Hospital Readmission Prediction System with LLM

Supraja Anagandula, Sai Krishna Bhamidipati, Mahesh Buragala, Karthik Patralapati and Sreenidhi Polineni

Department of Applied Data Science, San Jose State University

Data 298A : MSDA Project I

Dr. Simon Shim

May 13th , 2024

Abstract

Hospital readmission poses a substantial challenge to healthcare systems globally, resulting in escalated costs and patient morbidity. Identifying patients at high risk of readmission early on enables healthcare providers to implement proactive management strategies, thereby mitigating this risk. The project aims to address the critical challenge of predicting hospital readmission risk using electronic health records by developing large language models integrated with retrieval augmented generative architecture. Traditionally, machine learning models rely on structured data for model creation and evaluation, posing limitations when dealing with vast amounts of unstructured data. To address this, Large Language Models capable of processing unstructured data by predicting contextual information through masking and maintaining sentence relationships via next sentence prediction, are employed. A knowledge Vector base is used to store the health records in vector type which will be utilized by the retrieval system to retrieve relevant documents. The relevant documents along with the query are sent to the fine tuned LLM model for text generation. Model performance is evaluated using accuracy and confusion matrix, while the text generator is evaluated using rogue score which guides for further refinement and optimization. Python, in conjunction with UI frameworks like Django, enables seamless integration for making predictions using stored data. Potential applications include early identification of high-risk patients, decision support for healthcare providers, and customization of care plans. The project's impact includes reduced readmission rates, improved patient outcomes, cost savings, and enhanced healthcare delivery efficiency.

Introduction

1.1 Project Background and Executive Summary

Project Background, Needs, Importance, Target Problem, Motivations, and Goals

A patient readmission to the hospital affects the hospital economically as well as tells the effectiveness of the treatment. Detecting any readmission early is very helpful to reduce rates and be better prepared by the hospital and also give a good treatment to the patient. The project's motive is to improve patient care, quality, enhance resource allocation and mitigate economic burdens. Utilizing NLP and LLM's the project helps to analyze electronic health records and predict readmission of a patient within 30 days. The goal of the project is to help medical institutions to identify high risk patients by predicting readmission, which is beneficial for the patients as well as the healthcare systems.

Project Approaches and Methods

The project's primary architecture is Retrieval Augmented Generation, a process that integrates pre trained models with data stores. In the RAG, initially the retriever gets the similar documents to the input from the data store and then prediction and generation is based on these records and input. A variety of natural language processing (NLP) techniques and Large Language Models (LLMs), such as ClinicalBERT, Mistral, Llama2, and Gatortron, are employed and rigorously evaluated to determine the model that exhibits the highest performance for the given task. Hugging face transformers are utilized for the RAG pipeline.

Expected Project Contributions and Applications

The project endeavors to make substantial contributions to the healthcare domain by developing a cutting-edge hospital readmission prediction system driven by Large Language Models (LLMs). Through accurate identification of high-risk patients who may require

additional care, the system empowers healthcare providers to proactively implement tailored management strategies and interventions based on individual patient needs. The system's potential applications span a wide range of areas, including early detection of patients at elevated risk for readmission, decision support for healthcare professionals, personalization of care plans, and optimization of resource allocation within healthcare facilities. The project's anticipated impact encompasses a reduction in readmission rates, leading to improved patient outcomes and enhanced quality of care. Additionally, it holds the promise of significant cost savings for healthcare systems by mitigating the financial burden associated with preventable readmissions. Furthermore, the project aims to streamline healthcare delivery processes, fostering greater efficiency and enabling healthcare providers to allocate resources more effectively.By harnessing the power of advanced natural language processing (NLP) techniques and LLMs, the project seeks to unlock the wealth of information contained within electronic health records (EHRs), including unstructured clinical notes, patient demographics, medical histories, and readmission records. Through the analysis of these diverse data sources, the system aims to provide healthcare professionals with actionable insights, enabling them to identify high-risk patients proactively and implement targeted interventions to mitigate readmission risks. Ultimately, the project's overarching goal is to contribute to the enhancement of patient care quality, reduce the substantial financial burden imposed by readmissions, and minimize patient morbidity associated with these events. By leveraging cutting-edge technologies and fostering collaboration between healthcare providers, researchers, and technology experts, the project holds the potential to drive transformative change in healthcare delivery, paving the way for a more efficient, cost-effective, and patient-centric healthcare system.

1.2 Project Requirements

Functional requirements

The system should include mechanisms for user authentication and access control to safeguard sensitive healthcare data. It must also accommodate seamless data input and preprocessing, allowing for the integration of electronic health records (EHRs) and clinical notes while maintaining data integrity. A robust pipeline is required to finetune the model using the LLM by dividing the data in train,test and validation,then the data should be formatted into a question and answer prompt way to retrieve it from the RAG . Moreover, the system should deliver accurate predictions of hospital readmission risk, providing healthcare providers with actionable insights and decision support.

AI-powered requirements

The AI-driven project requirements necessitate rigorous testing and measurable evaluation metrics to ensure the efficacy and reliability of the predictive model. Model performance benchmarking involves comparing precision, recall, F1 score, AUC-ROC, and AUC-PR of different LLMs and NLP techniques and also it should consider the ROGUE score and BLEU score for the RAG system. Scalability and efficiency are assessed by measuring performance across varying dataset sizes and complexities, including training time, inference time, and memory usage.

Data requirements

Access to comprehensive and high-quality EHR datasets containing relevant clinical information is essential for model training and evaluation.Curating diverse and representative datasets ensures that the model captures variations in patient demographics, clinical conditions, and healthcare practices, enhancing its generalization and predictive accuracy. Additionally,

annotated datasets with ground truth labels facilitate the training and validation of the predictive model using the RAG ensuring consistency and accuracy in performance assessment.

1.3 Project Deliverables

The project has multiple stages and deliverables. Every deliverable has a description with information about the deliverable and a due date. The deliverables start with an abstract that gives an overview of the project's objective, methodology and strategies. Next, a work breakdown structure containing each task and its dependencies will be developed, along with the gantt charts showing dependencies and their dates. The following step is the data management plan, which describes the data collection methods, storage options and uses. Next, the primary phase of query engine development will start by using NLP techniques and Large Language Models (LLMs), including ClinicalBERT, Mistral 7B, Llama, and Gatortron. OpenAI IS then used to assess how accurate the generated results are. Finally, a project overview and report are provided, along with a presentation that summarizes the work. Table 1 lists all of the project's deliverables along with their respective deadlines and descriptions.

Table 1

Project Deliverables and Description

Deliverable	Description	Due Date
Abstract	An overview about the project's objective, goal, and methodology.	14 February, 2024
WBS	A breakdown displaying each project phase along with its subtasks	18 February, 2024

Gantt Chart	A chart visually representing project tasks along with their dependencies with clearly outlining the start and the end date of each task	28 February, 2024
Data Management Plan	A plan outlining data collection approaches, management and storage methods along with its usage	12 March, 2024
Model Development	This step outlines the development utilizing NLP techniques and Large Language Models (LLMs), including ClinicalBERT, Mistral 7B, Llama2, and Gatortron.	30 March, 2024
Model Evaluation	This step works on evaluating the results generated by different LLMs using OpenAI.	15 April, 2024
Final Project	A comprehensive project report summarizing the project objectives, findings, goals and methodology developed along with their results	13 May, 2024
Presentation	A detailed presentation describing the project objectives, goals, solution methodology, results obtained along with its conclusions.	14 May, 2024

1.4 Technology and Solution Survey

The technical survey investigates neural networks such as LSTMs, ClinicalBERT, gradient boosting machines, and other predictive models for hospital readmission. It assesses

metrics like AUROC and interpretability methods like self-attention maps with the goal of improving patient care through the use of data-driven approaches for precise and timely readmission forecasts.

Tung et al. (2022) investigates the application of Natural Language Processing (NLP) and BERT models for classifying medical records. It compares the performance of BERT, BioBERT, and PubMedBERT in classifying medical records into four clinical divisions. BERT, utilizing bidirectional representations from Transformers and a Masked Language Model pre-training approach, is contrasted with BioBERT, tailored for the biomedical domain, and PubMedBERT, trained from scratch using medical literature from PubMedCentral and PubMed. The study utilizes machine learning techniques, particularly NLP models, for text extraction and analysis in medical records. Evaluation metrics include accuracy, precision, recall, micro F1-score, macro F1-score, and weighted F1-score, along with confusion matrices and ROC curves for performance assessment. The findings reveal that PubMedBERT demonstrates the best performance, with 0.90 accuracy, 0.81 recall, and 0.80 weighted F1-score, while BioBERT also shows significant results. The study suggests future research directions, including cross-validation to investigate differences between cased and uncased representations in these models. In conclusion, specialized NLP models like BioBERT and PubMedBERT exhibit promising potential in extracting features from medical records, aiding medical decision-making across clinical divisions, and reducing medical errors, thus enhancing patient care.

Ali et al. (2023) explores the potential of large language models (LLMs), particularly ChatGPT, in revolutionizing healthcare by providing clinical insights and reducing doctors' workload. It discusses the various applications of ChatGPT in healthcare, including automating the generation of patient discharge reports, clinical vignettes, and radiology reports, as well as its

ability to pass medical licensing examinations. While highlighting the opportunities ChatGPT presents in improving patient outcomes and medical decision-making, the paper also addresses the associated risks such as error, misinformation, bias, lack of transparency, and privacy concerns. Case studies demonstrate ChatGPT's capabilities in generating medical reports, diagnosing diseases, and providing clinical reasoning. The paper emphasizes the importance of thorough clinical validation and caution in interpreting results due to potential errors. Additionally, it discusses ethical concerns, human emotions, data privacy, interpretability, standardization, integration challenges, and potential resistance from stakeholders. Despite the risks, ChatGPT offers significant potential to streamline healthcare tasks, enhance medical education, empower patients with personalized healthcare information, and advance medical research. The paper concludes by underscoring the need for ongoing evaluation, adaptation, and consideration of preventive measures in utilizing ChatGPT and other LLMs in healthcare.

Vasantharajan et al. (2022) introduces MedBERT, a novel pre-trained transformer-based model designed specifically for biomedical Named Entity Recognition (NER). NER is crucial in the biomedical domain for extracting entities such as proteins, genes, chemicals, diseases, and more from large volumes of text, facilitating tasks like information retrieval and knowledge extraction. MedBERT is pre-trained using a diverse corpus collected from multiple biomedical-related sources, including N2C2 challenges, BioNLP Corpus, CRAFT Corpus, and biomedical articles from Wikipedia. The pre-training corpus covers a broad spectrum of biomedical domains, ensuring MedBERT's ability to recognize entities across different disciplines. The pre-training process involves tokenization using the WordPiece algorithm and model training, with the network weights initialized from Bio ClinicalBERT. The pre-training corpus and methodology enable MedBERT to capture rich biomedical information.To evaluate

the effectiveness of MedBERT, the paper compares it with four publicly available pre-trained models (BERT, DistilBERT, BioBERT, and Bio ClinicalBERT) on ten biomedical datasets, including BioNLP and CRAFT challenges. The experiments involve fine-tuning each model on the datasets and evaluating their performance based on F1-micro scores. Results show that MedBERT consistently outperforms other models, achieving state-of-the-art performance on nine out of ten test sets. MedBERT achieves superior performance on diverse biomedical NER tasks, surpassing other pre-trained models. Experimental results highlight the importance of domain-specific pre-training for biomedical NLP tasks. The paper's contributions include the development and release of MedBERT, along with experimental results and open-source code for reproducibility and future research. In conclusion, the paper presents MedBERT as a highly effective pre-trained model for biomedical NER, demonstrating its superiority over existing models on various biomedical datasets. MedBERT's versatility and performance make it a valuable tool for biomedical text mining and information extraction tasks. The availability of pre-trained weights and code facilitates its adoption and further research in the biomedical NLP community.

Wang et al. (2021) proposes an automatic medical triage system to address the increasing pressure on hospital triage systems. By classifying patients' symptoms and questions into specific categories, the system aims to improve efficiency and alleviate the workload on healthcare professionals. TriageBert leverages BERT, a state-of-the-art transformer-based model, for text classification in the medical domain. Two models, TriageBertS and TriageBertL, are developed using different data preprocessing strategies and datasets. TriageBertS is trained on a subset of data with the five most frequent symptom tags, while TriageBertL utilizes a larger dataset with 20 categories. The models are fine-tuned using techniques such as cross-validation and early

stopping to optimize performance. The performance of TriageBert is evaluated using metrics such as top1 and top2 accuracies on medical question answering datasets. Results demonstrate high accuracy rates, indicating the effectiveness of the proposed approach in classifying medical texts. The paper also discusses the development of a web application to deploy the triage system for real-world use.TriageBert achieves high accuracy in classifying medical texts, showcasing its potential to streamline hospital triage processes. The paper contributes to the field by introducing TriageBert and providing insights into its development and evaluation. The web application developed for TriageBert facilitates its practical implementation in healthcare settings, enhancing patient triage and resource allocation.In conclusion, the paper presents TriageBert as a promising solution for automating medical triage processes and reducing the burden on hospital staff. The system's accuracy and versatility make it a valuable tool for improving patient care and optimizing healthcare resources. Further research and development can enhance TriageBert's performance and expand its applications in the medical domain.

Ganesh and Bansal (2023) compares four transformer-based models (BERT Base Uncased, Emilyalsentzer Bio_ClinicalBERT, RoBERTa, and DeBERTa) to determine the best backbone model for mapping free text in clinical notes to specific clinical concepts. DeBERTa emerges as the preferred model due to its disentangled attention and enhanced mask decoder. Meta pseudo labeling further enhances DeBERTa's performance. The study proposes the use of DeBERTa for patient note scoring in the USMLE Clinical Skills exam. The discontinuation of the USMLE Step 2 Clinical Skills exam in 2021 created opportunities for improving patient note-taking skills assessment. Extracting specific clinical concepts from free-text clinical notes remains challenging due to diverse representations. Transformer-based models offer potential solutions, but their effectiveness in clinical context needs exploration. Patient notes lack

structure, making traditional rule-based approaches ineffective. Transformer models like BERT, RoBERTa, and DeBERTa show promise but require evaluation for clinical concept mapping. DeBERTa, with its disentangled attention mechanism, outperforms other transformers in various tasks. Experimental research was conducted to compare transformer models. The NBME dataset from the USMLE Clinical Skills exam was used. Data processing involved correcting entries, merging datasets, and exploratory analysis. Models underwent installation, tokenization, training, and testing. Meta pseudo labeling was employed to expand training data. DeBERTa outperforms other models in F1-score, precision, and recall. Meta pseudo labeling significantly improves DeBERTa's performance. The model's attention mechanism, considering both word content and position, contributes to its superiority. Other models show limited improvement with meta pseudo labeling. DeBERTa emerges as the recommended model for clinical concept extraction. Clinical-specific BERT is deemed unnecessary. Future research should explore techniques like Debiased Self-Training and Replaced Token Detection to enhance model performance further.

Oxen et. al (2024) presents Mistral 7B is a recently released language model that is very advanced and capable of understanding and communicating in human language. It has over 7 billion pieces of information that allow it to perform extremely well on various tasks. Mistral 7B outperforms other large language models like Llama 2 and Llama 1 on many benchmarks and tests. It is remarkably good at not just understanding regular English text, but also at tasks involving computer code. Two key things make Mistral 7B so capable. First, it uses a technique called Grouped-query Attention which allows it to process information faster. Second, it has Sliding Window Attention which lets it handle very long pieces of text efficiently. The code and different versions of Mistral 7B are available for anyone to use without restrictions, under an open-source license. You can read more technical details about how it works in the research

paper. One can access and use Mistral 7B through various platforms like HuggingFace, Vertex AI, Replicate, and others. There's even a new way to use it on Kaggle without having to download anything, making it very convenient. Overall, Mistral 7B represents a major advancement in large language models, with impressive performance and the ability to handle complex tasks, all while being openly available for anyone to explore and utilize.

Moerschbacher and He (2023) addresses the critical issue of predicting 30-day readmission rates among ICU patients, which is vital for improving patient outcomes and hospital profitability. By leveraging both structured data (demographics, laboratory tests, comorbidities) and unstructured discharge summaries from the MIMIC-III database, the study evaluates various machine learning models. The best-performing model, Logistic Regression, achieves an AUROC of 75.7%, demonstrating the potential of machine learning and deep learning for predicting ICU readmissions. High ICU readmission rates signal healthcare quality issues and incur substantial costs for hospitals. Predicting ICU readmissions can mitigate these challenges. This study aims to predict 30-day ICU readmission rates using both structured and unstructured data from electronic health records. The study utilizes the MIMIC-III database, containing detailed patient information, including demographics, lab results, and discharge notes. Different machine learning algorithms are employed, including Logistic Regression, XGBoost, Random Forest, Feed Forward Neural Network, and Support Vector Classification, on three datasets: structured, unstructured, and combined. Performance metrics such as accuracy, precision, recall, and AUROC are used for evaluation. Among the models using structured data only, the Random Forest model achieves the highest AUROC of 73.9%. In contrast, the Logistic Regression model using unstructured data attains the highest AUROC of 75.7%. The combined dataset yields a Random Forest model with an AUROC of 70.4%. Deep learning models

generally underperform compared to non-deep learning models. The Random Forest model consistently outperforms other models. However, Logistic Regression with unstructured data exhibits superior performance. Hyperparameter tuning improves deep learning models' performance, but it requires significant time investment. This study demonstrates the efficacy of predicting ICU readmissions using structured and unstructured data. The Logistic Regression model with unstructured data achieves the highest AUROC, indicating the potential of leveraging unstructured data for predictive modeling in healthcare.

Imasogie (2023) provides a comprehensive guide on leveraging ClinicalBERT for predicting hospital readmission using early clinical notes. The paper highlights the significance of predicting hospital readmission using early clinical notes, emphasizing the importance of timely interventions by clinicians. ClinicalBERT, a modified version of BERT specifically trained on clinical notes, is introduced as a suitable model for this task. The authors utilize the MIMIC-III dataset, containing detailed clinical information of ICU patients, for training and evaluation. Detailed preprocessing steps are provided, including text normalization and segmentation of long clinical notes into smaller chunks. The paper describes the model configuration for fine-tuning ClinicalBERT for readmission prediction, including the addition of a classification layer. Strategies for handling long clinical notes, such as splitting them into smaller segments, are discussed. Evaluation metrics such as AUROC, AUPRC, and RP80 (recall at 80% precision) are used to assess the performance of the trained model. Results indicate promising performance with AUROC values of 0.748 for 2-day predictions and 0.758 for 3-day predictions. The paper emphasizes the importance of interpreting model predictions, especially in healthcare settings. A self-attention map visualization is presented to demonstrate which terms in clinical notes contribute to predicting patient readmission, providing insights into the model's

decision-making process. The paper concludes by highlighting the clinical relevance and potential applications of using ClinicalBERT for predicting hospital readmission. Future research directions, such as exploring additional interpretability techniques and integrating the model into clinical workflows, are suggested.

Table 2 provides a concise overview of the approaches, algorithms, comparison methods, and best-performing models discussed in above research papers. Each paper explores innovative solutions to challenges in healthcare using advanced technologies such as Natural Language Processing (NLP), large language models (LLMs), and deep learning.

Table 2

Comparison of the technologies discussed

Paper	Approach/ Model	Algorithm	Comparison Method	Best Performing Model
Multi-model Comparison for Classification of Medical Records	BERT, BioBERT, PubMedBERT	Natural Language Processing (NLP)	Performance metrics including accuracy, precision, recall, micro F1-score, macro F1-score, and weighted F1-score, confusion matrices, ROC curves	PubMedBERT
ChatGPT and	ChatGPT	Large Language	Case studies,	ChatGPT

Paper	Approach/ Model	Algorithm	Comparison Method	Best Performing Model
Large Language Models in Healthcare		Model (LLM)	evaluation of generated reports and diagnoses, discussion of opportunities and risks	
MedBERT: A Pre-trained Language Model for Biomedical NER	MedBERT	Transformer-based model	Comparison with other pre-trained models on biomedical datasets	MedBERT
Reduce the medical burden: An automatic medical triage system	TriageBert	Bidirectional Encoder Representations from Transformers (BERT)	Evaluation of top1 and top2 accuracies, discussion of system development	TriageBertS and TriageBertL
Building Prediction	Logistic Regression,	Machine Learning, Deep	Evaluation based on AUROC,	Logistic Regression with

Paper	Approach/ Model	Algorithm	Comparison Method	Best Performing Model
Models for 30-Day Readmissions	Random Forest, XGBoost, Feed Forward Neural Network, Support Vector Classification	Learning	precision, recall, accuracy, and comparison among models	unstructured data
ClinicalBERT: Using a Deep Learning Transformer Model	ClinicalBERT	Transformer-based model	Evaluation based on AUROC, AUPRC, and RP80, discussion of model interpretation	ClinicalBERT
Fine Tuning Mistral 7B	Mistral 7B, Gemma 7B	Sliding Window, RoPE	AUROC Curve	Mistral 7B

1.5 Literature Survey of Existing Research

The following Literature Survey explores current research and advancements in the use of Large Language Models (LLMs) and Retrieval Augmented Generation (RAG) for healthcare applications, examining developments in artificial intelligence (AI) for healthcare, including the integration of large language models (LLMs) and predictive analytics.

A brief literature study by Yu et al. (2023) aims to offer recommendations for incorporating Large Language Models (LLMs) and generative AI into medical practices and healthcare settings. It highlights the special mechanisms that set these technologies apart from conventional rule-based AI systems, including Reinforcement Learning from Human Feedback (RLFH). The potential of generative AI to enhance human capacities in a variety of information management fields, including healthcare, is also covered by the author. It draws attention to the value of language in medical conversations and the wealth of data found in electronic health records (EHRs). The study concludes by highlighting the revolutionary potential of generative AI and LLMs in healthcare, but also pointing out certain drawbacks, such as the scoping review's lack of exhaustiveness.

Hiren et al. (2023) looks at two very large and powerful language models called Mistral-7B and Llama-2-7B. These models are really good at understanding human language and images. These kinds of large language models are important because they can help computers communicate with people more naturally. The better they understand language, the more useful they can be for things like answering questions, giving advice, or helping with tasks. Mistral-7B and Llama-2-7B use transformers. This allows them to pay attention to different parts of a sentence and understand the order of words. They learn by reading tons of text from the internet and being trained on questions and answers. These models are massive, with billions of pieces of information called parameters. That's what makes them so powerful at understanding language.

The paper tests how well Mistral-7B and Llama-2-7B perform on different language tasks. Llama-2-7B turns out to be better overall, giving responses that are almost as good as a human. Now that Mistral-7B is available for people to use, the paper talks about how it could be useful in real situations. With more data and research, it could get even better at understanding language and images.Overall, this paper explains how these big language models work, how they compare to each other, and why it's important to keep improving them. As they get smarter, they'll be able to help people in all kinds of ways by communicating more naturally.

Ankur et. al (2024) stated that Google and Mistral AI have both created very advanced language models called Gemma 7B and Mistral 7B. These are like super-intelligent computer programs that can understand and communicate in human language. Gemma 7B is really good at writing computer code and solving math problems. Mistral 7B is better at logical reasoning and handling real-world situations that people might encounter. Google allows anyone to use Gemma 7B after agreeing to some rules about safety and privacy. Mistral 7B is also open for public use. Upon testing these language models, they found that each one performed better in certain areas. This shows that the right model to use depends on what kind of task you need it for. The work on these open-source language models highlights how important it is for people to collaborate and share AI technology openly. It allows many minds to contribute and make the systems smarter. As language models like Gemma 7B and Mistral 7B continue to improve, they could become incredibly useful tools to help humans in all kinds of ways by understanding our language and assisting with tasks. But it's crucial that their development remains open and accessible to everyone.

Goel et al. (2023) proposed a method to effectively produce ground truth labels for medical text annotation by fusing human experience with Large Language Models (LLMs). The

suggested method is a two-step process: Base Annotations are generated by the LLM and are subsequently enhanced to produce enhanced Annotations by medical annotation specialists. Through empirical examination, the effectiveness of this approach was confirmed with an emphasis on the medication extraction task. The outcomes demonstrated that LLMs can retain expert-level quality while speeding up the annotating process. In order to guarantee predetermined outputs, future work will investigate optimizing LLMs for particular workloads and incorporating limited decoding.

A ClinicalBERT model was developed by Huang et al. (2020) to process clinical notes and forecast 30-day hospital readmissions. ClinicalBERT discovers connections between medical concepts by using bidirectional encoder representations from transformers (BERT) trained to learn deep representations of clinical literature from the MIMIC-III dataset. In order to help clinicians make decisions, the model constantly changes risk scores based on patient notes.Predictions made by the model can be interpreted using its attention weights, which improves its usefulness and transparency in clinical contexts. Compared to other approaches, ClinicalBERT performed better for predicting readmission, particularly when early clinical notes or discharge summaries were used.

