive performance. We will also add CNN to CLI.

2. As mentioned, we want to move forward and summarize treatment plans of the predicted diseases for the user to read. We are considering the following means of doing this:
    a. Scraping wikipedia pages of diseases
    b. Scraping medical journal pages of diseases
    c. Scraping AI responses to potential treatment plans of a disease

3. Some examples in the models show high confidence in incorrect predictions. This behavior underscores the importance of high-quality, representative training data and potentially more robust data augmentation or ensemble methods to reduce overconfidence in edge cases. We will explore better ensembling methods and data augmentation

5. Conclusion

We believe our interim progress is promising in integrating BERT, LSTM, and CNN architectures—with both GloVe and BioBERT embeddings—for clinical diagnosis. Our innovative CLI selects the best-performing model for each input,

providing valuable insights into model performance. As we continue refining our approach, these early results pave the way for more robust and data-driven healthcare solutions.

6. Updates from initial project idea and timeline

6.1. Updates

This project initially aimed to develop an NLP-powered system that retrieved similar past clinical cases based on symptoms and diagnoses, and then summarized the treatments used in those cases. We stuck to our original concept as much as possible; however, we made a strategic pivot from retrieving similar past cases to focusing solely on disease prediction and treatment summarization.

By concentrating on predicting the most likely disease given a set of symptoms, we streamlined the problem and allowed ourselves to leverage advanced deep learning models more effectively. This shift not only simplified our system design but also enabled us to use GPUs more efficiently given the constraints.

This pivot has resulted in a more robust tool, while still preserving the core idea of using data to inform healthcare decisions.

6.2 Updated timeline

Below is an updated timeline that reflects the pivot from case retrieval to focusing on disease prediction and treatment summarization, along with further refinements to address model overconfidence. The changes were necessary because we streamlined our project to focus on direct prediction and actionable insights into treatments, which allows us to leverage deep learning models more effectively and improve predictive performance.

Task	Estimated Time
1. CLI Output Monitoring & Model Performance Analysis - Monitor the CLI outputs to identify which models consistently perform best and assess performance variations for different sentence types	2 days
2. Integrate CNN Model into CLI - Complete training of the CNN model and integrate its predictions into the CLI for a comprehensive model comparison	3 days
3. Develop Treatment Summarization Module - Explore scraping Wikipedia Pages: Collect treatment information from disease-related Wikipedia pages - Explore scraping Medical Journals: Extract treatment details from reputable medical journals - Explore scraping AI Responses: Experiment with using AI-generated summaries of treatment plans	2 weeks
4. Explore Ensemble Methods & Data Augmentation - Investigate ensemble techniques to combine predictions from different models and mitigate overconfidence - Implement data augmentation strategies to improve the representativeness of the training dataset, particularly for edge cases	1 week
5. Evaluate, Test, and Refine the Overall System - Comprehensive evaluation and refinement based on integrated CLI predictions, treatment summarization, and ensemble performance.	2 days
6. Write Report and Prepare Presentation	3 days

- Comprehensive evaluation and refinement based on integrated CLI predictions, treatment summarization, and ensemble performance.	