Hospital Readmission Prediction System With LLM

Jaya Lakshmi Gunji, Naveen Yadav Gongati, Neha Dabeeru, Priyanka Akula, Somna Sattoor

Department of Data Analytics, San Jose State University

Spring 24:DATA 298A-11

Simon Shim

May 13, 2024

ABSTRACT

Purpose

This study will, therefore, focus on predicting the rehospitalization of diabetic patients to reduce hospital readmissions. The developed predictive tool, after the analysis of the UC Irvine machine learning repository data, will help establish a reliable way of identifying diabetic patients who are at an increased risk of readmission, which may help in proactive management strategies to enhance outcomes in diabetes care.

Tasks

The main objectives include the following points:. Improve models' prediction by training on the electronic health records of patients who have diabetes to be able to identify important factors contributing to the risks associated with readmission.Building an EHR-Driven Hospital Readmission Prediction System using Python-based back-end systems that can effectively deal with a large amount of EHR data and for scalability. Implementation of advanced methods, like hyperparameter tuning and ensemble learning, to increase the training and performance of predictive models. Conduct in-depth evaluations using the appropriate metrics and quality-assessment protocols to maintain the reliability and effectiveness of the predictive system.

Outcomes

With this, a developed Hospital Readmission Prediction System could be used for identifying patients at risk and an estimation of their likely readmission. The model should be seamlessly integrated with the existing healthcare IT frameworks, and robust evaluation tools developed to demonstrate the effectiveness of the developed model to identify at-risk patients who may likely be readmitted. The predictive system is aimed at enabling preemptive interventions, personalized care plans, and better resource utilization to lower readmission rates and improve patient care in diabetes management.

Applications

The prediction system can be adopted in different healthcare settings to improve patient care and management. The healthcare provider will be able to apply prevention timely enough to stop the patient from being readmitted to the hospital. It is an integrated system that will improve risk assessment, care coordination, and bring the best outcome for a patient with diabetes. The focus is on fine-tuning predictive analytics in diabetes treatment for optimization of health care services and better health for the patient.

Project Background And Execute Summary

Hospitals have big problems in the health sector with respect to readmitting patients into the hospital that increases the cost and worsens the health of patients. This research work will develop an innovative Hospital Readmission Prediction System that exploits the state-of-the-art large language models such as ClinicalBERT, ChatGPT, LLAMA2, Mistral AI, ClinicalXLnet, among others. This system will then enable the healthcare professionals to do an early and proactive risk identification and management for the patients at risk of being readmitted, and in turn, lead to improved patient outcomes with a reduction of healthcare costs. It is primarily achieved through the development of predictive models, deployment of advanced data analytic techniques, and seamless integration with existing health care infrastructures using Python, having a user-friendly RAG application interface developed in Figma. This project is anticipated to reduce the rates of avoidable readmissions and improve the effectiveness of healthcare interventions with long-term benefits for health care management and predictive analytics.

Project Background, Needs And Importance

Hospital readmissions represent a significant financial burden and an issue of welfare in the healthcare industry. Adequate integration of fast-evolving medical technologies guarantees

the availability of on-time solutions to avoid readmissions. Such readmissions increase the healthcare costs and affect the physical and mental health of patients.

This is where the Hospital Readmission Prediction System with Large Language Models comes in. The system empowers healthcare providers with effective and accurate predictive tools, using the best of breed LLMs—be it ClinicalBERT, ChatGPT, LLAMA2, Mistral AI, and ClinicalXLnet—such that predictions are made to identify patients who are highly probable for readmission very early in advance, which will certainly improve patient outcomes and subsequently lower unnecessary hospital visits.

This will revolutionize patient care by integrating state-of-the-art technologies. Prediction of early readmission bestows a strategic edge to the system for better management of patient health. This proactive approach will not only reduce the costs of healthcare by decreasing repeated treatments but also set up new benchmarks in patient management and predictive healthcare analytics. The project will, to a large extent, improve long-term outcomes and help set new standards for technology integration into healthcare practices.

Project Problem, Motivations And Goals.

Hospital readmissions adversely affect patient health and raise costs in healthcare facilities. As medical technology has increased, preventable readmissions show the need for better treatment.

Towards this, this research initiative operationally lays out the Hospital Readmission Prediction System with Large Language Models using ClinicalBERT, ChatGPT, LLAMA2, Mistral AI, and ClinicalXLnet at the front line to assist health practitioners in identifying at-risk patients. The technology predicts readmission and helps in timely and effective treatment, resulting in improvement of patient outcomes and reduction of unnecessary hospital stays.

This study applies recent modeling to predict hospital readmissions and alter patient care. Such predictive modeling helps in cost reduction in terms of healthcare, since it helps in the reduction of repeated treatment and readmission that is not required. It also helps in betterment in treatments related to healthcare and advancements in the management of patients with the use of predictive healthcare analytics. The project will focus on new guidelines on the use of medical technology in order to improve healthcare practices over time.

Project Approaches And Methods.

The Hospital Readmission Prediction System, using Large Language Models (LLMs), is deliberately created to effectively bring forth advanced predictive analytics and data processing in dealing with the complete address of preventable hospital readmission. This model uses advanced language models like ClinicalBERT, ChatGPT, LLAMA2, Mistral AI, and ClinicalXLnet, amongst others. These models are then used as the basis of the predictive system in detecting those at risk by analyzing vast healthcare data, such as patient records and clinical outcomes. This approach develops the use of advanced data analytics techniques, such as Python for the back-end and pre-processing of data, to ensure that the system integrations are smooth and accurate within the available healthcare IT infrastructures. Additionally, the project supports a RAG application prototyped in Figma to ensure smooth and effortless use on a daily basis by medical professionals.

This would call for periodic and continuous training and testing with new and varied datasets to keep the effectiveness and adaptability of the system. Such an iterative process enhances the accuracy of the algorithms in readmission forecasting and allows timely and cost-effective treatments. The developed system is tested in a controlled healthcare environment with immediate input and incorporation where necessary. This feedback loop is critical in

improving performance and scaling the system for adaptability in dynamic requirements in the healthcare business. These are the ways that the project will seek to lower the rates of readmissions and the costs of healthcare, while at the same time creating new heights in patient care and predictive healthcare analytics.

Expected Project Contributions And Applications

The task assignment of the project team was very effective in providing each of the members with an opportunity to utilize their unique skills, although their schedule was a little varied based on their availability. Jaya Lakshmi has been effective in giving momentum to the project and led the work on modeling and data management. Naveen Yadav has been pretty critical in setting the requirements of the project, making sure all deliverables are met, and covering the initial data analysis work. This is the job that was taken on by Somna Sattoor in project management. She focused on planning and data analysis, to get data sets ready for close examination. Priyanka Akula was assigned creating the project's methods, going through academic study, and judging how well the models worked. Neha Dabeeru wasn't able to help all the time, but she helped at the first steps of data preparation, making the data better and getting it ready for further analysis. Of course, about 22% of the project's work was up to each team member. The method that was fair helped the group work well together and move the project forward.

Table 1

Team Contribution On Project

Team Member	Responsibilities and Contributions	Workload Share
Jaya lakshmi	supervised the fundamental elements and processes of the project, with a specific emphasis on sophisticated modeling approaches and data management.	22%
Naveen Yadav	Managed the intricate project requirements, prioritized the execution of deliverables, and assumed responsibility for the first data collection and analysis.And conducted a thorough evaluation of the application.	22%
Somna Sattoor	Supervised project operations and logistics, contributed to data trend analysis, and aided in data preparation for the final study.	22%
Priyanka Akula	Managed the project's schedule and approach structuring, made contributions to scholarly review, and oversaw performance metrics for models.I created a RAG application.	22%
Neha Dabeeru	Engaged in data manipulation and preparation activities, but had a diminished role due to restricted availability.	12%

Note. This Table Gives A Overview Of Tasks Done By Each Individual In A Team

Application. There is a new, groundbreaking application to further a hospital's ability to predict and plan for patient readmission. The application uses the latest innovations in large language models to comb through enormous amounts of information, including patient histories, treatment results, and various health markers. This software, by merging those models, identifies

trends and risk signals that might be left out by the traditional analysis methods. It helps health workers make better decisions so that patients get better care, with fewer preventable readmissions and better quality of life. At the same time, it greatly improves resource allocation in a hospital. More precise and effective treatment brings more welfare and happiness to patients.

Project Requirements

Functional Requirements That Can Be Tested And Measurable

This project will develop an automated Hospital Readmission Prediction System using the Diabetes dataset from the Health Facts Database by Cerner Corporation, Kansas City, MO and the US Healthcare Readmissions and Mortality dataset from Kaggle. The system will, in turn, utilize information, including but not limited to, that which is existing in the datasets on hand to quantify the risk of hospital readmissions by five state-of-the-art Large Language Models: ClinicalBERT, ChatGPT, LLAMA2, Mistral AI, and ClinicalXLnet. Data preprocessing will be done in Python within a Jupyter Notebook to extract and clean relevant information such as records of hospital readmission, medical history, and patient demographics, which will guarantee high-quality inputs during both the training and evaluation stages. The model performances are checked, ensuring metrics like an accuracy threshold of at least 85%, perplexity less than a set value, F1-score greater than 0.75, BLEU and ROUGE Scores to evaluate the linguistic quality and other domain-specific metrics that associate with the likelihood of readmission. Predictive algorithms are implemented in TensorFlow, while data storage is in Google Cloud Platform (GCP), with the interface for Chatbot and Readmission Assistance Guide (RAG) built-in Figma. The best model among these will be plugged into these applications so that health providers get predictions and insights on the patient instantly. In a dynamic way, visually portraying the data findings and the model performance, a Tableau

dashboard automatically updated, has been developed, to give a view to the stakeholders on the outcomes. This project thus intends to not only arm medical facilities with highly accurate hospital readmissions prediction but also to eliminate readmissions that could be prevented along with their associated costs and improve patient outcomes using data-driven interventions in the context of individual patient profiles.

Nonfunctional Project Requirements That Can Be Tested And Measurable.

The AI-powered development requirements for the Hospital Readmission Prediction System were to realize measurable outcomes in analytics for patient data and prediction accuracy. It used LLM models for ClinicalBERT, ChatGPT, LLAMA2, Mistral AI, and ClinicalXLNet. All these LLM models have been developed using advanced methods for natural language processing, specifically designed for clinical purposes, and they can extract invaluable information from unstructured clinical text data comprising electronic health records and medical notes.

Process and analyze the patient records from the datasets provided using ClinicalBERT, ChatGPT, LLAMA2, Mistral AI, and ClinicalXLNet to identify the associated clinically relevant features and the patterns of readmissions to the hospital. The models should detect readmission factors with high accuracy, considering the complex architecture and medical term understanding in context.

The further assessment of LLMs is done based on their capability to extract therapeutically important information from the datasets and forecast patient readmission, ensuring that the results have recall rates higher than 90%. It is thus the given stringent criteria of evaluation that ensure the robustness and reliability of the deployed models under practical deployment in healthcare settings.

They will also have continuous training and upgrading capabilities through feedback systems to further enhance the predictive power of the LLM models over time. The Hospital Readmission Prediction System is intended to provide healthcare providers with insight into the risks of readmission of the patients by using these special capabilities integrated into ClinicalBERT, ChatGPT, LLAMA2, Mistral AI, and ClinicalXLNet in processing clinical text data. This will eventually enhance patient outcomes and resource use in health facilities.

Two different datasets have been selected to build the models for predicting hospital readmissions. One is a dataset based on rates of hospital readmissions and deaths from Kaggle (The Devastator, 2023). It has patient records related to different diseases, including patient records for diabetes, of course. Additionally, this dataset provides the type of admission, length of stay, medical history, medications received, results of laboratory tests, diagnosis, and count of inpatient and outpatient visits. Thus, information on the age, gender, and race of the patient is abundant. The dataset includes missing values, so their treatment has to be done during the data preprocessing step.

The second dataset, from the source HealthFacts database by the Cerner Corporation, Kansas City, MO, relates to patients with diabetic-related information covering ten years of clinical treatment among 130 hospitals and integrated delivery networks within the US. Similarly, this database also consists of information on a large number of patients, consisting of the demographics of the patients, patient information regarding the admission of the patient, medical specialty, and lab findings, HbA1c test results, information regarding the diagnosis, the prescribed drug information, and the number of inpatient and outpatient visits. We build robust prediction models using a set of Large Language Models based on ClinicalBERT, ChatGPT, LLAMA2, Mistral AI, and ClinicalXLNet. Since the LLMs are catered for health-care

applications, the text information related to the patient can be the input to such LLMs to make them exceptional in getting meaningful information. Therefore, large amounts of text data from both datasets are going to be used in the model training process, where LLMs are fine-tuned and trained with the purpose of predicting the readmission to the hospital.

Project Deliverables

Therefore, the project, after much study, research, and understanding is poised to deliver the following factors, each having an acceptable estimation point:.

Project Proposal

Description. Attempted working on many problems statements and scenarios before concluding the best procedure to implement to address the problem. Hospital readmission is a formidable challenge for global healthcare in terms of patient outcome and cost. Aiming to develop an LLM-based advance hospital readmission prediction system to tackle the issue, the project will involve the following steps: evaluating patient data from selected datasets, and precise prediction of the hospital readmission probability. The project will apply the CRISP-DM Agile framework and will have the following stages: business understanding, data understanding, data preparation, modeling, evaluation, and deployment. Each stage will be achieved iteratively in smaller sprints in the light of Agile principles. The project will work with several LLM models, some of them including ClinicalBERT, ChatGPT, LLAMA2, Mistral AI, and ClinicalXLNet.

Due Date: 02/14/2024

Project Breakdown

Below is a project breakdown of the six stages:.

Business Understanding. This is the stage in which the goals and objectives of a project are identified and assessed through a general review of internet resources, including the identification of variables that will be important in predicting the patterns of re-admittance in hospitals. There will also be stakeholder consultations in order to learn more from their needs and expectations.

Data Understanding. The study uses two major datasets: the Diabetes dataset obtained from the Health Facts database (Cerner Corporation, Kansas City, MO) and the US Healthcare Readmissions and Mortality dataset obtained from Kaggle (The Devastator, 2023). Each dataset has several patient attributes such as age, gender, race, previous medical records, and results of treatment applied. Developing effective models requires an understanding of the structure and nature of these datasets. Appropriate exploratory data analysis approaches will be used to learn more about the distribution and correlations of the data.

Data Preparation. Pre-processing procedures for the data will include handling missing data, encoding categorical variables, and scaling numerical features. Other feature selection methods available to find relevant features to predict hospital readmissions. Data quality tests will also be done to ensure the datasets are consistent and have integrity.

Modeling. Among other LLMs, this project will train ClinicalBERT, ChatGPT, LLAMA2, Mistral AI, and ClinicalXLNet on the data it is provided. The fine-tuning and optimization of the model for prediction of hospital readmissions will be done. Ways to explore building further robustness and increasing the model generalization include cross-validation and hyperparameter tuning.

Evaluation. All LLMs will be evaluated in terms of Accuracy, Perplexity, F1-score, BLEU Score, ROUGE Score, and domain-specific metrics. This is done with the objective of

finding the model that performs best for the prediction of hospital readmission. In addition to this, model interpretation techniques will also be applied to gain insight into the variables that affect the risks of readmission.

Deployment: After a final model is chosen, it will be tested in a real-time application using LLM to predict the likelihood of hospital readmission. Based on the input derived from data about the patient, the application will draw the selected model and make predictions.

JIRA will be used in the management of project tasks and milestones. A WBS will also help in organizing project tasks, and Gantt and Pert charts will manage project durations and progress.

Due Date: 02/23/2024

Project Plan

The project plan follows the CRISP-DM approach and consists of six stages: business understanding, data understanding, data preparation, modeling, evaluation, and deployment. The first activity of the project is to establish the goal of the project and the needs of the stakeholders, that is, the hospital readmission prediction with LLMs. The required properties are searched for through an analysis of the datasets: Diabetes dataset from the Health Facts database by Cerner Corporation, Kansas City, MO; the US Healthcare Readmissions and Mortality dataset from Kaggle (The Devastator, 2023); others. Feature engineering and pretreatment of data are then performed to ready the data for model training. In turn, the right readmission prediction of various LLMs, including ClinicalBERT, ChatGPT, LLAMA2, Mistral AI, and ClinicalXLNet, is trained and fine-tuned. Model performance is measured through standard measures. Accuracy of the model is then identified. Finally, the developed model is integrated into the healthcare

systems to receive real-time predictions, and the development of visualization tools allows proper communication of the insights.

Due Date: 02/26/2024

Data Collection

At the data collection stage, datasets necessary for training and validating LLM-based hospital readmission prediction models will be obtained and pre-processed. In addition to the Diabetes dataset from the Health Facts database (Cerner Corporation, Kansas City, MO) and kaggle (The Devastator, 2023) offering the US Healthcare Readmissions and Mortality datasets, additional databases will be looked into, from where healthcare datasets will be collected. Such information may cover the history of hospital readmissions, patient demographics, results of laboratory tests, previous medication, and electronic health records. Additionally, relevant and publicly available medical libraries and research databases will be reviewed in an attempt to gather relevant datasets to ensure the stability and generalizability of the model. First, a follow-up is made to analyze the completeness, quality, and relevance of the dataset in relation to the set-out goals of the prediction. The data captured for training the models will be pre-processed by employing feature engineering, data cleaning, data imputation, and normalization.

Due Date: 03/01/2024

Prototype

The project involves the development of the system of predicting hospital readmission by using five state-of-the-art LLM models: ClinicalBERT, ChatGPT, LLAMA2, Mistral AI, and ClinicalXLNet. The five fine-tuned LLMs will be used for extracting the important content in the records of the patients.

Due Date: 03/15/2024

Testing

The assessment of the LLM model performance is based on the following: accuracy, perplexity, F1 score, BLEU score, ROUGE score, and domain specific metrics. Such a testing battery helps to determine strengths and weaknesses of all the LLM models and, hence, the most suitable model for hospital readmission prediction. Some of the classical and apt characteristics of medical data used include confusion matrix, accuracy rating, and F1 score calculation.

Due Date: 03/20/2024

Final Model

The final model in the hospital readmission prediction system will involve the ClinicalBERT, ChatGPT, LLAMA2, Mistral AI, and ClinicalXLNet models. This model will look through the datasets provided to make predictions related to the probability of readmission. Such a big move in the application of NLP will now make it easier for caregivers to clearly establish their at-risk patients by deriving insights from the medical information and patient records. The model will further be fine-tuned and optimized for the best potential accuracy in predicting the hospital readmission.

Due Date: 04/01/2024

Project Report

The project report will give a full description of the Hospital Readmission Prediction System with LLM and how the project has been targeted to be realized. It will describe the aims, methods, and how the LLM models are used. The study will report the problems faced during model creation, metrics of evaluation used to comprehend the assessed model performance, and

the insights acquired from predictive analysis. Project deliverables and timelines are shown in Table 2 below.

Due Date: 05/13/2024

Table 2

Project Deliverables And Timeline

Deliverable	Description	Due Date
Project Plan	creation of an all-encompassing project plan with deadlines and completion dates	02/26/2024
Data Collection	It allows us to get the Diabetes dataset and prepare it from the Health Facts database (Cerner Corporation, Kansas City, MO), along with the datasets US Healthcare Readmissions and Mortality from Kaggle(The Devastator, 2023), for processing, analysis, and training the models.	03/01/2024
Prototype	Develop LLM models using ClinicalBERT, ChatGPT, LLAMA2, Mistral AI, and ClinicalXLNet.	03/15/2024
Final Model	Development of the final prediction model for hospital readmission with LLMs.	04/01/2024
Project Report	Project documentation in APA format that includes the recommendations, findings, and conclusions.	05/13/2024

Note. This Table Covers The Primary Deliverables And Their Respective Due Dates For The Project Phases, As Well As A Brief Summary Of Each Deliverable.

Technology and Solutions Survey

Clinical BERT

This paper, therefore, deeply discusses how Clinical BERT transformation may be improved in the hospital readmission prediction systems according to two papers. The first paper under review is "Multimodal Prediction of COVID-19 Mortality with Clinical BERT," which

developed a multimodal model for predicting 30-day mortality upon hospital admission in COVID-19 patients by combining structured electronic health records and unstructured clinical notes. The research identifies the several data modalities and the update of Clinical BERT with structured data as being responsible through a systematic evaluation of multimodal fine-tuning for enhancing the predictivity of multiple data aspects. It further illustrates the problems with handling long clinical narratives and the proposed ways to ensure best performance for Clinical BERT text processing on long textual data.

The second paper, "Understanding Patient Complaint Characteristics Using Contextual Clinical BERT Embeddings," studies the use of Clinical BERT for understanding and classifying patient complaint characteristics, a critical aspect in predicting readmission risks (Saha et al., 2020). Clinical text was encoded with the subtlety of complaint features using a model combining Word2Vec with BERT, and the multi-label classification enabled with the LDA algorithm. From the experimental results, it is presented that the accuracy is increased by more than two times to the existing model, indicating the importance of Clinical BERT in reading complex clinical narratives for an accurate prediction of readmission risks. Overall, the present paper confirms Clinical BERT as a game changer in the prediction of readmission in hospitals and a powerful tool to meet the goal of improving patient care and containing readmission.

ChatGPT

The paper by Ali et al. "ChatGPT and Large Language Models in Healthcare: Opportunities and Risks" discusses the opportunities for transformation and the challenges associated with the use of ChatGPT and other large language models within healthcare. As demonstrated, the ability of ChatGPT to facilitate processes, ranging from the generation of patient discharge reports to clinical vignettes and support with preparation for USMLE for enhanced access, scalability, and effectiveness in the healthcare industry, cannot be

underestimated. Capable of integrating enormous amounts of medical information and offering clinical insight, ChatGPT could result in great improvement in patient care. Such risks, which accompany the implementation of AI when dealing with such sensitive matters as medical diagnostics and interactions with patients, include inaccuracy, bias, and problems with regard to ethics. Here, the authors stress that strong validation is needed and that ChatGPT has to be made part of formal healthcare systems to make its use safe and ethical within healthcare systems.

Gupta et al. (2023) assess the development of the ChatGPT model from the very first version to more sophisticated ones in the paper "ChatGPT: Exploring the Capabilities and Limitations of a Large Language Model for Conversational AI". These authors identify an increase in complexity from 117 million parameters in the first version to 175 billion in the latest. They also indicated significant improvements in the generation of coherent, context-aware responses that can take on challenges such as summarisation and translation. Such models have the potential to trickle into upcoming applications, and these advances are affecting a variety of industries from tech to healthcare and education. However, there are some issues with the model at the moment, and taking stock of the limits associated with it, such as emotional intelligence, domain expertise, and potential biases in the training data, they list those out.

Llama-2

The paper "Performance Analysis of LoRA Finetuning Llama-2" provides an in-depth analysis of Llama-2, a large language model (LLM) fine-tuned using the Low-Rank Adaptation (LoRA) technique, while comparing its performance with other models like GPT-4 (Pathak et al., 2023). It underscores Llama-2's potential in various research applications and its suitability for broader usage. Commencing with an overview of large language models and their burgeoning relevance across diverse fields, the study emphasizes the necessity to comprehend their mechanisms, evaluate their impact, and address ethical considerations.

Detailing the AI Transformer architecture, including encoder and decoder components, the paper introduces Llama-2 as a family of pretrained and fine-tuned LLMs with varied parameters, incorporating enhancements over its predecessor, such as training on a larger corpus and employing grouped-query attention. Moreover, it outlines the hyperparameters employed during Llama-2 training, encompassing the AdamW optimizer, learning rate schedule, weight decay, and gradient clipping. Experimental findings reveal Llama-2's robust performance in reward models, demonstrating competitiveness with models like ChatGPT, with its largest Chat model outperforming others on prompt sets. Additionally, the paper elucidates the concept of AI finetuning and presents human evaluation results, underlining Llama-2's significance in contemporary research and practical applications.

In the paper titled "Comprehensive Examination of Instruction-Based Language Models: A Comparative Analysis of Mistral-7B and Llama-2-7B," a comparative study between two LLMs, Mistral-7B and Llama-2-7B, in the realm of natural language processing (NLP) is presented (Thakkar et al., 2023). The analysis, employing performance indicators such as ROUGE, BLEU, and CIDEr, suggests Llama-2-7B's superior performance across various dimensions compared to Mistral-7B, albeit acknowledging Mistral-7B's potential for improvement with larger datasets due to its recent release. The paper introduces LLMs and delineates Mistral-7B's efficiency and versatility across language tasks alongside Llama-2-7B's prowess in discourse applications. The study carefully examines both approaches, using evaluation metrics like BLEU and ROUGE-L, in an effort to identify their advantages and disadvantages. Moreover, it clarifies the architecture and autoregressive character of LLMs and highlights their revolutionary influence on NLP fields, such as text summarization and machine translation.

By comparing Mistral-7B and Llama-2-7B, the paper aims to provide valuable insights into their characteristics and implications within the dynamic landscape of NLP.

Mistral AI

The paper by Tsai, Huang, and Kuo (2024), titled "Comparative Analysis of Automatic Literature Review Using Mistral Large Language Model and Human Reviewers," investigates the effectiveness of the Mistral Large Language Model (LLM) with Retrieval-Augmented Generation (RAG) in automating literature reviews compared to traditional human-led processes.

Analyzing 50 scientific papers from the OpenReview platform, the study evaluates the efficiency, scalability, and quality of Mistral LLM's review. Findings suggest that while Mistral LLM outperforms human efforts in terms of efficiency and scalability, it occasionally lacks the analytical depth and attention to detail exhibited by human reviewers. However, the study emphasizes the potential of Mistral LLM in standardizing preliminary literature reviews, advocating for a hybrid approach integrating Mistral LLM's capabilities with human expertise to enhance the literature review process.

The paper underscores the need for further advancements in AI technology to achieve deeper analytical insights and addresses ethical concerns and biases in AI-assisted research, highlighting the integration of LLMs like Mistral as a promising direction for redefining academic research methodologies.

In the paper "BioMistral: A Collection of Open-Source Pretrained Large Language Models for Medical Domains," BioMistral is introduced as an open-source large language model (LLM) specifically tailored for the biomedical domain (Labrak et al., 2024). Addressing challenges and opportunities in integrating LLMs into healthcare and medicine, the authors develop BioMistral 7B based on Mistral and pre-train it on the PubMed Central dataset. The paper conducts a comprehensive evaluation of BioMistral on a benchmark comprising 10

medical question-answering tasks in English, including analysis of different evaluation strategies. Additionally, the authors explore the development of lightweight models and model merging techniques, demonstrating BioMistral's superior performance compared to existing open-source medical models. The paper presents BioMistral as a powerful open-source LLM tailored for the biomedical domain, offering datasets, benchmarks, preprocessing scripts, and models freely available for further research and practical applications.