In their exploration of ChatGPT's potential in the healthcare industry, Ali et al. (2023) demonstrate how it can automate processes like creating medical reports and passing licensure examinations. The applications of ChatGPT in passing the USMLE, creating patient discharge reports, clinical vignettes, and radiology reports are illustrated via case studies completed by the authors. It draws attention to prospects for research advancement, medical education improvement, knowledge extraction from unstructured text, tailored health advice, and report summary.

According to Singhal et al. (2023), there has been a notable breakthrough in medical question answering with models like Mistral 7B2 exhibiting promising capabilities. Large language models (LLMs) have also been a focus of recent AI advancements. By utilizing advancements in basic LLMs, domain-specific fine-tuning, and creative prompting techniques, Mistral 7B2 fills in the gaps left by earlier models. Mistral 7B2 was assessed by the authors using MultiMedQA's multiple-choice and long-form medical question-answering datasets. Mistral 7B2's strengths in answering long-form and multiple-choice questions are demonstrated by human evaluations, which include physician assessments and adversarial question datasets. The fact that doctors typically choose Mistral 7B2's responses above those from other doctors shows that the system is moving closer to physician-level performance. The study does, however, recognize certain shortcomings, including the need for ongoing improvement in assessment techniques and the evaluation of empathy expressed by model outputs.

Jin et al. (2024) presented Health-LLM, a novel framework for intelligent healthcare that combines medical knowledge scoring and large-scale feature extraction to overcome shortcomings in conventional approaches. Based on patient health reports, the system uses machine learning, data analytics, and medical knowledge to forecast and prevent future health concerns. For illness prediction, the authors employed the XGBoost classification model, automated feature engineering, and the Llama Index framework. Health-performance LLM's comparative analysis with other LLMs shows how well it predicts diseases and provides individualized health advice. It draws attention to the notable advancements made possible by Health-LLM in terms of illness prediction and personalized health advice.

Ke et al. (2024) introduced a novel method for producing preoperative instructions in the healthcare industry that makes use of Large Language Models (LLMs) improved with Retrieval

Augmented Generation (RAG). The LLM-RAG model was created and evaluated using 35 preoperative criteria in comparison to responses that were created by humans. The findings showed non-inferiority, with the GPT4.0-RAG model achieving excellent accuracy (91.4%) in contrast to human-generated instructions (86.3%). In addition, the model had response speeds that were 15–20 seconds quicker than those of humans (10 minutes). These results demonstrate the potential of LLM-RAG pipelines in the healthcare industry, providing precision and effectiveness in producing intricate medical recommendations. The study also emphasizes the significance of scalability and grounded knowledge for successful implementation in clinical settings.

In order to accurately respond to medical questions about liver cancer, Qian et al. (2024) developed the Liver Cancer Question-Answering System (LCQAS), which makes use of the big model Mistral 7B 2 and next-generation artificial intelligence. According to the study, Mistral 7B 2, an enhanced version of the original model, showed notable advancements in medical question answering, attaining higher performance and accuracy levels. The authors highlighted the versatility and wide application of Mistral 7B 2 across a range of tasks and sectors when contrasting it with GPT-4. It also includes results demonstrating how Mistral 7B 2 performed better in terms of relevancy and quality of response to medical inquiries than both human doctors and its predecessor. This development represents a substantial step forward in the retrieval of medical information, helping patients, researchers, and healthcare providers by offering thorough and accurate answers to questions about liver cancer and chronic diseases.

Elgedawy et al. (2024) addressed the difficulties associated with extracting and interpreting data from electronic health records (EHRs) by presenting a conversational question-answering system driven by cutting-edge Large Language Models (LLMs). The authors

highlighted how LLMs' contextual understanding and generative capabilities could help them overcome the shortcomings of more established information retrieval techniques like TF-IDF, Boolean models, and probabilistic models in capturing the nuances and semantic context of natural language text. The study describes the process of creating a conversational question-answering system using LLMs, including trials carried out to maximize accuracy and speed. Future directions for this field of study are provided by the paper's discussion of model performance, assessment rigor, and real-world deployment constraints.

In order to support clinical diagnosis and medicine, Tung et al. (2022) investigated the efficacy of NLP (Natural Language Processing) and BERT (Bidirectional Encoder Representations from Transformers) models in extracting important data from medical records. The study emphasizes the value of medical records in documenting individuals' health journeys, but it also points out that the sheer volume and lack of uniformity in EHR systems make it difficult to reliably extract important information. Authors introduced and compared ordinary BERT models with two specialized pre-trained language representation models designed for biomedical domains: BioBERT and PubMedBERT. According to the results, PubMedBERT-case-unbased, with its high accuracy, recall, and weighted F1-score, outperformed the other BERT models. Following PubMedBERT's successful analysis of medical records due to its pre-training on professional biomedical literature from PubMed, BioBERT was also trained on biomedical corpora. was able to analyze medical records more successfully than BERT. BERT-base-cased, on the other hand, was less successful at comprehending medical records than it was in general-purpose NLP. In order to further differentiate between these models with cased and uncased representations, future studies may investigate cross-validation.

The problem of discrepancies between textual drug reviews and numerical ratings on websites such as Drugs.com was tackled by Shiju et al. (2022). In order to categorize medication review ratings based on textual reviews, the authors developed classification models, which included transformer-based deep learning models and conventional machine learning models. With an accuracy of 87%, the Bio_ClinicalBERT model surpassed the others. Additionally, the study investigates the semantic categories of Unified Medical Language System (UMLS) ideas found in reviews. Limitations include the overfitting and processing costs and future research will concentrate on multi-class identification and unscored social media data.

In example, Ganesh et al. (2023) compared transformer-based models for patient note scoring in the USMLE Clinical Skills exam, which involves mapping free text in clinical notes to certain clinical ideas. DeBERTa outperformed the other three models—BERT, Bio_ClinicalBERT, RoBERTa, and DeBERTa—because of its improved mask decoder and disentangled attention. The effects of meta pseudo labeling and context-specific embeddings on model performance were also investigated in this work. It is decided that BERT Base Uncased performs similarly to clinical-specific BERT, negating the need for it for USMLE grading. Among the suggestions are investigating Debiased Self-Training for mistake reduction and employing BERT distillation to minimize computing resources. Subsequent studies could examine additional improvements like using Replaced Token Detection (RTD) and separating classifier heads.

Yang et al. (2023) examined the LLM applications in the healthcare industry, concentrating on ChatGPT and other specific LLMs designed for the biological and clinical domains. The creation of domain-specific LLMs for the healthcare industry, including GatorTron, BioBERT, SCIBERT, PubMedBERT, and ClinicalBERT, is explored in this article.

To improve their performance in medical NLP tasks, these models were trained or adjusted on specialist datasets such as biological literature or clinical records. The significance of domain-specific information in enhancing LLM performance for clinical and biological applications is emphasized in the paper. The study also examines the conversational potential of LLMs in the healthcare industry, presenting examples such as Mistral 7B, ChatDoctor, and Baize-health care, which are intended to support doctors and patients through interactive communication.

GatorTronGPT, a generative clinical large language model (LLM) created especially for medical purposes, was presented by Cheng et al. (2023). A vast dataset of 277 billion words, which included both ordinary English text and clinical text from the University of Florida Health, was used to train GatorTronGPT. The results show that GatorTronGPT can produce synthetic clinical text and enhances biomedical NLP ability. GatorTronGPT-generated text is used to train synthetic NLP models, which perform better than models trained on actual clinical text. The importance of synthetic clinical text generation in resolving privacy-related barriers to large-scale clinical dataset access and sharing is discussed in the study. GatorTronGPT makes it possible to train NLP models without disclosing private medical data by producing synthetic text.

Li et al. (2023) addressed the shortcomings found in popular models such as ChatGPT in an effort to improve the precision of large language models (LLMs) in delivering medical advice. Using a dataset of 100,000 patient-doctor conversations from an online medical consultation platform, the authors improved the LLaMA model and included a self-directed information retrieval mechanism. The model's ability to comprehend patient demands and offer wise counsel was greatly enhanced by its fine-tuning. It saw significant gains in response accuracy when the

model was accessed for real-time information retrieval from offline medical databases and online sources like Wikipedia. Our suggested ChatDoctor paradigm, which shows enhanced comprehension of patient queries and precise advice delivery, constitutes a noteworthy breakthrough in medical LLMs. In the medical domain, where mistakes can have grave repercussions, these models' increased reliability is essential. To test and improve the ChatDoctor model for practical clinical usage, more investigation is required. Security measures that guard against errors and delusions should also be developed. However, ChatDoctor has the potential to increase access to high-quality medical consultations, especially in underprivileged areas, decrease the workload of medical personnel, and improve the accuracy of medical diagnoses.

Comparing pre-training a strictly clinical language model with different ways for adapting a general language model to the clinical domain was done by Lamprudis et al. in 2022. Domain-specific models outperform generic ones, according to the results of pretrained and fine-tuned three clinical language models for Swedish on many downstream clinical tasks. It takes less pre-training epochs to leverage an existing generic language model than to train a new model from the start, even though there is minimal difference in performance between the clinical language models. The study validates the advantages of domain-specific language models; the domain-adapted clinical language model with a clinical vocabulary is the best-performing model. Pretraining a new model within the domain and modifying a general language model for the clinical domain yield very little difference in terms of performance.

McCleary et al. (2024) conducted a study on applying LoRA (Low-Rank Adaptation) fine-tuning of small language models, specifically Mistral's 7B model, in order to perform TNM (Tumor, Lymph Node, Metastasis) staging in unstructured pathology reports of triple negative breast cancer cases. The dataset was provided by the Louisiana Tumor Registry and it has around

200 digital pathology reports. The data was manually labelled by subject matter experts as the reports did not have ground truth labels. They developed a process to generate new synthetic reports and data which are crucial for obtaining more training and validation data samples. Mistral's 7B Instruct model was fine-tuned using LoRA (Low-Rank Adaptation) and Axolotl library to target specific modules in the model. For fine tuning, various sample counts were used with the most extensive being 1600 samples for 4 epochs, taking around 17 hours on an RTX 4090 GPU. The model performed four passes during evaluation which are tumor measurement extraction, N category assessment, T category assessment, and M category assessment. Along with those, grammar based LLM sampling and JSON output enforcement were used in evaluation. The results showed that GPT-3.5 and GPT-4 demonstrated good performance. Mistral 7B Instruct fine tuning was also done with 100 samples for 16 epochs which increased performance by over 70%. The best result achieved has an accuracy of 96.5% for all three TNM categories. A detailed summary and comparison of the research is provided in Table 3.

Table 3

Comparison of the literature survey

Author	**Paper**	**Model**	**Techniques**	**Results**
Huang et al. (2020)	ClinicalBERT: Modeling Clinical Notes and Predicting Hospital Readmission	Clinical BERT, BERT, BI-LSTM	Self Attention Mechanism, clinical text embedding, comparison against word embedding models.	BERT BERT didn't perform well because it didn't have clinical data to train on.Clinical BERT outperformed other models and accurately predicts

Author	Paper	Model	Techniques	Results
				30-day readmissions.
Ankur et al. (2024)	Comparative Analysis Gemma 7B vs Mistral 7B	Gemma 7B, Mistral 7B	Multi Query Attention, RoPe, GeGLU	Gemma 7B works good for math and Mistral AI works good for diverse set of languages
Hiren et al. (2023)	Comprehensive Examination of Instruction-Based Language Models: A Comparative Analysis of Mistral-7B and Llama-2-7B	LLama2-7B, Mistral 7B	Grouped Query attention, Sliding window	Llama-2-7B turns out to be better overall giving responses that are almost as good as a human while Mistral 7B can be made better by training on more data

Author	Paper	Model	Techniques	Results
Jin et al. (2024)	Health-LLM: Personalized Retrieval-Augmented Disease Prediction System	Health-LLM, GPT-4, finetuned-LLaMa 2	LLaMa Index Framework, RAG, training XGBoost classification model for final disease prediction.	A combination of GPT-4 and RAG results in an accuracy of 0.68 and a F1 score of 0.71 for detecting disease. As for Health-LLM, accuracy was 0.833 and F1 was 0.762.
Ganesh et al. (2023)	Transformer-based Automatic Mapping of Clinical Notes to Specific Clinical Concepts	Clinical BERT, Emilyalsentzer Bio_Clinical BERT, RoBERTa, and DeBERTa	Annotation mapping, meta pseudo labeling,Masked language modeling.	In comparison with other models, DeBERTa achieved high recall, precision, and F1 scores because of its disentangling attention & enhancing masks decoder
McCleary et al. (2024)	TNMTumorClassification from Unstructured Breast Cancer Pathology Reports using LoRA Fine	Mistral 7B, GPT 3.5, GPT 4	Mistral 7B fine tuned using LoRA and Axolotl library	Fine tuning Mistral 7B Instruct model increased the performance compared to GPT 3.5 & GPT 4.

Author	Paper	Model	Techniques	Results
	Tuning of Mistral 7B			
Li et al. (2023)	ChatDoctor: A Medical Chat Model Fine-Tuned on a Large Language Model Meta-AI (LLaMA) Using Medical Domain Knowledge	Chat Doctor Model using LLaMA-7b model	Development using knowledge brain, patent-physician conversation dataset	It was demonstrated in a comparative study between ChatGPT and ChatDoctor that ChatGPT could not recognize the word Mpox while ChatDoctor was able to respond precisely.

Data and Project Management Plan

2.1 Data Management Plan

To begin with, the hospital readmission system using LLM's the system relies on the MIMIC III dataset , it contains around 50,000 patient records including their demographics, name, clinical notes, treatment procedures, post-discharge records, and mortality. As part of the admission prediction system two tables from the MIMIC III dataset were taken which are crucial in analysis as they possess the information about the patient demographics in the admission table and other one is note events table which has the recorded clinical notes of a patient during his presence in a hospital.Both these tables will be preprocessed and joined on the common columns and use that combined dataset to fine tune the LLM and also retrieve the data using RAG model.

The MIMIC III dataset was extracted from Physionet which is an open source on the internet.

The data is initially in compressed csv format, containing around 26 different files with a total size of about 81GB, with each file varying by size. Data is being managed using Google Drive by downloading the compressed csv format files and converting them to csv files by using the Python scripts.

Note events and Admission tables are taken from csv and then stored in Google drive with a limited access to teammates to ensure security , so users can access it easily and cost-effectively without downloading or storing it locally, which are going to be stored in different folders to differentiate formats and to make it easy to access depending on the task that should be performed.

Dataset

The MIMIC-III database is built upon a relational model consisting of 26 tables, each having a unique identifier indicated by the suffix 'ID'. An example of this is the SUBJECT_ID, which is used to identify a particular patient, as well as the HADM_ID which is used to identify a hospital admission shown in Figure 1, and the ICUSTAY_ID which represents an intensive care unit admission. 'Events' tables are used to store clinical information such as notes, laboratory tests, fluid balance, and fluid balance calculations. OUTPUTEVENTS is a table that contains measurements of output, while LABEVENTS is a table that contains results of laboratory tests. The tables that start with the prefix 'D_' are dictionary tables that provide definitions for identifiers found in the database. In the case of CHARTEVENTS, for instance, each row is associated with an ITEMID that represents a concept that will be measured. MIMIC is an iteratively developed data model that aims to simplify data models while maintaining fidelity to the underlying data sources at the same time. In terms of the NOTE_EVENTS table

the data present in that is the clinical notes of the patient that has been recorded for a patient during the time that he has spent in hospital shown in Figure 2.There are tables that contain patient stays, dictionaries that cross-reference codes, information on physiological measurements, observations by caregivers, and billing information contained in this database. The preference is to maintain the independence of some tables for clarity reasons, even though two tables could be merged.

Figure 1

Admissions Table Raw Dataset

ROW_ID	SUBJECT_ID	HADM_ID	ADMITTIME	DISCHTIME	DEATHTIME	ADMISSION_TYPE	ADMISSION_LOCATION	DISCHARGE_LOCATION	INSURANCE	LANGUAGE	RELIGION	MARITAL_STATUS	ETHNICITY
21	22	165315	2196-04-09 12:26:00	2196-04-10 15:54:00		EMERGENCY	EMERGENCY ROOM ADMIT	DISC-TRAN CANCER/CHLDRN H	Private		UNOBTAINABLE	MARRIED	WHITE
22	23	152223	2153-09-03 07:15:00	2153-09-08 19:10:00		ELECTIVE	PHYS REFERRAL/NORMAL DELI	HOME HEALTH CARE	Medicare		CATHOLIC	MARRIED	WHITE
23	23	124321	2157-10-18 19:34:00	2157-10-25 14:00:00		EMERGENCY	TRANSFER FROM HOSP/EXTRAM	HOME HEALTH CARE	Medicare	ENGL	CATHOLIC	MARRIED	WHITE
24	24	161859	2139-06-06 16:14:00	2139-06-09 12:48:00		EMERGENCY	TRANSFER FROM HOSP/EXTRAM	HOME	Private		PROTESTANT QUAKER	SINGLE	WHITE
25	25	129635	2160-11-02 02:06:00	2160-11-05 14:55:00		EMERGENCY	EMERGENCY ROOM ADMIT	HOME	Private		UNOBTAINABLE	MARRIED	WHITE
26	26	197661	2126-05-06 15:16:00	2126-05-13 15:00:00		EMERGENCY	TRANSFER FROM HOSP/EXTRAM	HOME	Medicare		CATHOLIC	SINGLE	UNKNOWN/NOT SPECIFIED
27	27	134931	2191-11-30 22:16:00	2191-12-03 14:45:00		NEWBORN	PHYS REFERRAL/NORMAL DELI	HOME	Private		CATHOLIC		WHITE
28	28	162569	2177-09-01 07:15:00	2177-09-06 16:00:00		ELECTIVE	PHYS REFERRAL/NORMAL DELI	HOME HEALTH CARE	Medicare		CATHOLIC	MARRIED	WHITE
29	30	104557	2172-10-14 14:17:00	2172-10-19 14:37:00		URGENT	TRANSFER FROM HOSP/EXTRAM	HOME HEALTH CARE	Medicare		CATHOLIC	MARRIED	UNKNOWN/NOT SPECIFIED
30	31	128652	2108-08-22 23:27:00	2108-08-30 15:00:00	2108-08-30 15:00:00	EMERGENCY	TRANSFER FROM HOSP/EXTRAM	DEAD/EXPIRED	Medicare		CATHOLIC	MARRIED	WHITE
31	32	175413	2170-04-04 08:00:00	2170-04-23 12:45:00		ELECTIVE	PHYS REFERRAL/NORMAL DELI	HOME	Medicaid		UNOBTAINABLE		WHITE
32	33	176176	2116-12-23 22:30:00	2116-12-27 12:05:00		EMERGENCY	EMERGENCY ROOM ADMIT	HOME	Medicare		PROTESTANT QUAKER	MARRIED	UNKNOWN/NOT SPECIFIED
33	34	115799	2186-07-18 16:46:00	2186-07-20 16:00:00		EMERGENCY	TRANSFER FROM HOSP/EXTRAM	HOME	Medicare	ENGL	CATHOLIC	MARRIED	WHITE
34	34	144319	2191-02-23 05:23:00	2191-02-25 20:20:00		EMERGENCY	CLINIC REFERRAL/PREMATURE	HOME HEALTH CARE	Medicare	ENGL	CATHOLIC	MARRIED	WHITE
35	35	166707	2122-02-10 11:15:00	2122-02-20 15:30:00		ELECTIVE	PHYS REFERRAL/NORMAL DELI	HOME HEALTH CARE	Medicare		CATHOLIC	DIVORCED	WHITE
36	36	182104	2131-04-30 07:15:00	2131-05-08 14:00:00		EMERGENCY	CLINIC REFERRAL/PREMATURE	HOME HEALTH CARE	Medicare	ENGL	NOT SPECIFIED	MARRIED	WHITE
37	36	122659	2131-05-12 19:49:00	2131-05-25 13:30:00		EMERGENCY	EMERGENCY ROOM ADMIT	REHAB/DISTINCT PART HOSP	Medicare	ENGL	NOT SPECIFIED	MARRIED	WHITE
38	36	165660	2134-05-10 11:30:00	2134-05-20 13:16:00		ELECTIVE	PHYS REFERRAL/NORMAL DELI	LONG TERM CARE HOSPITAL	Medicare	ENGL	NOT SPECIFIED	MARRIED	WHITE
39	37	188670	2183-08-21 16:48:00	2183-08-26 18:54:00		EMERGENCY	EMERGENCY ROOM ADMIT	HOME HEALTH CARE	Medicare		JEWISH	MARRIED	WHITE
40	38	185910	2166-08-10 00:28:00	2166-09-04 11:30:00		EMERGENCY	TRANSFER FROM HOSP/EXTRAM	LONG TERM CARE HOSPITAL	Medicare		CATHOLIC	WIDOWED	WHITE

Figure 2

Notevents Tables Raw Dataset

ROW_ID	SUBJECT_ID	HADM_ID	CHARTDATE	CHARTTIME	STORETIME	CATEGORY	DESCRIPTION	CGID	ISERROR	TEXT
174	22532	167853.0	2151-08-04	NaN	NaN	Discharge summary	Report	NaN	NaN	Admission Date: [**2151-7-16**] Dischar...
175	13702	107527.0	2118-06-14	NaN	NaN	Discharge summary	Report	NaN	NaN	Admission Date: [**2118-6-2**] Discharg...
176	13702	167118.0	2119-05-25	NaN	NaN	Discharge summary	Report	NaN	NaN	Admission Date: [**2119-5-4**] D...
177	13702	196489.0	2124-08-18	NaN	NaN	Discharge summary	Report	NaN	NaN	Admission Date: [**2124-7-21**] ...
178	26880	135453.0	2162-03-25	NaN	NaN	Discharge summary	Report	NaN	NaN	Admission Date: [**2162-3-3**] D...

2.2 Project Development methodology

For effective project management, it is imperative to establish clear project objectives, success criteria, and scope. To manage expectations and resources efficiently, specific goals and metrics must be defined to measure and track success. Further, a comprehensive literature review is essential for exploring existing research and methodologies associated with readmission

prediction in healthcare. As a result of this review, valuable insights can be gained that help to make the project more effective and to align it with modern best practices and cutting-edge advancements in the industry. In order to achieve success, the project must lay a solid foundation through meticulous planning and objective definition. A data-driven project must start with accessing and understanding the dataset. In this phase, one obtains access to the MIMIC-III dataset and becomes familiar with data access procedures via platforms such as PhysioNet. To ensure that the data is suitable for analysis, the data must be explored, cleaned, and transformed as soon as it is obtained. A well-prepared dataset can be modeled by performing tasks like exploratory data analysis (EDA) and feature engineering. During this phase, the data will be cleaned, organized, and prepared for subsequent analysis and model development. The project involves training and saving four large language models (LLMs) within the context outlined earlier: ClinicalBERT, LLama2, Mistral 7B, and Gatortron. The Hugging Face Transformers library will be utilized for streamlined deployment of each of these models in the Retrieval-Augmented Generation (RAG) setup. Clinical and medical context-related documents or passages will be included in a comprehensive knowledge base. A healthcare-specific dataset will be used to fine-tune the RAG model, optimizing its performance as a medical inquiry tool. Various metrics tailored to the healthcare domain, such as Precision, Recall, and F1 Scores, will be defined. The accuracy of the model will be evaluated by creating a dataset with queries and the corresponding ground truth answers. The strengths and weaknesses of each retriever in handling medical queries will be determined by analyzing the results. In order to refine the models iteratively, parameters will be updated or models retrained based on the analysis. Data will be split correctly, biases addressed, and models regularly updated to maintain their accuracy. For real-world deployment, a user interface (UI) will be created using frameworks like Django to

facilitate user interaction with the deployed models. An integrated LLM model will ensure seamless user interaction with the UI and facilitate effective stakeholder interaction with the model. The documentation of the entire project, including methodologies, processes, outcomes, and insights, is essential for knowledge dissemination and future reference. In addition to ensuring transparency and reproducibility, comprehensive documentation will facilitate stakeholders' access to knowledge. As a result of diligently documenting the project, the team can leverage the insights and learnings for future research and projects.