Clinical XLNet

The paper titled "Noise Reduction Learning Based on XLNet-CRF for Biomedical Named Entity Recognition" introduces two noise reduction models, Shared Labels and Dynamic Splicing, aiming to enhance the performance of Biomedical Named Entity Recognition (BioNER) systems (Chai et al., 2023). Addressing the limitations of existing BioNER systems, which suffer from noise interference and lack an effective label dependency decoder, the proposed models utilize XLNet encoding, a permutation language pre-training model, combined with Conditional Random Field (CRF) decoding. Through evaluation on 15 biomedical named entity recognition datasets, the models achieve an average F1-score improvement of 1.504 and 1.48, respectively, with state-of-the-art performance attained on 7 datasets. The paper contributes by proposing novel noise reduction models, showcasing their effectiveness in BioNER, and studying the impact of model structures on various feature datasets. It also discusses related work in biomedical named entity recognition, encompassing rules-based, dictionary-based, traditional machine learning, and deep learning methods.

In another study, researchers explore transformer-based learning for negation detection in biomedical texts, addressing negation sentence identification and negation scope recognition subtasks (Althari et al., 2022). Utilizing pre-trained transformer models such as BERT, ALBERT, XLNet, and ELECTRA, the evaluation is conducted on the BioScope corpus, reporting

performance metrics including accuracy, precision, recall, F1 score, and percentage of correct scopes (PCS). The findings highlight the potential of transformer-based learning for negation detection, achieving high accuracy for negation identification (99%) and a high PCS for negation scope recognition (95%). The paper compares their approach to previous methods such as rule-based systems, conventional machine learning, CNNs, and Bi-LSTMs, demonstrating superior performance and contributing to advancing natural language processing techniques in the biomedical domain, particularly in the context of negation detection. The below Table 3 shows Summarization of Technology and Solution Comparison.

Table 3

A Summarization of Technology and Solution Comparison

Author	Technical Approach	Models	Evaluation Metrics	Result	Conclusion
Pawar et al. (2022)	Supervised Fine-Tuning, Multimodal Integration	Clinical BERT	Accuracy, F1 Score	Improved prediction accuracy for 30-day mortality in COVID-19 patients	Demonstrates the benefits of multimodal models and fine-tuning with structured data to enhance Clinical BERT's accuracy in predicting hospital readmissions.
Saha et al. (2020)	Integration of Word2Vec and BERT, Multi-label Classification	Clinical BERT	Accuracy, Multi-label Classification Accuracy	Enhanced accuracy in understanding and classifying patient complaint characteristics	Highlights the importance of Clinical BERT in decoding complex clinical narratives for effective readmission risk prediction.

Author	Technical Approach	Models	Evaluation Metrics	Result	Conclusion
Ali et al. (2023)	Integration into Healthcare Workflows	ChatGPT	Clinical Task Performance Metrics	Optimized healthcare processes like patient discharge and USMLE preparation	Stresses the necessity of comprehensive validation and ethical integration of LLMs like ChatGPT into healthcare to improve patient care and reduce clinician documentation load.
Gupta et al. (2023)	Development Analysis, Capability Expansion	ChatGPT	Coherence, Context-Awareness, Task Handling	Marked improvements in model complexity and task handling capabilities	Indicates continuous advancements in LLMs like ChatGPT enhance their application across various industries, despite ongoing challenges in domain expertise and bias.
Pathak et al. (2023)	Low-Rank Adaptation (LoRA) Fine Tuning	Llama-2	Reward Model Performance, Human Evaluation Scores	Llama-2 shows robust performance in reward models, competitive with other leading LLMs	Llama-2's advancements suggest its significant potential in research and practical applications, needing careful ethical and impact evaluations
Thakkar et al. (2023)	Comparative Performance Analysis	Mistral-7B, Llama-2-7B	ROUGE, BLEU, CIDEr	Llama-2-7B outperforms Mistral-7B in NLP tasks, demonstrating superior discourse capabilities	Provides insights into the capabilities and applications of different LLMs in NLP, emphasizing the need for ongoing model and dataset enhancements.
Tsai, Huang, and Kuo (2024)	Automated literature review using a large language model with retrieval-augmented generation	Mistral LLM	Efficiency, scalability, quality of review	Mistral LLM outperforms humans in efficiency and scalability but sometimes lacks depth.	Promotes a hybrid approach of Mistral LLM with human expertise to enhance preliminary literature reviews.

Author	Technical Approach	Models	Evaluation Metrics	Result	Conclusion
Labrak et al. (2024)	Open-source pretrained large language model for the biomedical domain	BioMistral 7B	Performance on medical question-answering tasks	BioMistral shows superior performance compared to existing models.	Demonstrates the potential of LLMs specifically designed for medical domains; offers resources for further research.
Chai et al. (2023)	Noise reduction models for biomedical named entity recognition	XLNet-CRF with Shared Labels and Dynamic Splicing	F1-score	Achieved state-of-the-art performance on several datasets.	Effective at reducing noise and improving performance in biomedical NER tasks.
Althari et al. (2022)	Transformer-based learning for negation detection in biomedical texts	BERT, ALBERT, XLNet, ELECTRA	Accuracy, precision, recall, F1 score, PCS	High accuracy and PCS for negation detection tasks.	Superior to previous methods, advancing NLP techniques in the biomedical domain.

Note. This Table Covers Summarization Of Technology And Solution Comparison.

Literature Survey Of Existing Research

This segment presents an extensive review of hospital readmission prediction systems. Considerable time and effort were invested in exploring a variety of research papers to understand the project's intricacies. Each team member conducted meticulous independent research, enabling a thorough assessment of the predictive efficacy of different algorithms. The primary goal is to develop a robust system that can accurately predict hospital readmissions, allowing healthcare professionals to proactively intervene and prevent avoidable readmissions, thereby improving patient outcomes.

A study conducted in 2024 by Rahman et al. (2024) focused on the generalization ability of ClinicLLM, a large language model (LLM) tailored for healthcare applications. The authors highlighted the crucial role of LLMs in enhancing patient care, clinical decision-making, and workflow efficiency, while emphasizing the necessity for these models to perform effectively across diverse clinical settings and patient populations. The study evaluated ClinicLLM's effectiveness in predicting 30-day all-cause readmissions across different hospitals and patient demographics. It was observed that ClinicLLM's performance varied, particularly in hospitals with limited data and among certain patient groups, such as the elderly and those with complex medical conditions. Factors such as sample sizes, note content, patient characteristics, and healthcare system aspects were analyzed to understand this variability. Strategies like local fine-tuning, instance-based augmented fine-tuning, and cluster-based fine-tuning were explored to enhance ClinicLLM's generalization. Local fine-tuning emerged as the most effective approach, significantly improving predictive accuracy. The study provides valuable insights into deploying large language models in healthcare and improving their applicability across diverse patient populations and healthcare settings.

In 2017, researchers Baechle et al. (2017) addressed the issue of excessive readmissions in Chronic Obstructive Pulmonary Disease (COPD) cases in a paper on predicting 30-day hospital readmission. They advocated for predictive analytics to identify high-risk patients and proposed integrating cost considerations into readmission prediction models while evaluating the cost-effectiveness of existing models. The significant burden COPD imposes on healthcare costs and the legislative efforts to reduce readmissions, such as the ACA and HRRP, were emphasized. Despite the promise of predictive modeling techniques, existing models often overlook cost implications, prompting the need for improved integration. Various statistical learning algorithms

were reviewed, stressing the importance of models demonstrating substantial discriminative ability. The financial impact of readmission penalties imposed by CMS was highlighted, advocating for the incorporation of these considerations into model development to enhance relevance. Methodologies for integrating cost criteria into readmission models and advocating for dynamic cost evaluation were proposed.

In 2022, researchers (Pawar et al., 2022) investigated the creation of a multimodal model to predict 30-day mortality in COVID-19 patients upon hospital admission, leveraging structured electronic health records (EHRs) alongside unstructured clinical notes. The research highlighted the advantage of integrating diverse data modalities and updating Clinical BERT based on structured data, thereby enhancing predictive accuracy. The study also addressed the challenges associated with processing lengthy clinical narratives and proposed effective techniques to optimize Clinical BERT's performance in handling extensive textual data.

Researchers Ali et al. (2023) examined the potential and limitations of employing ChatGPT and other large language models (LLMs) in healthcare. They discussed how ChatGPT might improve the accessibility, scalability, and efficiency of healthcare services by streamlining processes like preparing clinical vignettes and patient discharge summaries and aiding in USMLE exam preparation. The study also addressed the risks associated with employing AI for medical diagnosis and patient relations, including potential biases, mistakes, and ethical issues. The authors emphasized the necessity of thorough validation and cautious integration within current healthcare frameworks to ensure ChatGPT's reliable and ethical application in medical settings.

Another study on paper, "Understanding Patient Complaint Characteristics Using Contextual Clinical BERT Embeddings," in 2020 (Saha et al., 2020), explored the utilization

of Clinical BERT to comprehend and classify patient complaint characteristics crucial for predicting readmission risks. By combining Word2Vec and BERT models, clinical text was encoded to capture subtle complaint features, followed by multi-label classification facilitated by the Linear Discriminant Analysis (LDA) algorithm. Experimental outcomes demonstrated significant improvements in accuracy compared to existing models, underscoring the importance of Clinical BERT in deciphering intricate clinical narratives and precisely predicting readmission risks.

A fresh approach to enhance readmission prediction by examining connections between features was proposed in paper titled "Hospital Readmission Prediction using Discriminative patterns" by Watson et al. (2020) . Researchers suggested utilizing discriminative pattern-based features extracted from electronic health records (EHRs) of hospitals. These patterns, occurring more frequently in one class (readmission or non-readmission) than the other, were employed as features in a classification model for predicting readmission. The study assessed the effectiveness of these pattern-based features across three datasets: diabetes, chronic kidney disease, and a general dataset encompassing all diseases. The experiments revealed that the features derived from cohorts with chronic diseases showed fewer distinctions between readmission and non-readmission classes compared to the dataset covering all diseases. While the proposed pattern-based model exhibited enhanced prediction performance for the all-disease cohort, the improvement was marginal for the diabetes and chronic kidney disease datasets.

In 2023, Gupta et al. explored the growth of the ChatGPT model from its original form to its present complicated iterations. The study highlighted the notable increase in model complexity, which has grown from 117 million parameters in the initial release to 175 billion in

the most recent one. This development has enabled the model to generate replies that are more logical and sensitive to context, making it capable of performing complex tasks like translation and summarization. The study also highlighted persistent issues such as emotional intelligence gaps, domain-specific knowledge gaps, and data biases in the training set.

Another study (Pathak et al., 2023) provided a thorough examination of Llama-2, a large language model (LLM) fine-tuned using the Low-Rank Adaptation (LoRA) technique, while comparing its performance with other models like GPT-4. The study highlighted Llama-2's potential across various research applications and its suitability for broader usage. The study stressed the importance of understanding the mechanisms of large language models, evaluating their impact, and addressing ethical considerations. Experimental results demonstrated Llama-2's robust performance in reward models, showcasing competitiveness with models like ChatGPT, with its largest Chat model outperforming others on prompt sets. Additionally, the study discussed AI finetuning and presented human evaluation results, underscoring Llama-2's significance in contemporary research and practical applications.

A comparative study (Thakkar et al., 2023) in between two LLMs, Mistral-7B and Llama-2-7B, within the realm of natural language processing (NLP) was conducted. The analysis, using performance indicators such as ROUGE, BLEU, and CIDEr, indicated Llama-2-7B's superior performance across various dimensions compared to Mistral-7B, while also acknowledging Mistral-7B's potential for improvement with larger datasets due to its recent release. The paper discussed the transformative impact of LLMs in NLP domains, including machine translation and text summarization, elucidating their architecture and autoregressive nature. Through the comparison of Mistral-7B and Llama-2-7B, valuable

insights into their characteristics and implications within the dynamic landscape of NLP were provided.

Tsai, Huang, and Kuo (2024) investigated the effectiveness of the Mistral Large Language Model (LLM) with Retrieval-Augmented Generation (RAG) in automating literature reviews compared to traditional human-led processes. Analyzing 50 scientific papers from the OpenReview platform, the study evaluated Mistral LLM's review efficiency, scalability, and quality. Results indicated that while Mistral LLM surpassed human efforts in terms of efficiency and scalability, it occasionally lacked the analytical depth and attention to detail exhibited by human reviewers. The study suggested the potential of Mistral LLM in standardizing preliminary literature reviews, proposing a hybrid approach that combines Mistral LLM's capabilities with human expertise to enhance the literature review process.

Chai et al. (2023) in a study introduced two noise reduction models, Shared Labels and Dynamic Splicing, aimed at enhancing the performance of Biomedical Named Entity Recognition (BioNER) systems. These models addressed the limitations of existing BioNER systems, which suffer from noise interference and lack an effective label dependency decoder. The proposed models utilized XLNet encoding, a permutation language pre-training model, combined with Conditional Random Field (CRF) decoding. Evaluation on 15 biomedical named entity recognition datasets demonstrated an average F1-score improvement of 1.504 and 1.48, respectively, with state-of-the-art performance achieved on seven datasets. The paper contributed by proposing novel noise reduction models, showcasing their effectiveness in BioNER, and studying the impact of model structures on various feature datasets.

An open-source large language model (LLM) specifically tailored for the biomedical domain was introduced by Labrak et al. (2024). BioMistral, developed based on Mistral and pre-trained on the PubMed Central dataset, was comprehensively evaluated on a benchmark consisting of ten medical question-answering tasks in English, including analysis of various evaluation strategies. The development of lightweight models and model merging techniques were also explored, demonstrating BioMistral's superior performance.

Affas et al. (2020), addressed the task of forecasting hospital readmissions occurring during a 30-day period. They proposed creating a prediction model using the Medical Information Mart for Intensive Care (MIMIC III) dataset, which includes extensive information on admissions for patients in the intensive care unit (ICU). The training process utilized several characteristics, including ICD9 embeddings, chart events, and demographic information, to train multiple classifiers such as Random Forest, Support Vector Machine, Logistic Regression, and Multi-Layer Perceptron. The Random Forest model stood out as the most efficient, with an accuracy of 0.65 and an Area Under the Curve (AUC) of 0.66. The research emphasized the importance of precise readmission prediction in improving patient care and reducing healthcare expenses, summarizing related research efforts and past investigations utilizing statistical methodologies and machine learning methods to forecast readmission.

Althari et al. (2022) used transformer-based learning for negation detection in biomedical texts, focusing on negation sentence identification and negation scope recognition subtasks. Pre-trained transformer models such as BERT, ALBERT, XLNet, and ELECTRA were utilized and evaluated on the BioScope corpus, reporting performance metrics including accuracy, precision, recall, F1 score, and percentage of correct scopes (PCS). The findings highlighted the potential of transformer-based learning for negation detection, achieving high

accuracy for negation identification (99%) and a high PCS for negation scope recognition (95%). The paper compared their approach to previous methods, demonstrating superior performance and contributing to advancing natural language processing techniques in the biomedical domain, particularly in the context of negation detection.

The research undertaken by Tang et al. (2023) focused on accurately forecasting hospital readmissions within a 30-day timeframe. They introduced the multimodal spatiotemporal graph neural network (MM-STGNN), which utilizes multimodal, longitudinal patient data and reflects patient similarity using a graph structure. The MM-STGNN combined several types of data collected from hospitalized patients, such as chest radiography and electronic health records, from two separate medical facilities. The MM-STGNN exhibited favorable performance in the experimental results, reaching an area under the receiver operating characteristic curve (AUROC) of 0.79 on both datasets. It surpassed the current clinical reference standard, LACE+, in terms of performance on the internal dataset. The MM-STGNN demonstrated notable enhancements compared to baseline models, especially for patients with heart disease. The authors also performed a qualitative interpretability analysis to comprehend the significant features influencing the model's predictions. The initial diagnoses of patients, even if not specifically used for training, contained vital information for making predictions. The MM-STGNN was positioned as a beneficial tool for clinical decision-making on patient discharge and identifying high-risk patients requiring intensive follow-up after release. The research introduced a novel method that accurately forecasts hospital readmissions within 30 days, outperforming current models by considering patient similarity and longitudinal data, highlighting the capacity of sophisticated methods to improve readmission

prediction accuracy and aid in healthcare decision-making. The below Table 4 shows Summarization of Literature Survey Comparison.

Table 4

A Summarization Of Literature Survey Comparison

Author	Paper	Models	Techniques	Result
Rahman et al. (2024)	Generalization in Healthcare AI: Evaluation of a Clinical Large Language Model	ClinicLLM	Local fine-tuning, Instance-based augmented fine-tuning, Cluster-based fine-tuning	Local fine-tuning was most effective; variability in performance across hospitals and demographics
Baechle et al. (2017)	A Cost Sensitive Approach to Predicting 30-Day Hospital Readmission in COPD Patients	Naive Bayes, Random Forest, SVM, k-NN, C4.5, Bagging, Boosting	Cost-Sensitive Classification, Cost Evaluation	Cost-sensitive classification significantly reduced misclassification costs, with notable improvements in Random Forest and Bagging models
Pawar et al. (2022)	Multimodal Prediction of COVID-19 Mortality with Clinical BERT	Clinical BERT	Multimodal fine-tuning	Enhanced predictive accuracy by integrating structured and unstructured EHR data.
Ali et al. (2023)	ChatGPT and Large Language Models in Healthcare: Opportunities and Risks	ChatGPT	RLHF, Case Studies, Predictive Modeling	Potential to improve healthcare efficiency; risks include biases and ethical concerns.
Saha et al. (2020)	Understanding Patient Complaint Characteristics Using Contextual Clinical BERT	Clinical BERT, Word2Vec	Multi-label classification with Linear Discriminant Analysis	Improved accuracy in predicting readmission risks by analyzing patient complaint characteristics.

Author	Paper	Models	Techniques	Result
	Embeddings			
Watson et al. (2020)	Hospital Readmission Prediction using Discriminative patterns	Discriminative Pattern Mining	Data Mining, Classification, Machine Learning Algorithms	Demonstrated the effectiveness of discriminative pattern-based features in predicting hospital readmissions, improving AUC by about 12% for the all-disease cohort compared to baseline models.
Gupta et al.(2023)	ChatGPT: Exploring the Capabilities and Limitations of a Large Language Model for Conversational AI	ChatGPT	Natural Language Processing, Few-shot Learning, Code Generation	Increased complexity and context sensitivity; issues with emotional intelligence and data biases.
Pathak et al. (2023)	Performance Analysis of LoRA Finetuning Llama-2	Llama-2, GPT-4	Low-Rank Adaptation (LoRA)	Llama-2 showed robust performance and competitiveness with ChatGPT; importance of fine-tuning discussed.
Thakkar et al. (2023)	Comprehensive Examination of Instruction-Based Language Models: A Comparative Analysis of Mistral-7B and Llama-2-7B	Mistral-7B, Llama-2-7B	Performance evaluation metrics (ROUGE, BLEU, CIDEr)	Llama-2-7B outperformed Mistral-7B; highlighted the importance of dataset size and language model development.
Tsai, Huang, and Kuo (2024)	Comparative Analysis of Automatic Literature Review Using Mistral Large Language Model and Human Reviewers	Mistral LLM with Retrieval-Augmented Generation	Retrieval-Augmented Generation (RAG)	Mistral LLM excels in efficiency and scalability, yet lacks depth compared to human reviewers; advocates hybrid approach.

Author	Paper	Models	Techniques	Result
Chai et al. (2023)	Noise Reduction Learning Based on XLNet-CRF for Biomedical Named Entity Recognition	XLNet-CRF	Shared Labels, Dynamic Splicing	Improved BioNER performance with F1-score improvements on multiple datasets; superior performance on 7 datasets.
Labrak et al. (2024)	BioMistral: A Collection of Open-Source Pretrained Large Language Models for Medical Domains	BioMistral	Pre-training, Lightweight models, Model merging	Demonstrated superior performance of BioMistral on medical question-answering tasks; innovation in LLM design for biomedical use.
Affas et al. (2020)	Predicting 30-day Hospital Readmissions Using MIMIC Data	Random Forest, SVM, Logistic Regression, MLP	ICD9 embeddings, chart events, demographic info	Random Forest most effective, highlighting the importance of predictive models in reducing healthcare costs and improving care.
Althari et al. (2022)	Transformer-based Learning for Negation Detection in Biomedical Texts	BERT, ALBERT, XLNet, ELECTRA	Transformer-based learning	Achieved high accuracy and PCS in negation detection, surpassing previous models including CNNs and Bi-LSTMs.
Tang et al. (2023)	Predicting 30-Day All-Cause Hospital Readmission Using Multimodal Spatiotemporal Graph Neural Networks	MM-STGNN	Multimodal Spatio Temporal Graph Neural Networks	MM-STGNN outperforms existing models with an AUROC of 0.79, demonstrating significant improvements for patients with heart disease.

Note. This Table Covers Summarization of Literature Survey Comparison.

Data Management Plan

Data Collection Approaches

Data were obtained from several sources. One of the diabetic datasets used in the work was obtained from the UC Irvine Machine Learning Repository. US Healthcare Readmissions and Mortality is another dataset from kaggle (Devastator, 2023). The Diabetic dataset was primarily concerned with collecting inpatient data from a national healthcare database known as the Health Facts database offered by Cerner Corporation. This collection contains substantial clinical records from hospitals in the United States that use the Cerner Electronic Health Record System. Participation in Health Facts is voluntary, and organizations can provide deidentified data in accordance with the Health Insurance Portability and Accountability Act of 1996, which protects patient privacy.

This dataset covers a decade of clinical care, from 1999 to 2008, and includes 130 hospitals and integrated delivery networks from several regions: 18 in the Midwest, 58 in the Northeast, 28 in the South, and 16 in the West. The bulk of these hospitals have bed capacities ranging from 100 to 499, with some having less than 100 and others exceeding 500 beds.

The database is organized into 41 tables using a fact-dimension schema, with a total of 117 different features. It contains data from 74,036,643 unique patient experiences, corresponding to 17,880,231 distinct patients and 2,889,571 providers. This dataset not only captures inpatient contacts but also includes outpatient data, including emergency department visits, for the same patient cohort, enabling a full perspective of both.

Data extraction and preparation were performed in two steps. Initially, relevant contacts were discovered and retrieved based on 55 essential criteria. This updated dataset has been made

publically available for research purposes and is currently being submitted to the UCI Machine Learning Repository for wider availability.

The major goal of using this information is to examine inpatient care for diabetic patients, including laboratory tests, drug regimes, and hospital stays of up to 14 days. The goal is to discover factors of early readmission within 30 days after release. This analysis is critical for optimizing diabetes treatment in hospital settings, aiming to reduce readmission rates, lower healthcare costs, and enhance patient outcomes by providing optimal diabetes care and hyperglycemia control during hospital stays.

The US Healthcare Readmissions and Mortality dataset from kaggle(The Devastator, 2023) is a collection of 30-day death and readmission measures collected on Kaggle, with the main source being the Health Care Cost and Utilization Project (HCUP), which is managed by the Health Care Research and Quality (AHRQ). The HCUP's Nationwide Readmissions Database contains inpatient data from US hospitals and includes a variety of metrics of health report.

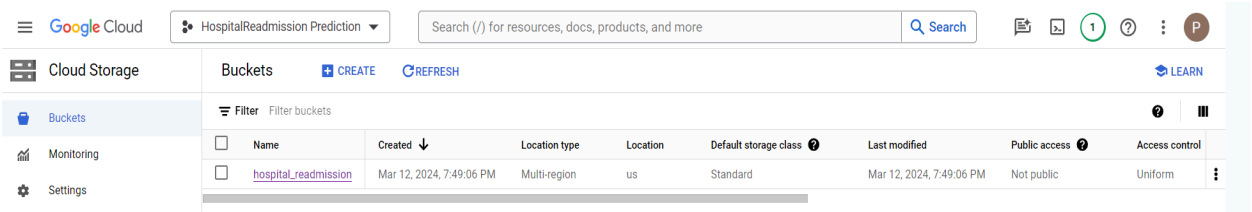
Data Management And Storage

Data management is a crucial aspect of dealing with data since data quality has a direct impact on how effectively a model functions. It is required to apply data transformation techniques, such as data cleansing, to resolve issues such as missing values or incorrect readings caused by malfunctioning tools. The data is preprocessed with Python tools before being gathered. Following that, an exploratory data analysis is performed to examine the data and derive fresh conclusions. The dataset is then separated into training and testing sets. Both datasets are subject to feature engineering. The data will be stored and managed using Google Cloud Platform (GCP). All of the data files will be transferred to a storage container that will be

set up in a specific location in Los Angeles. Which storage type is best for the bucket will be determined by factors such as frequency of access and speed of data recovery. There are two approaches to configure Identity and Access Management (IAM) functionalities. You can utilize the first method by assigning certain users special roles with specific permissions. The second option is to apply IAM policies at the bucket level. These policies grant team members limited access to specific buckets. "The Storage Legacy Bucket Owner" is a Google Cloud feature that grants users full access to and control over a certain bucket. This comprises all datasets and associated files that have been uploaded to the chosen bucket as shown in Figure 1. In addition, a copy of the bucket will be created in the Europe-North1 region to ensure that data can be retrieved in the event the system fails. The data is stored in the container in the format in which it was initially recorded, so its structure and content remain the same as when it was first extracted from the source.

Figure 1

Google Cloud Platform (GCP) Bucket

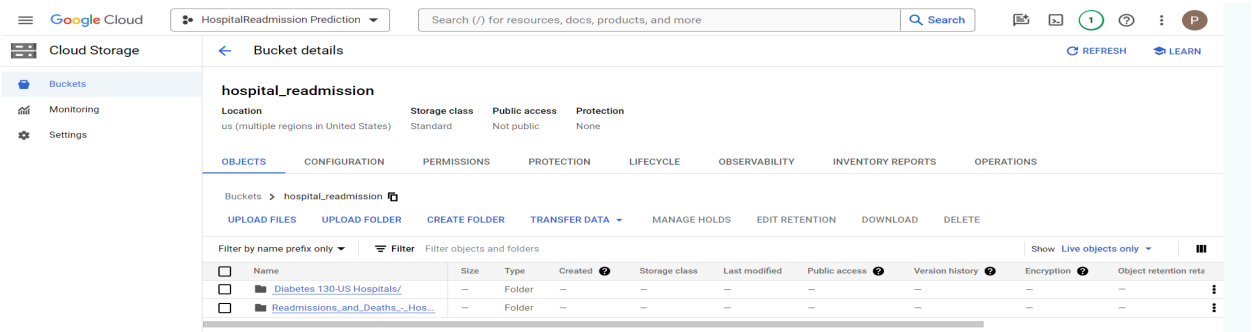


Note. Figure Giving An Overview Of Bucket Hospital Readmission In Cloud Storage.

In the hospital readmissions project, a root directory, known as the "bucket," was created. Additional folders were created within this bucket to accommodate various datasets as shown in Figure 2. The framework's first directory is named Diabetes 130-US Hospitals dataset, and the second directory is called Readmissions_and_Deaths_-_Hospital.The databases consist of 130 hospital inpatient health records.