2.3 Project Organization Plan

Work Breakdown Structure (WBS)

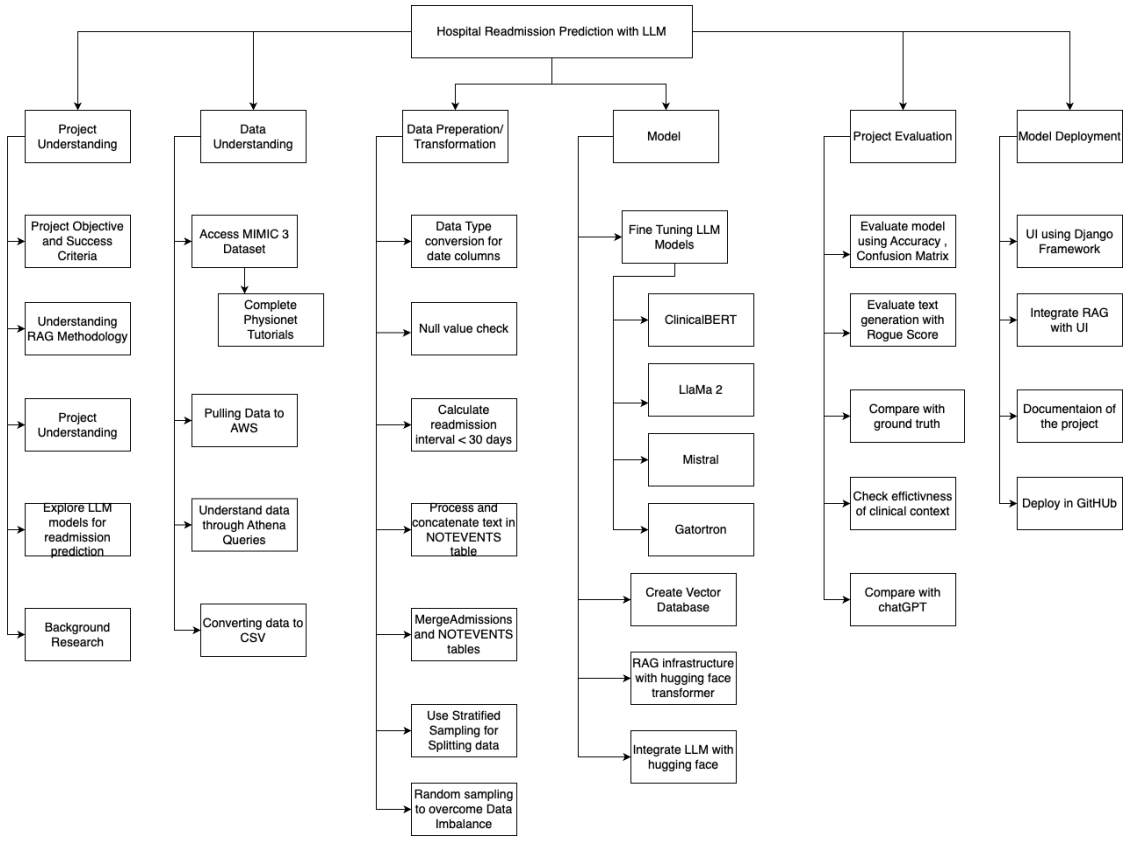
A work breakdown structure (WBS) acts as a roadmap for projects, breaking down main tasks into smaller, more manageable components. It's invaluable for understanding project scope and task requirements. Picture it as a detailed itinerary, guiding everyone through the project from beginning to end. By breaking down work into smaller parts, it clarifies objectives and makes them more attainable. Moreover, it facilitates planning, resource allocation, and progress tracking. Without a WBS, navigating a project might feel directionless, but with one, it can progress smoothly, knowing precisely which steps to take to reach the goals.

Within this project, the WBS arranges various aspects, including identifying business requirements, obtaining data, processing data, choosing models, training models, evaluating models, and deploying models. The project's framework follows the CRISP-DM methodology, where each CRISP-DM phase aligns with a stage in the WBS structure, employing the Waterfall model. The CRISP-DM methodology encompasses six phases, each mirroring a stage in the WBS. Subtasks are evenly distributed among team members to prevent uneven workload allocation.

The below Figure 3 showcases the different phases of the WBS. The work breakdown structure (WBS) for the project "Hospital Readmission Prediction using LLMS" is structured meticulously to cover all essential stages of the project lifecycle. It begins with Project Understanding, where the project objectives and success criteria are defined, alongside background research and a comprehensive exploration of the RAG methodology. This phase also includes a thorough literature survey and an examination of LLM models for readmission prediction, setting a strong foundation for subsequent tasks.

Figure 3

Work Breakdown Structure (WBS)



Moving into the Data Understanding phase, the WBS outlines steps for accessing the Mimic-3 dataset, completing tutorials on PhysioNet (HIPAA Certification) and converting to csv files, and pulling data for further analysis..

The Data Preparation/Transformation stage delves into data transformation, and setting up data for modeling, followed by crucial tasks like data ingestion, exploratory data analysis (EDA), feature engineering, and pre-processing. Further, splitting the dataset for training, validation, and testing ensures robust model evaluation, with documentation of the data preparation process promoting reproducibility and transparency.

The Project Modeling phase integrates LLMs into RAG, trains various models like ClinicalBERT, LLama, Mistral, and Gatortron, and fine-tuning it for optimal performance. RAG infrastructure with Hugging Face Transformers is established, and a knowledge base with vector tables is created to enrich the model's understanding. The Evaluation stage involves comparing results with clinical content and ground truth. Model performance is assessed using standard metrics like accuracy, precision, recall, and F1-score, with the most effective LLM model selected based on comparison.Text Generation is evaluated based on Rogue Score Interpretation of model predictions provides valuable healthcare insights, while comparisons with other models, like ChatGPT from OpenAI, offer additional context. Finally, Model Deployment encompasses creating a user interface using Django and documenting final workbooks and reports, ensuring seamless integration of models for user interaction.

2.4 Project Resource Requirement and Plan

This segment offers an extensive examination of the essential requirements for the project, covering hardware and software prerequisites, specialized equipment and licenses, and specific specifications, including associated costs.

Software Requirements

For this project, a suite of software tools and libraries tailored to meet specific requirements is essential for effectively managing, analyzing, and deploying predictive models

for hospital readmission prediction. Python serves as the core language for backend development, data preprocessing, and model implementation, leveraging its extensive ecosystem of libraries and tools. Frameworks such as TensorFlow and PyTorch offer comprehensive tools and algorithms for developing, training, and evaluating predictive models, ensuring robust performance and scalability. The Hugging Face Transformers library specializes in working with pre-trained language models, enabling integration of advanced NLP techniques such as retrieval-augmented generation into the predictive model pipeline. Django facilitates the development of user interfaces for model deployment, providing a seamless experience for healthcare providers to interact with the predictive system. Jupyter Notebooks offer an interactive coding environment, supporting collaborative data exploration, model prototyping, and documentation, enhancing the project's transparency and reproducibility. Table 4 gives an overview of the software requirements.

Table 4

Software Requirements

Software Component	**Description**	**Purpose**
Python	Programming language	Core language for backend development, data preprocessing, and model implementation,leveraging its extensive ecosystem of libraries and tools,and also important in downloading various packages

Software Component	Description	Purpose
Database Management System	Querying and managing datasets	Facilitates data exploration and analysis through efficient querying and management of datasets.
Machine Learning Frameworks	Libraries for model development (e.g., TensorFlow, PyTorch)	Offers comprehensive tools and algorithms for developing, training, and evaluating predictive models, ensuring robust performance and scalability.
Hugging Face Transformers	Library specialized in working with pre-trained language models	Enables integration of advanced NLP techniques such as retrieval-augmented generation into the predictive model pipeline.
Django	Web framework	Facilitates the development of user interfaces for model deployment, providing a seamless experience for healthcare providers to interact with the predictive system.

Hardware Requirements

The recommended hardware for the local machine, as outlined in Table 5, encompasses an 8-core CPU and an 8-core GPU, offering efficient data storage and computing capabilities. Alongside the CPU and GPU, a minimum of 8 GB RAM is deemed essential to support seamless data processing, model training, and overall system performance. Additionally, for optimal data accessibility and responsiveness, it is advisable to incorporate a solid-state drive (SSD) with a capacity of 256 GB. These hardware configurations aim to provide a balanced and capable system capable of meeting the computational demands of diverse data science tasks. Furthermore, exclusive access to High-Performance Computing (HPC) or GPU resources for students of the Department of Applied Data Science supplements the local machine resources, facilitating advanced computational tasks and training of sophisticated models.Table 5 gives an overview of the hardware requirements.

Table 5

Hardware Requirements

Hardware Component	**Description**	**Purpose**
Local Machine	CPU 8 Core GPU 8 Core RAM 8 GB SSD 256 GB	ForProject Management, Data Understanding, RAG and LLM development
HPC / GPU	High-performance computing (HPC) accelerator	Efficiently handles complex deep learning tasks, LLM and RAG

Tools and Licenses:

The selection of tools for this project is the result of careful consideration aimed at optimizing productivity and streamlining project workflows. Jupyter Notebook, known for its collaborative features and robust support for interactive coding and visualizations, serves as the

cornerstone for data exploration and model development. Jira, offered for free, acts as a central project management and issue tracking solution, ensuring efficient workflow management. GitHub, another open-source platform, is instrumental in providing robust version control for code management, fostering seamless collaboration among team members. Additionally, draw.io facilitates chart design, enhancing project visualization, while Google Docs facilitates documentation and team collaboration, promoting effective communication and knowledge exchange throughout the project lifecycle. These tools have been meticulously chosen based on their functionality, accessibility, and their significant contributions to the overall project success. Table 6 gives an overview of the tools and licenses.

Table 6

Tools and license

Tool	License	Justification
Django	Open source	Development of user interfaces for model deployment, ensuring a seamless user experience.
Jira	Free	Project management
GitHub	Open source	Code management and Version Control
AWS	Free Trial (Student Credits)	Scalable infrastructure and robust data management solutions for handling large-scale datasets.
draw.io	Free	Designing flow charts
Google Docs	Free	Documentation

Project Cost Estimation and Justification

The total cost for the project amounts to $0.00, as all resources and tools utilized are available for free or within the scope of a complimentary trial period. Leveraging local machine hardware, including a 64-bit machine, incurs no additional expenses. Data storage is managed through free resources such as S3 Bucket and Google Drive, while data transformation is facilitated by Athena, also available at no cost. GitHub serves as the version control system, Jira for project management, Zoom for project meetings, Draw.io for chart designing, and Physionet as the data source, all of which are accessible without financial implications. This cost-effective approach ensures efficient resource utilization, allowing for seamless project execution within budgetary constraints.

Table 7

Project Cost Estimation

Functionality	Type	Resource	Duration	Cost
Local Machine	Hardware	64-Bit machine	6 Months	Free
Data Storage	Software	S3 Bucket	6 Months	Free
Data Storage	Software	Google Drive	6 Months	Free
Functionality	**Type**	**Resource**	**Duration**	**Cost**

Data Transformation	Software	Athena	6 Months	Free
Version Control	Software	GitHub	6 Months	Free
Project Management	Software	Jira	6 Months	Free
Project Meeting	Software	Zoom	6 Months	Free
Charts	Software	Draw.io	6 Months	Free
Data Source	Software	Physionet	6 Months	Free
			Total	**$0.00**

2.5 Project Schedule

Gantt Chart

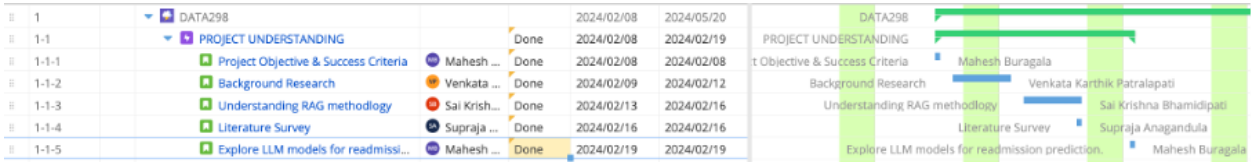
The Gantt chart aids in project scheduling and management by visually representing tasks, durations, and dependencies on a timeline. Using a Gantt chart, it will be able to view a clear overview of the task sequences and deadlines related to "Hospital Readmission Prediction Using LLMS" project. It helps allocate resources efficiently, track progress, and identify potential bottlenecks. In order to ensure smooth project execution, tasks such as data access, modeling training, and deployment are organized chronologically in order to ensure a smooth

process. As a result, the dependencies between tasks are easily visualized, allowing adjustments to be made in a timely manner. By using a Gantt chart, this project will be able to enhance communication between team members and promote accountability, which ultimately facilitates the successful completion of a project.

According to Figure 4, the Business/Project Understanding Gantt chart covers the period of February 8th through February 19th, 2024, outlining tasks related to project understanding for the period. An important part of this project is the definition of project objectives and success criteria, gathering background information, gaining an understanding of RAG methodology, conducting a comprehensive literature survey, as well as investigating LLM models for readmission prediction. During the assigned time frame, each task is assigned a specific duration, allowing a clear timeline for the completion of all activities necessary to complete the project understanding process. As a visual roadmap, the Gantt chart serves as a powerful tool for planning and coordinating the project's initial phases in an effective and efficient manner.

Figure 4

Project Understanding Gantt

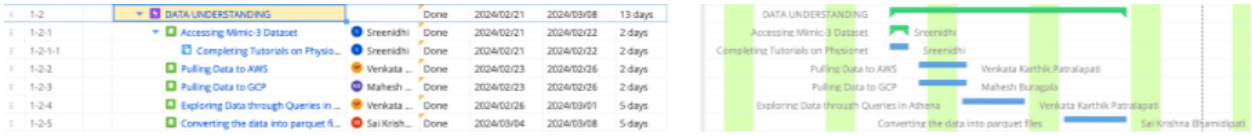


According to Figure 5, the Data Understanding Gantt chart covers the period from February 21st, 2024 to March 8th, 2024, and it focuses on tasks related to the understanding of the project's data during that period. As part of this curriculum, students will be responsible for an array of activities such as gaining access to Mimic-3 data, completing tutorials on Physionet, pulling data and converting data into csv files. A specific timeframe is assigned to each task within this period, which allows a structured approach to understanding the project's data

landscape to be created. It is intended to serve as a visual representation of the timeline for the data-related activities during this phase of the project, thereby facilitating efficient planning and implementation during this time period.

Figure 5

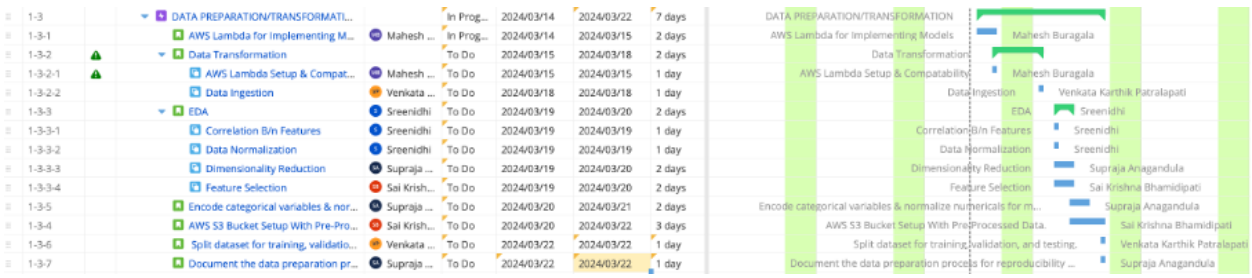
Data Understanding Gantt



According to Figure 6, the Data Preparation/Transformation Gantt chart spans from March 14th through March 22nd, 2024 and it highlights the steps that must be taken in order to prepare and transform the data for the project,By converting it into csv files, ingesting data, conducting exploratory data analysis (EDA), examining correlations between features, normalizing data, reducing dimensionality, selecting features, encoding categorical variables, and normalizing numerical data for modeling. As part of this project the pre-processed data, dividing the dataset for training, validation, and testing, and documenting the data preparation process in order to ensure reproducibility and transparency. Organizing data preparation and transformation tasks on a Gantt chart provides a structured timeline that can assist in the efficient management of projects during this phase by providing a structured timeline for executing key tasks.

Figure 6

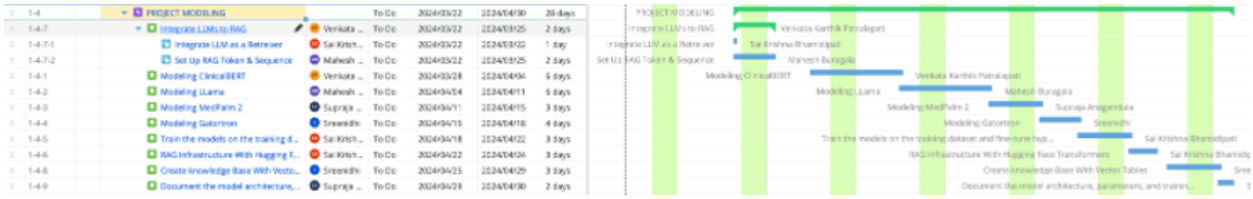
Data Preparation/Transformation Gantt



It can be seen in Figure 7 that the Project Modeling Gantt chart covers the period from March 22nd to April 30th, 2024, with the focus being on tasks related to modeling for the project during this period. As part of this process, LLMs are integrated into RAG, LLM is set up as a retriever, RAG tokens and sequences are established, various algorithms, such as ClinicalBERT, LLama2, Mistral 7B, and Gatortron, are modeled, trained on the training dataset, and hyperparameters are fine tuned for optimal performance. As part of the task, it'll have to create a knowledge base using vector tables, set up RAG infrastructure using Hugging Face Transformers, document the model architecture, parameters, and training process for each LLM, as well as setup all of the RAG infrastructure using Hugging Face Transformers. In order to facilitate efficient project management during this phase, prepared a Gantt chart that provides a structured timeline for executing key modeling tasks.

Figure 7

Project Modeling Gantt

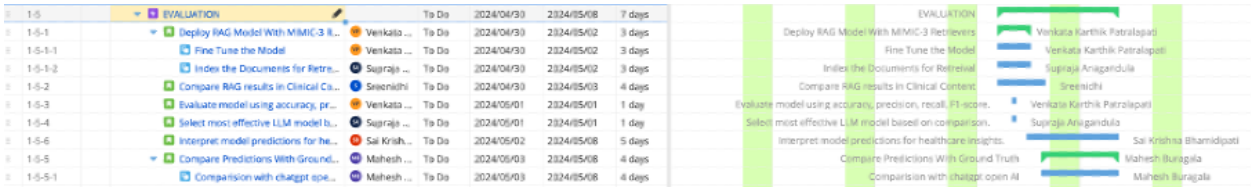


The Evaluation Gantt chart is shown in Figure 8 and spans from April 30th, 2024 to May 8th, 2024, illustrating the time period during which tasks related to evaluating the project's models and its results will be carried out. A key aspect of this phase is the deployment of the RAG model with MIMIC-3 retrievers, fine-tuning the model in order to achieve optimal performance, indexing documents in order to facilitate retrieval, and comparing RAG results in clinical content. Aside from evaluating models in terms of accuracy, precision, recall, and F1-score, tasks include comparing model predictions with ground truth data, interpreting model

predictions for healthcare insights, comparing predictions with ground truth data, and selecting the most effective LLM model based on comparisons. Additionally, comparing the ChatGPT service from OpenAI with the ChatGPT service from Nvidia adds valuable context to the evaluation process. In order to facilitate effective project management during this significant phase of the project, the Gantt chart provided here provides a structured timeline for executing key evaluation tasks.

Figure 8

Evaluation Gantt



The Model Deployment Gantt chart in Figure 9 illustrates the timing of the Model Deployment in the 298B semester, after the prototype has been completed. It will be the focus of this phase to deploy the project's models and integrate them into a user interface (UI) using the Django framework of the project. Additionally, this project aims to be responsible for documenting final workbooks and reports in order to ensure that they are transparent and reproducible. Models will be integrated with the user interfaces so that the user will be able to interact with the models and have a seamless experience. Although the project is currently at the prototype stage, the implementation of the complete user interface is expected to take place during the next semester of 2024. A Gantt chart showing the timeline for completing key tasks related to the deployment of the model and the integration of the user interface to facilitate effective project management for the upcoming semester is presented in this Gantt chart.

Figure 9

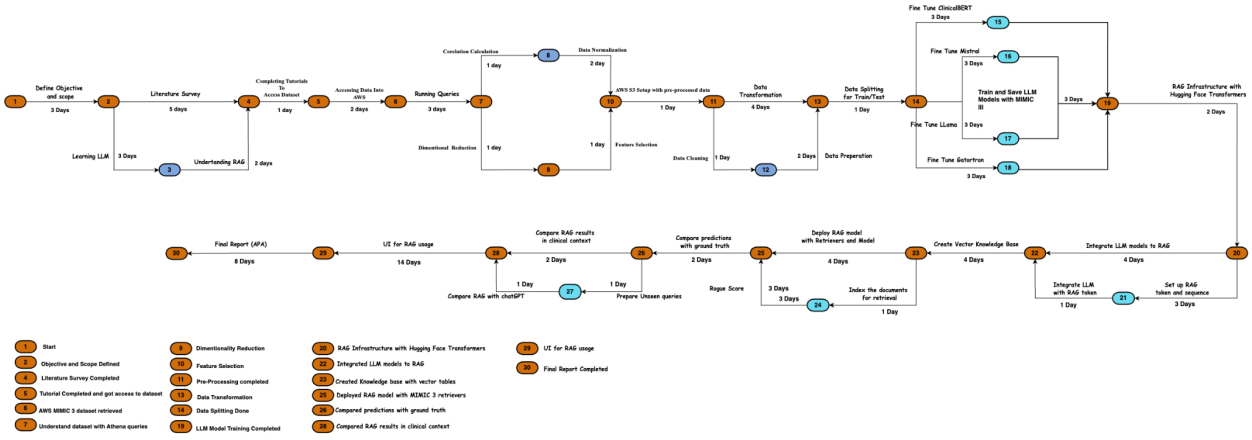
Model Deployment

Pert Chart

A Pert Chart is a project management tool used to schedule, organize, and coordinate tasks within a project. Pert Charts help visualize the sequence of tasks, their dependencies, and the critical path, ultimately aiding project managers in effectively allocating resources and managing project timelines. One of the primary advantages of Pert Charts is the ability to highlight the critical path, which represents the sequence of tasks that determines the minimum duration required to complete the project. By identifying the critical path, project managers can focus their attention on the most time-sensitive tasks, ensuring timely project delivery.

Figure 10

Pert Chart



In the context of the project, the critical path encompasses key activities essential for the successful implementation of the hospital readmission prediction system. Starting with defining the project's objective and scope, the critical path includes completing the literature survey and tutorials, obtaining access to the Physionet Mimic 3 dataset, understanding the dataset and

converting them to csv files, and performing dimensional reduction and feature selection. Preprocessing, data transformation, and data splitting are crucial steps preceding model training, which is followed by integrating the language models with Hugging Face Transformers and creating a knowledge base with vectors. Deployment of the model and comparison of predictions with ground truth data and clinical context are essential to validate the model's effectiveness. Finally, developing a user interface for model usage and preparing the final project report are integral steps concluding the critical path.

Data and Project Management Plan

3.1 Data Process

The Hospital Readmission Prediction system utilizes the MIMIC-III dataset available on a publicly available database provided by PhysioNet, containing de-identified health records of intensive care unit (ICU) patients at the Beth Israel Deaconess Medical Center (BIDMC) in Boston, MA. To access the MIMIC-III dataset, the team finished the tutorials provided by PhysioNet. The dataset was then downloaded and securely stored in Google Drive. The MIMIC-III database encompasses a wide range of healthcare data, comprising demographic information, patient admission records, bedside vital sign measurements, laboratory test results, medical procedures, medication records, caregiver notes, imaging reports, and mortality data. This project makes use of the NoteEvents table, which contains patient clinical notes and discharge summaries, and the Admissions table, which contains information about patient admission and discharge to build the hospital readmission prediction system.

The data processing workflow begins with loading the ADMISSIONS table in Jupyter Notebook, where cleaning and datatype conversion are performed. Subsequently, the number of days until the next admission is calculated for each patient. Next, the NOTEEVENTS table is

loaded, and all notes for each patient are concatenated into a single text. The datasets are merged based on patient identifiers (Subject_ID), ensuring each readmission is counted only once. An output label is generated to indicate readmissions within 31 days after discharge. Text preprocessing techniques including tokenization, stop word removal, lemmatization, and special character handling are applied to the clinical notes data. The dataset is balanced using SMOTE sampling technique. Finally, a Knowledge Base is constructed using preprocessed text data to facilitate efficient storage, retrieval, and analysis of clinical information, enhancing the capabilities of the Retrieval Augment Generator (RAG) and the pretrained Language Model to generate augmented outputs with improved relevance and clinical correctness.

3.2 Data Collection

The raw datasets utilized in this project were sourced primarily from the Medical Information Mart for Intensive Care III (MIMIC-III) database, a publicly available repository hosted on the PhysioNet website. MIMIC-III, an open-access relational database, contains tables of data pertaining to patients who were hospitalized in the intensive care units (ICUs) at Beth Israel Deaconess Medical Center. With information from about 41,000 patients admitted to intensive care units (ICUs) between 2001 and 2012, this public electronic health record database includes information from almost 53,000 admissions, including clinical notes and other pertinent data.The dataset, initially compressed in CSV format of about size 81GB, is downloaded from Physionet. ADMISSIONS and NOTEEVENTS files are considered from the MIMIC-III database for Hospital readmission prediction system.

Sources

This is a publicly available MIMIC-III database hosted on Physionet website.

Parameters

Document Type: Hospital admission and discharge records, clinical notes, vital sign measurements, laboratory test results, procedures, medications, caregiver notes, imaging reports, mortality statistics.