Figure 2

Google Cloud Platform (GCP) Root Directory

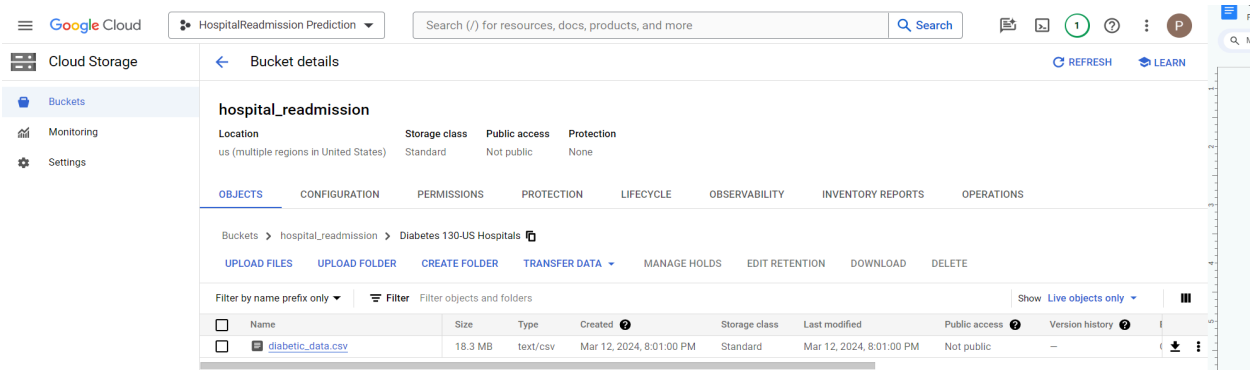


Note. Figure Giving A Overview Of Root Directory Of Hospital Readmission.

The file in the Diabetes 130-US Hospitals and Readmissions_and_Deaths_-_Hospital has patient records in .csv files as shown in Figure 3 and Figure 4.

Figure 3

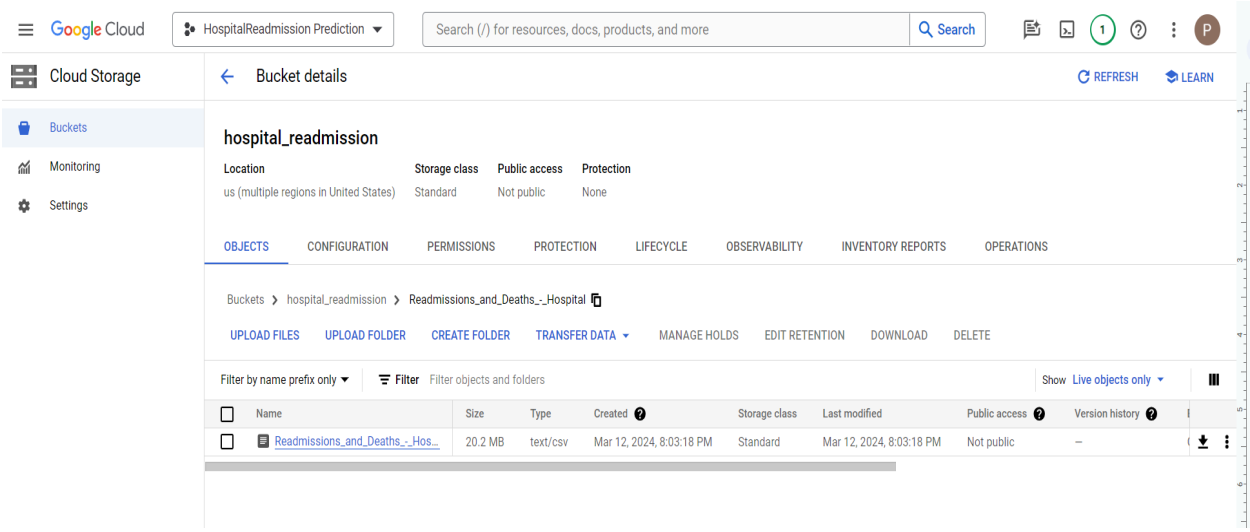
Google Cloud Platform (GCP) Diabetic_Data File



Note. Giving A Overview Of Diabetic Data File Of Hospital Readmission.

Figure 4

Google Cloud Platform (GCP) Readmissions_and_Deaths_-_Hospital File



Note. Giving A Overview Of Diabetic Data File Of Hospital Readmission.

Table 5 displays the dataset files in the bucket hospital readmission, including their names and purposes.

Table 5

Folder, Naming Convention, And Purpose

Folder Name	Purpose
Diabetes 130-US Hospitals	Stores .CSV Files
Readmissions_and_Deaths_-_Hospital	Stores .CSV Files

Note. Overview Of Folder Names And Purpose

Table 6 presents a comprehensive overview of the files included within the folders of the designated bucket, including pertinent details such as their respective naming conventions and file formats.

Table 6

Data Files

File	Naming Convention	Folder Name	File Format
Diabetes 130-US Hospitals	METR_LA_<source>_<v_num>	Diabetes 130-US Hospitals	CSV
Readmissions_and_Deaths_-_Hospital	METR_LA_<source>_<v_num>	Readmissions_and_Deaths_-_Hospital	CSV

Note. Overview Of Data Files,Naming Convention And Format.

Data Usage Mechanism

The project will involve developing five different large language models: Llama 2, Clinical XLNet, Mistral, Clinical BERT, and ChatGPT. The data collected will be analyzed using these algorithms, which will also generate accurate forecasts targeted at hospital readmission. To guarantee that other readers and users have a complete knowledge of the data and the procedures used to obtain it, documentation must be created. This guide will be written and stored as a ReadMe file in a GitHub repository. The documentation will include all pertinent information on the methodology used to gather the data from several sources, as well as the precise format and organization of the data.Along with the code implementation, it will also describe the planned uses of the data and include details on the libraries used for data preprocessing.

The documentation will also provide pertinent reference links on this particular topic. Future project updates will define the archiving and retention policies that apply to the uploaded files and buckets on the Google Cloud Platform. To encourage more research and exploration into this topic, the GitHub source will be open to the public as shown in Figure 5. Each team

member receives an equal amount of labor when implementing a data management plan (DMP). The allocation strategy for every member of the team is shown in Table 7.

Table 7

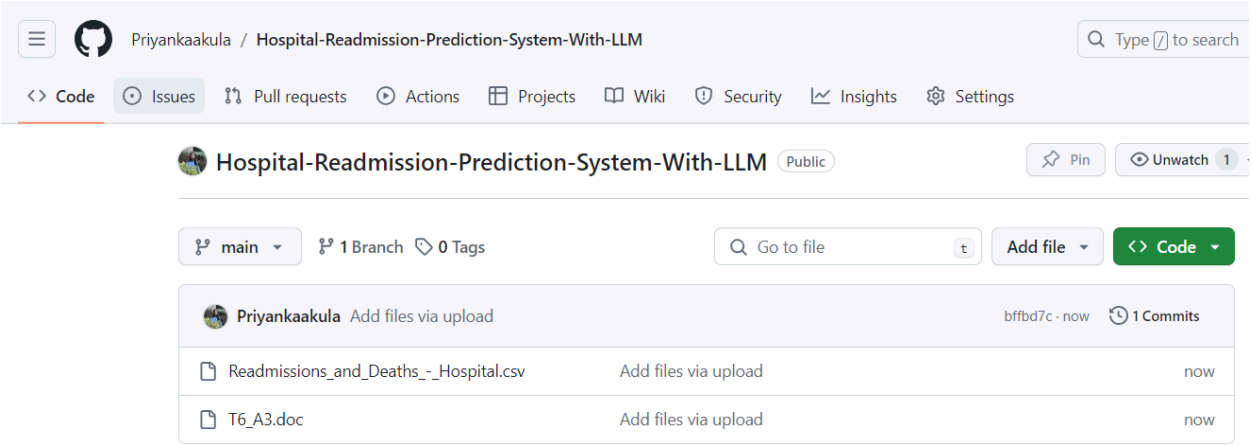
Comprehensive Responsibility And Resource Allocation Plan

DMP Phases	Resource	Responsibility
Data Acquisition	Priyanka	Use the Diabetes 130-US Hospitals and Readmissions_and_Deaths_-_Hospital datasets to prepare data for analysis.
Documentation	Neha	Document the whole data collection and analysis process.
EDA	Jaya Lakshmi	Conduct EDA on both datasets and detail the efforts.
Ethical and Legal adherence	Somna and Naveen	Ensure that any materials utilized are licensed and that the team adheres to the agreement's terms and conditions.
Data Sharing	Neha	Checkouts with data and codes are performed on a regular basis.
Data Retention and Archival	Naveen and Somna	Decision-making is required in order to determine which data to maintain and safeguard, as well as manage the costs and accessibility involved with preserving and archiving.

Note. This Table Shows The Detailed Information How Tasks are Divided Within The Team.

Figure 5

Github Repository



Note. The Figure Gives Overview OF The Repository

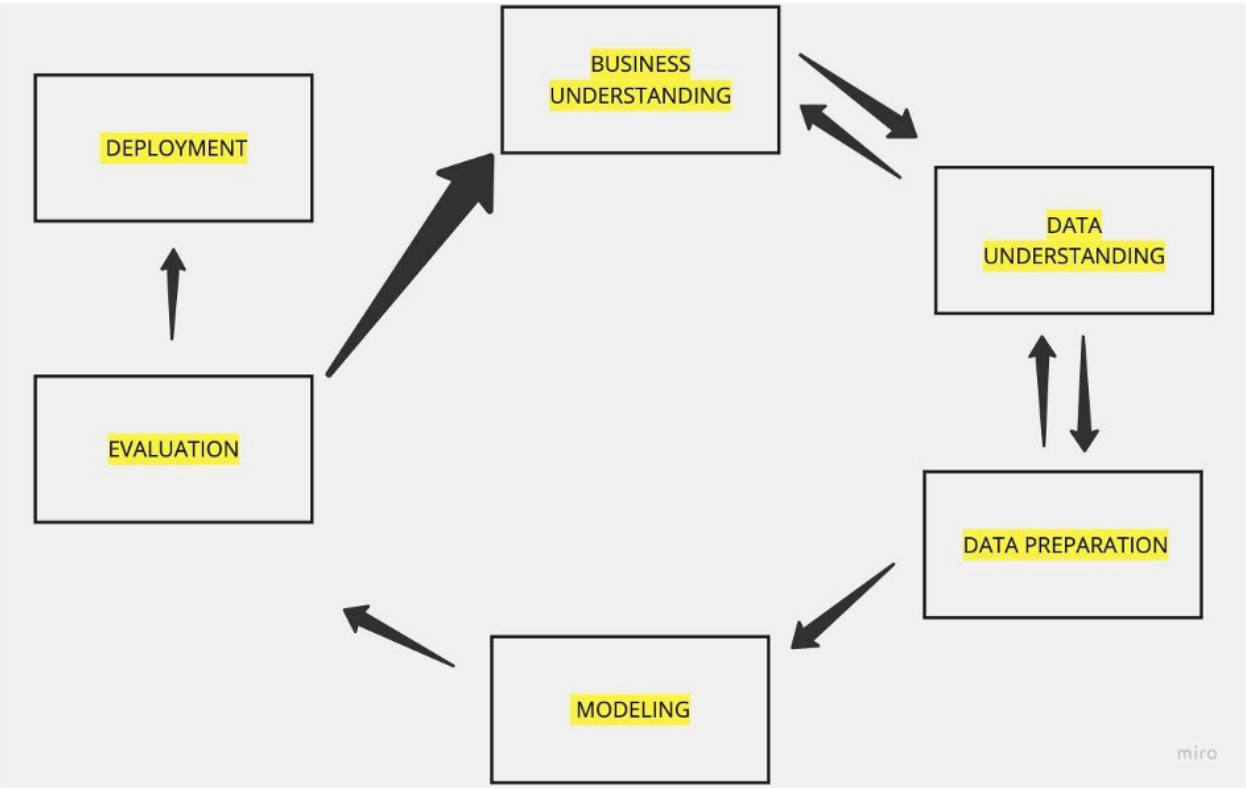
Project Development Methodology

Intelligent System Development Life Cycle

The CRISP-DM methodology is being implemented in this project's development. The acronym CRISP-DM represents the Cross-Industry Standard Process for Data Mining. This approach offers a structured and adaptable procedure for carrying out the undertaking. Business comprehension, data comprehension, data preparation, modeling, evaluation, and deployment are its six phases as shown in Figure 6.

Figure 6

CRISP-DM Methodology



Note. The Figure Shows Crisp-Dm Methodology.

Business Understanding. In the Business Understanding phase of the Hospital Readmission Prediction System With LLM project, following the CRISP-DM methodology, the primary objective is to mitigate hospital readmissions, especially among diabetic patients, by forecasting the likelihood of readmission within 30 days post-discharge. This endeavor not only aims to enhance patient care but also seeks to alleviate healthcare costs and elevate hospital efficiency. Leveraging data from the UCI Machine Learning Repository on Diabetes in 130-US hospitals from 1999-2008 and a Kaggle dataset on mortality and readmission rates, the plan is to employ advanced Large Language Models (LLMs) like ClinicalBERT, ChatGPT, LLAMA2, Mistral AI, and Clinical XLNet. These models will be instrumental in developing a predictive

model by analyzing patient data, treatment records, and other variables to pinpoint patterns and risk factors linked to readmissions. The project plan encompasses data preparation, model development, implementation, evaluation, and addressing potential risks like model bias and privacy concerns. By identifying patients at risk of readmission, the project aspires to forge a path toward improved healthcare outcomes and operational efficiency in hospitals.

Data Understanding. During the Data Understanding phase of the "Hospital Readmission Prediction System With LLM" project, a thorough examination and understanding of the two crucial datasets obtained from the UCI Machine Learning Repository and Kaggle are conducted. The files include extensive information on the care of diabetes in 130 US hospitals from 1999 to 2008, encompassing data from more than 100,000 hospital admissions & mortality rate and hospital readmission as shown in Figure 7, Figure 8 & Figure 9. Essential data components include patient characteristics, admission and discharge details, clinical test outcomes, diagnoses, medications, and readmission results. In this step, the datasets are analyzed comprehensively to gain a thorough understanding of the data's structure, quality, and distribution. This involves identifying any missing values, outliers, and potential biases that may impact the predictive modeling process. Moreover, it necessitates evaluating the significance of different attributes in forecasting hospital readmissions, including the duration of hospitalization, count of visits by inpatients, and diagnoses. The thorough analysis of this data provides the necessary foundation for the following data preparation, enabling the creation of a strong predictive model with the goal of decreasing hospital readmissions by promptly and accurately identifying patients at high risk.

Figure 7

Sample Diabetic Dataset

```
First few rows of the dataset:
   encounter_id  patient_nbr             race  gender      age weight  \
0       2278392      8222157        Caucasian  Female   [0-10)      ?
1        149190     55629189        Caucasian  Female  [10-20)      ?
2         64410     86047875  AfricanAmerican  Female  [20-30)      ?
3        500364     82442376        Caucasian    Male  [30-40)      ?
4         16680     42519267        Caucasian    Male  [40-50)      ?

   admission_type_id  discharge_disposition_id  admission_source_id  \
0                  6                        25                    1
1                  1                         1                    7
2                  1                         1                    7
3                  1                         1                    7
4                  1                         1                    7

   time_in_hospital  ... citoglipton insulin  glyburide-metformin  \
0                 1  ...          No      No                   No
1                 3  ...          No      Up                   No
2                 2  ...          No      No                   No
3                 2  ...          No      Up                   No
4                 1  ...          No  Steady                   No

   glipizide-metformin  glimepiride-pioglitazone  metformin-rosiglitazone  \
0                   No                        No                       No
1                   No                        No                       No
2                   No                        No                       No
3                   No                        No                       No
4                   No                        No                       No

   metformin-pioglitazone  change diabetesMed readmitted
0                      No      No          No         NO
1                      No      Ch         Yes        >30
2                      No      No         Yes         NO
3                      No      Ch         Yes         NO
4                      No      Ch         Yes         NO

[5 rows x 50 columns]
```

Note. The Figure Shows Sample Diabetic Dataset

Figure 8

Information About Diabetic Dataset

```
RangeIndex: 101766 entries, 0 to 101765
Data columns (total 50 columns):
 #   Column                    Non-Null Count   Dtype
---  ------                    --------------   -----
 0   encounter_id              101766 non-null  int64
 1   patient_nbr               101766 non-null  int64
 2   race                      101766 non-null  object
 3   gender                    101766 non-null  object
 4   age                       101766 non-null  object
 5   weight                    101766 non-null  object
 6   admission_type_id         101766 non-null  int64
 7   discharge_disposition_id  101766 non-null  int64
 8   admission_source_id       101766 non-null  int64
 9   time_in_hospital          101766 non-null  int64
 10  payer_code                101766 non-null  object
 11  medical_specialty         101766 non-null  object
 12  num_lab_procedures        101766 non-null  int64
 13  num_procedures            101766 non-null  int64
 14  num_medications           101766 non-null  int64
 15  number_outpatient         101766 non-null  int64
 16  number_emergency          101766 non-null  int64
 17  number_inpatient          101766 non-null  int64
 18  diag_1                    101766 non-null  object
 19  diag_2                    101766 non-null  object
 20  diag_3                    101766 non-null  object
 21  number_diagnoses          101766 non-null  int64
 22  max_glu_serum             101766 non-null  object
 23  A1Cresult                 101766 non-null  object
 24  metformin                 101766 non-null  object
 25  repaglinide               101766 non-null  object
 26  nateglinide               101766 non-null  object
 27  chlorpropamide            101766 non-null  object
 28  glimepiride               101766 non-null  object
 29  acetohexamide             101766 non-null  object
 30  glipizide                 101766 non-null  object
 31  glyburide                 101766 non-null  object
 32  tolbutamide               101766 non-null  object
 33  pioglitazone              101766 non-null  object
 34  rosiglitazone             101766 non-null  object
 35  acarbose                  101766 non-null  object
 36  miglitol                  101766 non-null  object
 37  troglitazone              101766 non-null  object
 38  tolazamide                101766 non-null  object
 39  examide                   101766 non-null  object
 40  citoglipton               101766 non-null  object
 41  insulin                   101766 non-null  object
 42  glyburide-metformin       101766 non-null  object
 43  glipizide-metformin       101766 non-null  object
 44  glimepiride-pioglitazone  101766 non-null  object
 45  metformin-rosiglitazone   101766 non-null  object
 46  metformin-pioglitazone    101766 non-null  object
 47  change                    101766 non-null  object
 48  diabetesMed               101766 non-null  object
 49  readmitted                101766 non-null  object
dtypes: int64(13), object(37)
memory usage: 38.8+ MB
```

Note. The Figure Shows Information about Diabetic Dataset.

Figure 9

Sample Us Healthcare Readmissions And Mortality Dataset

```
First few rows of the dataset:
   index  Provider ID             Hospital Name                   Address  \
0      0       230100  TAWAS ST JOSEPH HOSPITAL                200 HEMLOCK
1      1       230121       MEMORIAL HEALTHCARE      826 WEST KING STREET
2      2       230118      HURON MEDICAL CENTER  1100 SOUTH VAN DYKE ROAD
3      3       230121       MEMORIAL HEALTHCARE      826 WEST KING STREET
4      4       230133  OTSEGO MEMORIAL HOSPITAL          825 N CENTER AVE

         City State  ZIP Code County Name  Phone Number  \
0  TAWAS CITY    MI     48764       IOSCO    9893629301
1      OWOSSO    MI     48867  SHIAWASSEE    9897235211
2     BAD AXE    MI     48413       HURON    9892699521
3      OWOSSO    MI     48867  SHIAWASSEE    9897235211
4     GAYLORD    MI     49735      OTSEGO    9897312100

                                       Measure Name          Measure ID  \
0  Rate of readmission after discharge from hospi...  READM_30_HOSP_WIDE
1     Rate of readmission after hip/knee replacement   READM_30_HIP_KNEE
2            Pneumonia (PN) 30-Day Readmission Rate         READM_30_PN
3           Rate of readmission for stroke patients        READM_30_STK
4          Heart failure (HF) 30-Day Mortality Rate          MORT_30_HF

                  Compared to National Denominator Score Lower Estimate  \
0  No Different than the National Rate         438  13.9           12.6
1  No Different than the National Rate         150   4.0            2.8
2  No Different than the National Rate         205  16.7           13.8
3  No Different than the National Rate          67  11.5            8.7
4  No Different than the National Rate         102  13.4            9.9

  Higher Estimate Footnote Measure Start Date Measure End Date  \
0            15.6      NaN         07/01/2014       06/30/2015
1             5.7      NaN         07/01/2012       06/30/2015
2            19.9      NaN         07/01/2012       06/30/2015
3            14.9      NaN         07/01/2012       06/30/2015
4            17.7      NaN         07/01/2012       06/30/2015

                                          Location
0  200 HEMLOCK\nTAWAS CITY, MI 48764\n(44.274911,...
1  826 WEST KING STREET\nOWOSSO, MI 48867\n(43.00...
2  1100 SOUTH VAN DYKE ROAD\nBAD AXE, MI 48413\n(...
3  826 WEST KING STREET\nOWOSSO, MI 48867\n(43.00...
4  825 N CENTER AVE\nGAYLORD, MI 49735\n(45.03537...
```

Note. The Figure shows Sample US Healthcare Readmissions and Mortality Dataset.

Data preparation. The data preparation for the Hospital Readmission Prediction System project entails numerous crucial processes to optimize the data for analysis using advanced Large Language Models (LLMs) such as ClinicalBERT, CharGPT, LLAMA2, Mistral AI, and Clinical XLnet. Firstly, the dataset is processed to eliminate any irrelevant or redundant data, with a specific focus on the characteristics that are crucial for predicting hospital readmissions among diabetic patients. This involves standardizing the structure of patient demographic information, admission and discharge information, diagnostic codes, and laboratory test findings in order to establish a consistent and standardized dataset. Missing values are handled using imputation

techniques or by eliminating records as needed, to maintain the integrity of the dataset for training purposes. Feature engineering is utilized to extract and create pertinent characteristics from the unprocessed data, such as computing the duration between hospitalizations or converting diagnostic codes into clinically significant categories.

Categorical variables are suitably encoded to optimize their utilization in LLMs, whereas continuous variables are standardized to improve model performance. Textual data, like as clinical notes, undergo preprocessing techniques such tokenization, lemmatization, and removal of stopwords to prepare it for analysis by the LLMs. In addition, the dataset is divided into training, validation, and test sets, creating a strong framework for training and evaluating the model. Emphasis is placed on ensuring a balanced distribution of classes in the dataset to minimize any potential bias towards particular results. The careful and thorough process of preparing data is crucial in order to effectively utilize the improved capabilities of LLMs in properly forecasting hospital readmissions. This stage establishes a strong basis for the following phases of the CRISP-DM approach.

Modeling. During this phase, the project advances by choosing suitable modeling approaches, devising test procedures, creating prediction models, and assessing the performance of these models. The Hospital Readmission Prediction System utilizes sophisticated Large Language Models (LLMs) like ClinicalBERT, ChatGPT, LLAMA2, Mistral AI, and Clinical XLnet. These models are selected based on their expertise in comprehending and analyzing clinical narratives, as well as structured data from patient records. The dataset, which was created in the previous phase, is partitioned into training, validation, and testing subsets using an 80:10:10 split. This split ensures that there is enough data for training and allows for an unbiased evaluation.

Every chosen LLM receives training using the training set, during which hyperparameters are adjusted to correspond with the project's goal of reliably predicting hospital readmissions. The training method entails acquiring knowledge not only from organized data but also understanding the textual information contained in clinical notes, necessitating advanced natural language processing capabilities. After the training process, the models are assessed using the validation set to address any problems related to overfitting and to improve the model's capacity to apply new data.

The test set is used rigorously to evaluate the performance of each model at the final evaluation. Essential performance indicators, including accuracy, precision, recall, F1-score, and area under the ROC curve (AUC-ROC), are computed to assess the predictive capability of each model in forecasting hospital readmissions. The primary objective of this extensive modeling phase is to determine the LLM (Machine Learning Model) that exhibits the highest level of prediction accuracy. This is a crucial step towards minimizing avoidable hospital readmissions in patients diagnosed with diabetes.

Evaluation. Following the development and initial assessment of the models in the Hospital Readmission Prediction System project, the evaluation phase is critical for a detailed analysis of the outcomes. This phase scrutinizes the performance of the Large Language Models (LLMs) like ClinicalBERT, ChatGPT, LLAMA2, Mistral AI, and Clinical XLnet, focusing on their ability to predict hospital readmissions among diabetic patients accurately. The effectiveness of these models is rigorously evaluated using key metrics such as accuracy, precision, recall, Perplexity, BLEU Score, ROUGE Score and domain-specific metric and the F1 score, providing a comprehensive understanding of each model's predictive power and reliability. This evaluation ensures that the models align with the overarching project objectives,

particularly in terms of minimizing hospital readmissions and improving patient outcomes. By comparing the models against these metrics, insights and patterns are extracted, not only to gauge current model performance but also to identify areas for refinement and improvement. This evaluation phase is integral to the project, ensuring that the final model chosen not only meets but exceeds business requirements, thereby making a meaningful impact on healthcare practices and patient care.

Deployment. The last phase of the CRISP-DM methodology entails the model's implementation and its ongoing maintenance. The deployment setting is determined, focusing on seamless integration within the specified timeframe and milestones. A fitting Google Cloud Platform (GCP) environment is chosen for deployment, with a strategy formulated for effective integration. To track the deployed models' efficacy, Key Performance Indicators (KPIs) are utilized. Following deployment, a comprehensive report detailing the project's objectives and phases is compiled, marking the culmination of the project cycle.

Planned Project Development Processes and Activities

This project uses JIRA Software for project management and follows the Scrum methodology. Epics, Stories, Tasks, and Sub-Tasks have all been organized in accordance with the CRISP-DM architecture to expedite the project's development. Every stage of the project has a designated sprint. Six sprints total, five of which will last two weeks apiece and one final sprint that will last one week are planned for the project. Every phase in the CRISP-DM process is classified as an Epic. Sub-tasks are subsequently developed under the main tasks, which are then articulated as either Stories or Tasks. At every project step, these planned measures are presently being carried out.

Business Understanding. Understanding the business requirements necessary for assessing and forecasting readmissions in the healthcare industry is the main emphasis of the hospital readmission prediction project's early phase. In order to better understand this important healthcare issue, it is necessary to identify the main variables causing readmissions. It's critical to assess the state-of-the-art healthcare technology and solutions, especially those that pertain to readmission prediction. It is imperative to define the technical and functional needs for the study and to establish any prerequisites, such as locating pertinent datasets from healthcare databases. In addition, it is crucial to examine current predictive models, algorithms, and academic research about hospital readmissions in order to guide the project's approach and course of action.