Patient Identification: Each patient is identified by a unique SUBJECT_ID.

Admission Identification: Each hospital admission is assigned a unique HADM_ID.

Time Period: Data covers the period from 1 June 2001 to 10 October 2012.

The ADMISSIONS table provides details about a patient's hospital admission, where a distinct HADM_ID is allocated to each hospital visit. The various columns that make up the Admissions table are depicted in Figure 11.These columns include SUBJECT_ID, which is a unique identifier for each patient, HADM_ID, which is a single patient's admission to the hospital, ADMITTIME, DISCHTIME, DEATHTIME, ADMISSION_TYPE, which describes the type of admission ('ELECTIVE,' URGENT,' NEWBORN, or EMERGENCY,'), Admission Location,Discharge Location, Diagnosis, and Hospital expire flag, which indicates whether the patient passed away during the specified hospitalization.

Figure 11

Raw sample of Admissions Table

ROW_ID	SUBJECT_ID	HADM_ID	ADMITTIME	DISCHTIME	DEATHTIME	ADMISSION_TYPE	ADMISSION_LOCATION	DISCHARGE_LC	INSURANCE	LANGU	RELIGION	MARITAL_S	ETHNICITY	DIAGNOSIS	HOSPITAL_EXPIRE_FLAG
21	22	165315	4/9/2196 12:26	4/10/2196 15:54		EMERGENCY	EMERGENCY ROOM ADMIT	DISC-TRAN CAI	Private		UNOBTAIN	MARRIED	WHITE	BENZODIAZEPINE OVEI	0
22	23	152223	9/3/2153 7:15	9/8/2153 19:10		ELECTIVE	PHYS REFERRAL/NORMAL D	HOME HEALTH	Medicare		CATHOLIC	MARRIED	WHITE	CORONARY ARTERY DIS	0
23	23	124321	10/18/2157 19:34	10/25/2157 14:00		EMERGENCY	TRANSFER FROM HOSP/EXT	HOME HEALTH	Medicare	ENGL	CATHOLIC	MARRIED	WHITE	BRAIN MASS	0
24	24	161859	6/6/2139 16:14	6/9/2139 12:48		EMERGENCY	TRANSFER FROM HOSP/EXT	HOME	Private		PROTESTA	SINGLE	WHITE	INTERIOR MYOCARDIAL	0
25	25	129635	11/2/2160 2:06	11/5/2160 14:55		EMERGENCY	EMERGENCY ROOM ADMIT	HOME	Private		UNOBTAIN	MARRIED	WHITE	ACUTE CORONARY SYN	0
26	26	197661	5/6/2126 15:16	5/13/2126 15:00		EMERGENCY	TRANSFER FROM HOSP/EXT	HOME	Medicare		CATHOLIC	SINGLE	UNKNOWI	V-TACH	0
27	27	134931	11/30/2191 22:16	12/3/2191 14:45		NEWBORN	PHYS REFERRAL/NORMAL D	HOME	Private		CATHOLIC		WHITE	NEWBORN	0
28	28	162569	9/1/2177 7:15	9/6/2177 16:00		ELECTIVE	PHYS REFERRAL/NORMAL D	HOME HEALTH	Medicare		CATHOLIC	MARRIED	WHITE	CORONARY ARTERY DIS	0
29	30	104557	10/14/2172 14:17	10/19/2172 14:37		URGENT	TRANSFER FROM HOSP/EXT	HOME HEALTH	Medicare		CATHOLIC	MARRIED	UNKNOWI	UNSTABLE ANGINA\CA	0
30	31	128652	8/22/2108 23:27	8/30/2108 15:00	8/30/2108 15:00	EMERGENCY	TRANSFER FROM HOSP/EXT	DEAD/EXPIRED	Medicare		CATHOLIC	MARRIED	WHITE	STATUS EPILEPTICUS	1
31	32	175413	4/4/2170 8:00	4/23/2170 12:45		ELECTIVE	PHYS REFERRAL/NORMAL D	HOME	Medicaid		UNOBTAINABLE		WHITE	TRACHEAL STENOSIS/S	0
32	33	176176	12/23/2116 22:30	12/27/2116 12:05		EMERGENCY	EMERGENCY ROOM ADMIT	HOME	Medicare		PROTESTA	MARRIED	UNKNOWI	SEPSIS;TELEMETRY	0
33	34	115799	7/18/2186 16:46	7/20/2186 16:00		EMERGENCY	TRANSFER FROM HOSP/EXT	HOME	Medicare	ENGL	CATHOLIC	MARRIED	WHITE	CHEST PAIN\CATH	0
34	34	144319	2/23/2191 5:23	2/25/2191 20:20		EMERGENCY	CLINIC REFERRAL/PREMATI	HOME HEALTH	Medicare	ENGL	CATHOLIC	MARRIED	WHITE	BRADYCARDIA	0
35	35	166707	2/10/2122 11:15	2/20/2122 15:30		ELECTIVE	PHYS REFERRAL/NORMAL D	HOME HEALTH	Medicare		CATHOLIC	DIVORCED	WHITE	AORTIC VALVE DISEASE	0
36	36	182104	4/30/2131 7:15	5/8/2131 14:00		EMERGENCY	CLINIC REFERRAL/PREMATI	HOME HEALTH	Medicare	ENGL	NOT SPEC	MARRIED	WHITE	CORONARY ARTERY DIS	0

The NOTEEVENTS table, which contains all patient clinical notes, is another table taken into consideration for this project. It also includes Subject_ID and HADM_ID, CHARTDATE (which records the note's charting date), CHARTTIME (which records the note's charting time),

STORETIME (which records the note's saving time into the system), CATEGORY, and DESCRIPTION (which define the type of note recorded), the caregiver who entered the note is identified by their CGID. The note text is contained in TEXT, and a '1' in the ISERROR column signifies that a doctor has determined that this note is incorrect. The Noteevents table's example raw data is displayed in Figure 12.

Figure 12

Raw sample of Noteevents table

ROW_ID	SUBJECT_ID	HADM_ID	CHARTDATE	CHARTTIME	STORETIME	CATEGORY	DESCRIPTION	CGID	ISERROR	TEXT
174	22532	167853.0	2151-08-04	NaN	NaN	Discharge summary	Report	NaN	NaN	Admission Date: [**2151-7-16**] Dischar...
175	13702	107527.0	2118-06-14	NaN	NaN	Discharge summary	Report	NaN	NaN	Admission Date: [**2118-6-2**] Discharg...
176	13702	167118.0	2119-05-25	NaN	NaN	Discharge summary	Report	NaN	NaN	Admission Date: [**2119-5-4**] D...
177	13702	196489.0	2124-08-18	NaN	NaN	Discharge summary	Report	NaN	NaN	Admission Date: [**2124-7-21**] ...
178	26880	135453.0	2162-03-25	NaN	NaN	Discharge summary	Report	NaN	NaN	Admission Date: [**2162-3-3**] D...

3.3 Data Pre-processing

The primary objective of our Clinical Notes data preparation endeavor was to transform disorganized tables into structured data suitable for subsequent research and analysis further.

MIMIC III dataset contains around 26 tables that contain different aspects of patient information, including clinical data. The ADMISSIONS and NOTEEVENTS tables within this dataset have been identified as being relevant to further modeling and analysis as it contains the patient demographics and clinical notes.

An admission table in Figure 13 shows a comprehensive set of attributes that provides valuable context when considering how the patient's journey within the hospital can be understood. The attributes listed in the figure include admission details, demographics, clinical condition, and indicators of outcome.

Figure 13

Admission Dataframe

	ROW_ID	SUBJECT_ID	HADM_ID	ADMITTIME	DISCHTIME	DEATHTIME	ADMISSION_TYPE	ADMISSION_LOCATION	DISCHARGE_LOCATION	INSURANCE	LANGUAGE	RELIGION	MARITAL_STATUS	ETHNICITY	EDREGTIME	EDOUTTIME	DI
0	21	22	165315	2196-04-09 12:26:00	2196-04-10 15:54:00	NaN	EMERGENCY	EMERGENCY ROOM ADMIT	DISC-TRAN CANCER/CHLDRN H	Private	NaN	UNOBTAINABLE	MARRIED	WHITE	2196-04-09 10:06:00	2196-04-09 13:24:00	BENZODI/ OVI
1	22	23	152223	2153-09-03 07:15:00	2153-09-08 19:10:00	NaN	ELECTIVE	PHYS REFERRAL/NORMAL DELI	HOME HEALTH CARE	Medicare	NaN	CATHOLIC	MARRIED	WHITE	NaN	NaN	CORONARY DISEASE\COI ARTERY B
2	23	23	124321	2157-10-18 19:34:00	2157-10-25 14:00:00	NaN	EMERGENCY	TRANSFER FROM HOSP/EXTRAM	HOME HEALTH CARE	Medicare	ENGL	CATHOLIC	MARRIED	WHITE	NaN	NaN	BRA
3	24	24	161859	2139-06-06 16:14:00	2139-06-09 12:48:00	NaN	EMERGENCY	TRANSFER FROM HOSP/EXTRAM	HOME	Private	NaN	PROTESTANT QUAKER	SINGLE	WHITE	NaN	NaN	IN MYO INFA
4	25	25	129635	2160-11-02 02:06:00	2160-11-05 14:55:00	NaN	EMERGENCY	EMERGENCY ROOM ADMIT	HOME	Private	NaN	UNOBTAINABLE	MARRIED	WHITE	2160-11-02 01:01:00	2160-11-02 04:27:00	ACUTE COI SY

A clinical note in the form of text is recorded during patient admission within a hospital which is displayed in the columns of Figure 14, the notesevents table. The other columns are used as part of data pre-processing to handle the time frame for setting up the readmission prediction These notes contain important information that can be used as a foundation for using the models that will be suggested at the start of this procedure to analyze patient conditions, therapies, and medical histories.

Figure 14

Notevents Dataframe

	ROW_ID	SUBJECT_ID	HADM_ID	CHARTDATE	CHARTTIME	STORETIME	CATEGORY	DESCRIPTION	CGID	ISERROR	TEXT
0	174	22532	167853.0	2151-08-04	NaN	NaN	Discharge summary	Report	NaN	NaN	Admission Date: [**2151-7-16**] Dischar...
1	175	13702	107527.0	2118-06-14	NaN	NaN	Discharge summary	Report	NaN	NaN	Admission Date: [**2118-6-2**] Dischar...
2	176	13702	167118.0	2119-05-25	NaN	NaN	Discharge summary	Report	NaN	NaN	Admission Date: [**2119-5-4**] D...
3	177	13702	196489.0	2124-08-18	NaN	NaN	Discharge summary	Report	NaN	NaN	Admission Date: [**2124-7-21**] ...
4	178	26880	135453.0	2162-03-25	NaN	NaN	Discharge summary	Report	NaN	NaN	Admission Date: [**2162-3-3**] D...
...	...	...	...	...	...	...	...	...	...	...	...
2083175	2070657	31097	115637.0	2132-01-21	2132-01-21 03:27:00	2132-01-21 03:38:00	Nursing/other	Report	17581.0	NaN	NPN\n\n\n#1 Infant remains in RA with O2 sats...
2083176	2070658	31097	115637.0	2132-01-21	2132-01-21 09:50:00	2132-01-21 09:53:00	Nursing/other	Report	19211.0	NaN	Neonatology\nDOL #5, CGA 36 weeks.\n\nCVR: Con...
2083177	2070659	31097	115637.0	2132-01-21	2132-01-21 16:42:00	2132-01-21 16:44:00	Nursing/other	Report	20104.0	NaN	Family Meeting Note\nFamily meeting held with ...
2083178	2070660	31097	115637.0	2132-01-21	2132-01-21 18:05:00	2132-01-21 18:16:00	Nursing/other	Report	16023.0	NaN	NPN 1800\n\n\n\n#1 Resp: [**Known lastname 2243*...
2083179	2070661	31097	115637.0	2132-01-21	2132-01-21 18:05:00	2132-01-21 18:31:00	Nursing/other	Report	16023.0	NaN	NPN 1800\nNursing Addendum:\n[**Known lastname...

2083180 rows × 11 columns

The ADMISSIONS and NOTEEVENTS tables clinical data was first kept in a Google Drive repository. Python scripts were utilized to pre-process and alter the data in the background while utilizing Google Colab. This included leveraging tools like Pandas and NumPy to standardize data formats, handle missing values, and eliminate duplicates. Additionally, feature

engineering was conducted to extract datetime features and pre-process clinical notes for input into the proposed models. of subsequent analysis tasks.

Admissions Data Pre-processing

Handling null values. Calculated and showed the count of null versus non-null values in the admission data's DEATHTIME column. This analysis is required to exclude deceased patients from readmission prediction tasks as they are irrelevant as part of the analysis. The use of null values in Figures 15 and 16 guarantees that forecasts are limited to patients who are alive, improving the accuracy and usefulness of the predictive models.

Figure 15

Non-null value count in the column Deathtime

```
admission_df.count()

ROW_ID                  58976
SUBJECT_ID              58976
HADM_ID                 58976
ADMITTIME               58976
DISCHTIME               58976
DEATHTIME                5854
ADMISSION_TYPE          58976
ADMISSION_LOCATION      58976
DISCHARGE_LOCATION      58976
INSURANCE               58976
LANGUAGE                33644
RELIGION                58518
MARITAL_STATUS          48848
ETHNICITY               58976
EDREGTIME               30877
EDOUTTIME               30877
DIAGNOSIS               58951
HOSPITAL_EXPIRE_FLAG    58976
HAS_CHARTEVENTS_DATA    58976
dtype: int64
```

Figure 16

Non-null value count after removing the non-null values in Deathtime

```
ROW_ID                  53122
SUBJECT_ID              53122
HADM_ID                 53122
ADMITTIME               53122
DISCHTIME               53122
DEATHTIME                   0
ADMISSION_TYPE          53122
ADMISSION_LOCATION      53122
DISCHARGE_LOCATION      53122
INSURANCE               53122
LANGUAGE                30603
RELIGION                52761
MARITAL_STATUS          43655
ETHNICITY               53122
EDREGTIME               26792
EDOUTTIME               26792
DIAGNOSIS               53098
HOSPITAL_EXPIRE_FLAG    53122
HAS_CHARTEVENTS_DATA    53122
dtype: int64
```

Appropriate Data Types. Convert the admission_df DataFrame's date columns to datetime type. Specifically, the ADMITTIME, DISCHTIME, and DEATHTIME columns are converted using the pd.to_datetime() function. The format argument guarantees that the dates are properly processed, and any parsing problems are converted to NaT (Not a Time) with the errors='coerce' parameter. This conversion in Figures 17 and 18 facilitates the processing and analysis of date-related information in later tasks.

Figure 17

Initial Data types

```
ROW_ID                  int64
SUBJECT_ID              int64
HADM_ID                 int64
ADMITTIME              object
DISCHTIME              object
DEATHTIME              object
ADMISSION_TYPE         object
ADMISSION_LOCATION     object
DISCHARGE_LOCATION     object
INSURANCE              object
LANGUAGE               object
RELIGION               object
MARITAL_STATUS         object
ETHNICITY              object
EDREGTIME              object
EDOUTTIME              object
DIAGNOSIS              object
HOSPITAL_EXPIRE_FLAG    int64
HAS_CHARTEVENTS_DATA    int64
dtype: object
```

Figure 18

Datatype after converting columns Admit, Discharge and Deathtime

```
ROW_ID                          int64
SUBJECT_ID                      int64
HADM_ID                         int64
ADMITTIME              datetime64[ns]
DISCHTIME              datetime64[ns]
DEATHTIME              datetime64[ns]
ADMISSION_TYPE                 object
ADMISSION_LOCATION             object
DISCHARGE_LOCATION             object
INSURANCE                      object
LANGUAGE                       object
RELIGION                       object
MARITAL_STATUS                 object
ETHNICITY                      object
EDREGTIME                      object
EDOUTTIME                      object
DIAGNOSIS                      object
HOSPITAL_EXPIRE_FLAG            int64
HAS_CHARTEVENTS_DATA            int64
dtype: object
```

Figure 19 shows the DataFrame admission_df sorted by the columns SUBJECT_ID and ADMITTIME using the sort_values() method. This sorting guarantees that the data is sorted in ascending order by subject ID and admission time. The DataFrame was then grouped according

to SUBJECT_ID using the groupby() method. This grouping separates the data into discrete groups based on topic IDs, allowing for subsequent analysis and actions inside each group.

Figure 19

Sorted Dataframe on SubjectID and Admittime

	ROW_ID	SUBJECT_ID	HADM_ID	ADMITTIME	DISCHTIME	DEATHTIME	ADMISSION_TYPE	ADMISSION_LOCATION	DISCHARGE_LOCATION	INSURANCE	LANGUAGE	RELIGION	MARITAL_STATUS	ETHNICITY	EDREGTIME	EDOUTTIME	
211	1	2	163353	2138-07-17 19:04:00	2138-07-21 15:48:00	NaT	NEWBORN	PHYS REFERRAL/NORMAL DELI	HOME	Private	NaN	NOT SPECIFIED	NaN	ASIAN	NaN	NaN	
212	2	3	145834	2101-10-20 19:08:00	2101-10-31 13:58:00	NaT	EMERGENCY	EMERGENCY ROOM ADMIT	SNF	Medicare	NaN	CATHOLIC	MARRIED	WHITE	2101-10-20 17:09:00	2101-10-20 19:24:00	
213	3	4	185777	2191-03-16 00:28:00	2191-03-23 18:41:00	NaT	EMERGENCY	EMERGENCY ROOM ADMIT	HOME WITH HOME IV PROVIDR	Private	NaN	PROTESTANT QUAKER	SINGLE	WHITE	2191-03-15 13:10:00	2191-03-16 01:10:00	FEVER,D
214	4	5	178980	2103-02-02 04:31:00	2103-02-04 12:15:00	NaT	NEWBORN	PHYS REFERRAL/NORMAL DELI	HOME	Private	NaN	BUDDHIST	NaN	ASIAN	NaN	NaN	
215	5	6	107064	2175-05-30 07:15:00	2175-06-15 16:00:00	NaT	ELECTIVE	PHYS REFERRAL/NORMAL DELI	HOME HEALTH CARE	Medicare	ENGL	NOT SPECIFIED	MARRIED	WHITE	NaN	NaN	CHRONI
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	
56435	58972	99985	176670	2181-01-27 02:47:00	2181-02-12 17:05:00	NaT	EMERGENCY	EMERGENCY ROOM ADMIT	HOME HEALTH CARE	Private	ENGL	JEWISH	MARRIED	WHITE	2181-01-26 23:35:00	2181-01-27 04:18:00	
56436	58973	99991	151118	2184-12-24 08:30:00	2185-01-05 12:15:00	NaT	ELECTIVE	PHYS REFERRAL/NORMAL DELI	HOME	Private	ENGL	NOT SPECIFIED	MARRIED	WHITE	NaN	NaN	
56437	58974	99992	197084	2144-07-25 18:03:00	2144-07-28 17:56:00	NaT	EMERGENCY	CLINIC REFERRAL/PREMATURE	SNF	Medicare	ENGL	CATHOLIC	WIDOWED	WHITE	2144-07-25 13:40:00	2144-07-25 18:50:00	
56565	58975	99995	137810	2147-02-08 08:00:00	2147-02-11 13:15:00	NaT	ELECTIVE	PHYS REFERRAL/NORMAL DELI	HOME	Medicare	ENGL	NOT SPECIFIED	MARRIED	WHITE	NaN	NaN	
56566	58976	99999	113369	2117-12-30 07:15:00	2118-01-04 16:30:00	NaT	ELECTIVE	PHYS REFERRAL/NORMAL DELI	SNF	Medicare	SPAN	JEHOVAH'S WITNESS	SEPARATED	HISPANIC OR LATINO	NaN	NaN	SPC

52219 rows × 19 columns

Following the sorting and grouping stages, the next objective in Figure 20 was to calculate the next admission time and the number of days between admissions for each patient. This technique most likely used the pandas shift() function to shift admission times within each subject's group, allowing for the estimation of the time interval between consecutive admissions. This research sheds light on patient readmission trends and intervals by determining both the next admission time and the number of days between admissions.

Figure 20

Calculating Next Admission Time and Time Interval between Admissions

_LOCATION	DISCHARGE_LOCATION	INSURANCE	...	RELIGION	MARITAL_STATUS	ETHNICITY	EDREGTIME	EDOUTTIME	DIAGNOSIS	HOSPITAL_EXPIRE_FLAG	HAS_CHARTEVENTS_DATA	next_admit_time	days_between_admits
PHYS /NORMAL DELI	HOME	Private	...	NOT SPECIFIED	NaN	ASIAN	NaN	NaN	NEWBORN	0	1	NaT	NaT
ICY ROOM ADMIT	SNF	Medicare	...	CATHOLIC	MARRIED	WHITE	2101-10-20 17:09:00	2101-10-20 19:24:00	HYPOTENSION	0	1	NaT	NaT
ICY ROOM ADMIT	HOME WITH HOME IV PROVIDR	Private	...	PROTESTANT QUAKER	SINGLE	WHITE	2191-03-15 13:10:00	2191-03-16 01:10:00	FEVER,DEHYDRATION,FAILURE TO THRIVE	0	1	NaT	NaT
PHYS /NORMAL DELI	HOME	Private	...	BUDDHIST	NaN	ASIAN	NaN	NaN	NEWBORN	0	1	NaT	NaT
PHYS /NORMAL DELI	HOME HEALTH CARE	Medicare	...	NOT SPECIFIED	MARRIED	WHITE	NaN	NaN	CHRONIC RENAL FAILURE/SDA	0	1	NaT	NaT
...	...	...	...	...	...	...	...	...	...	...	...	...	...
ICY ROOM ADMIT	HOME HEALTH CARE	Private	...	JEWISH	MARRIED	WHITE	2181-01-26 23:35:00	2181-01-27 04:18:00	FEVER	0	1	NaT	NaT
PHYS /NORMAL DELI	HOME	Private	...	NOT SPECIFIED	MARRIED	WHITE	NaN	NaN	DIVERTICULITIS/SDA	0	1	NaT	NaT
CLINIC IEMATURE	SNF	Medicare	...	CATHOLIC	WIDOWED	WHITE	2144-07-25 13:40:00	2144-07-25 18:50:00	RETROPERITONEAL HEMORRHAGE	0	1	NaT	NaT
PHYS /NORMAL DELI	HOME	Medicare	...	NOT SPECIFIED	MARRIED	WHITE	NaN	NaN	ABDOMINAL AORTIC ANEURYSM/SDA	0	1	NaT	NaT
PHYS /NORMAL DELI	SNF	Medicare	...	JEHOVAH'S WITNESS	SEPARATED	HISPANIC OR LATINO	NaN	NaN	SPONDYLOLISTHESIS/SDA	0	1	NaT	NaT

Determined and sorted the dataframe to include only the rows where the next_admit_time is not null as shown in Figure 21. This process successfully finds cases in which patients have future admissions documented in the dataset. By focusing on these specific rows, additional analysis can be performed to investigate patterns and trends in patient readmissions.

Figure 21

Identifying Subsequent Admissions: Filtering Data for Further Analysis

)CATION	DISCHARGE_LOCATION	INSURANCE	...	RELIGION	MARITAL_STATUS	ETHNICITY	EDREGTIME	EDOUTTIME	DIAGNOSIS	HOSPITAL_EXPIRE_FLAG	HAS_CHARTEVENTS_DATA	next_admit_time	days_between_admits
PHYS NORMAL DELI	HOME HEALTH CARE	Private	...	CATHOLIC	MARRIED	WHITE	NaN	NaN	PATIENT FORAMEN OVALE\ PATENT FORAMEN OVALE MI...	0	1	2135-05-09 14:11:00	133 days 06:56:00
PHYS NORMAL DELI	HOME HEALTH CARE	Medicare	...	CATHOLIC	MARRIED	WHITE	NaN	NaN	CORONARY ARTERY DISEASE\CORONARY ARTERY BYPASS...	0	1	2157-10-18 19:34:00	1506 days 12:19:00
R FROM EXTRAM	HOME	Medicare	...	CATHOLIC	MARRIED	WHITE	NaN	NaN	CHEST PAIN\CATH	0	1	2191-02-23 05:23:00	1680 days 12:37:00
CLINIC MATURE	HOME HEALTH CARE	Medicare	...	NOT SPECIFIED	MARRIED	WHITE	NaN	NaN	CORONARY ARTERY DISEASE\CORONARY ARTERY BYPASS...	0	1	2131-05-12 19:49:00	12 days 12:34:00
Y ROOM ADMIT	REHAB/DISTINCT PART HOSP	Medicare	...	NOT SPECIFIED	MARRIED	WHITE	2131-05-12 17:26:00	2131-05-12 22:17:00	CHEST PAIN/SHORTNESS OF BREATH	0	1	2134-05-10 11:30:00	1093 days 15:41:00
...	...	...	...	...	...	...	...	...	...	...	...	...	...
R FROM EXTRAM	HOME	Medicare	...	PROTESTANT QUAKER	MARRIED	WHITE	NaN	NaN	CARDIOMYOPATHY\CARDIAC CATH	0	1	2132-09-15 00:36:00	265 days 06:55:00
CLINIC MATURE	HOME	Private	...	7TH DAY ADVENTIST	MARRIED	BLACK/HAITIAN	2181-08-05 22:31:00	2181-08-06 04:13:00	HYPERGLYCEMIA	0	1	2182-07-03 19:50:00	331 days 17:28:00
CLINIC MATURE	HOME	Private	...	CATHOLIC	MARRIED	WHITE	2201-02-23 15:54:00	2201-02-23 21:58:00	HYPONATREMIA	0	1	2201-05-15 13:12:00	80 days 16:30:00
PHYS NORMAL DELI	HOME HEALTH CARE	Medicare	...	CATHOLIC	MARRIED	WHITE	NaN	NaN	TVR	0	1	2157-01-05 17:27:00	38 days 05:31:00
CLINIC MATURE	HOME	Medicare	...	CATHOLIC	MARRIED	WHITE	2157-01-05 14:03:00	2157-01-05 18:50:00	SHORTNESS OF BREATH	0	1	2157-02-16 17:31:00	42 days 00:04:00

The 'days_between_admits' column in the sorted DataFrame was changed from a timedelta type to an integer indicating the number of days between admissions. This translation, as illustrated in Figure 22, simplifies subsequent calculations and analyses involving the time interval between consecutive admissions.