Data Understanding. During the hospital readmission prediction project's data understanding phase, gathering and analyzing data is the main goal in order to have a thorough grasp of the dataset that is derived from medical records and databases. Python libraries are used to make it easier to access and extract data, which allows for a more in-depth analysis of the dataset. After a thorough examination of the dataset's functionality, information that is judged relevant for readmission analysis is extracted and processed further. The missing data points in the dataset are located, and methods for addressing them—such as feature engineering to generate new features for more thorough analysis—are developed. In addition, an outlier analysis is carried out to guarantee the accuracy and consistency of the data, providing the foundation for next phases of predictive modeling.

Data Preparation. In order to prepare the dataset for further modeling and analysis, the hospital readmission prediction project places a strong emphasis on cleaning, converting, and formatting the data. To maintain data integrity, methods including outlier detection and addressing missing values through imputation or removal are used. Python libraries are essential

for carrying out formatting and data transformation operations. To find underlying patterns and connections in the dataset and provide important insights for the predictive modeling process, extensive data exploration is also conducted.

Modeling. Our hospital readmission prediction study made use of a set of sophisticated language models created especially for the examination of medical data. ClinicalBERT and ClinicalXLNet are excellent at deciphering the complex subtleties of clinical narratives and extracting important data that is crucial to accurately forecast readmissions. By offering insights into patient experiences and facilitating individualized therapies, ChatGPT improves our comprehension of patient interactions and attitudes. To improve predictive accuracy, Mistral AI uses state-of-the-art AI approaches to recognize intricate patterns and risk variables linked to readmissions. Sophisticated data analysis tools are provided by LLAMA2, which is specifically designed for healthcare applications. These tools enable the extraction of actionable insights and support the building of strong predictive models for hospital readmission prediction.

Evaluation. Appropriate evaluation metrics are allocated in the hospital readmission prediction study evaluation phase, taking into account the particulars of the predictive task. Metrics like recall, accuracy, precision, and F1-score are used to evaluate each model's performance in detail. The models' accuracy in identifying patients who are at risk of readmission is ascertained by a thorough review and comparison with these measures. The model with the best predictive performance will be chosen to anticipate hospital readmissions in the future, with an emphasis on producing precise forecasts to enable proactive interventions and enhance patient outcomes.

Deployment. The chosen models are incorporated into the RAG (Readmission Analysis and Guidance) program during the deployment stage of our Hospital Readmission Prediction

project. Customized libraries and frameworks, like Hugging Face Transformers for ClinicalBERT, ClinicalXLNet, ChatGPT, Mistral AI, and LLAMA2, are used to guarantee smooth application interaction and functionality. In the healthcare field, the forecasts produced by the RAG program are extremely useful for informing clinical judgment calls and allocating resources in a way that efficiently reduces readmissions. Additionally, interactive dashboards and visual representations are made using data visualization technologies like Tableau, which help medical professionals comprehend readmission trends and patterns more naturally. By providing healthcare professionals with practical information, this all-encompassing strategy seeks to improve patient care and effectively utilize hospital resources.

Project Organization Plan

The Work Breakdown Structure (WBS) divides complex tasks into smaller components. It guides the group through all phases of data mining, including business understanding and implementation.The work breakdown structure (WBS) helps identify potential project difficulties. In this project, the CRISP-DM methodology is used. Identifying critical CRISP-DM phases is the initial step.WBS approach involves understanding the company, preparing data, modeling, assessing, and deploying the results. To assure project completion, break subphases into smaller tasks assigned to team members.

The Work Breakdown Structure (WBS) divides complex tasks into smaller components, guiding the group through all phases, including business understanding and implementation. According to a paper on the effectiveness of work breakdown structures in data mining projects by Feng et al. (2009), the work breakdown structure (WBS) is pivotal in identifying potential project difficulties. In this project, the CRISP-DM methodology is employed, with the identification of critical CRISP-DM phases as the initial step. The WBS approach involves

comprehending the company, preparing data, modeling, evaluating, and deploying the results. By breaking down subphases into smaller tasks assigned to team members, the project's successful completion is assured, as highlighted in the context of utilizing WBS for effective project management.

The Work Breakdown Structure (WBS) begins with Business Understanding as shown in Figure 10, which defines the project's scope. This involves defining the project aim and problem statement. After reviewing research papers, four machine learning models were chosen for the project. During this step, information regarding data sets, project management methods, and algorithms was obtained through a literature review.

In the Data Interpretation stage, meticulous analysis was conducted on datasets from the UCI Machine Learning Repository and Kaggle, focusing on diabetes patient treatments and outcomes in 130 US hospitals from 1999-2008. These datasets, rich in patient demographics, hospital admissions, clinical tests, diagnoses, medications, and readmission rates, are foundational. Thorough data exploration was undertaken using Python to clean and ensure the quality of the data, addressing null and missing values, and outliers. This stage was crucial for preparing the ground for employing advanced Large Language Models, such as ClinicalBERT, ChatGPT, LLAMA2, Mistral AI, and Clinical XLnet, aiming to enhance accuracy in reducing hospital readmissions. The meticulous data interpretation underpins model development, ensuring predictive analytics are grounded in reliable, relevant data

During the Data Preparation phase, the focus was on refining and optimizing the datasets sourced from the UCI Machine Learning Repository and Kaggle, detailing diabetes patient care and outcomes across 130 US hospitals. This crucial step involved advanced data wrangling techniques to clean and preprocess the data, addressing missing values and ensuring data

integrity. Significant features relevant to hospital readmissions were extracted, employing Python libraries for data manipulation, preparing the datasets for the next stage of modeling. This preparation is foundational for leveraging Large Language Models to accurately predict hospital readmissions and contribute to improved patient care and healthcare resource management.

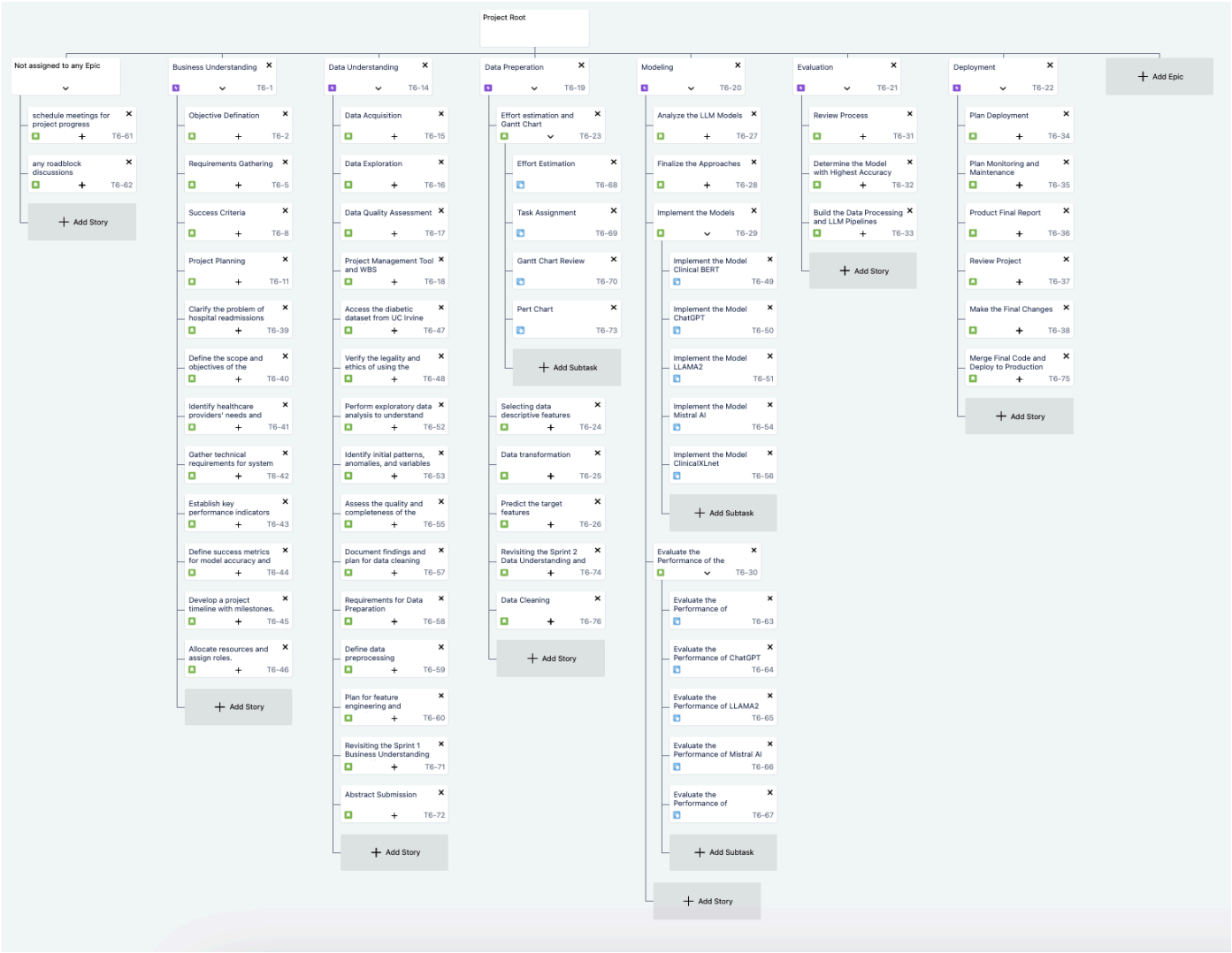
In the Data Modeling phase, predictive models will be built and analyzed using Large Language Models (LLMs) such as ClinicalBERT, ChatGPT, LLAMA2, Mistral AI, and Clinical XLnet, in addition to traditional machine learning techniques. This strategy intends to take advantage of the advanced models' comprehensive knowledge of clinical narratives and organized healthcare data. The goal is to compare these models based on their ability to forecast hospital readmissions accurately, using rich datasets. This step entails rigorous training, testing, and validation procedures to determine which models best represent patient data and readmission risk factors.

In the Evaluation phase that follows, the principal aim is to methodically evaluate the performance of these models by employing metrics including accuracy, precision, recall, Perplexity, BLEU Score, ROUGE Score and domain-specific metric and the F1 score. This entails utilizing reliable statistical tools and methods to assess the effectiveness of each model in forecasting readmissions, while placing significant emphasis on reducing the occurrence of false positives and negatives. By conducting an exhaustive examination of model outputs and error rates, the objective is to enhance and optimize the models in order to guarantee their clinical applicability and precision. By virtue of its superior performance as measured by these evaluation metrics, the model will be chosen for implementation in an actual healthcare environment, where its effects on the reduction of readmissions will be meticulously observed and appraised. By employing machine learning and iterative modeling and evaluation in

accordance with the CRISP-DM methodology, there is a guarantee that the efforts methodically progresses in its pursuit of enhancing patient outcomes and diminishing healthcare expenditures via the predictive capabilities of LLMs.

Figure 10

Work Breakdown Structure (WBS)



Note. The Figure Shows Work Breakdown Structure presenting the hierarchical and incremental decomposition of the project into phases, deliverables and work packages.

Project Resource Requirements and Plan

This section pertains to the resources utilized for the implementation of a project and the manner in which they will be allocated. Centralizing the budget is a crucial aspect of project management to ensure timely completion of the project.

Hardware Requirements

The hardware requirements section details the technological resources utilized in the Hospital Readmission Prediction System With LLM project. This component of the project necessitated the employment of various computing devices, which were consistently used by the team throughout the project's duration. The specifications regarding the Storage and CPU requirements are described in Table 8 provided below.

Table 8

Hardware Requirements Of The Project

Hardware	Specifications	Cost	Justification
Storage (Data Management)	10GB	Varies by size	offers reliable and expandable solutions to satisfy a range of data storage needs.
CPU Instances (Processing)		Varies by size	makes it easier to manipulate data effectively and quickly deploy models for real-time applications.
Load Balancer		Varies by size	balances the load between servers to increase system speed and dependability..

Note. The Table Shows Hardware Requirements Of The Project.

Software Requirements

Python, TensorFlow, PyTorch, CUDA and Med7 are used in the development and execution of the hospital readmission prediction system as shown in Table 9. The primary programming language is Python, and TensorFlow and PyTorch are utilized to create predictive models. While Flask makes it easier for medical practitioners to use web-based interfaces, CUDA improves computing efficiency. Streamlit, Git & GitHub, Google Colab, and Med7 facilitate collaborative development, version control, and interactive data visualization, respectively. Med7 also helps extract medical entities from clinical text. When utilized in tandem, these software technologies enable the system to more precisely forecast hospital readmissions and enhance patient care.

Table 9

Software Requirements Of The Project

Software	Specifications	Cost	Purpose
Python	Open Source	Free	Programming Language for data processing, and more
TensorFlow	Open Source	Free	Developing and training machine learning models.
PyTorch	Open Source	Free	Flexible framework for prototyping and implementing architectures
CUDA	Open Source	Free	To accelerate computations on NVIDIA GPUs.
Flask	Open Source	Free	Developing web applications and APIs, including for AI model deployment.
Med7	Open Source	Free	For Clinical text data extraction for medical entities
LLM	Llama index	Free	Manages large datasets using Elasticsearch effectively.
Git & GitHub	Open Source	Free	Manages code revisions and supports collaboration, ensuring project continuity and integrity.
Streamlit	Open Source	Free	Enhances UX with Dash and Streamlit interfaces.

Note. The Table Shows Software Requirements Of The Project.

Tools and Licenses

For the creation of the Hospital Readmission Prediction System With LLM, a variety of programming languages and software tools are essential. It is crucial to verify the necessity of licenses for each tool deployed in this project. The array of tools utilized is detailed in Table 10.

Table 10

Tools And Licenses For The Project

Tools/License	Specifications	Costs	Justification
Visual Studio Code	Open source	Free	A widely-used code editor with multi-language support and versatile features.
Microsoft Office 365	Open source	Free	Essential for documentation and reporting purposes
Jira	Tool	Free	Offers visual and flexible task and project organization capabilities.
Zoom	Communication pathway	Free	Provides a platform for video meetings with robust features.
Google Colab Pro Plus	Open source	$49/month	A collaborative, cloud-based Python code editor for LLM Models
Draw.io	Open source	Free	For creating WBS, Pert chart, Figures
Google Cloud Platform	Open source	Free	Data Storage

Note. Table Shows Tools And Licenses For The Project.

Project Schedule

Project schedules can be visualized using tools like Gantt Charts and PERT charts, aiding in planning and tracking tasks, dependencies, and timelines effectively.

Gantt Chart

The Gantt Chart serves as a pivotal project management tool, offering a visual representation of task timelines. Within the project framework, this chart is constructed through Jira Software. Here, epics based on CRISP DM topics are first defined, which are then broken down into stories and tasks. These tasks are further segmented into manageable subtasks, each spanning 1-2 days. Assigning each subtask to an individual ensures clear accountability for progress, facilitating seamless monitoring by team members. Moreover, the chart effectively showcases task dependencies, thereby aiding in risk mitigation, refining time management strategies, scoping efforts, and fostering enhanced communication across the team.

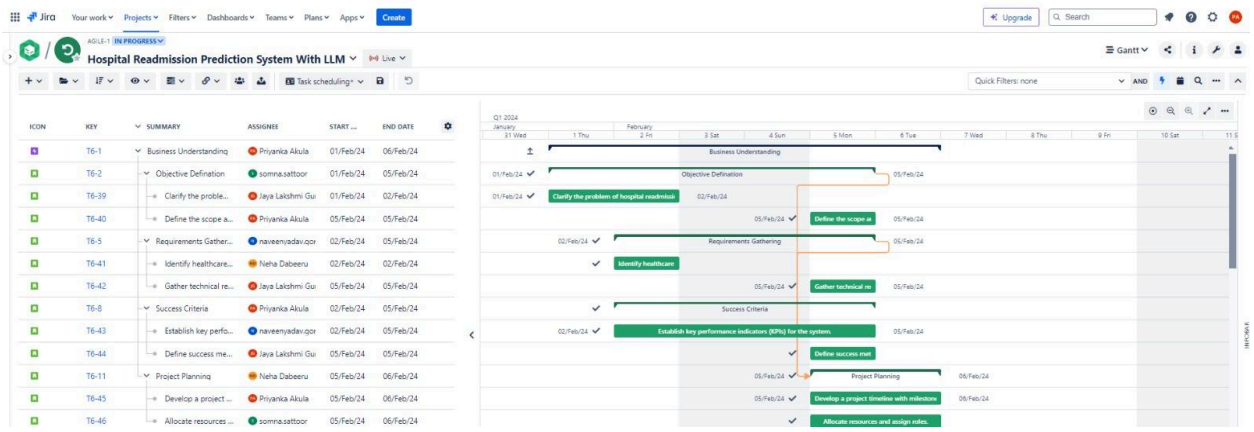
Business Understanding. The CRISP-DM project's 'Business Understanding' phase, running from February 1 to February 6, 2024, serves as the groundwork with a comprehensive 4-day analysis. In this initial stretch, the 'Objective Definition' task from February 1 to February 5 sets the project's clear goals. This includes the critical assessment of hospital readmissions issues over two days from the start and a single day dedicated to defining the predictive system's scope on February 5.

Concurrent tasks comprise 'Requirements Gathering' from February 2 to February 5, which investigates healthcare providers' expectations and the project's technical needs, allocating one day to each task. Simultaneously, 'Success Criteria' begins on February 2, with two days earmarked for establishing KPIs and one day, February 5, to specify success metrics. After the project's objectives and requirements are well-established, 'Project Planning' initiates with an

end-to-start dependency, immediately following the 'Requirements Gathering'. This phase, spanning February 5 and 6, is pivotal for developing a timeline replete with milestones and for the distribution of roles and resources as shown in Figure 11. This sequential dependency ensures that planning is informed by defined requirements, thereby securing a clear and systematic progression for the project's early stages.

Figure 11

Business Understanding Phase



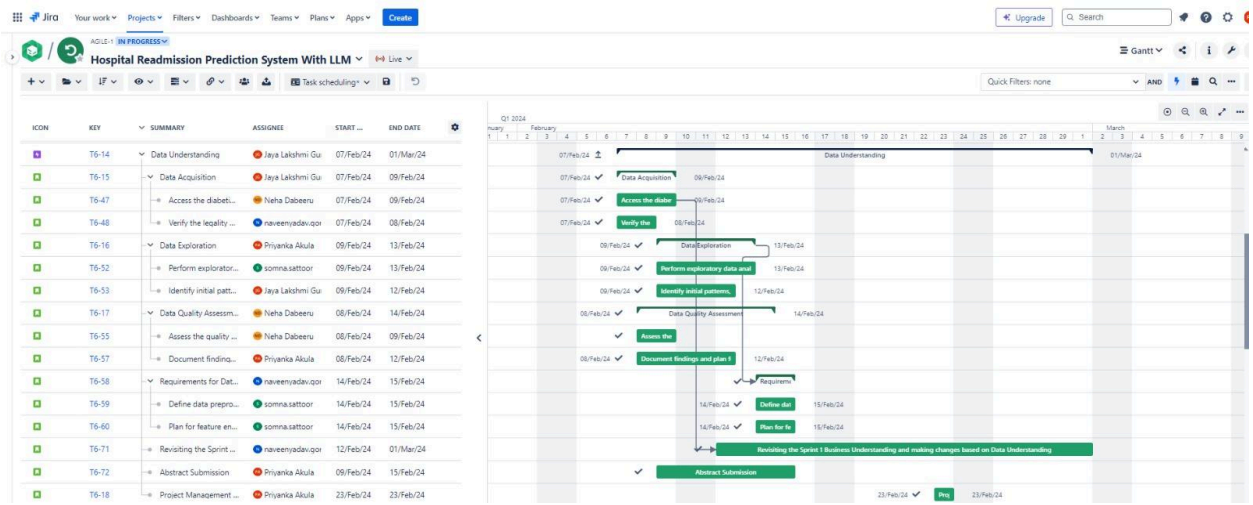
Note. Figure Gives A Detailed Overview Of The Business Understanding.

Data Understanding. The 'Data Understanding' phase, spanning from February 7 to March 1, 2024, is dedicated to comprehending the context and quality of the data. Initial steps involve 'Data Acquisition' from February 7 to 9, accessing the diabetic dataset from UC Irvine and Kaggle. This is followed by a 2-day verification of its legality and ethics. 'Data Exploration', occurring from February 9 to 13, entails scrutinizing the dataset through exploratory analysis over a 3-day period and pinpointing key patterns, anomalies, and relevant variables within a 2-day timeframe. Subsequently, 'Data Quality Assessment' from February 8 to 14 focuses on evaluating the data's integrity, followed by documenting the findings and strategizing data cleaning efforts for 5 days and 3 days, respectively.

Preparation for data processing is outlined from February 14 to 15, emphasizing setting preprocessing requirements, feature engineering, and selection within concise 2-day windows. The process then revisits the 'Business Understanding' phase from February 12 to March 1 for refinement over a comprehensive 15-day period, once the task of exploring the datasets is completed, as there is an end-start dependency built as shown in Figure 12. Lastly, the 'Abstract Submission' task, scheduled from February 9 to 15, culminates the phase with a 5-day period for formal documentation and submission. This structured approach ensures effective establishment of data-driven insights for subsequent project phases.

Figure 12

Data Understanding Phase



Note. Figure Gives a Detailed Overview of The Data Understanding.

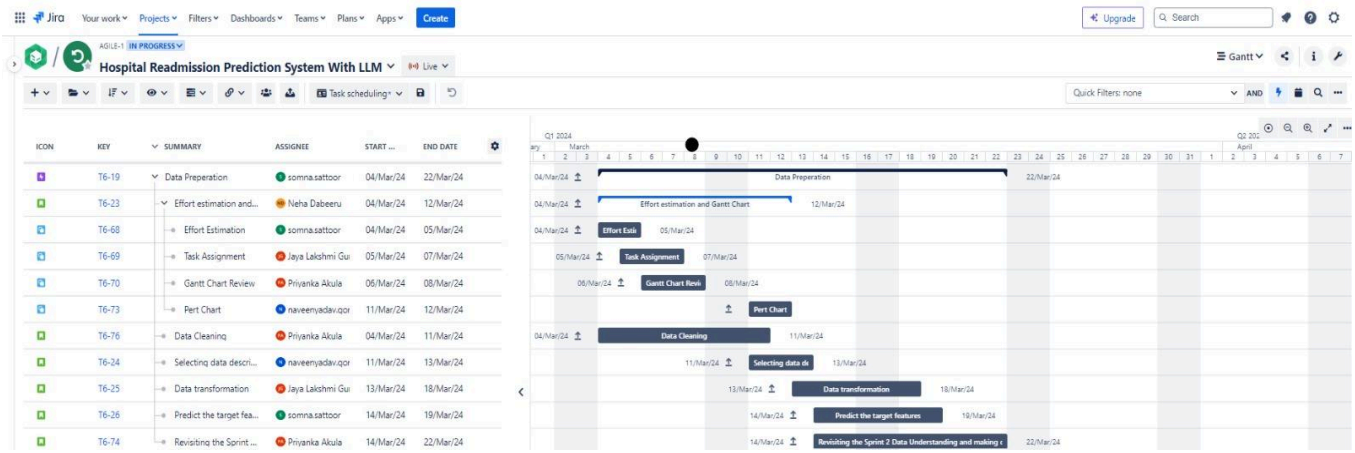
Data Preparation. The 'Data Preparation' phase in the CRISP-DM project extends from March 4 to March 22, 2024, spanning 15 days. This stage commences with 'Effort estimation and Gantt Chart' tasks, stretching over 7 days from March 4 to March 12. It breaks down into 'Effort Estimation' from March 4 to 5 and 'Task Assignment' from March 5 to 7, each task scheduled over two and three days respectively. Following this, 'Gantt Chart Review' is slotted for three

days, starting on March 6 and concluding on March 8. The project management tasks continue with 'Pert Chart' analysis on March 11 and 12, a two-day activity aimed at optimizing task sequences and timelines. The most important task. ‘Data Cleaning’, is scheduled from March 4 to 11 (6 days), which consists of tasks like handling noises, removing null values, identifying outliers, etc. 'Selecting data descriptive features' is scheduled from March 11 to March 13, focusing on identifying key data characteristics. This is followed by 'Data transformation', which occurs from March 13 to March 18, dedicated to converting the data into a suitable format for analysis as shown in Figure 13.

The task 'Predict the target features', occupies the dates from March 14 to 19, dedicated to forecasting essential variables. The phase wraps up with a review and iteration period, 'Revisiting the Sprint 2 Data Understanding and making changes based on Data Preparation.', spanning 7 days from March 14 to 22. This critical task allows for refinement and adjustments, ensuring that the data understanding is as thorough and accurate as possible, setting a strong foundation for subsequent analysis.

Figure 13

Data Preparation Phase



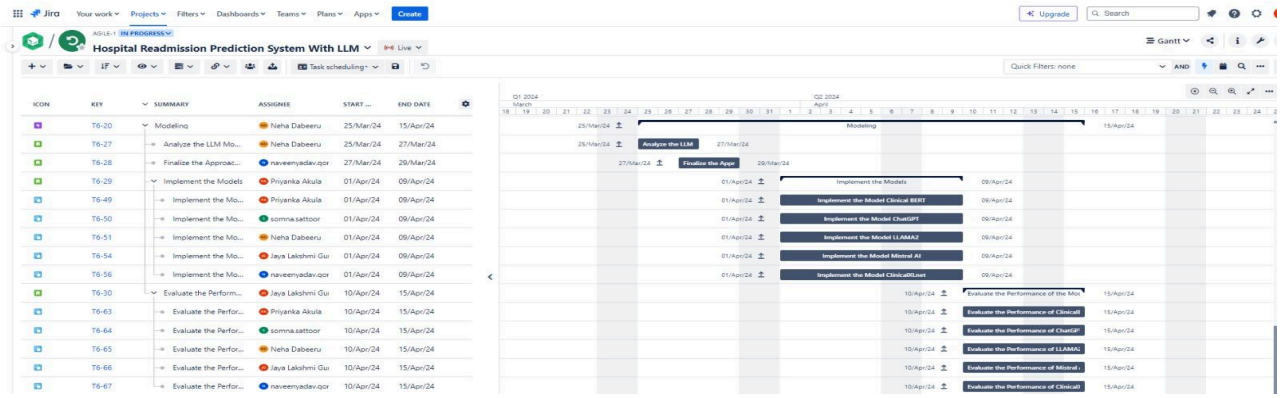
Note. Figure Gives a Detailed Overview of The Data Preparation Phase.

Modeling. The 'Modeling' phase of the CRISP-DM project unfolds from March 25 to April 15, 2024, across a span of 16 days. The phase starts with analyzing the LLM (Large Language Models) models for a period of 3 days, from March 25 to March 27. This analysis phase is crucial for understanding the strengths and limitations of the models before application.