Figure 22

Converting Time Interval between Admissions to Integer Format

LOCATION	DISCHARGE_LOCATION	INSURANCE	...	RELIGION	MARITAL_STATUS	ETHNICITY	EDREGTIME	EDOUTTIME	DIAGNOSIS	HOSPITAL_EXPIRE_FLAG	HAS_CHARTEVENTS_DATA	next_admit_time	days_between_admits
PHYS /NORMAL DELI	HOME	Private	...	NOT SPECIFIED	NaN	ASIAN	NaN	NaN	NEWBORN	0	1	NaT	NaN
ICY ROOM ADMIT	SNF	Medicare	...	CATHOLIC	MARRIED	WHITE	2101-10-20 17:09:00	2101-10-20 19:24:00	HYPOTENSION	0	1	NaT	NaN
ICY ROOM ADMIT	HOME WITH HOME IV PROVIDR	Private	...	PROTESTANT QUAKER	SINGLE	WHITE	2191-03-15 13:10:00	2191-03-16 01:10:00	FEVER,DEHYDRATION,FAILURE TO THRIVE	0	1	NaT	NaN
PHYS /NORMAL DELI	HOME	Private	...	BUDDHIST	NaN	ASIAN	NaN	NaN	NEWBORN	0	1	NaT	NaN
PHYS /NORMAL DELI	HOME HEALTH CARE	Medicare	...	NOT SPECIFIED	MARRIED	WHITE	NaN	NaN	CHRONIC RENAL FAILURE/SDA	0	1	NaT	NaN
...	...	...	...	...	...	...	...	...	...	...	...	...	...
ICY ROOM ADMIT	HOME HEALTH CARE	Private	...	JEWISH	MARRIED	WHITE	2181-01-26 23:35:00	2181-01-27 04:18:00	FEVER	0	1	NaT	NaN
PHYS /NORMAL DELI	HOME	Private	...	NOT SPECIFIED	MARRIED	WHITE	NaN	NaN	DIVERTICULITIS/SDA	0	1	NaT	NaN
CLINIC EMATURE	SNF	Medicare	...	CATHOLIC	WIDOWED	WHITE	2144-07-25 13:40:00	2144-07-25 18:50:00	RETROPERITONEAL HEMORRHAGE	0	1	NaT	NaN
PHYS /NORMAL DELI	HOME	Medicare	...	NOT SPECIFIED	MARRIED	WHITE	NaN	NaN	ABDOMINAL AORTIC ANEURYSM/SDA	0	1	NaT	NaN
PHYS /NORMAL DELI	SNF	Medicare	...	JEHOVAH'S WITNESS	SEPARATED	HISPANIC OR LATINO	NaN	NaN	SPONDYLOLISTHESIS/SDA	0	1	NaT	NaN

By changing the days_between_admits column to indicate the number of days between admissions, the study was able to get insight into the average time between successive admissions. This information is useful in defining the optimal timeframe for readmission prediction. Understanding the average or median time between admissions allows healthcare providers to set a threshold or time frame within which readmissions are likely to occur. This understanding influences the design and deployment of readmission prediction algorithms, improving their accuracy and efficacy in identifying at-risk patients.

Noteevents Data Pre-processing

Appropriate Data Types. First step is to convert the CHARTDATE and CHARTTIME columns in the noteevents_df to datetime type. It specifies the format of the date and time strings and handles any parsing errors by coercing them to NaT (Not a Time). This conversion facilitates easier manipulation and analysis of date and time-related information in the dataset.

Figure 23

Noteevents Dataframe

	ROW_ID	SUBJECT_ID	HADM_ID	CHARTDATE	CHARTTIME	STORETIME	CATEGORY	DESCRIPTION	CGID	ISERROR	TEXT
0	174	22532	167853.0	2151-08-04	NaN	NaN	Discharge summary	Report	NaN	NaN	Admission Date: [**2151-7-16**] Dischar...
1	175	13702	107527.0	2118-06-14	NaN	NaN	Discharge summary	Report	NaN	NaN	Admission Date: [**2118-6-2**] Discharg...
2	176	13702	167118.0	2119-05-25	NaN	NaN	Discharge summary	Report	NaN	NaN	Admission Date: [**2119-5-4**] D...
3	177	13702	196489.0	2124-08-18	NaN	NaN	Discharge summary	Report	NaN	NaN	Admission Date: [**2124-7-21**] ...
4	178	26880	135453.0	2162-03-25	NaN	NaN	Discharge summary	Report	NaN	NaN	Admission Date: [**2162-3-3**] D...
...	...	...	...	...	...	...	...	...	...	...	...
2083175	2070657	31097	115637.0	2132-01-21	2132-01-21 03:27:00	2132-01-21 03:38:00	Nursing/other	Report	17581.0	NaN	NPN\n\n\n#1 Infant remains in RA with O2 sats...
2083176	2070658	31097	115637.0	2132-01-21	2132-01-21 09:50:00	2132-01-21 09:53:00	Nursing/other	Report	19211.0	NaN	Neonatology\nDOL #5, CGA 36 weeks.\n\nCVR: Con...
2083177	2070659	31097	115637.0	2132-01-21	2132-01-21 16:42:00	2132-01-21 16:44:00	Nursing/other	Report	20104.0	NaN	Family Meeting Note\nFamily meeting held with ...
2083178	2070660	31097	115637.0	2132-01-21	2132-01-21 18:05:00	2132-01-21 18:16:00	Nursing/other	Report	16023.0	NaN	NPN 1800\n\n\n#1 Resp: [**Known lastname 2243*...
2083179	2070661	31097	115637.0	2132-01-21	2132-01-21 18:05:00	2132-01-21 18:31:00	Nursing/other	Report	16023.0	NaN	NPN 1800\nNursing Addendum:\n[**Known lastname...

2083180 rows × 11 columns

The DataFrame 'noteevents_df' in Figure 23 was sorted by SUBJECT_ID, CHARTDATE, and CHARTTIME to arrange the data chronologically for each patient. The text items within each topic group were then concatenated into a single string, yielding a new DataFrame called 'concatenated_text_df'. DataFrame's rows each reflect a unique topic ID, as well as the concatenated text elements connected with it.

Then the Data Frames were merged based on the SUBJECT_ID column to combine the information from both datasets into a single cohesive dataset in Figure 24. This merging process facilitates comprehensive analysis by consolidating relevant data points from different sources into one unified DataFrame, and it was named as

Figure 24

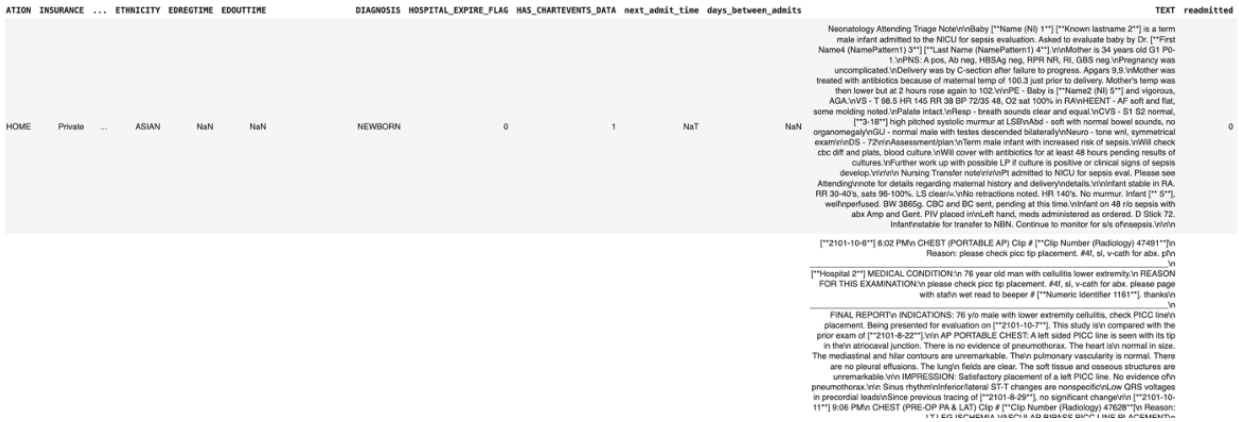
Integrating Patient Text Data: Merging and Consolidating Information

	SUBJECT_ID	TEXT
0	2	Neonatology Attending Triage Note\n\nBaby [**N...
1	3	[**2101-10-6**] 6:02 PM\n CHEST (PORTABLE AP) ...
2	4	[**2191-3-15**] 4:20 PM\n CHEST (PORTABLE AP) ...
3	5	NNP Triage Note\n\nBB [**Known lastname 6**] d...
4	6	[**2175-5-25**] 10:52 AM\n CHEST (PRE-OP PA & ...
...	...	...
46141	99985	Sinus rhythm. Normal ECG. Since the previous t...
46142	99991	[**2184-12-27**] 11:35 AM\n CHEST (PA & LAT) ...
46143	99992	[**2144-7-10**] 8:04 AM\n CHEST PORT. LINE PLA...
46144	99995	[**2147-1-10**] 8:15 AM\n CAROTID SERIES COMPL...
46145	99999	Sinus rhythm. Normal tracing. Compared to th...

46146 rows × 2 columns

Figure 25

Creating Binary Readmission Target Based on Time Interval



ATION	INSURANCE	...	ETHNICITY	EDREGTIME	EDOUTTIME	DIAGNOSIS	HOSPITAL_EXPIRE_FLAG	HAS_CHARTEVENTS_DATA	next_admit_time	days_between_admits	TEXT	readmitted
HOME	Private	...	ASIAN	NaN	NaN	NEWBORN	0	1	NaT	NaN	Neonatology Attending Triage Note\n\nBaby [**Name (NI) 1**] [**Known lastname 2**] is a term male infant admitted to the NICU for sepsis evaluation. Asked to evaluate baby by Dr. [**First Name4 (NamePattern1) 3**] [**Last Name (NamePattern1) 4**].\n\nMother is 34 years old G1 P0-1.\nPNS: A pos, Ab neg, HBSAg neg, RPR NR, RI, GBS neg.\nPregnancy was uncomplicated.\nDelivery was by C-section after failure to progress. Apgars 9,9.\nMother was treated with antibiotics because of maternal temp of 100.3 just prior to delivery. Mother's temp was then lower but at 2 hours rose again to 102.\n\nPE - Baby is [**Name2 (NI) 5**] and vigorous, AGA.\nVS - T 98.5 HR 145 RR 38 BP 72/35 48, O2 sat 100% in RA\nHEENT - AF soft and flat, some molding noted.\nPalate intact.\nResp - breath sounds clear and equal.\nCVS - S1 S2 normal, [**3-18**] high pitched systolic murmur at LSB\nAbd - soft with normal bowel sounds, no organomegaly\nGU - normal male with testes descended bilaterally\nNeuro - tone wnl, symmetrical exam\n\nDS - 72\n\nAssessment/plan:\nTerm male infant with increased risk of sepsis.\nWill check cbc diff and plats, blood culture.\nWill cover with antibiotics for at least 48 hours pending results of cultures.\nFurther work up with possible LP if culture is positive or clinical signs of sepsis develop.\n\n\n\n Nursing Transfer note\n\n\nPt admitted to NICU for sepsis eval. Please see Attending\nnote for details regarding maternal history and delivery\ndetails.\n\nInfant stable in RA. RR 30-40's, sats 96-100%. LS clear\=.\nNo retractions noted. HR 140's. No murmur. Infant [** 5**], well\nperfused. BW 3665g. CBC and BC sent, pending at this time.\n\nInfant on 48 rito sepsis with abx Amp and Gent. PIV placed in\nLeft hand, meds administered as ordered. D Stick 72. Infant\nstable for transfer to NBN. Continue to monitor for s/s of\nsepsis.\n\n\n	0
											[**2101-10-6**] 6:02 PM\n CHEST (PORTABLE AP) Clip # [**Clip Number (Radiology) 47491**]\n Reason: please check picc tip placement. #4f, sl, v-cath for abx. p\n _____\n [**Hospital 2**] MEDICAL CONDITION:\n 76 year old man with cellulitis lower extremity.\n REASON FOR THIS EXAMINATION:\n please check picc tip placement. #4f, sl, v-cath for abx. please page with stat\n wet read to beeper # [**Numeric Identifier 1161**]. thanks\n _____\n FINAL REPORT\n INDICATIONS: 76 y/o male with lower extremity cellulitis, check PICC line\n placement. Being presented for evaluation on [**2101-10-7**]. This study is\n compared with the prior exam of [**2101-8-22**].\n\n AP PORTABLE CHEST: A left sided PICC line is seen with its tip in the\n atriocaval junction. There is no evidence of pneumothorax. The heart is\n normal in size. The mediastinal and hilar contours are unremarkable. The\n pulmonary vascularity is normal. There are no pleural effusions. The lung\n fields are clear. The soft tissue and osseous structures are unremarkable.\n\n IMPRESSION: Satisfactory placement of a left PICC line. No evidence of\n pneumothorax.\n\n Sinus rhythm\n\nInferior/lateral ST-T changes are nonspecific\nLow QRS voltages in precordial leads\nSince previous tracing of [**2101-8-29**], no significant change\n\n [**2101-10-11**] 9:06 PM\n CHEST (PRE-OP PA & LAT) Clip # [**Clip Number (Radiology) 47628**]\n Reason:	

In Figure 25 a new column readmitted was created in the merged DataFrame based on the condition days_between_admits <= 30. This criterion determines if the number of days between admissions is fewer than or equal to 30. If the criterion is satisfied, the 'readmitted' column is set to 1, indicating that the patient was readmitted. Otherwise, it is assigned the value 0, indicating no readmission.

3.4 Data Transformation

Data transformation is critical in this Hospital Readmission Prediction study because it allows raw clinical data to be refined to fit LLM models. Raw data frequently has irregularities, missing numbers, and a variety of formats, reducing LLM performance. Data transformation addresses these challenges by pre-processing, standardizing, and extracting characteristics, resulting in optimal LLM training. Key phases include managing missing values, which was already done in previous rounds, encoding variables, and extracting features. The objective is to provide an improved dataset that allows for reliable readmission prediction using LLMs.

Figure 26

Standardizing Text Data: Lowercasing and Removing Formatting Characters

```
merged_df['TEXT'].iloc[0]
```

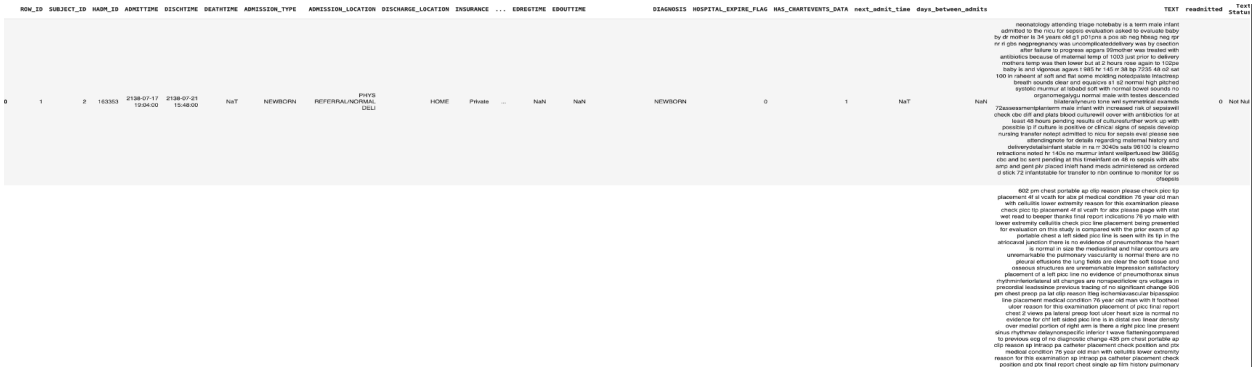
```
"neonatology attending triage notebaby [**name (ni) 1**] [**known lastname 2**] is a term male infant admitted to the nicu for sepsis evaluation.  asked to evaluate baby by dr. [**first name4 (namepattern1) 3**] [**last name (namepattern1) 4**].mother is 34 years old g1 p0-1.pns:  a pos, ab neg, hbsag neg, rpr nr, ri, gbs neg.pregnancy was uncomplicated.delivery was by c-section after failure to progress.  apgars 9,9.mother was treated with antibiotics because of maternal temp of 100.3 just prior to delivery.  mother's temp was then lower but at 2 hours rose again to 102.pe - baby is [**name2 (ni) 5**] and vigorous, aga.vs -  t 98.5 hr 145 rr 38 bp 72/35 48, o2 sat 100% in raheent - af soft and flat, some molding noted.palate intact.resp - breath sounds clear and equal.cvs - s1 s2 normal, [**3-18**] high pitched systolic murmur at lsbabd - soft with normal bowel sounds, no organomegalygu - normal male with testes descended bilaterallyneuro - tone wnl, symmetrical exameds - 72assessment/plan:term male infant with increased risk of sepsis.will check cbc diff and plats, blood culture.will cover with antibiotics for at least 48 hours pending results of cultures.further work up with possible lp if culture is positive or clinical signs of sepsis develop. nursing transfer notept admitted to nicu for sepsis eval. please see attendingnote for details regarding maternal history and deliverydetails.infant stable in ra. rr 30-40's, sats 96-100%. ls clear/=.no retractions noted. hr 140's. no murmur. infant [** 5**], wellperfused. bw 3865g. cbc and bc sent, pending at this time.infant on 48 r/o sepsis with abx amp and gent. piv placed inleft hand, meds administered as ordered. d stick 72. infantstable for transfer to nbn. continue to monitor for s/s ofsepsis."
```

The data transformations on the TEXT column in the DataFrame merged_df includes standardizing the text data to ensure uniformity and cleanliness. Initially, all text entries are transformed to lowercase to guarantee text case consistency. The text is then stripped of newlines ('\n') and carriage returns ('\r'). These changes, as seen in Figure 26, strive to eliminate extraneous formatting and standardize the text data, so improving its quality and usefulness for further analysis and modeling operations.

The next step is to obtain needed resources from the Natural Language Toolkit (NLTK) library. This involves getting Punkt tokenizer models, a stopwords corpus, and WordNet. These resources are essential for many natural language processing jobs. The Punkt tokenizer models help in tokenization by dividing text into separate words or tokens. The stopwords corpus includes frequent words with limited semantic significance, which are often deleted during text analysis. WordNet also functions as a lexical database, categorizing English words into synonym sets and semantic relationships. By obtaining these resources, the NLTK library prepares for future text processing tasks.

Figure 27

Final Dataframe after NLTK Resource Acquisition and Text Data Preprocessing



The following data transformation includes deleting rows from the DataFrame merged_df that have null values in the ‘TEXT’ column. By removing missing text data cases, this stage guarantees data integrity and completeness for further analysis. With the subset option set to ['TEXT'] in this instance, about 0.3% of the rows had null values in the TEXT column and were therefore successfully removed from the DataFrame. This guarantees that there are no missing text data in the dataset, enabling more accurate and dependable analysis.

The next data transformation step involves sanitizing the 'TEXT' column in the DataFrame by removing personally identifiable information (PII) placeholders. This is achieved using regular expressions to identify and replace patterns corresponding to PII placeholders with an empty string. By sanitizing the text data in this manner, sensitive information is anonymized or removed, ensuring data privacy and compliance with privacy regulations. This transformation enhances the suitability of the DataFrame for analysis and modeling tasks, while also safeguarding the confidentiality of individuals' information.

Further to remove special characters from text data, another stage in the data transformation process is to define a function named remove_special_characters. This method finds characters that are neither alphanumeric or whitespace and replaces them with an empty string using a regular expression pattern. The special characters are then eliminated from each text entry in the DataFrame's TEXT' column by using this function. As seen in Figure 27, this procedure guarantees that the text data is cleaned and standardized, making it more appropriate for activities involving analysis and modeling later on.

Figure 28

Final Data frame required for further preparation and modeling of the data

As seen in Figure 28, a new DataFrame called model_df is produced by choosing particular columns (SUBJECT_ID, HADM_ID, TEXT, readmitted) from the combined DataFrame merged_df. The predictive modeling challenge is based on these chosen columns. The processed textual data is contained in the TEXT column, whereas the SUBJECT_ID and HADM_ID columns most likely function as patient and hospital admission IDs, respectively. The target variable that indicates whether a patient was readmitted is represented by the readmitted column. The dataset is ready for additional data preparation processes like train-test split and model training by generating this DataFrame.

As the dataset is imbalanced, random over sampling is performed to handle the imbalance in data classification. As seen in the below figure there are equal number of records for both readmitted and not readmitted

Figure 29

Balanced data after Random Over Sampling

```
Number of records before sampling: 41965
Number of records after sampling: 81308

[ ] unique_counts = model_df_balanced["readmitted"].value_counts()
    print(unique_counts)

readmitted
0    40654
1    40654
Name: count, dtype: int64
```

3.5 Data Preparation

The MIMIC-III (Medical Information Mart for Intensive Care III) dataset, collected between 2001 and 2012 from the Beth Israel Deaconess Medical Center's intensive care units, is comprised of de-identified health data sourced from electronic health records, monitoring devices, administrative systems, and potentially research databases. This comprehensive dataset encompasses a wide array of patient information, including demographics, vital signs, laboratory results, medications, procedures, and administrative records, providing researchers with a rich resource for studying critical care outcomes and treatments. The data collection process involves careful de-identification to protect patient privacy while ensuring compliance with relevant regulations such as HIPAA, with access granted to approved researchers for specified research purposes.

The pre-processing begins by loading two CSV files, 'ADMISSIONS.csv' and 'NOTEEVENTS.csv', into Pandas DataFrames, followed by exploratory data analysis to understand their structure. It then performs data cleaning tasks such as converting date columns to datetime format, sorting admission data, and calculating the time between consecutive admissions. Text data processing steps involve concatenating text notes for each patient, merging them with admission data, and normalizing text by converting it to lowercase and removing special characters. NLTK library is used for tokenization, stopword removal, and lemmatization. The data is split into training, validation, and test sets, and class imbalance is addressed using

SMOTE. Finally, TF-IDF vectorization is applied to convert text data into numerical features for predictive modeling tasks, particularly in predicting readmissions. A sample of dataset ADMISSIONS.csv and NOTEEVENTS.csv are given in Figures 30 and Figure 31 respectively.

Figure 30

Sample of ADMISSIONS.csv

	ROW_ID	SUBJECT_ID	HADM_ID	ADMITTIME	DISCHTIME	DEATHTIME	ADMISSION_TYPE	ADMISSION_LOCATION	DISCHARGE_LOCATION	INSURANCE	LANGUAGE	RELIGION	MAR
0	21	22	165315	2196-04-09 12:26:00	2196-04-10 15:54:00	NaN	EMERGENCY	EMERGENCY ROOM ADMIT	DISC-TRAN CANCER/CHLDRN H	Private	NaN	UNOBTAINABLE	
1	22	23	152223	2153-09-03 07:15:00	2153-09-08 19:10:00	NaN	ELECTIVE	PHYS REFERRAL/NORMAL DELI	HOME HEALTH CARE	Medicare	NaN	CATHOLIC	
2	23	23	124321	2157-10-18 19:34:00	2157-10-25 14:00:00	NaN	EMERGENCY	TRANSFER FROM HOSP/EXTRAM	HOME HEALTH CARE	Medicare	ENGL	CATHOLIC	
3	24	24	161859	2139-06-06 16:14:00	2139-06-09 12:48:00	NaN	EMERGENCY	TRANSFER FROM HOSP/EXTRAM	HOME	Private	NaN	PROTESTANT QUAKER	
4	25	25	129635	2160-11-02 02:06:00	2160-11-05 14:55:00	NaN	EMERGENCY	EMERGENCY ROOM ADMIT	HOME	Private	NaN	UNOBTAINABLE	

Figure 31

Sample of NOTEEVENTS.csv

```
noteevents_df.head()
```

	ROW_ID	SUBJECT_ID	HADM_ID	CHARTDATE	CHARTTIME	STORETIME	CATEGORY	DESCRIPTION	CGID	ISERROR	TEXT
0	174	22532	167853.0	2151-08-04	NaN	NaN	Discharge summary	Report	NaN	NaN	Admission Date: [**2151-7-16**] Dischar...
1	175	13702	107527.0	2118-06-14	NaN	NaN	Discharge summary	Report	NaN	NaN	Admission Date: [**2118-6-2**] Discharg...
2	176	13702	167118.0	2119-05-25	NaN	NaN	Discharge summary	Report	NaN	NaN	Admission Date: [**2119-5-4**] D...
3	177	13702	196489.0	2124-08-18	NaN	NaN	Discharge summary	Report	NaN	NaN	Admission Date: [**2124-7-21**] ...
4	178	26880	135453.0	2162-03-25	NaN	NaN	Discharge summary	Report	NaN	NaN	Admission Date: [**2162-3-3**] D...