Following this, the project transitions to finalizing the approaches for 3 days, from March 27 to March 29, a process that sets the stage for practical implementation. Once the strategies are established, the implementation of various models takes place, each scheduled for 7 days from April 1 to April 9. The models to be implemented include Clinical BERT, ChatGPT, LLAMA2, Mistral AI, and ClinicalXLnet, signifying a parallel process of development for these diverse systems. The phase culminates with the 'Evaluate the Performance of the Models' task from April 10 to April 15, lasting 4 days. This evaluation is segmented into subtasks, each dedicated to assessing the performance of an individual model: ClinicalBERT, ChatGPT, LLAMA2, Mistral AI, and ClinicalXLnet, ensuring a thorough analysis of each model's effectiveness and readiness for deployment as shown in Figure 14. This methodical process underscores the meticulous attention to detail and rigorous standards upheld throughout the modeling phase of the project.

Figure 14

Modeling Phase

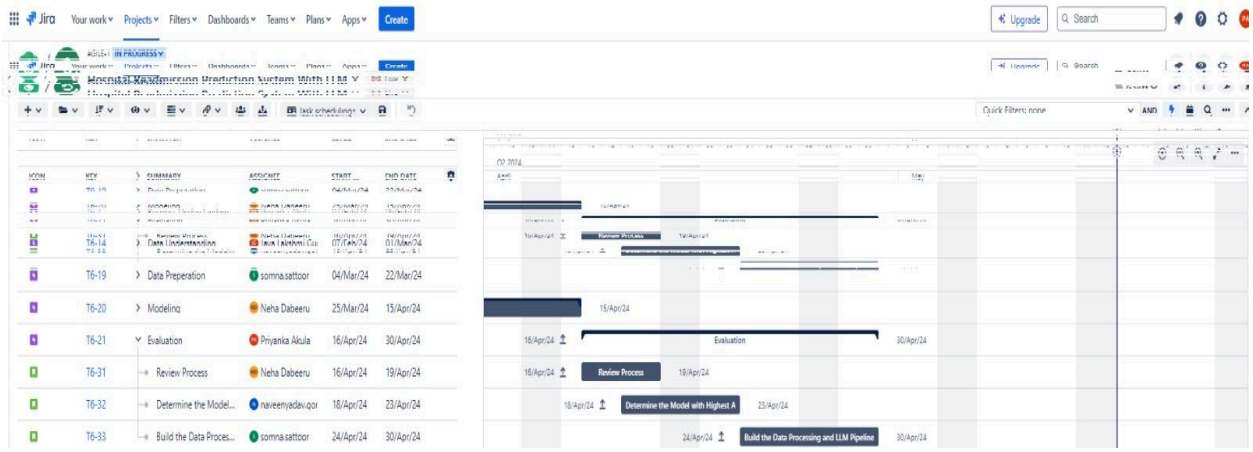


Note. Figure Gives a Detailed Overview of The Modeling Phase.

Evaluation. This phase in the CRISP-DM project timeline is set from April 16 to April 30, 2024, encompassing 11 days. This phase involves a 'Review Process' that starts on April 16 and ends on April 19, spanning 4 days. The objective is to thoroughly assess the models and methods applied in the project. Following the review, attention shifts to 'Determine the Model with Highest Accuracy', a task assigned 4 days from April 18 to April 23. This step is vital for identifying the most effective model based on performance metrics as shown in Figure 15. The final task, 'Build the Data Processing and LLM Pipelines', begins on April 24 and concludes on April 30, extending over 5 days. This involves constructing the necessary data processing structures and integrating the Large Language Models into the project workflow, a crucial step for operationalizing the chosen models. This sequence of tasks ensures that the project moves forward with the most capable models and robust data handling mechanisms.

Figure 15

Evaluation Phase



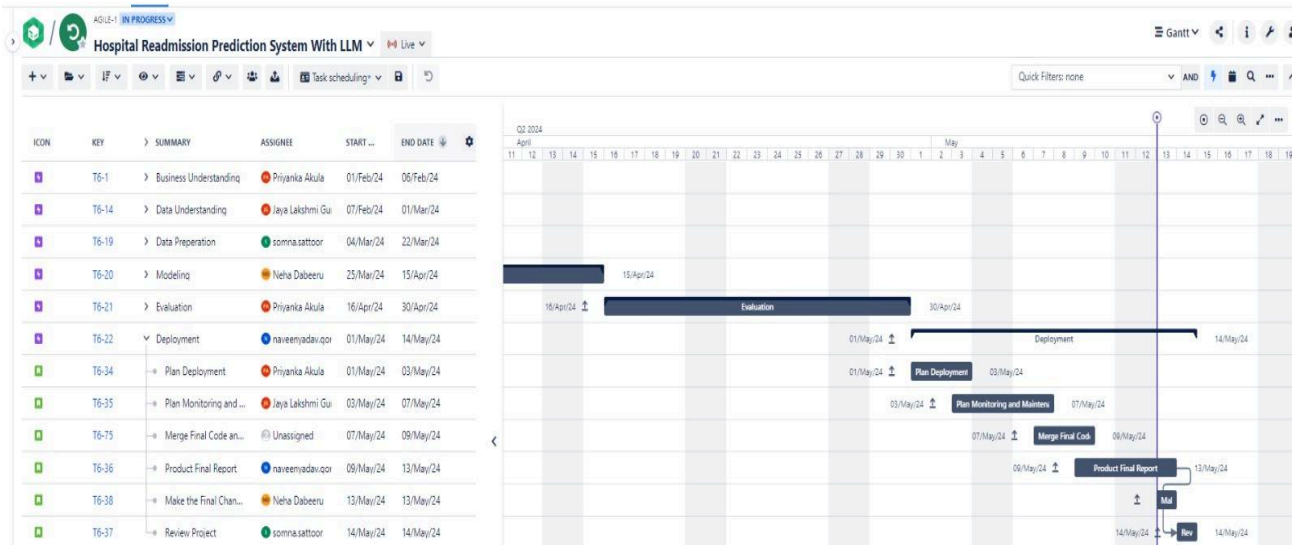
Note. Figure Gives a Detailed Overview of The Evaluation Phase.

Deployment. This phase spans from May 1 to May 15, 2024, comprising an 11-day window. This phase begins with 'Plan Deployment' from May 1 to May 3, dedicating 3 days to strategize the rollout of the project deliverables. Following this, 'Plan Monitoring and

Maintenance' is scheduled from May 3 to May 7, also allocated 3 days, ensuring the project's long-term success and addressing any potential issues post-deployment. Concurrently, from May 7 to May 10, 'Make the Final Changes' is assigned 4 days, indicating a period for last-minute adjustments before finalization as shown in Figure 16. 'Product Final Report' is then set from May 9 to May 13, spanning 3 days, aimed at documenting the project outcomes and insights. The phase concludes with a 'Review Project' task, a 2-day review on May 14 and 15, which serves as a final assessment of the project before its closure. This stage is critical for ensuring that the project meets its intended goals and is prepared for a successful launch.

Figure 16

Deployment Phase



Note. Figure Gives a Detailed Overview of The Deployment Phase.

PERT Chart

Project development tools like Gantt charts and PERT charts offer distinct visualization approaches. While Gantt charts focus on task timelines, PERT charts, or Program Evaluation Review Technique charts, emphasize task relationships and dependencies. Creating a PERT chart

involves breaking down tasks, identifying dependencies, estimating task durations, and determining the critical path—the longest path of dependent tasks that establishes the project's minimum duration. By following these steps, project managers gain insights into task interdependencies, estimate project duration, and identify critical tasks crucial for effective project planning and risk management, ultimately contributing to project success.

Critical path. From the provided PERT chart, a sequential flow of tasks is being established, which is required for the effective completion of the Hospital Readmission Prediction System with the LLM project. Following the CRISP-DM methodology, the chart is organized into six distinct stages: business understanding, data understanding, data preparation, modeling, evaluation, and deployment. The critical path is identified with red arrows, showing the sequence of hierarchically independent tasks. Within the business understanding stage, the tasks from 'Objective Definition' to 'Project Planning' are deemed critical. These steps are essential and must be completed for the project to advance.

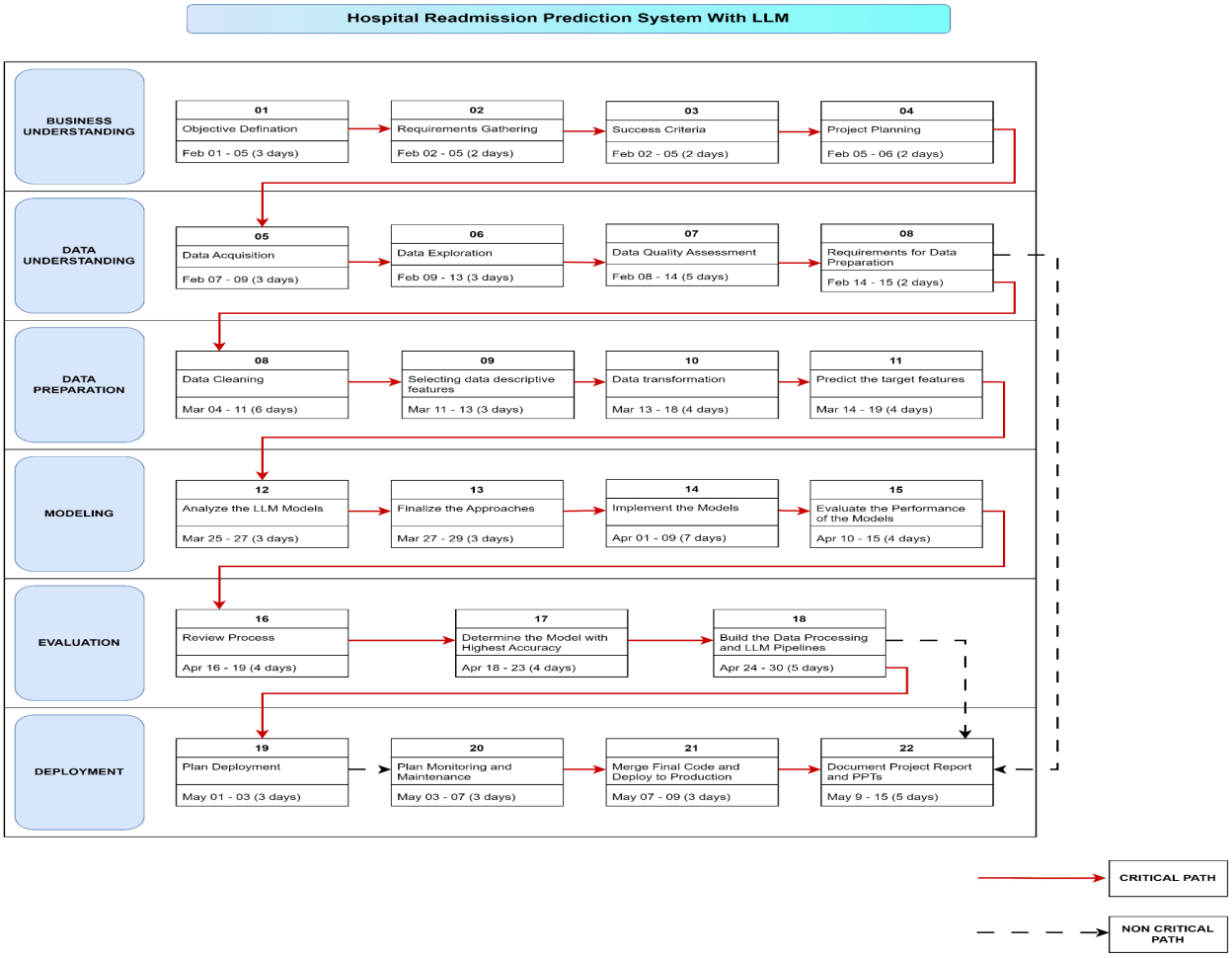
In the data understanding stage, tasks like 'Data Acquisition', 'Data Exploration', and 'Data Quality Assessment' are critical. These are foundational for ensuring that the data preparation stage, which includes 'Data Cleaning', 'Selecting data descriptive features', 'Data transformation', and 'Predict the target features', can proceed effectively. Then comes the modeling stage, 'Analyze the LLM Models', 'Finalize the Approaches', and 'Implement the Models' are all critical tasks that contribute to the robust development of the system. For the evaluation stage, 'Review Process' and 'Determine the Model with Highest Accuracy' are critical to ensure the model’s effectiveness before deployment.

Finally, in the deployment stage, tasks like 'Plan Deployment', 'Plan Monitoring and Maintenance', and 'Merge Final Code and Deploy to Production' are on the critical path,

indicating their importance for the successful launch and maintenance of the prediction system. The tasks connected by black-colored arrows represent the non-critical path, indicating tasks with zero dependencies as shown in Figure 17. Delays in these tasks, such as 'Document Project Report and PPTs', would not influence the timely success of other tasks or the overall project completion. Summarizing the PERT chart's critical analysis, recognizing that most tasks are critical to the project's timely and successful completion, while a few, identified by black arrows, can be considered non-essential to the core project timeline.

Figure 17

PERT Chart - Hospital Readmission Prediction System With LLM



Note. Figure Gives a Detailed Overview of The PERT Chart

Data Engineering

Data Process

Large language models are employed to identify instances of patient readmission within hospitals through the analysis of two datasets. The initial dataset, sourced from Kaggle (The Devastator, 2023), focuses on hospital readmissions and mortality rates, comprising comprehensive patient records diagnosed with various ailments, including diabetes. This dataset includes extensive information such as patient demographics (age, gender, race), type of admission, duration of stay, medical history, prescribed medications, laboratory test results, diagnosis, and the number of inpatient and outpatient visits. Utilizing a substantial volume of data is essential for effectively leveraging large language models for data analysis.

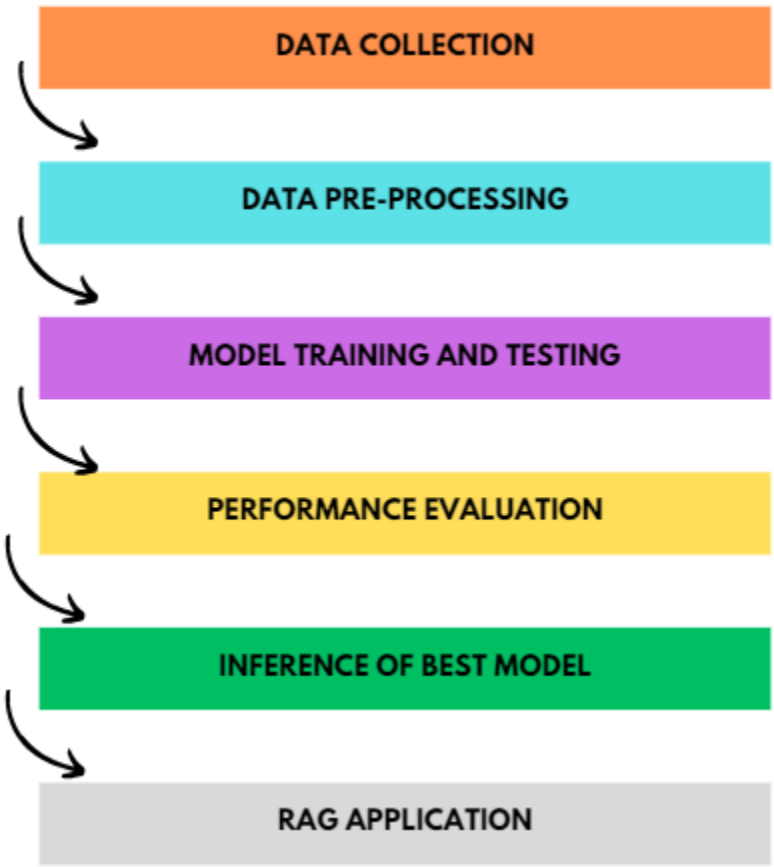
The second dataset, obtained from the Health Facts database (Cerner Corporation, Kansas City, MO), focuses on diabetic patients and spans a ten-year clinical treatment period in 130 US hospitals and integrated delivery networks. Similar to the first dataset, it contains detailed information on patient demographics, admission details, medical specializations, laboratory findings, patient HbA1c test outcomes, diagnoses, prescribed medications, and inpatient and outpatient visits.

Figure 18 illustrates the subsequent step of data pre-processing, which involves conducting exploratory data analysis (EDA). This stage enhances data quality and accuracy by addressing issues such as missing values, inconsistent entries, and noise in the data, making the dataset more consistent and appropriate for analysis. The datasets from the Health Facts database and the US Healthcare Readmissions and Mortality dataset were loaded into a pandas DataFrame for manipulation and evaluation. An initial assessment revealed substantial records and columns, suggesting a rich source of information. Each column was examined for missing values,

identified by the presence of '?' in various categorical and numeric fields. Missing values for categorical attributes such as 'race' and 'gender' were imputed using the mode, while missing values in numerical attributes like 'time_in_hospital' and 'number_diagnoses' were substituted with the mean and median values of the columns.

Figure 18

Workflow Diagram



Note. Workflow Of The Processes For Hospital Readmission Prediction.

Data validation techniques were applied to detect outliers, especially in numerical attributes where values outside acceptable ranges were excluded. For example, entries in the 'time_in_hospital' attribute with durations beyond a reasonable range (e.g., 0 to 30 days) were

removed. Additionally, categorical variables were standardized to ensure uniform formatting, eliminating variations and discrepancies resulting from data entry errors. This meticulous correction of missing data, inconsistencies, and noisy entries refined the dataset, making it suitable for further analysis and modeling.

Data transformation involved cleansing the data to remove inaccuracies or inconsistencies. All numerical data were standardized to manage inconsistencies in scale, ensuring uniform processing across all numerical inputs. Categorical variables were transformed using binary and one-hot encoding methods. Techniques like Principal Component Analysis (PCA) were applied to decrease the dataset's dimensionality, focusing on the most significant features while reducing computational load. The target variable 'Readmission' was simplified from multiple classes to a binary format, reducing the model's complexity. Age data were converted from categorical to numerical format, allowing for finer quantitative evaluations. Finally, The dataset was then strategically divided into training, validation, and testing segments following an 80-10-10 distribution. This division enabled comprehensive training of the model and rigorous evaluation of its predictive performance.

Data Collection

The Health Facts database, managed by Cerner Corporation in Kansas City, MO, is devoted to tracking patients with diabetes across a ten-year span. This repository incorporates data from 130 U.S. hospitals and integrated delivery networks, comprising a dataset size of 19 MB with 50 columns and 101,766 instances as shown in Figure 19. It encompasses detailed patient information, including demographics, admission details, medical specialty, laboratory findings, HbA1c test results, diagnoses, prescriptions, and the count of inpatient and outpatient visits.

An additional dataset by "The Devastator, 2023" has been sourced from Kaggle. This dataset, sized at 2 MB with 64,764 instances, focuses on hospital readmissions and mortality rates, containing patient records for a variety of ailments including diabetes. Comprehensive data elements within this dataset cover patient demographics (age, gender, and race), admission type, duration of stay, medical history, prescribed medications, laboratory test results, diagnoses, and the number of inpatient and outpatient visits. It should be noted that this dataset does contain missing values that will be addressed during the subsequent data preparation phase. Accessibility to both databases is granted to the entire project team, ensuring collaborative access and review. Given the extensive array of data columns, only a representative sample is included in the data collection table to maintain clarity and focus on key variables.

Figure 19

Data Collection Plan

1	**Data Collection Plan**						
2	Project number	Team 6	Project title:	Hospital Readmission Prediction System With LLM	Project leader:	Priyanka Akula	Date:	04/17/2024
3	**Description of the data collection**						
4		Diabetes Health Facts was produced by Kansas City-based Cerner Corporation. 130 US hospitals and IDNs contributed data for ten years. This database contains demographics, admissions, medical specialties, laboratory results, HbA1c test results, diagnosis, prescriptions, and inpatient and outpatient visits. "The Devastator, 2023," another Kaggle dataset, tracks hospital readmissions and deaths. Dataset patients include diabetics. This dataset includes patient demographics (age, gender, and race), admission type, duration of stay, medical history, prescribed medications, laboratory test results, diagnosis, and inpatient and outpatient visits, which are stored in GCP and our local to train and test models.					
6	**What will be done with the data once it has been collected?**						
7		yes	Identify how well the process is meeting customer needs		yes	Analyze a problem, or identify the causes of variation	
8		yes	Obtain exploratory view of the process		yes	Test a hypothesis about the process output	
9		yes	Evaluate the measurement system		yes	Test a hypothesis about the effects of one or more inputs	
10		yes	Check the stability of the process (is it in control?)		yes	**Control a process input or monitor a process output**	
11		yes	Conduct a capability study			**Other reason…**	
12	**Key Variables - A summary of the chosen input variables (Y's) and/or output variables (X's)**						
13				**1**	**2**	**3**	
14		**Variable title**		readmitted	AGE	Gender	
15		**Input (X) or output (Y) va**		X	Y	Y	
16		Unit of measurement		N/A	N/A	N/A	
17		Data type		CSV	CSV		
18		Collection method		Manual	Manual	Manual	
19	What?			The Diabetes Health Facts database. Over ten years, 130 US hospitals and integrated delivery networks provided data. Demographics, admission details, medical specialty, laboratory findings, HbA1c test results, diagnosis, prescriptions, and inpatient and outpatient visits are in this database.			
20		If manual		"The Devastator, 2023," another Kaggle dataset, covers hospital readmissions and death. Diabetics are among the patients in the dataset.			
21		Gauge/instrument		N/A	N/A	N/A	
22		Location		N/A	N/A	N/A	
23		Gauge calibrated?		N/A	N/A	N/A	
24		Measurement system checke		N/A	N/A	N/A	
25		Precision (R&R) adequate?		N/A	N/A	N/A	
26	MSA	Accuracy adequate?		N/A	N/A	N/A	
27		Historical data exist?		N/A	N/A	N/A	
28		Source of historical data		N/A	N/A	N/A	
29		Historical data representativ		N/A	N/A	N/A	
30		Mean and Standar deviation		N/A	N/A	N/A	
31		Upper and lower specificati		N/A	N/A	N/A	
32	Historical data	Target		N/A	N/A	N/A	
33		Minimum sample size (MSS		N/A	N/A	N/A	
34		Sampling frequency		N/A	N/A	N/A	
35		Sub-grouping needed?		N/A	N/A	N/A	
36		Sub-group size		N/A	N/A	N/A	
37	Sampling	Stratification needed?		N/A	N/A	N/A	
38		**Data collector**		Priyanka Akula	Priyanka Akula	Priyanka Akula	
39	Who?	Operational definition exist? Data collector trained? Resources available for data collector?		There is no operational definition for the gathered to make sure it is consistent, and it doesn't need to be documented as meta data. To understand the facts, you don't need to know anything about the domain.			
40		**Start date**		12-Feb	12-Feb	12-Feb	
41		**Due date**		14-Feb	14-Feb	14-Feb	
42	When?	Duration (in days)		2	2	2	

Note. The Data Collection Plan's Properties, Data Sources, And Project Implementation Dates Are Listed Above.

Dataset samples. As said, both datasets contain patient data in.csv format. The dataset variable table from our local storage is shown in Figure 20. As illustrated in Figure 21, the csv file has 50 columns with categories and numeric data.

Figure 20

Sample Of Variables Table From UC Irvine Machine Learning Repository

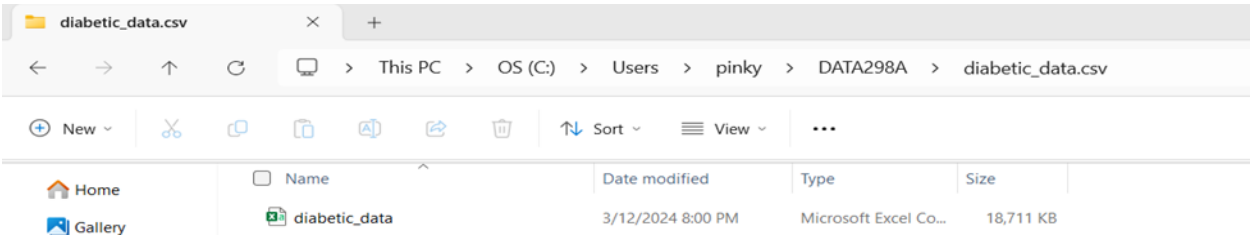
Variables Table

Variable Name	Role	Type	Demographic	Description	Units	Missing Values
encounter_id	ID			Unique identifier of an encounter		no
patient_nbr	ID			Unique identifier of a patient		no
race	Feature	Categorical	Race	Values: Caucasian, Asian, African American, Hispanic, and other		yes
gender	Feature	Categorical	Gender	Values: male, female, and unknown/invalid		no
age	Feature	Categorical	Age	Grouped in 10-year intervals: [0, 10), [10, 20),..., [90, 100)		no
weight	Feature	Categorical		Weight in pounds.		yes
admission_type_id	Feature	Categorical		Integer identifier corresponding to 9 distinct values, for example, emergency, urgent, elective, newborn, and not available		no
discharge_disposition_id	Feature	Categorical		Integer identifier corresponding to 29 distinct values, for example, discharged to home, expired, and not available		no
admission_source_id	Feature	Categorical		Integer identifier corresponding to 21 distinct values, for example, physician referral, emergency room, and transfer from a hospital		no
time_in_hospital	Feature	Integer		Integer number of days between admission and discharge		no

Note. The Above Figure Shows The Entire Variables Table From The Source.

Figure 21

Sample Of Raw Dataset Saved In Local System



Note. The Above Figure Shows The Raw Dataset Saved In Local System.

The below Figure 22 and Figure 23 gives the Shape of diabetic dataset which contains a total of 50 columns in it. Overall 49 are input columns and 1 readmitted is an output column and 101766 rows.

Figure 22

The Number Of Rows And Columns In The Dataset

```
print('The shape of the Dataset is :', df.shape, 'with', df.shape[0], 'records and', df.shape[1], 'columns')
The shape of the Dataset is : (101766, 50) with 101766 records and 50 columns
```

Note. The Above Figure Shows The Shape Of The Dataset.

Figure 23

Columns In The Raw Dataset

```
Index(['encounter_id', 'patient_nbr', 'race', 'gender', 'age', 'weight',
       'admission_type_id', 'discharge_disposition_id', 'admission_source_id',
       'time_in_hospital', 'payer_code', 'medical_specialty',
       'num_lab_procedures', 'num_procedures', 'num_medications',
       'number_outpatient', 'number_emergency', 'number_inpatient', 'diag_1',
       'diag_2', 'diag_3', 'number_diagnoses', 'max_glu_serum', 'A1Cresult',
       'metformin', 'repaglinide', 'nateglinide', 'chlorpropamide',
       'glimepiride', 'acetohexamide', 'glipizide', 'glyburide', 'tolbutamide',
       'pioglitazone', 'rosiglitazone', 'acarbose', 'miglitol', 'troglitazone',
       'tolazamide', 'examide', 'citoglipton', 'insulin',
       'glyburide-metformin', 'glipizide-metformin',
       'glimepiride-pioglitazone', 'metformin-rosiglitazone',
       'metformin-pioglitazone', 'change', 'diabetesMed', 'readmitted'],
      dtype='object')
```

Note. The Above Figure Shows The Columns In The Dataset.

Below the second raw dataset was collected from kaggle.com as shown in Figure 24 and Figure 25 shows the shape of the dataset whereas Figure 26 gives the column view of the dataset which is of 20 columns in it.

Figure 24

Sample Of Raw Dataset From Kaggle

Readmissions_and_Deaths_-_Hospital.csv (21.29 MB)

Detail Compact Column 20 of 20 columns

# index	Provider ID	Hospital N...	Address	City	State	# ZIP Code
1	230121	MEMORIAL HEALTHCARE	826 WEST KING STREET	OWOSSO	MI	48867
2	230118	HURON MEDICAL CENTER	1100 SOUTH VAN DYKE ROAD	BAD AXE	MI	48413
3	230121	MEMORIAL HEALTHCARE	826 WEST KING STREET	OWOSSO	MI	48867

Note. The Above Figure Shows The Raw Dataset Which Is In .CSV File.

Figure 25

The Number Of Rows And Columns In The Dataset

```
hospital.shape
```

```
(64764, 20)
```

Note. The Above Figure Shows The Shape Of The Second Dataset.

Figure 26

Columns In The Raw Dataset

```
Index(['index', 'Provider ID', 'Hospital Name', 'Address', 'City', 'State',
       'ZIP Code', 'County Name', 'Phone Number', 'Measure Name', 'Measure ID',
       'Compared to National', 'Denominator', 'Score', 'Lower Estimate',
       'Higher Estimate', 'Footnote', 'Measure Start Date', 'Measure End Date',
       'Location'],
      dtype='object')
```

Note. The Figure Above Shows Second Dataset Columns.

Data Pre-Processing

Data preprocessing is essential for refining raw data into a format that is more suitable for analysis. This step improves the dataset's quality and accuracy by resolving issues like missing values, inconsistent entries, and noisy data, thereby ensuring consistency and reliability for subsequent modeling efforts. For this project, the raw datasets were thoroughly examined to identify any patterns and discrepancies. Subsequently, a plan for data exploration was formulated to extract significant insights and guide the application of data transformation and cleaning methods to the datasets in question.

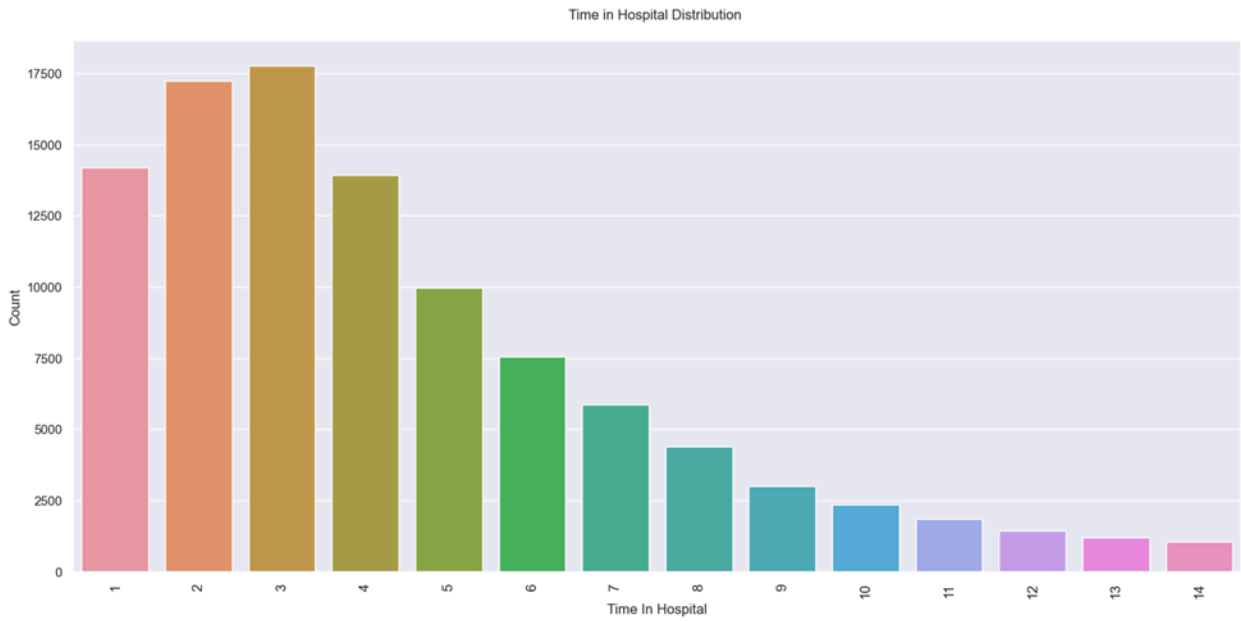
The dataset, known as the Diabetes 130-US hospitals for years 1999-2008 Data Set, is hosted in the UCI repository. It consists of data gathered from 130 U.S. hospitals over a decade (1999-2008), encompassing 101,766 observations across 50 features, which include demographic details, diagnostic data, medications, and outcomes related to hospitalization. An additional dataset used in this study is the US Healthcare Readmissions and Mortality dataset, available on Kaggle. This dataset offers an extensive view of 30-day readmission and mortality rates within U.S. hospitals, featuring 64,765 records and 50 columns.

In Figure 27, each bar of the histogram indicates the duration of hospital stays, which vary from one to 14 days. The x-axis denotes the duration in days, while the y-axis represents the

patient count. The chart shows that the majority of hospitalizations are brief, typically peaking at around three days. As the duration extends, the frequency of such occurrences declines, suggesting that longer hospital stays are uncommon. This visual representation yields valuable insights into the patterns of hospital stay durations within the dataset, aiding in the optimization of healthcare resource distribution, discharge planning, and the enhancement of patient care management strategies.

Figure 27

Time In Hospital Distribution



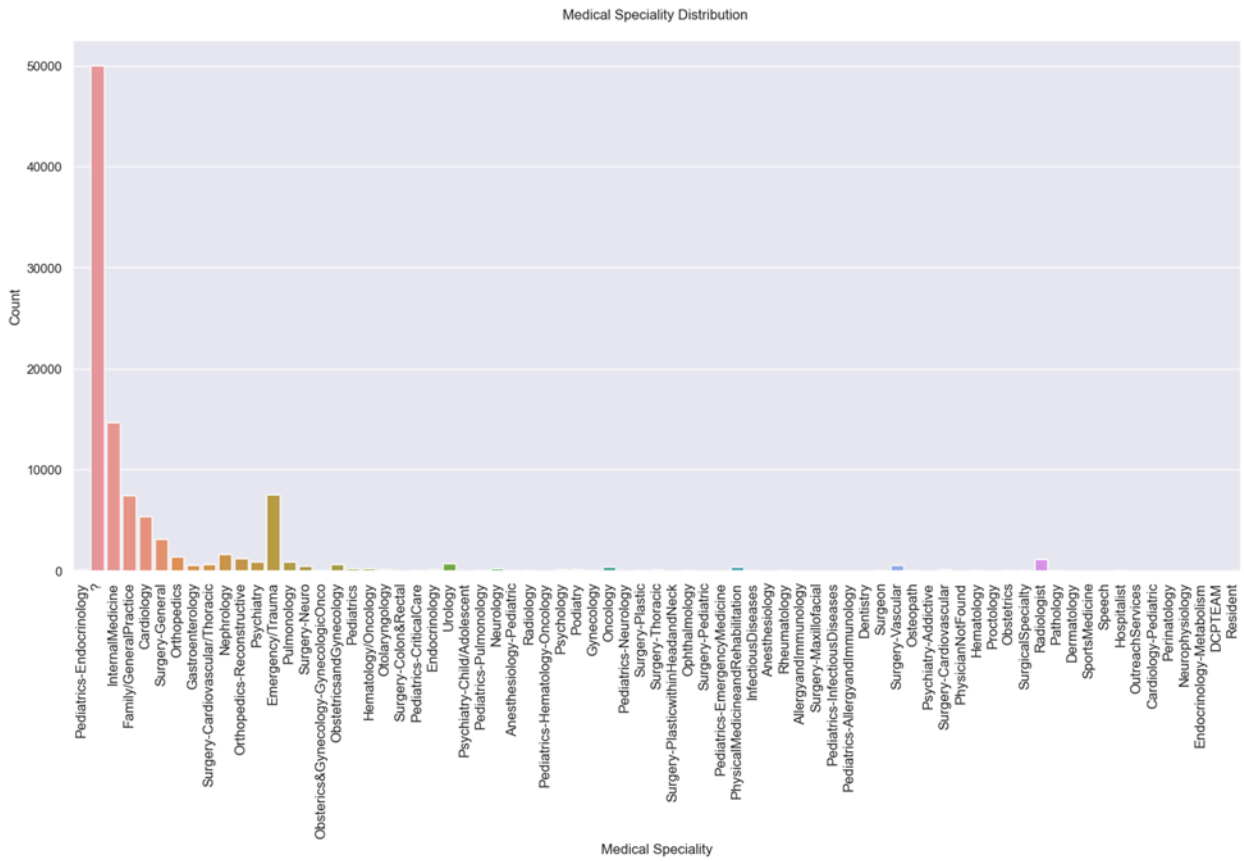
Note. The Plot Illustrates The Distribution Of Time Spent In The Hospital, Providing Insights Into Patient Hospitalization Patterns.

The graph in Figure 28 provides insights on the prevalence of different medical disciplines within the dataset. By examining the distribution of bars, one can determine which specialties are more prevalent among the dataset's records. This visualization helps healthcare

workers and analysts comprehend the structure of medical data by offering significant insights about the distribution of specialties and their representation in healthcare datasets.

Figure 28

Medical Specialty Distribution

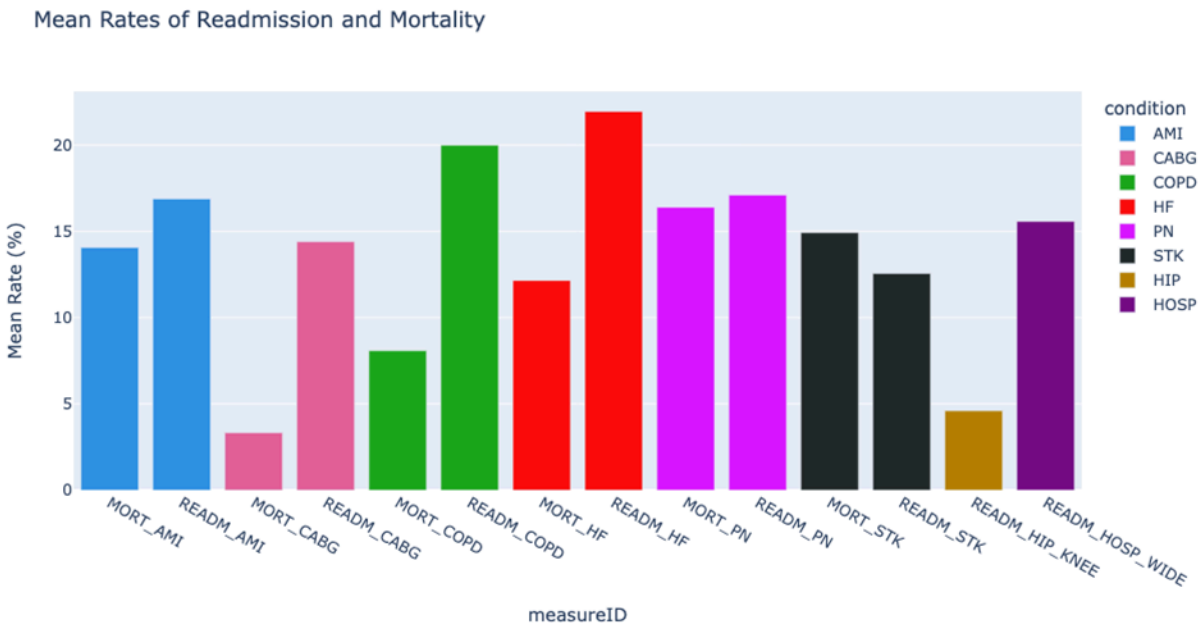


Note. The Plot Depicts The Distribution Of Medical Specializations, Providing Insights Into Their Prevalence In The Dataset.

The bar chart in Figure 29 depicts the mean rates of readmission and mortality across different conditions. Each bar represents a specific condition, with the x-axis indicating the corresponding measure ID and the y-axis representing the mean rate. The bars are color-coded based on the condition, allowing for easy visual differentiation.

Figure 29

Mean Rates Of Readmission And Mortality

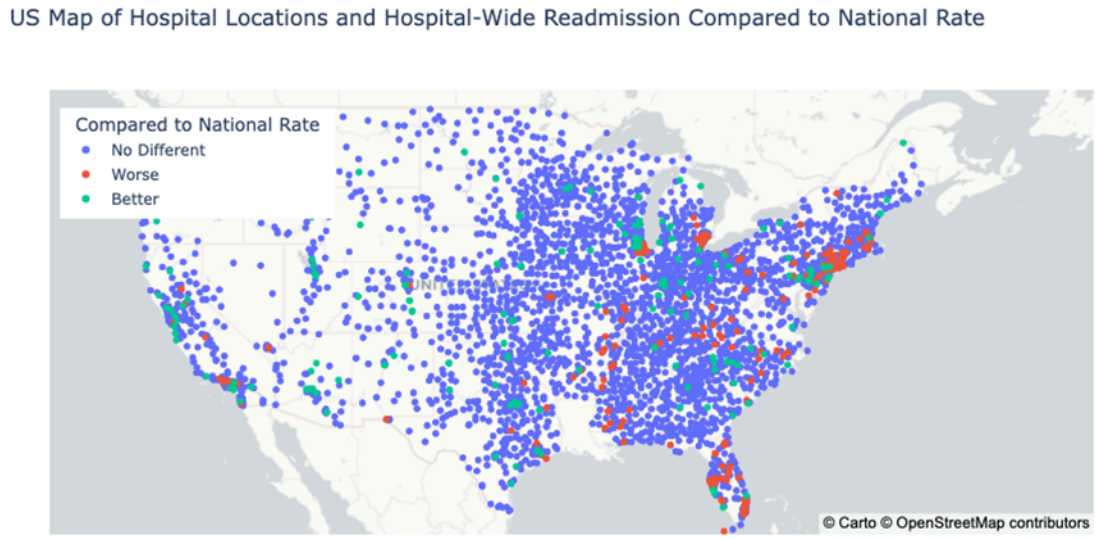


Note. The Bar Chart Depicts The Average Rates Of Readmission And Mortality For Various Illnesses, Allowing For Comparison Analysis.

This Plotly scatter map in Figure 30 depicts hospital sites across the United States, with colors reflecting hospital-wide readmission rates relative to the national rate. The graph's content is clearly described by the title "US Map of Hospital Locations and Hospital-Wide Readmission Compared to National Rate". Each point represents a hospital, whose location is determined by latitude and longitude coordinates. The color scale shows whether the hospital's readmission rate is better, worse, or equal to the national rate. Hovering over each point displays additional information such as the hospital name and score. The map's legend clarifies the color-coding technique, which improves interpretability.

Figure 30

Us Map Of Hospital Locations And Readmission Rates



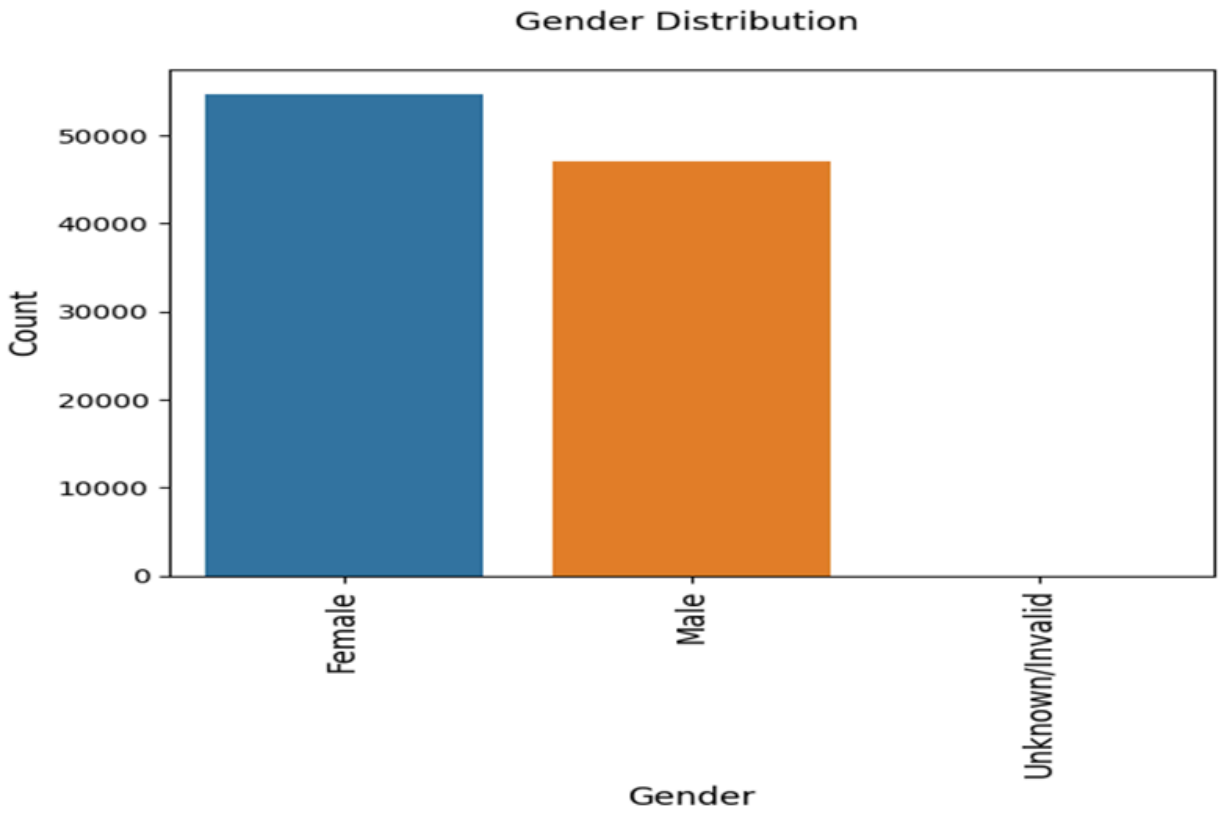
Note. The Scatter Map Depicts Us Hospital Locations And Readmission Rates In Comparison To The National Average, Allowing For More Detailed Geographical Study.

Handling Missing Data. In the initial stages of dataset exploration, several preprocessing techniques were utilized to confirm data integrity and ready the dataset for more detailed analysis. The datasets from the Health Facts database (Cerner Corporation, Kansas City, MO) and the US Healthcare Readmissions and Mortality dataset from Kaggle (The Devastator, 2023) were loaded into a pandas DataFrame for manipulation and evaluation. An initial assessment of the dataset structure from the Health Facts database disclosed a substantial number of records and columns, suggesting a rich source of information. Each column was examined for missing values, identified by the presence of '?' in various categorical and numeric fields. A systematic approach was adopted to address these missing entries. Missing values for categorical attributes such as 'race' and 'gender' were imputed using the mode (most frequent value) of each column, as illustrated in Figure 31. Figure 32 and 33 demonstrate the handling of the data. In

parallel, missing values in numerical attributes like 'time_in_hospital' and 'number_diagnoses' were substituted with the mean and median values of the columns. This methodical approach is designed to preserve data integrity and ensure statistical reliability in subsequent analyses.

Figure 31

Gender Distribution



Note. The Above Bar Plot Shows The Distribution Of Gender Over The Patients.

Figure 32

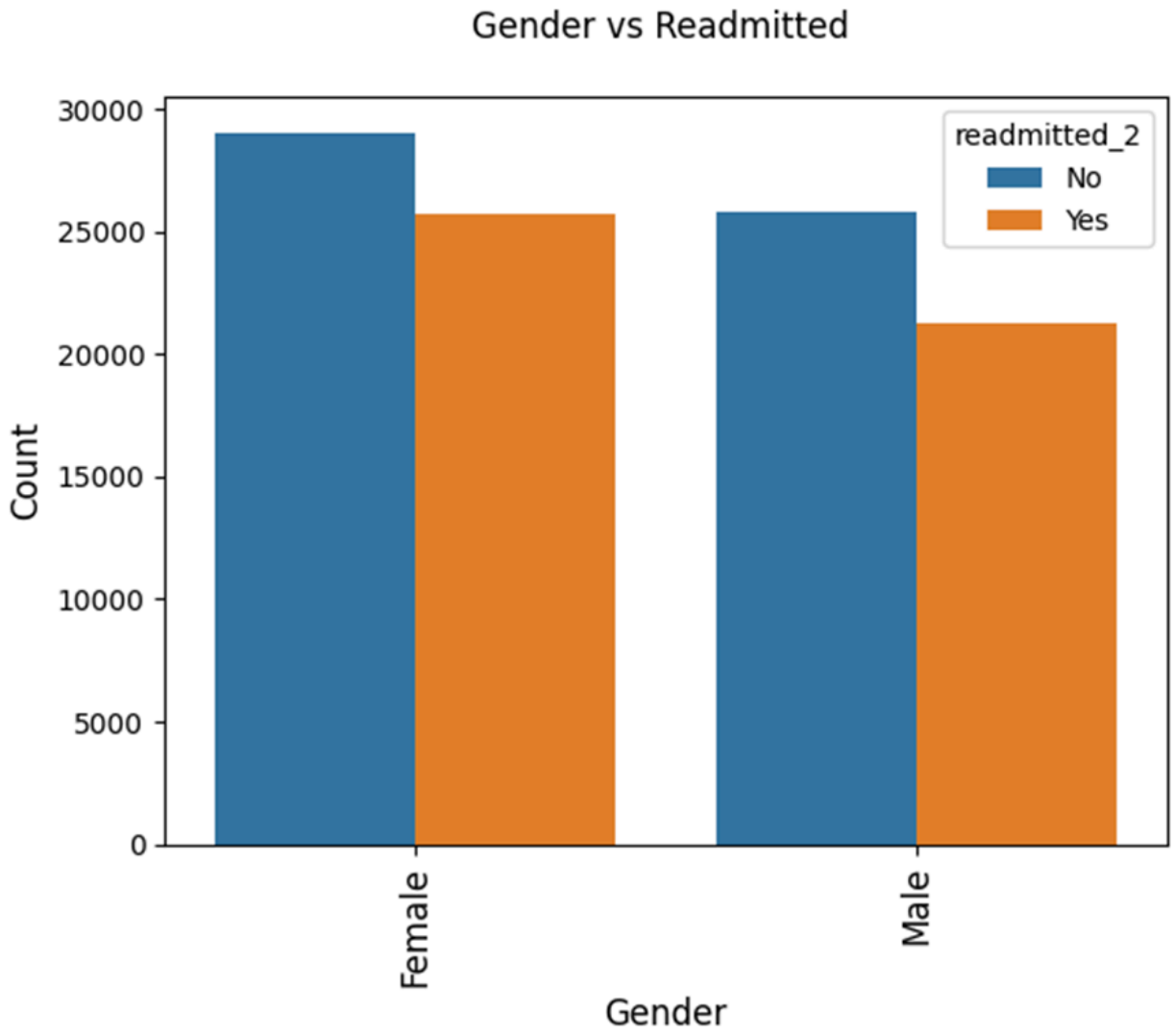
Gender Value Counts

```
Female              54708
Male                47055
Unknown/Invalid         3
Name: gender, dtype: int64
```

Note. The Above Figure Shows That There Are 3 Unknown/Invalid Gender Types

Figure 33

Gender Vs Readmitted



Note. The Above Figure Illustrates The Distribution Of Readmission Status Among Different Genders In The Dataset.

To deal with missing data in the US Healthcare Readmissions and Mortality dataset from Kaggle (The Devastator, 2023), rows in the hospital Data Frame's score column that have "Not Available" values in them should be removed as shown in Figure 34 and Figure 35. This method preserves data integrity and guards against potential biases by ensuring that only accurate and

comprehensive data is used for analysis. Furthermore, by guaranteeing data type integrity, transforming certain columns to numeric types with pd.to_numeric() makes additional analysis easier.

Figure 34

Analysis Of Health Care Providers

```
Health care providers:  64764
Rows:  4626
```

Note. The Dataset Comprises 64764 Health Care Providers, With 4626 Unique Provider Ids.

Figure 35

Analysis Of Health Care Providers After Removing Values

```
Health care providers:  41785
Rows:  4423
```

Note: After removing rows with unavailable scores, The dataset comprises 41785 health care providers, with 4423 unique provider IDs.

Handling Inconsistencies and Noisy Data. The dataset from the Health Facts database (Cerner Corporation, Kansas City, MO) underwent a comprehensive review to identify inconsistencies and noisy data that could impact analytical results. Data validation techniques were applied to detect outliers, especially in numerical attributes where values outside acceptable ranges were considered potentially erroneous and subsequently excluded. To mitigate the effects of outliers, entries in the 'time_in_hospital' attribute with durations beyond a reasonable range (e.g., 0 to 30 days) were removed from the dataset. Additionally, efforts were made to address noisy data, particularly in categorical attributes such as 'race', through standardization processes. This involved ensuring uniform formatting, such as converting all text to lowercase, to eliminate

variations and discrepancies resulting from data entry errors. The meticulous correction of missing data, inconsistencies, and noisy entries refined the dataset, making it suitable for further analysis and modeling and laying the foundation for generating valuable insights and forecasts.

Further, by standardizing categorical variables across the dataset, any inconsistencies within those values were rectified. For instance, phrases like "No Different than the National Rate" were simplified to "No Different," with similar adjustments made for "Worse than the National Rate" and "Better than the National Rate." This standardization ensures that variations in category representations do not influence modeling and subsequent analyses, thereby enhancing data quality. This process also facilitates the management and interpretation of the data by simplifying cross-category comparisons. Moreover, non-numeric characters in payment data are modified prior to conversion to floating-point format, ensuring consistency in data representation and supporting accurate quantitative analysis.

Data Transformation

The data transformation stage is essential in the Hospital Readmission Prediction System project, ensuring that the complex dataset on hospital readmissions is optimally prepared for analysis using a Large Language Model (LLM). Initially, the data is cleansed to remove any inaccuracies or inconsistencies, establishing a reliable foundation for analysis. All numerical data are standardized to manage inconsistencies in scale, ensuring uniform processing across all numerical inputs. Categorical variables are transformed using binary and one-hot encoding methods, making them suitable for effective modeling. Techniques like Principal Component Analysis (PCA) are applied to decrease the dataset's dimensionality, focusing on the most significant features while reducing computational load. The target variable, which classifies readmission status, is simplified from multiple classes to a binary format, reducing the model's

complexity. Age data is also converted from categorical to numerical format, allowing for finer quantitative evaluations. To enhance model efficiency and uniformity, automated feature engineering tools such as Feature Tools are deployed to identify and select predictive features crucial for assessing patient readmission risks. This phase is vital for refining the data, thus enhancing the model's performance and streamlining computational efforts while preserving essential information.