The dataset thus obtained is named model_data and has a total of 52954 records. A sample of the dataset is given in figure 32.

Figure 32

Sample of the dataset Model_data

	SUBJECT_ID	HADM_ID	TEXT	readmitted
0	2	163353	neonatology attending triage notebaby is a term male infant admitted to the nicu for sepsis evaluation asked to evaluate baby by dr mother is 34 years old g1 p01pns a pos ab neg hbsag neg rpr nr ri gbs negpregnancy was uncomplicateddelivery was by csection after failure to progress apgars 99mother was treated with antibiotics because of maternal temp of 1003 just prior to delivery mothers temp was then lower but at 2 hours rose again to 102pe baby is and vigorous agavs t 985 hr 145 rr 38 bp 7235 48 o2 sat 100 in raheent af soft and flat some molding notedpalate intactresp breath sounds clear and equalcvs s1 s2 normal high pitched systolic murmur at lsbabd soft with normal bowel sounds no organomegalygu normal male with testes descended bilaterallyneuro tone wnl symmetrical examds 72assessmentplanterm male infant with increased risk of sepsiswill check cbc diff and plats blood culturewill cover with antibiotics for at least 48 hours pending results of culturesfurther work up with possible lp if culture is positive or clinical signs of sepsis develop nursing transfer notept admitted to nicu for sepsis eval please see attendingnote for details regarding maternal history and deliverydetailsinfant stable in ra rr 3040s sats 96100 ls clearno retractions noted hr 140s no murmur infant wellperfused bw 3865g cbc and bc sent pending at this timeinfant on 48 ro sepsis with abx amp and gent piv placed inleft hand meds administered as ordered d stick 72 infantstable for transfer to nbn continue to monitor for ss ofsepsis	0

The dataset is split into training, testing and validation in the ratio 70:15:15 using stratified sampling. The code snippet describing this is given in Figure 33.

Figure 33

Code Snippet of Stratified Sampling for splitting the data into train, test and validation

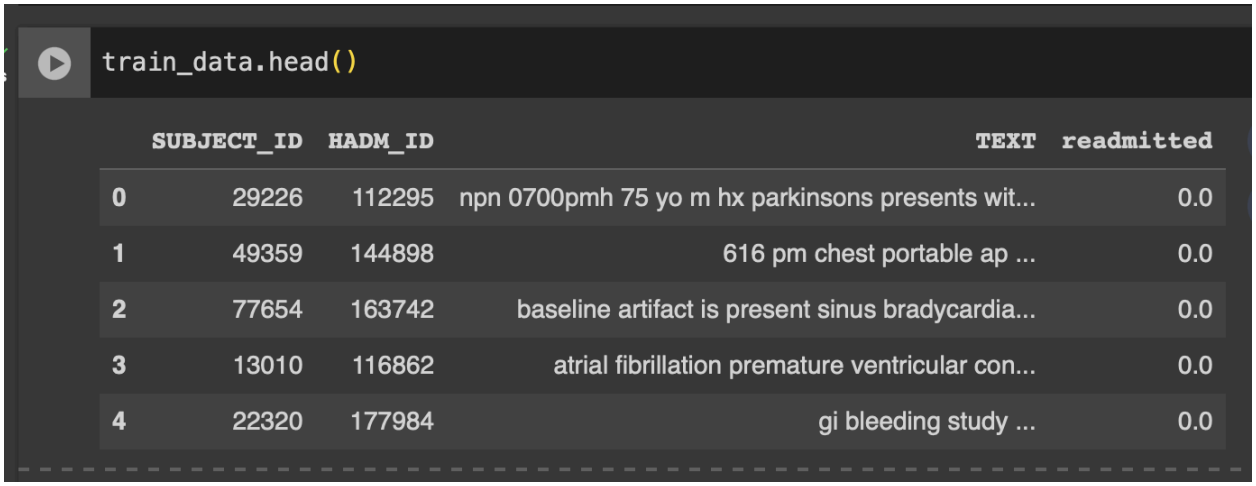
```
[ ] import sklearn
    from sklearn.model_selection import train_test_split

    # Split into train (70%) and temp (30%)
    train_df, temp_df = train_test_split(model_df, test_size=0.3, random_state=42, stratify=model_df['label'])

    # Further split temp into test (50%) and validation (50%)
    test_df, validation_df = train_test_split(temp_df, test_size=0.5, random_state=42, stratify=temp_df['label'])

    # Print the shapes of the split DataFrames
    print("Train set shape:", train_df.shape)
    print("Test set shape:", test_df.shape)
    print("Validation set shape:", validation_df.shape)
```

```
Train set shape: (37067, 4)
Test set shape: (7943, 4)
Validation set shape: (7944, 4)
```

Samples of training, testing and validation datasets are given in figure 34, figure 35 and figure 36 respectively.

Figure 34

Sample of the train_data



```
train_data.head()
```

	SUBJECT_ID	HADM_ID	TEXT	readmitted
0	29226	112295	npn 0700pmh 75 yo m hx parkinsons presents wit...	0.0
1	49359	144898	616 pm chest portable ap ...	0.0
2	77654	163742	baseline artifact is present sinus bradycardia...	0.0
3	13010	116862	atrial fibrillation premature ventricular con...	0.0
4	22320	177984	gi bleeding study ...	0.0

Figure 35

Sample of the test_data

```
[16] test_data.head()
```

	SUBJECT_ID	HADM_ID	TEXT	readmitted
0	14225	191798	1244 pm chest preop pa lat ...	0.0
1	16966	123375	1041 am mr l spine scan ...	0.0
2	46114	120212	323 pm chest preop pa lat ...	0.0
3	32483	139440	633 pm chest portable ap ...	0.0
4	98595	133357	758 am abdomen supine erect ...	0.0

Figure 36

Sample of the validation_data

```
validation_data.head()
```

	SUBJECT_ID	HADM_ID	TEXT	readmitted
0	55008	193168	1023 am chest preop pa lat ...	0.0
1	59388	122391	130 am ct head wo contrast ...	0.0
2	3739	193286	resp care notept received from er intub placed...	0.0
3	79673	187424	154 pm chest preop pa lat ...	0.0
4	31334	156101	ccu npnplease see fhpa for pmh events leading ...	0.0

3.6 Data Statistics

'Admission' and 'note events' CSV files were extracted from Physionet's MIMIC III for the Hospital Readmission Prediction System project. A bar chart representing the count of null values and not null values is given in figure 37.

Figure 37

Bar Graph presenting Count of null values

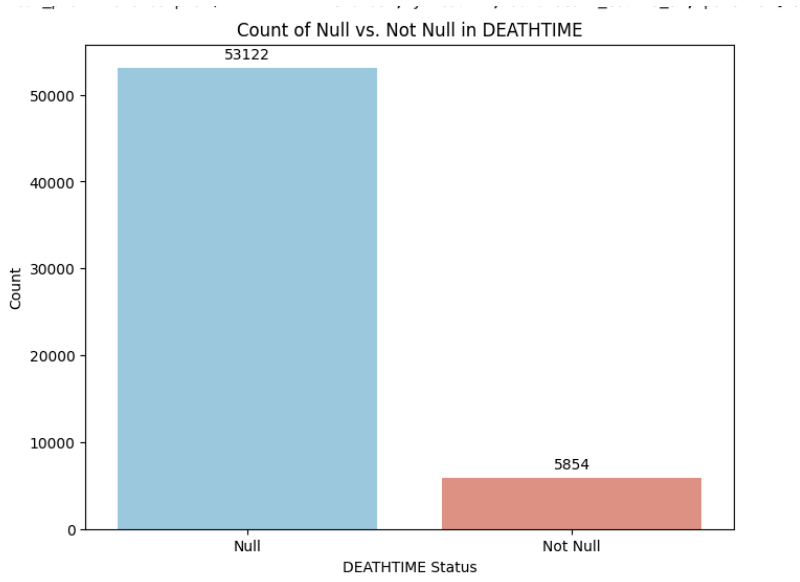


Figure 37 depicts the count of non-null and null values for the 'DEATHTIME' column. This count distinguishes between recorded death times (non-null) and instances where no death times

were recorded (null). Such insights are essential for analyzing patient outcomes like readmission, as patients with recorded death times may be excluded from predictive models. Therefore, understanding the distribution of null and non-null values for 'DEATHTIME' informs decisions on data inclusion and model development.

Figure 38

Count of patients by Readmission Status

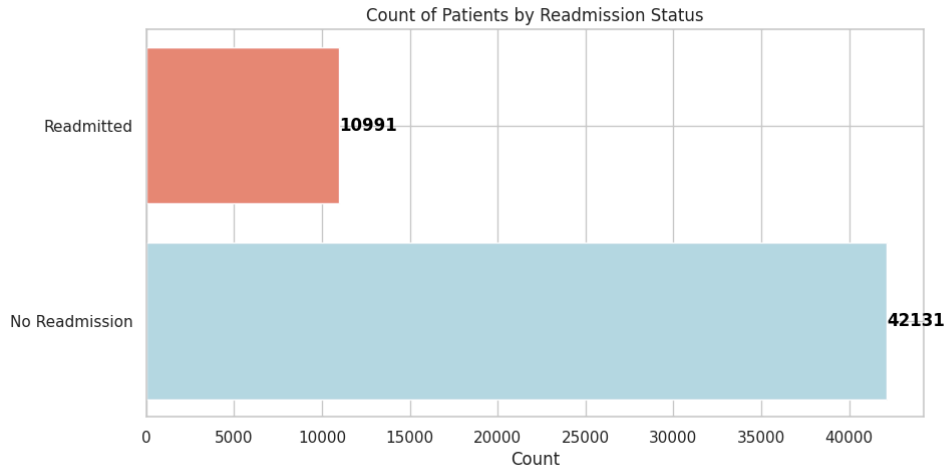
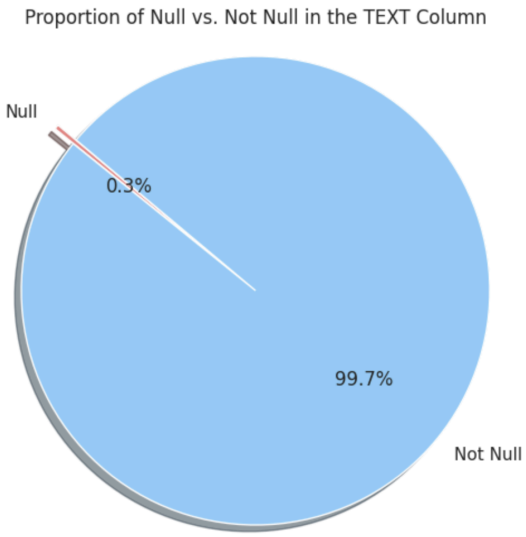


Figure 38 also presents the distribution of readmission statuses. Specifically, it reveals that out of the total entries, 10,991 instances are labeled as "Readmitted," indicating cases where patients were readmitted within a specific time frame. Conversely, 42,131 entries are categorized as "No readmission," indicating instances where patients were not readmitted within the specified period.

A pie chart representing the proportion of null values in the dataset is represented in figure 39.

Figure 39

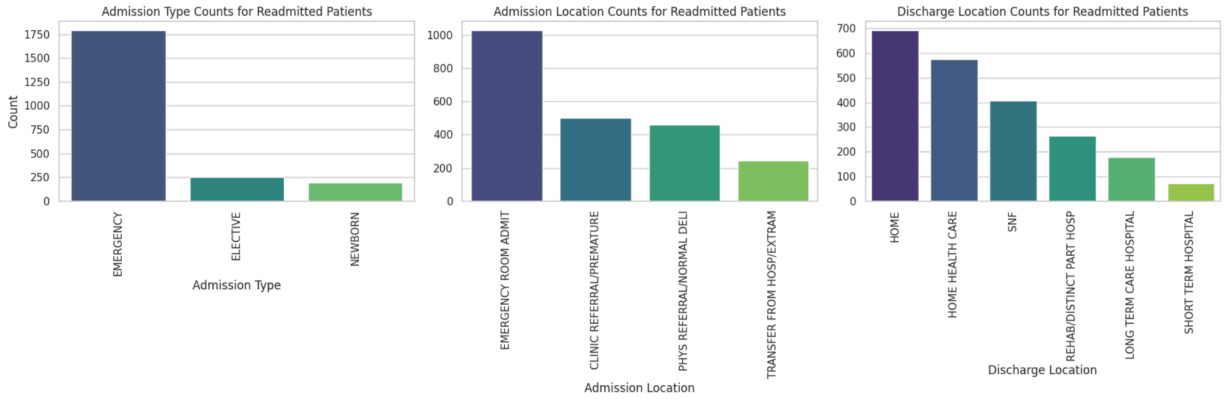
Pie Chart representing the proportion of null values



The bar charts representing admission type counts, admission location counts for readmitted patients discharge location counts for readmitted patients is shown in the figure 40.

Figure 40

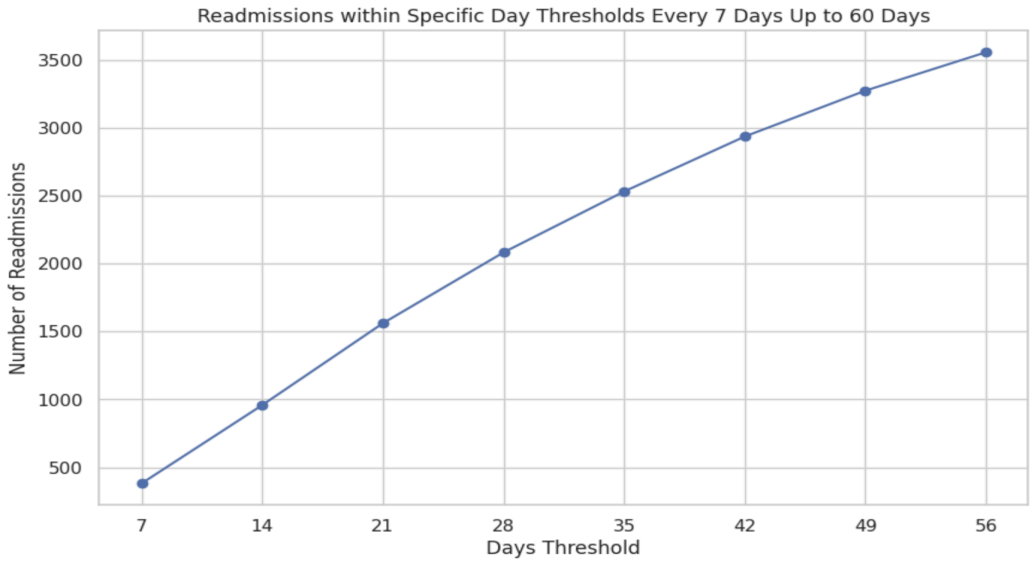
Bar charts representing admission type counts, admission location counts for readmitted patients discharge location counts for readmitted patients



A line chart representing readmissions within Specific Day Thresholds Every 7 Days Up to 60 Days is given in figure 41.

Figure 41

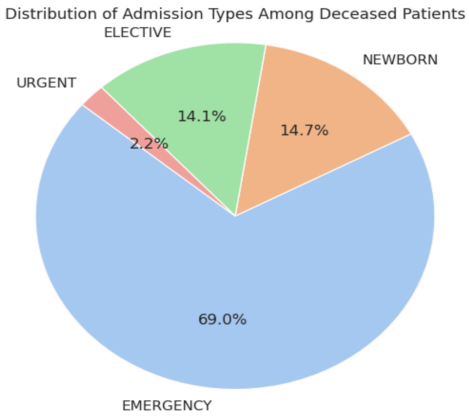
Readmissions within Specific Day Thresholds (7-Day Increments, Up to 60 Days)



Further, a pie chart representing distribution of admission types among deceased patients is given in figure 42.

Figure 42

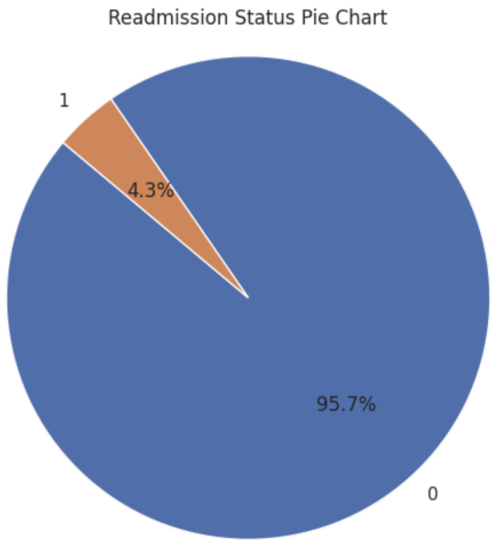
Pie chart representing distribution of admission types among deceased patients



The model_dataset has 52954 records and has 4 features that are Subject ID, HADM ID, Text and readmission status. A pie chart is plotted on the readmission status and is shown in figure 43.

Figure 44

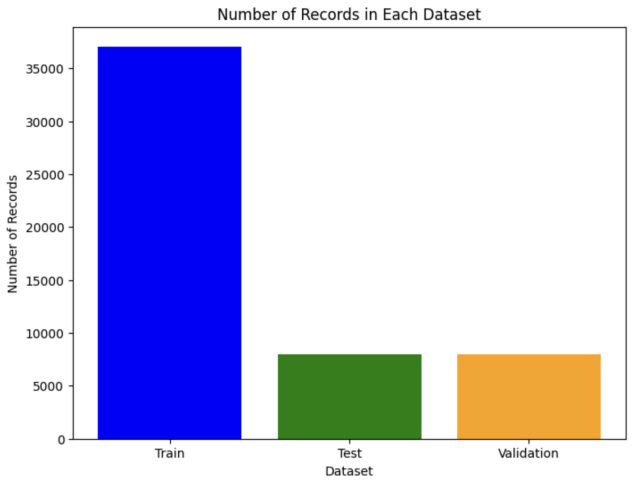
Pie Chart representing the readmission count



The model_data is then divided into train, test and validation datasets in the proportion of 70:15:15. A bar graph representing this shown in figure 45.

Figure 46

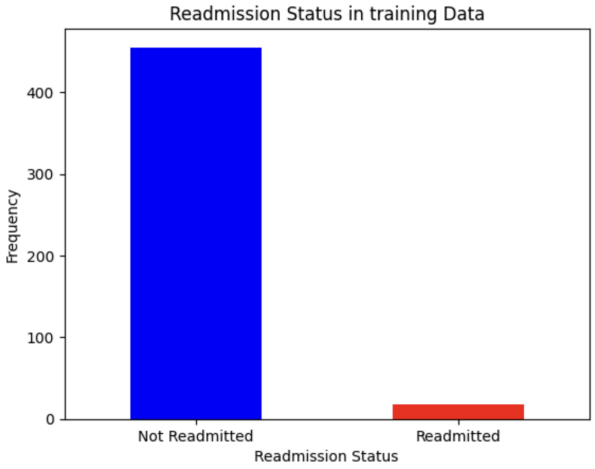
Bar Graph representing the total number of records in each dataset



The train dataset has 37067 records. A graph representing the count of readmission status is given in figure 47.

Figure 47

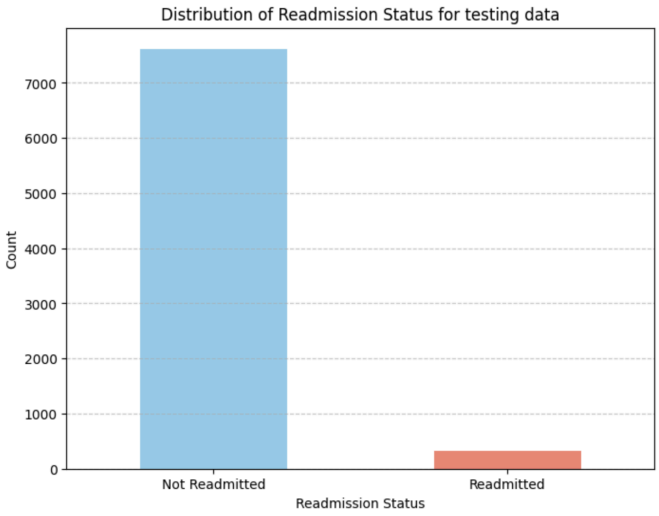
Bar graph resenting the readmission status in training data.



The test dataset has 7943 records. A bar graph representing the count of readmission status is given in figure 48.

Figure 48

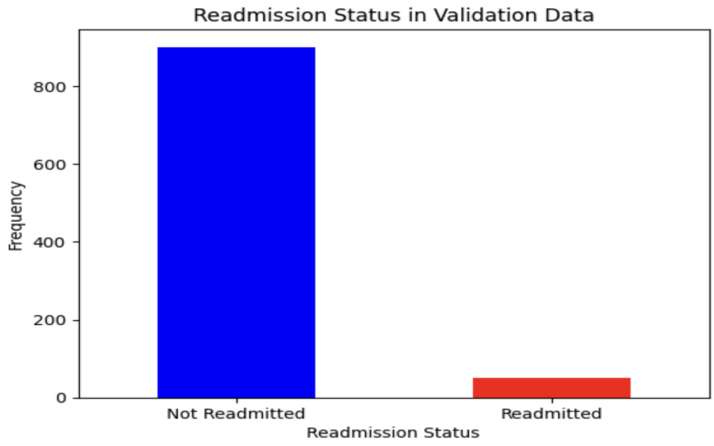
Bar graph resenting the readmission status in testing data.



The validation dataset has 7944 records. A bar graph representing the count of readmission status is given in the figure 49.

Figure 49

Bar graph resenting the readmission status in validation data.



4.1 Model Proposals

The 'admission' and 'noteevents' CSV files have been used to extract the MIMIC III dataset from Physionet for the Hospital Readmission Prediction System project, with the aim of identifying trends related to hospital readmissions. To create a single patient profile, the first step is to link two different files together using the'subject_id' as a key. This allows you to combine information about individual patient admissions and the clinical notes that go with them. Following merger, the data undergoes a thorough preprocessing step that includes normalizing text entries, cleaning the data to eliminate errors and inconsistencies, and converting categorical data into a format that can be analyzed. The extraction of pertinent variables that may affect readmission risks, including as diagnosis codes, treatment information, and discharge summaries, is another step in the transformation process. The emphasis is on a 31-day period after discharge, examining data to find significant elements and trends that might result in a patient being readmitted. The window was selected with the goal of capturing the most essential time prone to readmissions, taking into account clinical significance and statistical analysis. The preprocessed

structured data serves as the basis for constructing a RAG in order to obtain the necessary readmission predictions.

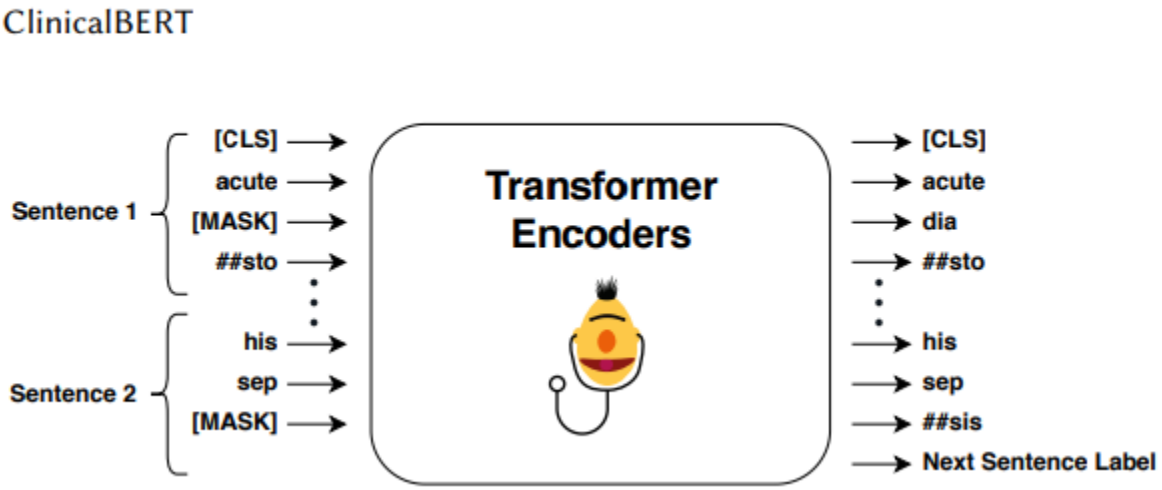
Clinical BERT

Imasogie(2023) utilized the Clinical BERT in predicting the readmission chances of a patient using the clinical notes in the MIMIC III dataset and compared the study with the other model which is BILSTM and the model performed by displayed an AUROC of around 6.5% for a 2 day prediction and the study also involves 3 day prediction which resulted in an 9.6% AUROC when compared with the BILSTM

Clinical BERT is a large language model which is built on BERT(Bidirectional Encoder Representations from Transformers) model which is trained on clinical datasets, which typically include electronic health records, medical research articles, and other types of medical documentation. This specialized training helps the model understand and interpret the nuances and complexities of medical terminology and clinical language.

Figure 50

Clinical Bert Transformer Encoder Working



*Note.*This picture is taken from the paper by Huang(2019)

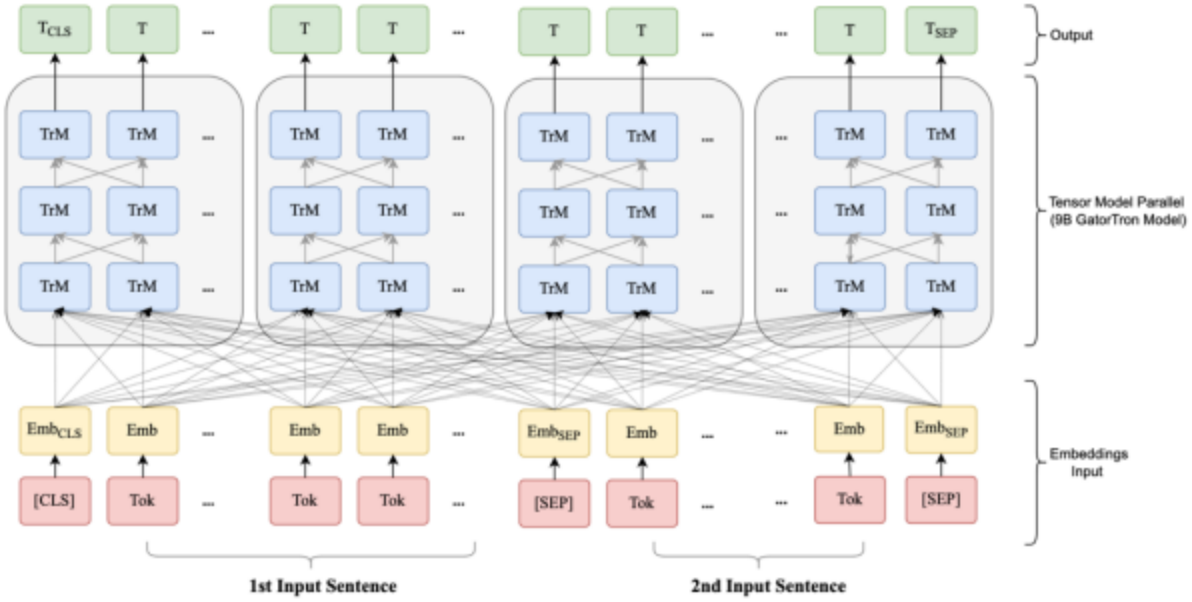
Huang(2019) explained Clinical BERT develops detailed representations of clinical text through two unsupervised learning tasks: masked language modeling, where certain input tokens are omitted and must be predicted, and next sentence prediction, where the model determines if two given sentences are sequentially connected shown in figure 50.It processes clinical notes by breaking the text into subword tokens, each represented by a combination of token embeddings, segment embeddings, and position embeddings. Segment embeddings distinguish between different sequences, while position embeddings indicate the token's location in the sequence. A special [CLS] token is added at the beginning of each sequence to facilitate classification tasks.

Gatatron

Cheng et al. (2023) developed GatorTronGPT, a large language model based on GPT-3 architecture, for medical research and healthcare. Through Turing evaluation and synthetic text generation, GatorTronGPT exhibited competitive performance in biomedical question answering. It achieved a high score of 0.451 compared to BioLinkBERT on the MedQA dataset and approached BioGPT's top score on the PubMedQA dataset, scoring 0.776. These results highlight GatorTronGPT's potential for advancing biomedical research and healthcare applications.

Figure 51

Gatatron Encoder Working



Note. This figure is taken from the paper by Yonghui et al.(2022)

Yonghui et al.(2022) suggested the GatorTron model is an advanced NLP model based on the BERT architecture, pre-trained from scratch on a corpus exceeding 90 billion words using the byte pair encoding algorithm. GatorTron was trained using two self-supervised learning tasks: masked language modeling (MLM) and sentence-order prediction (SOP) shown in Figure 51 . For MLM, the model masks 15% of the input tokens with a special [MASK] token, then predicts these masked tokens. In the SOP task, the model is given two consecutive text segments in random order and must predict if they are in the correct sequence. The GatorTron model has variants including a large version with 8.9 billion parameters, which is trained using model parallelism across multiple GPUs due to its size, while the base and medium models are trained without this slicing approach. The model uses the standard loss function defined in the original BERT model.

$$P_i = \frac{e^{C_i}}{\sum_{j=1}^{N} e^{C_j}} \quad (1)$$

In equation 1 Yonghui et al.(2022) calculated the probability of a given sample classified to specific category where N is the number of categories, Ci is the score generated by a transformer model for category i, where Pi is probability of a given sample to be classified to category i.

LLAMA2

Touvron et al. (2023b) introduced a series of openly released language models that demonstrate competitive performance with leading foundation models. Notably, LLaMA2-13B surpasses GPT-3 while being significantly smaller, and LLaMA2-65B matches the performance of Chinchilla-70B and PaLM-540B. These achievements were realized using solely publicly available data, eschewing proprietary datasets. The release aims to spur further development in the language model field, enhancing robustness and addressing issues like toxicity and bias. Promising results were also seen from instruction-based finetuning, with plans for future exploration and the development of even larger models trained on more extensive corpora.

LLAMA2 (Longform Language Model with Amnesia) is a large language model developed by Meta AI. Based on the transformer architecture, it utilizes attention mechanisms to capture long-range dependencies in text data. LLAMA2 was pre-trained on a massive corpus of internet data, including websites and books, with filtering for offensive content. It tokenizes input text into sequences processable by the model. During pre-training, LLAMA2 employed a self-supervised objective of predicting the next token given previous tokens.

Touvron et al. (2023b) suggested LLAMA2 possesses a unique aspect which is the amnesia mechanism, where it was trained to forget specific input parts, aiming to improve generalization

and abstract reasoning. Efficient parallelism techniques allowed LLAMA2 to scale to larger model sizes while maintaining computational efficiency. Like other language models, LLAMA2 can be fine-tuned on specific tasks using task-specific data. During inference, it generates text by predicting the next token iteratively. While LLAMA2 exhibits strong performance, it may inherit biases from its training data, necessitating careful deployment considerations.

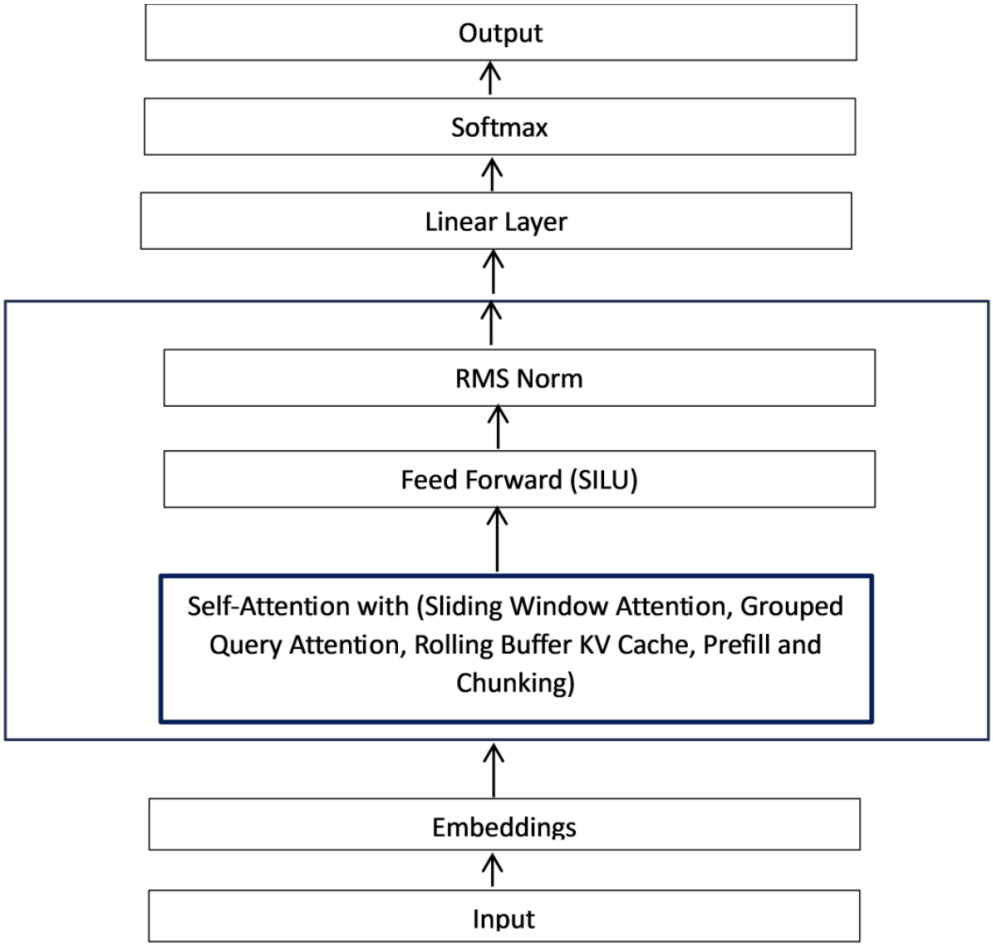
Mistral 7B

Singhal et al. (2023) suggested Mistral-7B, an advanced AI model, demonstrates significant improvements in medical question answering, scoring 86.5% on the MedQA dataset, a 19% increase from its predecessor. It also shows superior performance across other medical datasets. Employing a combination of refined large language model technologies, medical-specific finetuning, and novel prompting methods, Mistral-7b excels in detailed human evaluations, often outperforming physician responses.

Singhal et al. (2023) mentioned Mistral-7b is an advanced AI model developed for medical question answering, significantly built upon the Mistral base model and enriched with targeted medical domain-specific finetuning. This model utilizes several sophisticated prompting techniques, including instruction finetuning—where it is trained across a variety of datasets for both multiple-choice and long-form questions to enhance uniformity in its responses. It also employs few-shot prompting to prime the model with example inputs and outputs, thereby aiding contextual understanding. Further refining its capabilities, the Chain-of-Thought (CoT) prompting adds step-by-step reasoning to the prompts, enhancing transparency and logic in the responses. The Self-Consistency (SC) method generates multiple answers to aggregate the most reliable response through internal consensus.

Figure 52

Architecture of Mistral 7B



Note. This figure is taken from the paper by Singhal et al. (2023)

Lastly, figure 52 describesEnsemble Refinement (ER) is utilized in a two-stage process where initial answers are generated and then refined through repetitive sampling, particularly enhancing multiple-choice question answering by producing more accurate and thorough responses shown in figure 38. Collectively, these methodologies enable Mistral 7B to achieve substantial performance improvements, marking significant progress in AI-driven medical diagnostics and decision support.

4.2 Model Supports

As part of the hardware requirements of project a local computer equipped with 8 GB of RAM, a 256 GB SSD, an 8-core CPU, and an 8-core GPU is utilized to obtain the dataset regarding the storage of dataset which is obtained from Physionet website, google drive is used to store the dataset in a private folder with limited access to only teammates.In addition to local computer, a high performance system from university is also utilized with a RAM of 128GB and 16-core processor of 4.5GHz and

Google Drive is employed as the primary data storage solution for its scalable cloud storage capabilities and robust access control options, to restrict to only teammates. This strategy helps to safeguard the patients demographics data and the respective clinical notes.

Jupyter Notebook is utilized as an interactive coding environment for live code execution, data visualization. Its integration with Python ensures compatibility with a plethora of libraries including TensorFlow, scikit-learn, PyTorch, and PySpark making it an irreplaceable environment in working with the complex projects.PySpark is critical in processing large datasets through in-memory computation essential for the large-scale data handling typical in hospital readmission analyses Both Matplotlib and Seaborn are useful for data visualization, enabling the production of various graphs and charts with improved statistical visualization features. One essential library is Scikit-learn (Sklearn), which provides a whole toolkit for mining, evaluating, and choosing machine learning models.

TensorFlow and PyTorch are employed to manage the demands of deep learning models involved in the project, such as ClinicalBERT and GatorTron. These libraries support model development and training through functionalities like automatic differentiation and large-scale optimization. They are optimized for GPU utilization, ensuring efficient training of deep learning models.

The transformers library from Hugging Face is central to implementing BERT-based architectures such as ClinicalBERT, GatorTron, Mistral 7B, Llama2 and integrate it with the RAG architectures, offering access to pre-trained models adaptable for specific tasks like hospital readmission prediction.

Table 8

Libraries required for Hospital Readmission System

Library	**Method**	**Purpose**
pandas	Data manipulation and analysis	Data manipulation
matplotlib	Data visualization	Data visualization
numpy	Numerical computing	Numerical computing
transformers	ClinicalBERT and other NLP models	Natural Language Processing
sklearn	Machine learning algorithms	Machine learning
sklearn.model_selection	Train-test split and cross-validation	Model selection
sklearn.feature_extraction.text	Feature extraction using HashingVectorizer	Feature extraction
imbalanced-learn	Handling imbalanced datasets	Imbalanced dataset handling
spacy	NLP Library	Natural language processing
nltk	Natural Language Toolkit	Natural language processing

Library	Method	Purpose
re	Regular expressions	Text pattern matching
torch	Deep learning framework	Deep learning
tensorflow	Deep learning framework	Deep learning
keras	Deep learning framework	Deep learning
llama2	Biomedical text mining framework	Biomedical text mining
pedpalm	Framework for analyzing pediatric gait data	Pediatric gait analysis
GATATron	Graph Attention Networks framework	Graph Attention Networks
DataLoader	Data loading utility	PyTorch data loading
RandomSampler	Random sampling utility	PyTorch random sampling
TensorDataset	Dataset utility	PyTorch dataset creation
BertTokenizer	Tokenization for BERT models	Tokenization for BERT models
BertForSequenceClassification	BERT model for sequence classification	BERT-based sequence classification

Library	Method	Purpose
AdamW	Optimizer for BERT models	AdamW optimizer for models
Elasticsearch	Knowledge Base Indexing & Searching	To index and retrieve clinical data and literature that can provide context or supporting information to enhance prediction accuracy.
FAISS	Efficient Similarity Search	For fast retrieval of relevant documents or notes based on vector similarity,enhancing the model's contextual awareness.

Table 8 outlines a broad spectrum of libraries and frameworks that are crucial in the fields of data science, machine learning, and natural language processing. Foundational libraries such as pandas, matplotlib, and numpy provide essential tools for data manipulation, visualization, and numerical analysis. More specialized resources include transformers, which support advanced NLP models like ClinicalBERT, and other NLP-focused libraries such as spacy and nltk that enhance language processing capabilities. The sklearn library enriches the machine learning landscape with robust algorithms and tools for model selection and feature extraction, while imbalanced-learn specifically tackles the challenges posed by imbalanced datasets. For deep learning endeavors, frameworks like torch, tensorflow, and keras are pivotal in facilitating the

development, training, and deployment of neural networks. Additionally, domain-specific tools such as llama2 for biomedical text mining and ped palm for pediatric gait analysis cater to specialized needs. PyTorch utilities like DataLoader, RandomSampler, and TensorDataset play key roles in efficient data handling and processing, crucial for managing large datasets. The use of BERT-specific tools such as BertTokenizer and BertForSequenceClassification illustrates the integration of advanced model-based techniques in sequence classification tasks, bridging traditional machine learning with cutting-edge NLP technology.

Model Architecture and DataFlow

The knowledge base in the Retrieval-Augmented Generation (RAG) model is a critical component that enables the system to augment its language generation capabilities with information retrieved from a vast, pre-encoded dataset. Typically, this knowledge base is composed of a large corpus of documents or data such as Wikipedia, which is processed and embedded into a dense vector space using techniques like the Dense Passage Retrieval (DPR) method. During the model's operation, when RAG receives a query or a prompt, it performs a vector-based search in this dense space to retrieve the most relevant documents or passages. These retrieved segments then inform and guide the generative model, enhancing its responses with factual, context-specific information, thereby combining the benefits of neural generative models with those of information retrieval systems to generate more accurate and contextually appropriate outputs.

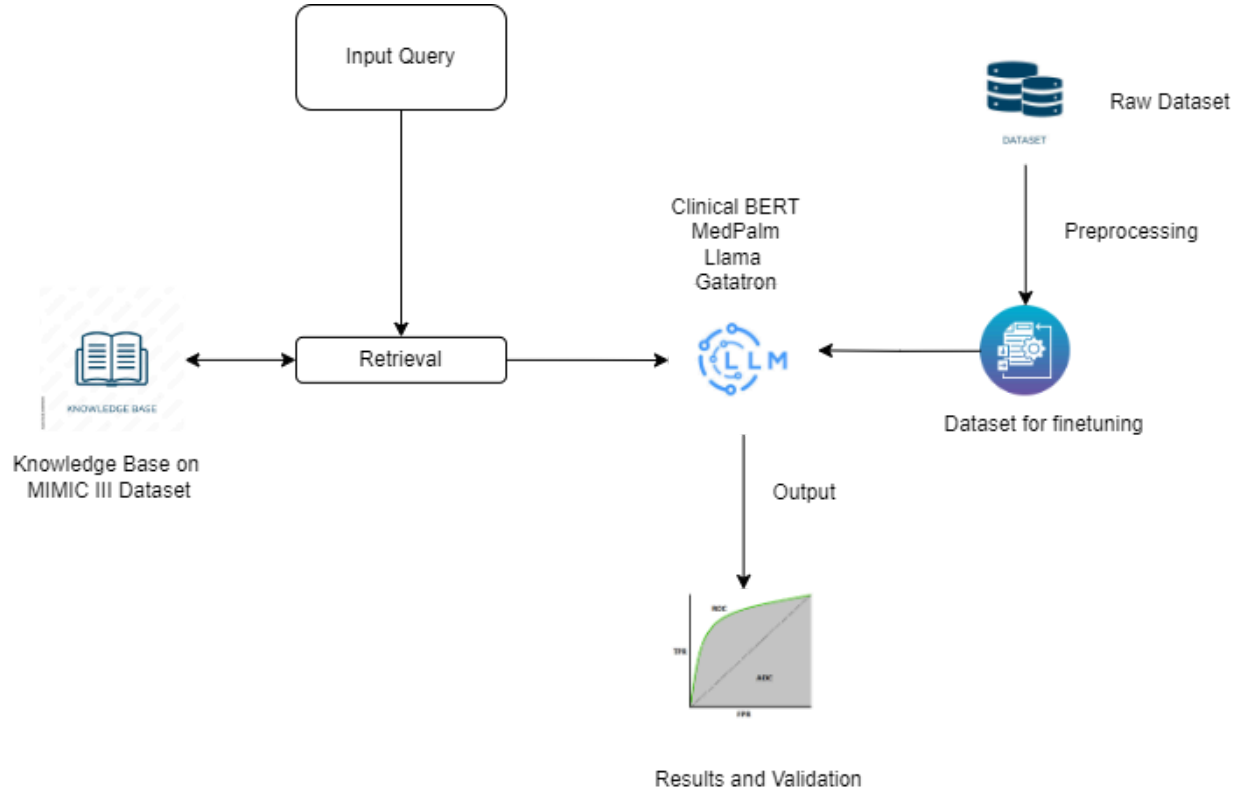
In the RAG model, the retriever component plays a pivotal role by sourcing pertinent information from an external knowledge base to enhance response accuracy and relevancy. The process begins with the pre-training of a Dense Passage Retrieval (DPR) model, which encodes all documents in the knowledge base into high-dimensional vectors, creating a vector space

where documents are indexed for efficient retrieval. When a query or prompt is presented to the RAG system, the retriever promptly encodes this input into a similar vector format using the same DPR model. This encoded query is then used to perform a similarity search in the pre-encoded vector space of documents. The search identifies and retrieves the most relevant documents based on vector closeness, effectively bridging the query with related external information. These retrieved documents are then passed to the generator component of RAG, which synthesizes the information into coherent and contextually enriched responses, leveraging both the generative capabilities of neural language models and the informational depth of retrieval systems.

In figure 53 data flow of the hospital readmission system using the different large language models such as Clinical BERT,Mistral 7B, Llama2, Gatatron are being used. After establishing a knowledge base filled with relevant documents,by using the MIMIC III dataset and index this using tools like Elasticsearch or FAISS for efficient retrieval. Subsequently retrieval method needs to be implemented by choosing the Dense Passage Retrieval as retrieval system .A raw dataset needs to be taken which will be pre processed to prepare it for fine tuning and train it with query-document pairs to create a meaningful embedding space.

Figure 53

Dataflow and Model Architecture of Readmission Prediction System

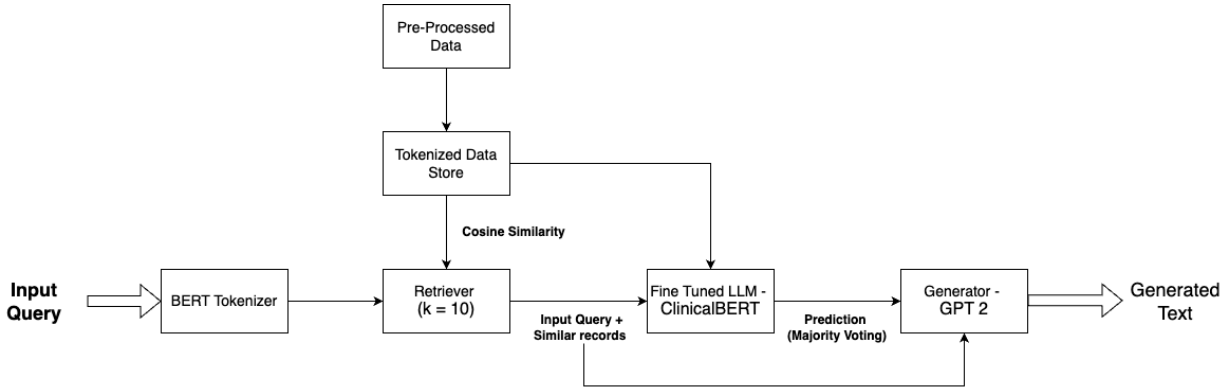


Integrate the retriever with the LLM by having the retriever fetch relevant documents based on input queries, which are then provided to the LLM, modifying its inputs to include both the original prompt and the content of the retrieved documents, and ensuring the LLM can handle the additional context effectively. Optionally, some implementations of RAG support end-to-end training of both the retriever and generator. Evaluate the system using relevant metrics like accuracy or F1 score, and iterate on the process by refining the knowledge base, improving the retriever, or further fine-tuning the LLM based on performance feedback.

ClinicalBERT RAG Architecture

Figure 54

ClinicalBERT RAG Architecture

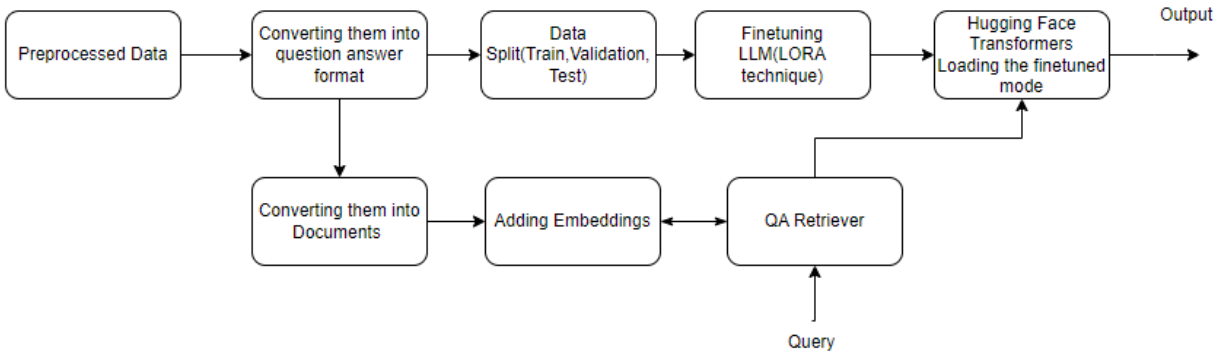


Once the data pre-processing is completed, the data is tokenized using BERT Tokenizer and stored as a data store which can further be used for fine tuning the model as well as be used by the retriever. The input query is also tokenized using BERT Tokenizer and passed to the retriever to fetch similar records from the data store. Cosine similarity is used for comparison and top 10 relevant records are fetched from the data store. The 10 records along with the original query are then passed to the fine tuned ClinicalBERT model for predictions on readmission. Majority voting is conducted and final prediction is noted and it is passed to the GPT2 generator along with all the queries for Text generation.