One-hot Encoding. In the data preparation phase of the Hospital Readmission Prediction System project, categorical variables were transformed to suit the analysis requirements of machine learning algorithms using binary and one-hot encoding techniques. Binary encoding was applied to variables such as change, diabetesMed, and readmitted_2. This method converts categorical labels into a binary format, where '1' represents the presence of a feature and '0' its absence, simplifying the data and aiding algorithms in identifying patterns linked to diabetic medication management and hospital readmissions, as shown in Figure 56 which presents the original data. Additionally, the insulin variable was subjected to one-hot encoding, creating separate columns for each insulin dosage category (insulin_No, insulin_Up, insulin_Steady). Each column is labeled with Boolean values to indicate whether a specific insulin regimen was followed. This restructuring makes the data more understandable for Large Language Models (LLMs), thereby enhancing the model’s ability to process and analyze data based on insulin administration patterns, as illustrated in Figure 37. Such transformations are crucial for developing an effective model that can accurately predict patient readmissions, thereby improving patient care management and outcomes. The restructuring also enhances the precision of predictive analytics and supports the development of a sophisticated system capable of efficiently predicting hospital readmission rates.

Figure 36

Original Data

id	time_in_hospital	num_lab_procedures	...	pioglitazone	rosiglitazone	acarbose	miglitol	insulin	glyburide-metformin	change	diabetesMed	readmitted	readmitted_2
7	3	59	...	No	No	No	No	Up	No	Ch	Yes	>30	Yes
7	2	11	...	No	No	No	No	No	No	No	Yes	NO	No
7	2	44	...	No	No	No	No	Up	No	Ch	Yes	NO	No
7	1	51	...	No	No	No	No	Steady	No	Ch	Yes	NO	No
2	3	31	...	No	No	No	No	Steady	No	No	Yes	>30	Yes

Note. The Above Figure Shows The Head() Of The Data Set Before Transforming

Figure 37

Transformed Data

```
Data after One-Hot Encoding (Dense Representation):
[[0. 0. 1. 0. 0. 1. 0. 0. 0. 1. 0. 0. 0. 1. 0. 0. 1. 0. 0. 0. 1. 0. 0. 0.
  1. 0. 0. 0. 1. 0. 0. 0. 1. 0. 0. 0. 1. 0. 0. 0. 1. 0. 0. 0. 1. 0. 0. 0.
  1. 0. 0. 0. 1. 0. 0. 0. 1. 0. 0. 0. 0. 0. 1. 0. 1. 0. 0. 1. 0. 0. 1.]
 [1. 0. 0. 0. 0. 1. 0. 0. 0. 1. 0. 0. 0. 1. 0. 0. 1. 0. 0. 0. 1. 0. 0. 0.
  1. 0. 0. 0. 1. 0. 0. 0. 1. 0. 0. 0. 0. 1. 0. 0. 1. 0. 0. 0. 1. 0. 0. 0.
  1. 0. 0. 0. 1. 0. 0. 0. 1. 0. 0. 0. 1. 0. 0. 0. 1. 0. 0. 0. 1. 0. 1.]
 [0. 0. 1. 0. 0. 0. 1. 0. 0. 1. 0. 0. 0. 1. 0. 0. 1. 0. 0. 0. 1. 0. 0. 0.
  1. 0. 0. 0. 1. 0. 0. 0. 1. 0. 0. 0. 1. 0. 0. 0. 1. 0. 0. 0. 1. 0. 0. 0.
  1. 0. 0. 0. 1. 0. 0. 0. 1. 0. 0. 0. 0. 0. 1. 0. 1. 0. 0. 1. 0. 0. 1.]
 [0. 0. 1. 0. 0. 0. 1. 0. 0. 1. 0. 0. 0. 1. 0. 0. 1. 0. 0. 0. 1. 0. 0. 0.
  1. 0. 0. 0. 1. 0. 0. 0. 1. 0. 0. 0. 0. 1. 0. 0. 1. 0. 0. 0. 1. 0. 0. 0.
  1. 0. 0. 0. 1. 0. 0. 0. 1. 0. 0. 0. 0. 1. 0. 0. 1. 0. 0. 1. 0. 0. 1.]
 [0. 0. 1. 0. 0. 0. 1. 0. 0. 1. 0. 0. 0. 1. 0. 0. 1. 0. 0. 0. 1. 0. 0. 0.
  1. 0. 0. 0. 1. 0. 0. 0. 1. 0. 0. 0. 1. 0. 0. 0. 1. 0. 0. 0. 1. 0. 0. 0.
  1. 0. 0. 0. 1. 0. 0. 0. 1. 0. 0. 0. 0. 1. 0. 0. 1. 0. 0. 0. 1. 0. 1.]]
```

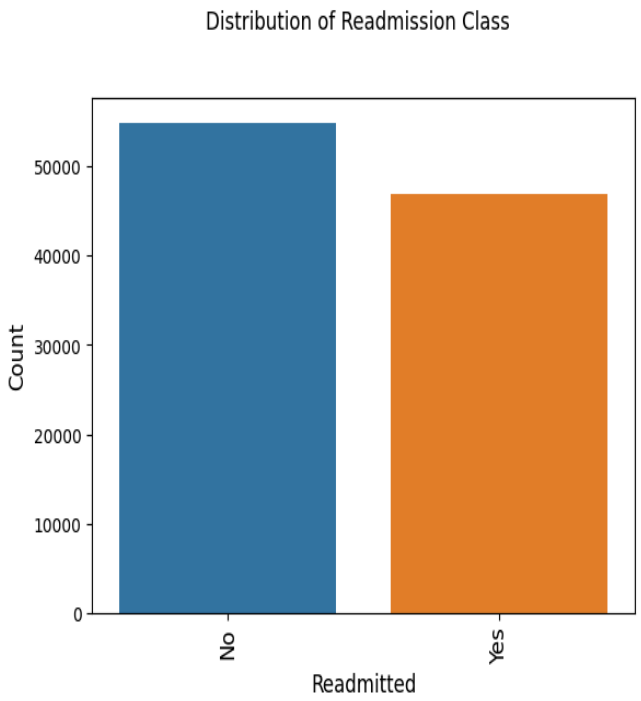
Note. The Above Figure Shows The Transformed Data After Applying One-Hot Encoding.

Encoding the outcome variable. Transforming the 'Readmission' variable was a key step in preparing the data for the Hospital Readmission Prediction System project. The 'Readmission' column initially had three categories: '<30' for patients readmitted within 30 days, '>30' for those readmitted after 30 days, and 'No Readmission' for those who were not readmitted. To simplify the modeling process, the '>30' and 'No Readmission' categories were combined into a single 'Not Readmitted' category. This merger was performed using the pandas' `replace` function, where both '>30' and 'No Readmission' were transformed into 'Not Readmitted' in the dataset.

The transformation reduced the 'Readmission' column to two categories: '<30' and 'Not Readmitted'. This change is visualized in the updated dataset representation, where the data is now divided into a binary variable format, clearly distinguishing early readmissions from all other cases, as illustrated in Figure 38. Such a binary classification significantly improves analysis efficiency, especially when employing machine learning algorithms that perform optimally with binary outcomes. By reducing the complexity of the outcome variable, the model is better positioned to focus on identifying patterns that indicate early hospital readmissions, enhancing its predictive accuracy and reliability in forecasting patient readmission risks.

Figure 38

Transformed Data After Encoding The Outcome Variable

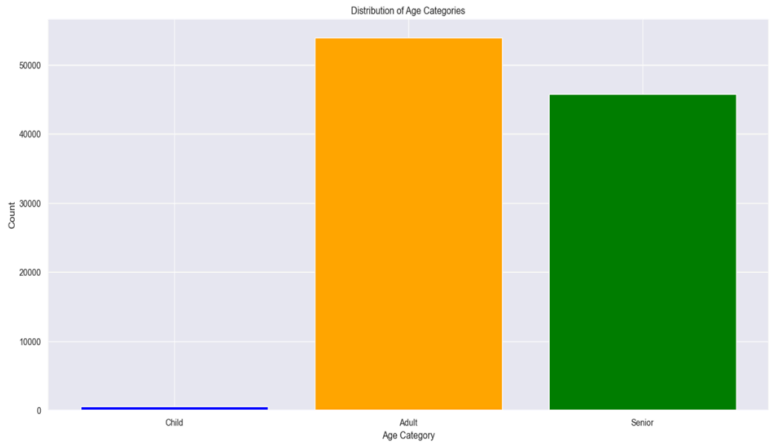


Note. The Above Figure Shows The Transformed Data After Encoding The Outcome Variable.

Feature Engineering. In refining the dataset for analysis, the age variable was initially transformed from a categorical format based on age ranges to a numerical format. This involved extracting the numerical values from the age ranges and converting them to integers, allowing for more accurate analysis of age as a factor. Subsequently, binary variables were developed to differentiate between three specific age groups: 'children' for individuals under 18, 'adults' for those between 18 and 65, and 'seniors' for individuals aged 65 and above. These categories were encoded such that a '1' represents an individual belonging to a particular age group, and '0' indicates otherwise. This categorical breakdown establishes clear divisions within the dataset, streamlining the analysis. The bar plot in Figure 39 illustrates the distribution across these age groups, with blue, orange, and green representing children, adults, and seniors, respectively. The visualization shows that the majority of the dataset consists of adults and seniors, with a smaller representation of children. This distribution is important as age is a key variable in many studies, particularly in healthcare. The chart offers an immediate visual insight into the age composition of the data, underscoring the importance of these demographic factors in subsequent analyses.

Figure 39

Transformed Data After Feature Engineering

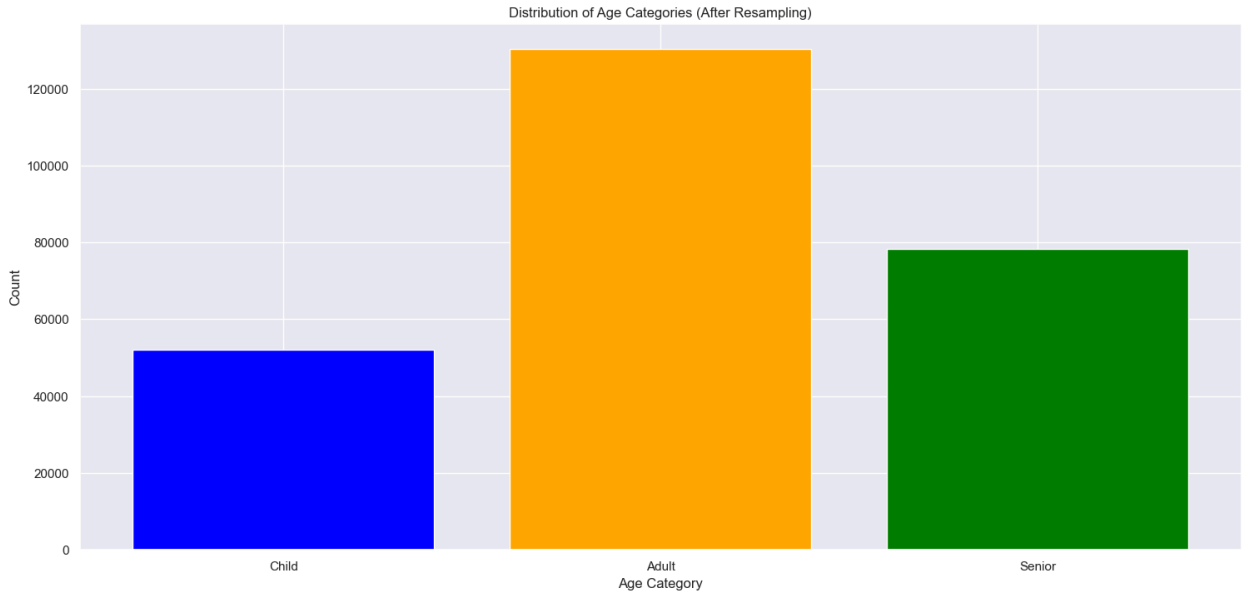


Note. The Above Figure Shows The Transformed Data After Feature Engineering.

The above Figure 39 depicts that the Child category values are very less compared to Adult and Senior. The Random Oversampling method from the imbalanced-learn module is used to handle the class imbalance problem in the dataset. In order to achieve balance in the class distribution between the majority classes ("Adult" and "Senior") and the minority class ("Child" category), this strategy replicates instances from the minority class at random. A distribution plot as shown in Figure 40 is created to show the balance attained following resampling and recalculating the counts for each age group. Ensuring a more equitable representation of age groups within the dataset will provide the model with an improved chance of assimilating the unique attributes and trends linked to every age group, including youngsters. Rather than preferring the majority classes, this method assures that the model performs well across all age groups and reduces the possibility of biased predictions.

Figure 40

Distribution Of Age Categories After Resampling



Note. The Above Figure Shows The Transformed Data After Resampling the Child categoty.

Standardization of Numerical Data. The dataset underwent standardization to ensure that all numerical data operate on a consistent scale, addressing scale inconsistencies across different metrics. This process involves adjusting the values so that the dataset features have a mean of zero and a standard deviation of one. As shown in the provided data snapshot, each feature's value has been transformed to reflect this standardization, facilitating more effective analysis by removing the influence of differing scales. This standardization is crucial Figure 41 or algorithms that are sensitive to the scale of input data, such as many machine learning models which assume data is normally distributed. The transformed data ensures that each feature contributes equally to the analysis, preventing any one feature with larger scale from dominating. This approach not only improves the performance of predictive models but also enhances the reliability of the results obtained from subsequent analyses.

Figure 41

Transformed Data after Feature Scaling

```
Data after Feature Scaling:
[[-3.29012721 -0.47482174  0.80370808 -0.78920088  0.2318987   0.81704169
  13.66445127 -1.07925836 -0.91667738]
 [-2.64848324 -0.80913096 -1.633924    2.1332546  -0.38413752 -0.81383658
  -0.07318259  0.9265622  -0.91667738]
 [-2.00683927 -0.80913096  0.04194806 -0.20470978 -0.01451579 -0.27021049
  -0.07318259  0.9265622  -0.91667738]
 [-1.3651953  -1.14344018  0.39743607 -0.78920088 -1.00017374 -1.35746268
  -0.07318259  0.9265622  -0.91667738]
 [-0.72355133 -0.47482174 -0.61824396  2.7177457  -0.01451579  0.81704169
  -0.07318259  0.9265622  -0.91667738]]
```

Note. The Above Figure Shows The Transformed Data After Feature Scaling.

Dimensionality Reduction using PCA. The dataset was processed using Principal Component Analysis (PCA) to condense its structure, concentrating on the most essential features while eliminating the less important ones. This process is illustrated in Figure 42, where the original variables of the dataset are transformed into a new, more compact dimensional space. Each row in the snapshot represents a version of the original data that has been condensed to

capture the core information with fewer dimensions. The application of PCA is aimed at improving the efficiency of the modeling process by simplifying the data structure and lowering computational requirements. By reducing the feature set, PCA focuses attention on the most significant attributes, thereby enhancing the effectiveness of machine learning algorithms. This method is particularly advantageous when dealing with datasets that have a large number of dimensions, which can lead to problems such as overfitting and increased computational burdens. The streamlined dataset not only accelerates the analytical process but also preserves the data's essential qualities by retaining the most impactful features.

Figure 42

Transformed Data after Dimensionality Reduction using PCA

```
Data after PCA Transformation:
[[ 13.76589041  51.85276638]
 [-34.02726097  39.95180388]
 [ -0.59880173  31.32104432]
 [  5.58341479  21.73658089]
 [-12.53147251  10.75123418]]
```

Note. The Above Figure Shows The Transformed Data After PCA.

Data Preparation

In the study on predicting hospital readmissions, two expansive datasets were meticulously curated. The initial dataset encompassed a decade's worth of patient records from a network of U.S. hospitals, while the second dataset provided detailed accounts of patient demographics, health outcomes, and readmission instances. The data underwent thorough cleaning to rectify any missing or inconsistent information. This process included using statistical methods to impute missing values and ensure uniformity across various data categories. During the feature engineering phase, meaningful variables were extracted to illuminate patterns relevant to readmissions. This involved creating new variables and indicators

to deepen the understanding of the data. To maintain consistency across the numerical data, normalization and standardization techniques were applied, preventing any single feature from disproportionately influencing the model due to its magnitude. The dataset was then strategically divided into training, validation, and testing segments following an 80-10-10 distribution. This division enabled comprehensive training of the model and rigorous evaluation of its predictive performance. The entire process was geared towards ensuring the data was clean, well-structured, and enriched with insights, providing a robust foundation for subsequent analysis.

Data Statistics

The data preparation process begins with the acquisition of raw data from two primary sources. The first dataset, sourced from Kaggle, offers extensive patient information covering various demographics and medical interactions. The second dataset, obtained from the Health Facts database, includes a decade of clinical data from 130 US hospitals. Both datasets are rich in detail but contain numerous inconsistencies and missing values that require thorough scrutiny and correction.

In the pre-processed data stage, exploratory data analysis (EDA) is conducted to identify and rectify outliers, missing values, and other anomalies that could compromise data integrity. This is followed by extensive data cleaning efforts aimed at resolving issues related to incomplete, inconsistent, and noisy data, ensuring the quality and reliability of the datasets. As the process transitions to the transformed data phase, the focus shifts to making the data conducive for modeling. This involves normalizing and standardizing numerical features to eliminate scale-induced biases and encoding categorical variables to facilitate computational

processing. Additionally, dimensionality reduction techniques are employed to simplify the data structure, thereby enhancing model efficiency and accuracy.

The final phase, the prepared dataset, involves sophisticated feature engineering to develop new variables that more accurately reflect the patterns associated with hospital readmissions. The dataset is strategically divided into training, validation, and testing segments, following an 80-10-10 split to ensure comprehensive training and rigorous performance evaluation. The culmination of these stages is demonstrated through detailed statistical visualizations that showcase the evolution of the dataset as highlighted in Table 11, which illustrates changes in class distribution, data balance, and reductions in dataset size after processing. These visualizations provide a clear and concise representation of the data's transformation from raw inputs to an analytically robust format, ready for effective model training and validation. This systematic preparation emphasizes precision and rigor in modeling, establishing a robust foundation for subsequent analyses and insights.

Table 11

Data Statistics

Stage	Step	Applicable Statistics
Raw Data	Kaggle (The Devastator, 2023) and diabetics dataset from UC Irvine machine learning repository	19 MB and 2 MB
Sampled Data	Manual Sampling of Excel Data	19MB of CSV file with 101766 instances 2 MB with instances of 64764

Stage	Step	Applicable Statistics
	Inconsistent Data Handling	101766 in diabetes dataset and 64764 in US healthcare and mortality rate dataset
	Data Imputation	101766 in diabetes dataset and 64764 in US healthcare and mortality rate dataset
Data transformation	One hot encoding	Insulin variable of 101766 instances
	Data Encoding	Readmission of 101766
	Feature Engineering	Age of each range to single variables of 101766
	Dimensionality Reduction using PCA	Entire dataset
	Standardization of Numerical Data	'age', 'time_in_hospital', 'num_lab_procedures', 'num_procedures', 'num_medications', 'number_diagnoses', 'is_child', 'is_adult', 'is_senior'
Preparation	Train	80%
	Validate	10%
	Test	10%

Note. The Above Table Shows The Statistical Data Of The Dataset Used For The Project.

Following the transformation and cleaning procedure, the columns for the Diabetic and US Healthcare and Mortality Datasets were modified. As can be seen in Figure 43, Figure 44, Figure 45 and Figure 46, the dataset's shape is depicted both before and after these procedures are applied. There are 38 columns total in diabetic and 15 columns in US Healthcare and Mortality Datasets.

Figure 43

Shape Of The Dataset

```
The shape of the Dataset is : (101766, 50) with 101766 records and 50 columns
```

Note. The Diabetes Dataset Contains 101766 Rows And 50 Columns

Figure 44

Shape Of The Dataset After Applying Cleaning And Transformations

```
(100241, 35)
```

Note. After Applying All The Transformation Techniques And Cleaning The Data , 100241 Rows And 35 Columns Were Obtained.

Figure 45

Shape of The US Healthcare and Mortality Dataset

```
(64764, 20)
```

Note. The US Healthcare and Mortality Dataset contains 64764 rows and 20 columns

Figure 46

Shape Of The Us Healthcare And Mortality Dataset After Applying Cleaning And Transformation

```
(41785, 15)
```

Note. After Applying All The Transformation Techniques And Cleaning The Data , 41785Rows And 15 Columns Were Obtained.

Model Development

Model Proposals

To achieve the best model selection for predicting hospital readmissions, a robust approach is necessary to effectively handle complicated and diverse tabular data from .csv files. The optimal model must proficiently handle large datasets, examine patient records across several dimensions, and generate very precise predictions regarding the probability of readmission. Four advanced architecture models, namely ClinicalBERT, Medpalm-2, llama2, and clinicalXLNET, are presented to fulfill the given objectives. Additionally, the integration of a customized AI model called Mistral AI is suggested. All proposed models utilize sophisticated language model architectures specifically engineered to process medical data.

The thorough examination and manipulation of these models depend on a profound comprehension of natural language processing as it is applied to the medical domain, fine-tuning of the models, and the particular difficulties connected with hospital readmission data. Methods such as addressing imbalanced classes, managing missing values, and encoding categorical variables are essential. These specialized models are specifically developed to comprehend and analyze medical language. They are capable of efficiently integrating both organized and unstructured data from patient records. The objective of this strategy is to utilize the capabilities

of medical-specific language comprehension in order to enhance the precision and dependability of readmission predictions.

ClinicalBERT. The Figure 47 pseudocode presents a series of instructions for utilizing the ClinicalBERT architecture to forecast hospital readmissions based on patient data stored in a .csv file. During the initial phase of data import, the system loads the .csv file, which typically contains organized clinical information like demographics, past admissions, and perhaps unstructured data like doctor notes. Next, data preprocessing takes place, during which the raw data is purified and structured to ensure compatibility with machine learning procedures. This entails resolving the issue of missing or duplicated values, identifying pertinent factors that may impact the risk of readmission, and transforming natural language text into a tokenized format that can be understood by ClinicalBERT. Feature extraction is an essential phase in this process, since it includes analyzing the several forms of data to uncover significant patterns and signs of readmission.

After the preprocessing stage is finished, the ClinicalBERT model is implemented. This model has undergone prior training using substantial medical datasets, enabling it to comprehend the nuances of clinical language. The training method consists of repeatedly inputting the tokenized data into the model, which then converts it into numerical embeddings. The embeddings traverse through numerous layers of the transformer neural network, which is the basic architecture of ClinicalBERT. This network is capable of identifying intricate correlations within the data. The results generated by ClinicalBERT are further utilized by a dedicated classification layer, specifically incorporated for the purpose of readmission prediction. This layer calculates the probability of a patient being readmitted at different time intervals throughout their hospitalization. The model undergoes refinement during multiple epochs, with

each iteration involving the adjustment of internal parameters to minimize predicted errors. The model's performance is consistently assessed by employing a distinct validation dataset to confirm its accurate prediction of readmission, while avoiding overfitting to the training data. Ultimately, the refined ClinicalBERT model is implemented in a clinical environment. During this operational phase, the model analyzes fresh patient data by implementing the identical preprocessing and tokenization procedures to produce predictions on the likelihood of readmission. Healthcare practitioners can utilize these predictions to identify patients who are more likely to be readmitted and customize their care to minimize these risks. The outputs of the model at various time points provide real-time insights about a patient's likelihood of being readmitted throughout their hospitalization, enabling prompt interventions. The pseudocode demonstrates the advanced utilization of natural language processing and machine learning to address a critical healthcare issue, showcasing the potential of AI to assist in decision-making within clinical settings.

Figure 47

Pseudocode Of ClinicalBert

```
// Pseudocode for ClinicalBERT architecture to predict hospital readmission from .csv file input

// Step 1: Data Ingestion
Load CSV file containing patient data including clinical notes and various healthcare interactions

// Step 2: Data Preprocessing
FOR EACH record in the CSV file DO
   Clean missing values and handle duplicates
   Extract features from clinical notes and other structured data
   Tokenize textual data into a format suitable for ClinicalBERT
END FOR
// Step 3: Load the ClinicalBERT Model
Initialize ClinicalBERT model with pre-trained weights
Prepare the model for training with a classification layer for readmission prediction

// Step 4: Training Process
FOR EACH epoch in training epochs DO
   FOR EACH batch in training data DO
      Pass tokenized input through ClinicalBERT to get embeddings
      Process embeddings through ClinicalBERT transformer blocks
      Apply classification layer to output readmission probability
      Calculate loss function and perform backpropagation
      Update ClinicalBERT weights and classification layer parameters
   END FOR
   Evaluate model on validation set to monitor performance
END FOR

// Step 5: Model Evaluation
Evaluate the trained model on a separate validation dataset
Optimize model parameters to improve prediction accuracy

// Step 6: Deployment
Deploy the trained and optimized ClinicalBERT model for operational use

// Step 7: Making Predictions
FOR EACH new patient record DO
   Preprocess data and tokenize as per model requirements
   Input data through ClinicalBERT to receive readmission probability
   Output readmission probability and store/log the result
END FOR
```

Note. The Figure Above Shows In Detail How The .CSV was Loaded, Processed , Trained Using ClinicalBERT Model In Each Step.

ClinicalXLNet. ClinicalXLNet is an effective tool for predicting hospital readmissions using patient data. As shown in Figure 48 the process begins with importing data from a.csv file that contains structured clinical information such as demographics, previous admissions, and maybe unstructured data such as doctor notes. Preprocessing follows, which ensures that the data is pure and compatible with machine learning algorithms. This includes processing missing or

duplicate values, finding key attributes that influence readmission risk, and converting text into a tokenized format suited for ClinicalXLNet.

A critical stage is feature extraction, which entails examining numerous data formats to identify patterns that indicate readmission risk. ClinicalXLNet is installed after the preprocessing is complete. It is trained on large medical datasets and understands clinical nuances. The model turns tokenized data into numerical embeddings by passing through layers of ClinicalXLNet's transformer neural network architecture, which detects detailed data association. The results are fed into a specific classification layer for readmission prediction, which is refined across several epochs to minimize mistakes.

To achieve accurate predictions without overfitting, performance is evaluated on a separate validation dataset. In the operational phase, the enhanced ClinicalXLNet examines new patient data using preprocessing and tokenization to predict readmission likelihood. Healthcare practitioners use these findings to tailor patient care and reduce readmission risk. Real-time predictions throughout the hospitalization enable timely actions. This pseudocode demonstrates AI's potential in clinical decision-making by utilizing advanced natural language processing and machine learning with ClinicalXLNet.