Llama2 RAG Architecture

Figure 55

LLama2 RAG Architecture for Prediction System



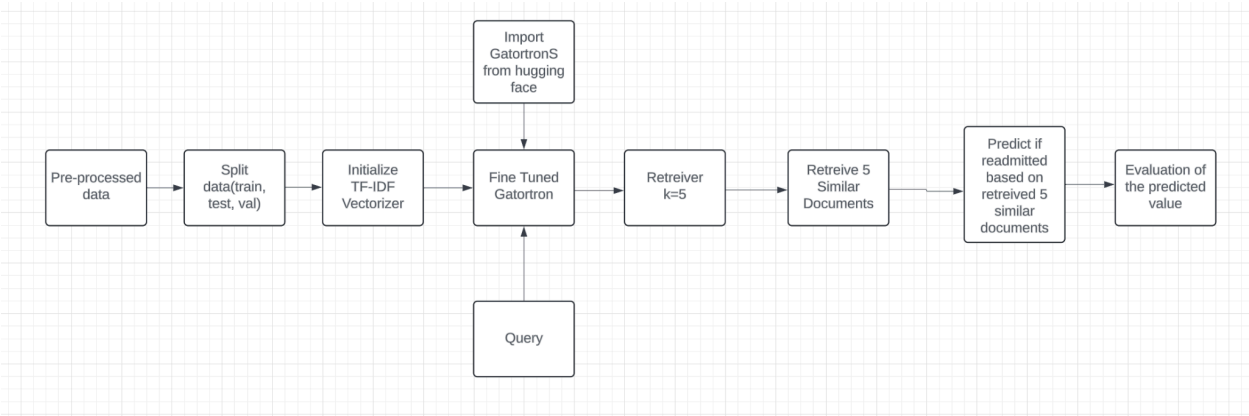
Once the preprocessing of the data has been completed it is being converted into question answer format which is being split into train,validation and test dataset to fine tune the Llama and Mistral model using LoRA technique. During this process the parameters are fine tuned to achieve the best accuracy and that model will be sent to Hugging Face Transformers library by creating a repo which can be accessed through the specific token pertaining to the repo.

In regarding with RAG entire dataset will be considered which is converted into question and answer format which will be considered as the vector database by converting them into documents by adding the necessary embeddings.QA Retriever from the LangChain is used as a retriever for which input query will be given inbuilt package will convert that query into tokens and it hits the vector database and database will search all its documents and send the nearest index document to the retriever which will communicate with fine-tuned LLM model to generate the predicted output in a text format and it also generates the context using the LLama indexing.

Gatortron

Figure 56

Gatortron RAG Architecture for Prediction System



To build a RAG using Gatortron, the data is divided into training, testing, and validation sets during this pre-processing stage. In order to translate textual data into matrix forms, a TF-IDF Vectorizer is initialised.It will convert the TEXT column into vectors and then matrices. After that, the query is run through GatortronS, model that was fine-tuned after taken from the Hugging Face library. Using a k value of 5, a retriever is used to obtain the top 5 similar documents based on the query. Based on the data in these retrieved documents, a prediction is made as to whether or not a patient will be readmitted. Lastly, a performance and accuracy comparison between the predicted values and the ground truth labels is made. The model can predict patient readmission by using contextual data from related documents thanks to this workflow.

4.3 Model Comparison and Justification

ClinicalBERT

ClinicalBERT, as detailed by Huang et al. (2020), is meticulously designed for clinical prediction tasks, specifically in forecasting 30-day hospital readmissions. It leverages the sophisticated architecture of bidirectional encoder representations from transformers (BERT) and employs attention mechanisms to make predictions more interpretable. This focus makes it particularly adept at processing clinical notes, where its ability to dynamically adjust risk scores based on

continuous patient data proves invaluable for clinicians' decision-making. In contrast, LLama extends its capabilities not only by integrating large language models with generative AI but also by actively learning from human feedback, presenting a broader scope of applications beyond clinical predictions. Similarly, GatorTron prioritizes synthetic text generation for biomedical applications and addresses privacy concerns but does not cater directly to predictive tasks in clinical settings. On the other hand, Mistral 7B 2 excels in delivering unparalleled accuracy and relevance in medical responses, although it is not specifically designed for predictive tasks like ClinicalBERT. Despite ClinicalBERT's specialized approach offering clear advantages in clinical environments, it also faces challenges such as the need for extensive fine-tuning and task-specific optimization, which requires significant technical expertise, contrasting with the user-ready nature of models like Mistral 7B 2 that are pre-optimized for broader accessibility and user interaction.

LLama2

LLama2, as highlighted by Yu et al. (2023), excels in advancing generative AI capabilities through the strategic integration of large language models (LLMs) with generative AI techniques. This combination allows it to effectively process and utilize vast amounts of data, especially from electronic health records (EHRs), enhancing its application across various domains, including healthcare. LLama2's strength lies in its ability to augment human capacities, facilitating improvements not just in data handling but also in decision-making processes. In contrast to ClinicalBERT, which is finely tuned for specific predictive tasks in clinical settings, LLama2 adopts a broader approach. While ClinicalBERT excels at interpreting early clinical notes or discharge summaries for predicting hospital readmissions, LLama2 leverages its generative capabilities to provide insights across a wider spectrum of healthcare issues, learning

continuously from human feedback to refine its outputs. Additionally, compared to models like GatorTron, which focuses on synthetic text generation for privacy preservation in biomedical data, and Mistral 7B 2, which targets medical question answering, LLama2's approach is more holistic. It not only engages with the generative aspects of AI but also embeds adaptability and scalability, making it suitable for a diverse range of applications beyond those of its peers. This broad scope and adaptability underscore LLama2's potential to enhance healthcare AI applications, as it not only addresses immediate clinical needs but also provides a foundation for future innovations in the field.

GatorTron

GatorTron, as detailed by Cheng et al. (2023), excels in two specialized areas within the realm of healthcare AI: generating synthetic clinical text and enhancing biomedical natural language processing (NLP) capabilities. This model is uniquely designed to safeguard privacy by training on synthetic data that does not expose sensitive medical information, distinguishing it from other AI models that may require access to real patient data. Unlike ClinicalBERT, which is geared towards precise clinical predictions such as hospital readmissions, GatorTron prioritizes the creation of synthetic text that mimics real patient narratives without compromising privacy. This feature is particularly valuable in environments where data sharing is restricted by confidentiality concerns. Moreover, in comparison to LLama2, which leverages generative AI for a broad range of applications, GatorTron's focus is sharply defined within the synthetic text generation domain, aiming to improve the security and privacy of data used in biomedical research and healthcare applications. GatorTron's approach combines advanced deep learning techniques with large-scale feature extraction, enabling it to handle complex biomedical data effectively. This methodology, however, does lead to variability in performance depending on the

dataset and task at hand, as the quality of synthetic text can differ based on numerous factors. This variability is a critical point of consideration for researchers and practitioners deploying GatorTron in diverse clinical and research settings, underscoring its niche but essential role in the landscape of healthcare AI.

Mistral 7B

Jiang et al. (2023) introduced Mistral 7B, a 7 billion parameter language model and is an open source language model. It is an efficient and high performing 7-B language model having strong capabilities for various NLP tasks. It makes use of grouped-query attention (GQA) for faster inference speed and lower memory usage when decoding. It also makes use of sliding window attention (SWA) to manage lengthy sequences efficient and at a reduced computing cost. Mistral 7B is not domain-specific, in contrast to clinical BERT, which is limited to the medical domain. By fine tuning the Mistral 7B model it can be suitable for a variety of tasks. Clinical BERT has higher accuracy in the healthcare domain because it was trained on vast amounts of medical text data. Llama2 can produce imaginative text structures that could strike a balance between ability and efficiency. Mistral 7B exhibits superior performance compared to Llama2 across various language tasks, suggesting it may be more capable for fundamental language tasks. While Llama2 excels in conversational and generative abilities, its emphasis on dialogue generation could potentially compromise the deep medical comprehension provided by specialized models like clinical BERT.

Table 9

Model Comparisons

Model	ClinicalBERT	LLama 2	GatorTron	Mistral
Basic Architecture	BERT-based encoder-decoder model	LLMs integrated with generative AI	Large-scale feature extraction, synthetic text generation	Hybrid of transformers and reinforcement learning
Types of Data	Clinical literature, patient notes	Electronic health records (EHRs), healthcare data	Clinical text, synthetic clinical text	Textual Data Medical datasets, clinical records |
Known Issues	Fine-tuning requirements, task specificity	Scoping review exhaustiveness	Privacy-related concerns with real clinical data	Generalization against overfitting
Data Size	Moderate to Large	Large	Varies	Moderate to Large
Complexity	Medium to High	Medium	Varies	High
Strengths	Accurate readmission predictions, interpretable predictions	Enhancing generative AI capabilities, vast data utilization	Enhancing biomedical NLP, privacy preservation	Improved question answering, relevancy, and quality

Model	ClinicalBERT	LLama 2	GatorTron	Mistral
Limitations	Fine-tuning overhead, task-specific tuning	Scoping review comprehensiveness	Privacy challenges with real data, performance variability	Require more training periods to achieve optimal performance

4.4 Model Evaluation Methods

The evaluation of a model's efficacy relies on assessing specific parameters or metrics, especially in research focused on classification. In this study, the effectiveness of the RNN model is measured using essential metrics like Precision , Accuracy, F1 Score, Recall and ROC/AUC. These metrics are crucial in assessing the model's capacity for accurate classification and its overall performance.

Figure 57

Class designation		Actual class	
		True (1)	False (0)
Predicted class	Positive (1)	TP	FP
	Negative (0)	FN	TN

Confusion Matrix

Note: Referred from paper by Vujović (2021)

The metrics Precision , Accuracy, F1 Score, Recall are calculated from a confusion matrix, utilized for assessing the performance of a classification model, is a table that helps evaluate its effectiveness in classification tasks. The confusion matrix utilizes four fundamental components to gauge a model's classification accuracy: True Positives (correct predictions of the

positive class), False Positives (incorrect predictions of the positive class in actual negative cases), False Negatives (incorrect predictions of the negative class in actual positive cases), and True Negatives (correct predictions of the negative class).

Precision

As discussed by Vakili et al. (2021), Precision serves as an indicator of how accurately an algorithm predicts positive outcomes. It is calculated by dividing the number of true positive predictions (TP) by the total positive predictions made (TP + FP). It is calculated by dividing the number of true positive predictions by the sum of true positive and false positive predictions. Precision can range from a perfect score of 1.0 to a minimum of 0.0, reflecting the percentage of relevant selections that are actually correct.

$$Precision = \left(\frac{TP}{TP+FP} \right) \qquad (2)$$

Recall

According to Vakili1 et al. (2021), Recall illustrates the algorithm's proficiency in identifying positive observations, showing how many relevant data items were accurately selected. Precision is determined by dividing the number of true positive predictions by the sum of true positives and false positives. This calculation helps gauge the algorithm's effectiveness in identifying positive instances within the dataset.

$$Recall = \left(\frac{TP}{TP+FN} \right) \qquad (3)$$

F1-Score

In binary classification, the F1 Score acts as a balanced metric, merging precision and recall into a single value that reflects the overall performance of the model. It provides a comprehensive

evaluation by taking into account both false positives and false negatives, offering a unified assessment of the model's effectiveness across different scenarios.

$$F1\ Score = \left(\frac{2 \times (Precision \times Recall)}{(Precision + Recall)} \right) \quad (4)$$

Accuracy

As highlighted by Vakili1 et al. (2021), Accuracy is frequently used and typically the first choice for evaluating algorithm performance in classification tasks. The accuracy of a model is calculated by summing up the number of true positive and true negative predictions (TP + TN) and then dividing this sum by the total size of the dataset (P + N). This calculation offers a direct measure of the model's correctness in its predictions.

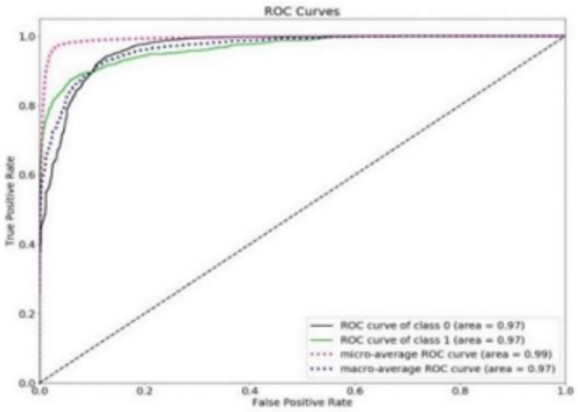
$$Accuracy = \left(\frac{TP + TN}{P+N} \right) \quad (5)$$

ROC/AUC Score

The ROC/AUC Score metric, which stems from ROC analysis, showcases the connection between the true positive rate (also known as sensitivity or recall) and the false positive rate. Vujović (2021) notes that this score reflects the model's ability to rank predictions accurately, offering insight into its ranking performance.

Figure 58

ROC Curve



Note: Above Image referenced from Vujović (2021)

Figure 58 displays The ROC curve illustrates the equilibrium between the False Positive Rate (FPR) and the True Positive Rate (TPR) across various classification thresholds. Each point on this curve reflects how each threshold affects the correct identification of positive cases as opposed to the incorrect labeling of negative ones. Ideally, a curve that approaches the top-left corner indicates high sensitivity and minimal false positives. Curves that tilt towards the left suggest better performance, characterized by higher sensitivity and fewer false positives. The ROC AUC score quantifies the model's overall performance in classification tasks to accurately rank predictions, where TPR is defined as the Sensitivity at the ith threshold and FPR is the False Positive Rate at the same threshold.

$$AUC = \sum_{i=1}^{n-1} \frac{(TPR_I + TPR_{I+1}) \times (FPR_{I+1} - FPR_I)}{2} \qquad (6)$$

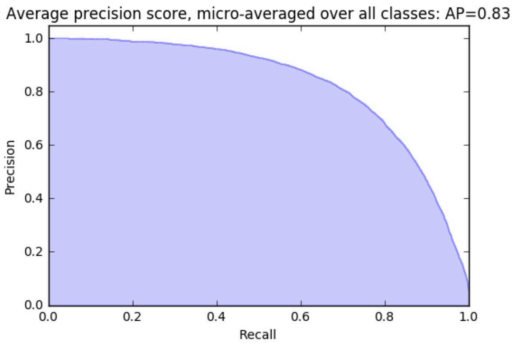
AUPRC

AUPRC, or Area Under the Precision-Recall Curve, is a crucial metric for assessing classification models in machine learning, particularly useful when working with imbalanced datasets where the positive class is significantly outnumbered. This metric focuses on the ability

of a model to identify the positive class accurately, crucial in situations where reducing false positives is vital, such as in medical diagnostics or fraud detection.

Figure 59

PRC Curve



Note: Above Image referenced from windweller (2018)

The AUPRC provides a comprehensive single-value summary of a classifier's performance, emphasizing its capability to maintain high precision and recall. This means the model not only identifies a significant number of true positives but does so with minimal incorrect positive predictions. Thus, AUPRC offers a more nuanced understanding of model effectiveness, especially in critical scenarios where detecting the positive cases accurately is paramount, and is generally more informative than metrics like accuracy or the area under the ROC curve.

Rogue Score

Max Grusky (2023) notes that the text similarity metric ROUGE is now among the most widely used evaluation tools in natural language processing. Initially developed for assessing summarization models, ROUGE is quite adaptable and can evaluate various generative tasks, including question answering. ROUGE assesses the similarity between the generated text and the reference text by comparing units like words, n-grams, or sequences. This comparison aids in evaluating how accurately the generated content reflects the essence of the reference materials.

References

A. Shiju and Z. He, "Classifying Drug Ratings Using User Reviews with Transformer-Based Language Models," 2022 IEEE 10th International Conference on Healthcare Informatics (ICHI), Rochester, MN, USA, 2022, pp. 163-169, doi: 10.1109/ICHI54592.2022.00035

Ali, H., Qadir, J., Alam, T., Househ, M., & Shah, Z. (2023). ChatGPT and Large Language Models in Healthcare: Opportunities and Risks. *IEEE*. https://doi.org/10.1109/aibthings58340.2023.10291020

Cheng, P., Yang, X., Chen, A., Smith, K. E., PourNejatian, N., Costa, A., Martin, C., Flores, M. G., Zhang, Y., Magoč, T., Lipori, G., Mitchell, D. A., Ospina, N. S., Ahmed, M. M., Hogan, W. R., Shenkman, L., Guo, Y., Bian, J., & Wu, Y. (2023). A study of generative large language models for medical research and healthcare. Npj Digital Medicine, 6(1). https://doi.org/10.1038/s41746-023-00958-w

Elgedawy, R., Srinivasan, S. K., & Danciu, I. (2024). Dynamic Q&A of Clinical Documents with Large Language Models. arXiv (Cornell University). https://doi.org/10.48550/arxiv.2401.10733

Ganesh, J., & Bansal, A. (2023). Transformer-based Automatic Mapping of Clinical Notes to Specific Clinical Concepts. *IEEE*. https://doi.org/10.1109/compsac57700.2023.00080

Goel, A., Gueta, A., Gilon, O., Liu, C., Erell, S., Nguyen, L. H., Hao, X., Jaber, B. a. M., Reddy, S. R. N., Kartha, R., Steiner, J. L., Laish, I., & Feder, A. (2023). LLMs accelerate annotation for medical information extraction. *arXiv (Cornell University)*. https://doi.org/10.48550/arxiv.2312.02296

H. Ali, J. Qadir, T. Alam, M. Househ and Z. Shah, "ChatGPT and Large Language Models in Healthcare: Opportunities and Risks," 2023 IEEE International Conference on Artificial

Intelligence, Blockchain, and Internet of Things (AIBThings), Mount Pleasant, MI, USA, 2023, pp. 1-4, doi: 10.1109/AIBThings58340.2023.10291020

Huang, K., Altosaar, J., & Ranganath, R. (2020). ClinicalBERT: Modeling clinical notes and predicting hospital readmission. CHIL ’20: ACM Conference on Health, Inference, and Learning; Workshop Track, 9. https://arxiv.org/pdf/1904.05342.pdf

Imasogie, N. (2023) ClinicalBERT: Using a deep learning transformer model to predict hospital readmission. *ICDM*. https://ieee.icdm.com/clinicalbert-using-deep-learning-transformer-model-to-predict-hospital-readmission-c82ff0e4bb03

J. Ganesh and A. Bansal, "Transformer-based Automatic Mapping of Clinical Notes to Specific Clinical Concepts," 2023 IEEE 47th Annual Computers, Software, and Applications Conference (COMPSAC), Torino, Italy, 2023, pp. 558-563, doi: 10.1109/COMPSAC57700.2023.00080

Jin, M., Yu, Q., Zhang, C., Shu, D., Zhu, S., Du, M., Zhang, Y., & Meng, Y. (2024). Health-LLM: Personalized Retrieval-Augmented Disease Prediction Model. arXiv (Cornell University). https://doi.org/10.48550/arxiv.2402.00746

Ke, Y., Jin, L., Elangovan, K., Abdullah, H. R., Liu, N., Sia, A. T. H., Soh, C. R., Tung, J. Y. M., Ong, J. C. L., & Ting, D. S. W. (2024). Development and testing of retrieval augmented generation in large language models -- a case study report. arXiv (Cornell University). https://doi.org/10.48550/arxiv.2402.01733

Lamproudis, A., Henriksson, A., & Dalianis, H. (2022, June 1). Evaluating pretraining strategies for clinical BERT models. ACL Anthology. https://aclanthology.org/2022.lrec-1.43

Li, Y., Li, Z., Zhang, K., Ruilong, D., & Zhang, Y. (2023). ChatDoctor: A medical chat model Fine-Tuned on a large language model Meta-AI (LLAMA) using medical domain knowledge. arXiv (Cornell University). https://doi.org/10.48550/arxiv.2303.14070

McCleary, K., Ghawaly, J., Louisiana State University, & LSU Health New Orleans. (2024). TNM Tumor Classification from Unstructured Breast Cancer Pathology Reports using LoRA Fine Tuning of Mistral 7B. TNMTumorClassification From Unstructured Breast Cancer Pathology Reports Using LoRA Fine Tuning of Mistral 7B.

Moerschbacher, A., & He, Z. (2023). Building Prediction Models for 30-Day Readmissions Among ICU Patients Using Both Structured and Unstructured Data in Electronic Health Records. *IEEE*. https://doi.org/10.1109/bibm58861.2023.10385612

N. -Y. Tung, H. -W. Hu, T. -W. Chang, Y. -M. Hu and H. -M. Lin, "Multi-model Comparison for Classification of Medical Records using the BioBERT Models," 2022 IEEE 4th Eurasia Conference on Biomedical Engineering, Healthcare and Sustainability (ECBIOS), Tainan, Taiwan, 2022, pp. 221-224, doi: 10.1109/ECBIOS54627.2022.9945029.

Qian, J., Jin, Z., Zhang, Q., Cai, G., & Liu, B. (2024). A liver Cancer Question-Answering system based on Next-Generation Intelligence and the large model Mistral 7B 2. International Journal of Computer Science and Information Technology, 2(1), 28–35. https://doi.org/10.62051/ijcsit.v2n1.04

Singhal, K., Tu, T., Gottweis, J., Sayres, R., Wulczyn, E., Hou, L., Clark, K., Pfohl, S., Cole-Lewis, H., Neal, D., Schaekermann, M., Wang, A., Amin, M., Lachgar, S., Mansfield, P., Prakash, S., Green, B., Dominowska, E., Arcas, B. a. Y., . . . Natarajan, V. (2023). Towards Expert-Level Medical Question Answering with Large Language Models. arXiv (Cornell University). https://doi.org/10.48550/arxiv.2305.09617

Tung, N., Hu, H. T., Chang, T., Hu, Y., & Lin, H. (2022). Multi-model Comparison for Classification of Medical Records using the BioBERT Models. *IEEE*. https://doi.org/10.1109/ecbios54627.2022.9945029

Vasantharajan, C., Tun, K. Z., Ho, T., Jain, S., Tong, R., & Siong, C. E. (2022). MEDBERT: a pre-trained language model for biomedical named entity recognition. *2022 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*. https://doi.org/10.23919/apsipaasc55919.2022.9980157

Wang, X., Tao, M., Wang, R., & Zhang, L. (2021). Reduce the medical burden: An automatic medical triage system using text classification BERT based on Transformer structure. *Ieee*. https://doi.org/10.1109/icbase53849.2021.00133

Yang, R., Tan, T. R., Lu, W., Thirunavukarasu, A. J., Ting, D. S. W., & Liu, N. (2023). Large language models in health care: Development, applications, and challenges. HealthCare Science, 2(4), 255–263. https://doi.org/10.1002/hcs2.61

Yu, P., Xu, H., Hu, X., & Deng, C. (2023). Leveraging Generative AI and large language Models: A Comprehensive Roadmap for Healthcare integration. Healthcare, 11(20), 2776. https://doi.org/10.3390/healthcare11202776

Ran Elgedawy, Sudarshan Srinivasan, Ioana Danciu (2024). Dynamic Q&A of Clinical Documents with Large Language Models, https://doi.org/10.48550/arXiv.2401.10733

Windweller (2023), Cross Validated https://stats.stackexchange.com/users/204873/windweller

Grusky, M. (2023). Rogue Scores. Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers) https://doi.org/10.18653/v1/2023.acl-long.107

Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M., Lacroix, T., Rozière, B., Goyal,

N., Hambro, E., Azhar, F., Rodriguez, A., Joulin, A., Grave, É., & Lample, G. (2023b). LLAMA: Open and Efficient Foundation Language Models. *arXiv (Cornell University)*. https://doi.org/10.48550/arxiv.2302.13971

Vakili, M., Ghamsari, M., & Rezaei, M. K. (2020). Performance analysis and comparison of machine and deep learning algorithms for IoT data classification. *arXiv (Cornell University)*. https://doi.org/10.48550/arxiv.2001.09636

Vujović, Ž. (2021). Classification model evaluation metrics. *International Journal of Advanced Computer Science and Applications/International Journal of Advanced Computer Science & Applications*, *12*(6). https://doi.org/10.14569/ijacsa.2021.0120670

APPENDIX

Google Drive Link: https://drive.google.com/drive/u/1/folders/0ABdqz6S2chBaUk9PVA