Figure 48

Pseudocode Of ClinicalXLNet

```
// Pseudocode for ClinicalXLNet architecture to predict hospital readmission from .csv file input

// Step 1: Data Ingestion
LOAD CSV file containing patient data including clinical notes and various healthcare interactions

// Step 2: Data Preprocessing
FOR EACH record in the CSV file DO
   Clean missing values and handle duplicates
   Extract features from clinical notes and other structured data
   Tokenize textual data into a format suitable for ClinicalXLNet
END FOR

// Step 3: Load the ClinicalXLNet Model
Initialize ClinicalXLNet model with pre-trained weights
FOR EACH tokenized input DO
   Pass tokenized input through ClinicalXLNet to get embeddings \( E_i \)
END FOR

// Step 4: Process with Bidirectional LSTM
Initialize Bidirectional LSTM
FOR EACH embedding \( E_i \) DO
   Pass embeddings through Bidirectional LSTM to capture sequential information \( H_{xl} \)
END FOR

// Step 5: Prediction Layer
Initialize Predictor Neural Network
FOR EACH LSTM output \( H_{xl} \) DO
   Pass through Predictor Neural Network to output readmission probability \( P_r \)
END FOR

// Step 6: Training Process
FOR EACH epoch in training epochs DO
   FOR EACH batch in training data DO
      Calculate loss function using \( P_r \) and actual labels
      Perform backpropagation to update ClinicalXLNet, LSTM, and Predictor Neural Network weights
   END FOR
   Evaluate model on validation set to monitor performance
END FOR

// Step 7: Model Evaluation
Evaluate the trained model on a separate validation dataset
Optimize model parameters to improve prediction accuracy

// Step 8: Deployment
Deploy the trained and optimized ClinicalXLNet model for operational use

// Step 9: Making Predictions
FOR EACH new patient record DO
```

Note. The Figure Above Shows In Detail How The .CSV Was Loaded, Processed , Trained Using Clinical Xlnet Model In Each Step.

Mistral AI. In healthcare, managing hospital readmissions effectively is especially important for patients with chronic conditions like diabetes, as these readmissions can signify

gaps in care and impose significant costs. Utilizing advanced technologies, such as the hypothetical Mistral AI, could significantly improve predictions of such events and enhance patient outcomes. Mistral AI, designed around the transformer architecture, is adept at handling the large, complex datasets typically found in electronic health records (EHRs). This capacity is essential for analyzing the extensive documents these records often comprise and is critical for the model's success in various medical applications.

The envisioned deployment of Mistral AI involves a meticulous process as shown in Figure 49. Initially, the model would be fine-tuned on targeted datasets specific to the medical field, enhancing its ability to predict patient readmission risks accurately. After this fine-tuning, the model would be integrated into hospital IT systems to provide real-time insights, a step that requires ongoing testing to ensure accuracy and reliability.

Moreover, the implementation of such a model requires continuous refinement and monitoring to adapt to new data and maintain effectiveness in a dynamic healthcare environment. Protecting patient data and ensuring the model's predictions are understandable to medical professionals are paramount, ensuring that the model's use is both ethical and practical. Research such as that by Tsai, Huang, and Kuo (2024) highlights the potential for such technology to streamline processes like literature reviews, suggesting that while models like Mistral AI can outperform humans in efficiency, they often lack depth, indicating the value of a hybrid approach. Additionally, developments like BioMistral, which Labrak et al. (2024) discuss, demonstrate the adaptability of such models to the nuances of the medical domain.

Figure 49

Pseudocode Of Mistral AI

```
# Pseudocode for a system that processes multimedia data to generate semantic metadata

# Step 1: Process multimedia data to extract features and semantic concepts
def process_multimedia_data(multimedia_data):
    speech_features = extract_speech_features(multimedia_data['speech'])
    video_features = extract_video_features(multimedia_data['video'])
    text_features = extract_text_features(multimedia_data['text'])
    additional_features = extract_additional_features(multimedia_data['additional'])
    return {
        'speech_features': speech_features,
        'video_features': video_features,
        'text_features': text_features,
        'additional_features': additional_features
    }

# Step 2: Integrate extracted features to form a comprehensive metadata profile
def integrate_features(features):
    integrated_metadata = integrate(features['speech_features'], features['video_features'],
                                    features['text_features'], features['additional_features'])
    return integrated_metadata

# Step 3: Generate semantically rich metadata from integrated data
def generate_semantic_metadata(integrated_metadata):
    semantically_rich_metadata = enrich_metadata(integrated_metadata)
    return semantically_rich_metadata

# Step 4: Use the semantically rich metadata in various semantic applications
def use_semantic_metadata(semantic_metadata, applications):
    for application in applications:
        result = application.use_metadata(semantic_metadata)
        visualize(result)

# Main execution
if __name__ == '__main__':
    multimedia_data = load_multimedia_data()
    features = process_multimedia_data(multimedia_data)
    integrated_metadata = integrate_features(features)
    semantic_metadata = generate_semantic_metadata(integrated_metadata)
    applications = load_semantic_applications()
    use_semantic_metadata(semantic_metadata, applications)
```

Note. The Figure Above Shows In Detail How The .CSV was Loaded Processed , Trained Using Mistral AI Model In Each Step.

Llama2. The Llama2 model is an innovative solution designed to predict hospital readmissions, leveraging the power of advanced natural language processing. It targets the complexities of chronic diseases like diabetes by meticulously analyzing electronic health records (EHRs). This model is a result of a carefully planned pipeline, beginning with the initialization of a robust computational environment. In the figure, essential libraries are loaded, and GPU configurations are set to support the heavy demands of medical data processing. In the data preparation stage, EHRs are imported and thoroughly preprocessed. This involves cleaning

the data to remove any duplicates and handling missing values, a vital step in ensuring the accuracy of the model's predictions as shown in Figure 50. Numerical data is normalized or standardized, and categorical variables are encoded to create a uniform dataset. NLP techniques are employed to tokenize clinical notes, extracting key features such as medical history, patterns of medication adherence, and symptoms, all of which are indicative of readmission risks.

The architecture of Llama2 is composed of multiple Transformer blocks, each tailored to process medical text and discern subtle indicators of patient health trajectory. The model is fine-tuned with a focus on these features, employing algorithms such as gradient descent and backpropagation to optimize its predictive performance. Given the common challenge of imbalanced datasets in the medical field, Llama2 incorporates a weighted loss function to ensure sensitivity to rare but critical outcomes. The pseudocode for the model underscores its systematic approach to training, which includes updating weights and parameters through backpropagation—a calculation where the gradient of the loss function is propagated back through the network to update the weights, following the Equation $\boldsymbol{W_{new} = W_{old} - \alpha \cdot \partial_{Loss} / \partial_{W}}$

Where W_{new} and W_{old} are the updated and previous weights, α is the learning rate, and $\partial_{Loss} / \partial_{W}$ represents the gradient of the loss function with respect to the weights.

Post-training, Llama2's performance is rigorously evaluated against a validation dataset using metrics such as accuracy, precision, recall, and the ROC-AUC curve. The deployment phase sees the model integrated into the hospital’s IT system, where it begins offering real-time

predictions on readmission risks. Continuous monitoring and periodic updates are critical in this phase, ensuring the model remains accurate and reliable.

This practical application of Llama2 underscores the ethical and operational considerations inherent in deploying AI in healthcare. By safeguarding patient data and ensuring predictions are interpretable for healthcare professionals, the model aligns with the industry's ethical standards. The integration of Llama2 not only exemplifies technical excellence but also highlights the transformative impact of AI technologies in enhancing healthcare delivery for chronic disease management.

Figure 50

Pseudocode Of Llama-2

```
Pseudocode: Llama2 Model for Predicting Hospital Readmissions

//step 1. Initialize Environment
   Load necessary libraries (e.g., TensorFlow, PyTorch)
   Set up environment settings (e.g., GPU configuration)

//step 2. Data Preparation
   Load Data
     -Import Electronic Health Records (EHR) data from the hospital database
   Preprocess Data
     - Clean data (remove duplicates, handle missing values)
     - Normalize/standardize numerical data
     - Encode categorical variables
     - Extract relevant features from text data using NLP techniques
   Split Data
     - Divide the data into training, validation, and test sets

// step 3. Model Configuration
   Define Llama2 Model Architecture
     - Configure model with appropriate number of layers and parameters
     - Customize the transformer blocks as required for the medical domain
   Set Hyperparameters
     - Define learning rate, batch size, number of epochs
     - Set up optimizers and loss functions (e.g., AdamW optimizer, cross-entropy loss)

// step 4. Model Training
   Load Llama2 pre-trained model
   Fine-tune on diabetic patient EHR data
     - Train model using training data
     - Validate model using validation data to monitor performance and adjust hyperparameters
   Apply techniques to handle imbalanced data (e.g., weighted loss functions)

// step 5. Model Evaluation
   Evaluate model on test dataset
     - Calculate performance metrics (e.g., accuracy, precision, recall, ROC-AUC)
   Perform error analysis
     - Analyze cases of mispredictions and refine features or model parameters

// step 6. Model Deployment
   Deploy model into production
     - Integrate model into hospital’s IT system for real-time prediction
   Monitor Model
     - Continuously check model performance
     - Update model periodically with new data or retrain if performance degrades

// step 7. Reporting
   Generate reports on model performance and impact on readmission rates
   Provide insights and recommendations to healthcare staff for patient management

// step 8. End
```

Note. The Figure Above Shows In Detail How The .CSV Is Loaded, Processed, Trained, And Deployed Using The Llama-2 Model.

ChatGPT. Chronic illnesses like diabetes, heart disease, and respiratory issues make hospital readmissions a major healthcare management challenge. Readmissions are costly and may indicate patient care and management issues. Predictive analytics using large language models (LLMs) like ChatGPT can enhance readmission risk identification. Hospitals can improve predictive accuracy by integrating ChatGPT with hospital IT systems to examine vast EHRs. This procedure requires careful data integration and preparation to protect patient privacy and health standards. ChatGPT will fine-tune datasets based on the hospital's usual patient demographics and common readmission circumstances after data preparation. The model should recognize linguistic and clinical markers of higher readmission chances. ChatGPT will work in real time after training, giving healthcare professionals with relevant insights and alerts regarding high-risk patients, enabling early and successful treatments as shown in Figure 51.

The deployment of ChatGPT in predicting hospital readmissions promises not just better patient outcomes through personalized care plans but also a reduction in operational costs by minimizing unnecessary readmissions. The system’s capability to continuously learn from new patient data ensures that the model remains accurate and relevant in a dynamic healthcare environment. Furthermore, the outputs provided by ChatGPT will be designed to be easily interpretable by medical staff, ensuring that the predictive insights are used effectively in clinical decision-making. Alongside these technological and clinical benefits, ethical considerations are paramount; the system will adhere strictly to data privacy laws such as HIPAA and include robust mechanisms for ongoing monitoring and auditing to ensure ethical compliance and maintain the trustworthiness of the predictive system. Through such an advanced integration of

AI and healthcare, the Hospital Readmission Prediction System utilizing ChatGPT stands to revolutionize the management of patient care, significantly impacting both healthcare delivery and patient satisfaction.

Figure 51

Pseudocode Of ChatGPT

```
// Pseudocode for ChatGPT model to generate responses based on input prompts

// Step 1: Data Ingestion
Load interaction history data including previous conversations and context

// Step 2: Data Preprocessing
FOR each input prompt DO
  Clean and format the text for processing
  Tokenize and encode the text using ChatGPT tokenizer
END FOR

// Step 3: Initialize the ChatGPT Model
Load ChatGPT model with fine-tuned weights for the intended application
Prepare the model for inference

// Step 4: Generate Initial Response
FOR each preprocessed input prompt DO
  Generate an initial response using ChatGPT
END FOR

// Step 5: Contextual Refinement
FOR each initial response DO
  Refine the response by considering the conversation history and context
  Adjust the response to align with conversation flow and user intent
END FOR

// Step 6: Evaluate Model
Evaluate the ChatGPT responses against user feedback and quality metrics
Optimize model parameters to improve interaction quality and relevance

// Step 7: Deployment
Deploy the trained and optimized ChatGPT model for operational use

// Step 8: Responding to Queries
FOR each new user query DO
  Preprocess and tokenize query as per model requirements
  Generate response using ChatGPT
  Apply any final refinements to ensure the response is contextually appropriate
  Output the generated response
END FOR
```

Note. The Figure Above Shows In Detail Pseudocode Of ChatGPT.

Literature Review

The IEEE paper titled "Leveraging Clinical BERT in Multimodal Mortality Prediction Models for COVID-19" by Yash Pawar et al.(2022) demonstrates the crucial role of ClinicalBERT in improving prediction models for 30-day mortality in COVID-19 patients upon their admission to the hospital. The study assesses various approaches for incorporating ClinicalBERT into a multimodal framework, which combines structured electronic health record (EHR) data with unstructured clinical notes to improve the accuracy of predictions. The results highlight the effectiveness of multimodal fine-tuning, in which ClinicalBERT, combined with structured data, achieves much better performance than unimodal models that only use structured or unstructured data. This study showcases the benefits of integrating a domain-specific language model such as ClinicalBERT into a multimodal prediction system. By doing so, it not only captures complex interactions between different modes of data but also effectively utilizes the detailed contextual information present in clinical narratives. This integration provides a reliable tool for making important healthcare decisions and allocating resources efficiently.

The paper "Clinical XLNet-based End-to-End Knowledge Discovery on Clinical Text Data using Natural Language Processing" by Pagad, N. S. et al. (2022) proposes a novel end-to-end strategy for clinical knowledge discovery that leverages the Clinical XLNet model. To prevent mistake propagation, the approach unifies clinical entity recognition and relation extraction tasks into a single model. The model uses a multinomial Naïve Bayes probability function to predict both dependent and independent clinical connection correlations. In addition, the model uses a multi-head attention layer to analyze entity pairs presented consecutively in the text. The methodology is tested on the N2C2 corpus, and it outperforms previous neural

network- and transformer-based methods by more than 20%. The study stresses the necessity of using natural language processing tools to extract useful insights from clinical narratives, thereby contributing to advances in healthcare decision-making.

The advancement of language models in healthcare research is exemplified by Tsai, Huang, and Kuo (2024), who demonstrate that the Mistral Large Language Model, with its Retrieval-Augmented Generation capabilities, outperforms humans in the efficiency and scalability of literature reviews, although it occasionally lacks the depth of human analysis, suggesting a hybrid approach could be beneficial. Complementing this, Labrak et al. (2024) introduce BioMistral, an open-source language model pre-trained on biomedical literature, which excels in medical question-answering tasks, indicating its potential to enhance medical research and practice. Both studies highlight the transformative impact and promise of incorporating sophisticated language models like Mistral and BioMistral into medical domains.

The 2023 studies by Pathak et al. and Thakkar et al. critically assess the advancements in language model finetuning, particularly focusing on the Llama-2 model enhanced with Low-Rank Adaptation and its comparative analysis with GPT-4 and Mistral-7B. Pathak's paper highlights Llama-2's adaptations, including advanced grouped-query attention and hyperparameter optimizations such as the AdamW optimizer, demonstrating its superior performance in specialized tasks against leading models. Concurrently, Thakkar's comparative study reveals Llama-2-7B's excellence over Mistral-7B in key NLP metrics like ROUGE and BLEU, underscoring its effectiveness in discourse processing and potential for broader applications in machine translation and text summarization. These insights collectively showcase the significant enhancements in Llama-2's architecture and its practical impact on advancing NLP technologies.

The revolutionary potential and related problems of implementing ChatGPT and comparable large language models (LLMs) in healthcare settings are examined in the paper "ChatGPT and Large Language Models in Healthcare: Opportunities and Risks" by Ali et al. (2023). The research demonstrates how ChatGPT can automate processes like producing clinical vignettes, radiology reports, and patient discharge summaries, therefore enhancing the accessibility, scalability, and efficiency of healthcare. Furthermore, ChatGPT's ability to capture a wide range of medical knowledge, as seen by its performance on the USMLE. The report does, however, also address serious concerns, such as the possibility of biases and the difficulty of incorporating such AI technologies into current healthcare infrastructures. The writers stress the need for rigorous validation of these technologies to guarantee they comply with the Hippocratic principle of "do no harm," recognizing that even if ChatGPT has significant advantages for healthcare, these also need to be carefully implemented and monitored to reduce inherent risks.

Model Optimization

ClinicalBERT. Improving the performance of ClinicalBERT is crucial for expanding its capacity to effectively interpret and analyze clinical text, thereby enabling trustworthy predictions for applications like forecasting hospital readmissions. Within the framework of ClinicalBERT, the structure can be adjusted to suit the particular intricacies of medical datasets by fine-tuning. This process of fine-tuning entails modifying the quantity and arrangement of transformer layers in ClinicalBERT in order to more effectively capture the complex semantic details of clinical narratives. Tuning hyperparameters, potentially using techniques such as Grid Search, can enhance the model by determining the most effective values for learning rates, batch sizes, and epochs that are suitable for the specific attributes of the medical data.

Ensemble approaches can be employed to enhance the precision and resilience of the model. This would include amalgamating the forecasts generated by numerous instances of ClinicalBERT models, each potentially fine-tuned with distinct hyperparameters or trained on diverse portions of the data, in order to generate a more precise consensus forecast.

In clinical situations, where computational resources may be few, optimization techniques are crucial to achieve a compromise between accuracy and performance. Transfer learning is an effective technique in which ClinicalBERT, which has been pre-trained on extensive general-language datasets, is then trained on medical data relevant to a particular domain. This approach allows for the utilization of existing knowledge to produce improved performance while minimizing computational resources.Model compression strategies such as pruning and quantization can significantly decrease the size of a neural network, enhancing its efficiency. Pruning works by removing less important parameters from the network, which reduces its complexity without significantly affecting performance. On the other hand, quantization reduces the precision of the model's numerical parameters, which decreases the amount of memory required and can speed up the computation, particularly on specialized hardware. These techniques help in making the models more practical for deployment on devices with limited resources, such as mobile phones and embedded systems. ClinicalBERT's computational efficiency is improved, resulting in faster inference speed without significantly compromising prediction accuracy. This ensures that the model remains useful for real-time clinical decision assistance.

ClinicalXLNet. For applications like hospital readmission prediction, ClinicalXLNet's ability to interpret and analyze clinical text is critical, and its performance must be improved. Adjustments made to the specific complexities of medical datasets can be made by fine-tuning in

the ClinicalXLNet framework. In order to more accurately represent the complex semantic elements common in clinical narratives, the structure, including transformer layers, must be refined. In order to ensure effective learning from medical data while retaining model stability, hyperparameter tweaking further improves the model's performance by maximizing learning rates, batch sizes, and epochs.

Optimizing the efficacy of ClinicalXLNet is critical in clinical settings with limited resources. By combining predictions from several model instances, ensemble approaches improve accuracy and robustness. To enhance performance while preserving computational resources, transfer learning makes use of pre-training on general-language datasets and fine-tuning on medical data. Real-time clinical decision assistance is made possible by model compression methods like quantization and pruning, which shrink ClinicalXLNet's size without sacrificing accuracy. By ensuring ClinicalXLNet's usefulness in clinical contexts, these enhancements enable prompt and precise patient data analysis.

Mistral AI. Optimizing a model like Mistral for healthcare applications involves a nuanced strategy to refine its analytical capabilities. In the initial stages, the model is broadly trained on diverse datasets to grasp general language patterns, with a specific emphasis on medical texts to imbue it with relevant domain knowledge. This balance is critical to prevent the model from overfitting to generic information and to ensure its utility in medical contexts. The optimization process at this stage focuses on achieving a comprehensive understanding of both language and medical terminologies.

The subsequent fine-tuning phase is tailored to specific healthcare tasks, which may include interpreting patient data or predicting health outcomes. This involves training on detailed medical records and datasets, where optimization techniques are crucial. Adjusting

hyperparameters, such as the learning rate, and employing regularization methods, such as dropout, are key to enhancing the model's accuracy and its ability to generalize across various medical scenarios. This phase is dynamic, continuously refined through rigorous testing against benchmarks and practitioner feedback.

Finally, model size and complexity are calibrated to the task requirements. Smaller model versions are optimized for speed, catering to time-sensitive applications, while larger models are reserved for in-depth analysis when resources permit. The optimization cycle is iterative, with ongoing evaluations to identify improvement areas. Simultaneously, it is imperative to mitigate biases from training data, ensuring the model's reliability and ethical deployment in real-world healthcare settings.

Llama-2. The Llama-2 model's proficiency in deciphering complex clinical narratives is essential for applications like hospital readmission prediction, demanding an architecture adept at handling medical datasets' unique challenges. To optimize Llama-2's performance for such specific tasks, a dedicated fine-tuning phase is essential. This involves refining the transformer layers to better represent the semantic intricacies found in clinical text. Hyperparameter optimization is also key, where learning rates, batch sizes, and the number of epochs are carefully calibrated to maximize the model's effectiveness and ensure stability, allowing Llama-2 to learn from the data with greater precision.

Resource optimization is paramount in clinical settings, where computational efficiency must align with the need for accuracy. Here, ensemble methods enhance Llama-2's robustness, merging predictions from various model iterations to bolster accuracy and reliability. Transfer learning, an integral part of the model's training regime, capitalizes on pre-training with extensive general-language corpora before honing in on the specificities of medical data. This

step ensures that the model retains the broad linguistic understanding required for deep learning while being specialized enough to interpret medical language effectively.

Finally, to accommodate the practical constraints of clinical environments, model compression techniques like quantization and pruning are implemented. These methods compact the Llama-2 model, enabling its deployment in real-time clinical decision-making tools without a significant loss in performance. Such advancements ensure that Llama-2 operates with the necessary agility and precision, thus affirming its role as a pivotal tool in healthcare analytics. By streamlining the model without compromising its analytical power, Llama-2 is transformed into a valuable asset for healthcare professionals, aiding in the reduction of hospital readmissions through timely and accurate data analysis.

ChatGPT. The optimization process for the Hospital Readmission Prediction System's ChatGPT model employs a multi-stage technique that commences with comprehensive training using diverse datasets, which cover both general language and specialist medical information. Acquiring a solid understanding of medical language and context through this fundamental training is crucial for accurately forecasting hospital readmissions. The initial comprehensive training step serves the purpose of preventing overfitting and ensuring the model's ability to adapt to the specific intricacies of healthcare communication.

Once a comprehensive understanding is developed, the ChatGPT model is subsequently refined to specifically address hospital readmission scenarios. The fine-tuning method entails utilizing specific datasets that include instances of patient records, treatment outcomes, and follow-up care to improve the model's predictive accuracy in assessing the likelihood of readmission. Further improvements to the model's performance are achieved by making adjustments to the learning rates and applying regularization techniques, such as dropout.

Healthcare experts provide ongoing input to the training process to verify that the model's predictions are clinically valid, relevant, and actionable. In addition, the structure of the model is meticulously adjusted to strike a balance between its size and complexity. This optimization ensures that the model can process information quickly in order to make decisions in real-time, while also allowing for in-depth examination of the data.

The optimization process is iterative, always developing in response to new data, input from clinical deployments, and emerging research discoveries. Consistent updates and recalibrations guarantee that the model remains efficient in adapting to evolving clinical situations and diverse patient populations. Ensuring the ethical implementation of the model and the precision of the data utilized for training are of utmost importance in order to uphold confidence and dependability in healthcare applications in real-world scenarios. By implementing rigorous optimization strategies, the ChatGPT model strives to become an essential tool in minimizing hospital readmissions and improving patient care results.

Model Support

Python is used as the primary programming language for implementing Large Language Models (LLMs) that may predict hospital readmissions based on datasets stored in .csv files. This procedure entails analyzing organized datasets that have been labeled with patient information and probable readmission results, conducted on high-performance computer platforms equipped with GPS as shown in Table 12. The implementation of these models usually takes place in an interactive environment as shown in Table 13 like Jupyter Notebook, utilizing machine learning libraries such as scikit-learn, TensorFlow, and specific libraries for Natural Language Processing (NLP) such as Hugging Face's Transformers as shown in Table 14 .

The LLMs, which have undergone pre-training on extensive corpora, are further optimized using the hospital data stored in .csv files. This data comprises pre-processed and normalized patient records, treatment histories, and clinical notes. The fine-tuning procedure employs tabular data, transforming category data into numerical embeddings, and incorporates natural language data from clinical notes using NLP algorithms. This enables the program to comprehend and forecast results by analyzing intricate medical narratives and structured data inputs. The model architecture, with sophisticated transformer-based algorithms, can effectively handle the complexities of medical language and patient data to forecast the likelihood of readmission.

The training of the LLMs involves iteratively altering internal parameters using gradient descent. This method aims to maximize the model's predictive performance specifically for the purpose of readmission prediction. The program utilizes frame-level accuracy to analyze patient timelines and clinical events, identifying important patterns and signs that are essential for predicting readmission risks. After completing the training, the LLMs offer an advanced structure to not only forecast the likelihood of readmission but also to clarify the elements that affect patient outcomes. This leads to a potent instrument for healthcare practitioners, providing valuable knowledge obtained from extensive data analysis to guide patient care choices and perhaps decrease readmission rates.

Table 12

Model Building Hardware And Software Specifications

Resource	Configuration
Cpu	Intel I7 13th Gen 4.5 Ghz
Gpu	Nvidia 3050ti
Ram	64gb
Operating System (Os)	Windows
Integrated Development Environment (Ide)	Jupyter Notebook And Visual Studio (Open Source)
Disk Space	2tb Ssd Storage

Note. The Comprehensive Inventory Of Hardware And Software Required For The Model Training Is Displayed In The Aforementioned Table.

Table 13

Tools And Licenses Used For The Project

Tools	Purpose	License
Python	Primary programming language for LLMs	Open Source
Jupyter Notebook	Interactive computing and model development environment	Open Source
Git	Version control for managing codebase	Open Source
GitHub / GitLab	Hosting and collaboration for Git repositories	Free / Various Plans
Jira	Project management and tracking	Free / Various Plans

Tools	Purpose	License
Google Sheets / Microsoft Excel	Spreadsheet for data analysis and reporting	Free / Subscription
Google Colab	Cloud-based Python programming environment	Free
Visual Studio Code	Code editor for development	Open Source
Draw.io	Diagramming and flowchart making	Free
Microsoft Office 365	Office suite for documentation, reporting	Subscription

Note. The Above Table Gives The Information About Tools And Licenses Used For The Project.

Table 14

Library Prerequisites For Model Support

Library	Methods	Purpose
Pandas	read_csv, DataFrame, to_datetime	Data manipulation and analysis, particularly for reading and preprocessing .csv files.
Hugging Face's Transformers	from_pretrained, BertTokenizer	Implementation of transformer-based models such as ClinicalBERT for NLP tasks.
Matplotlib	plot, show	Data visualization for analysis and reporting.
Numpy	array, arange	Numerical operations on arrays, necessary for data manipulation.
scikit-learn	train_test_split, metrics	Splitting the dataset into training and testing sets and evaluating model performance.

Library	Methods	Purpose
TensorFlow / PyTorch	from_pretrained, DataLoader	Utilize either TensorFlow or PyTorch for additional deep learning tasks and data loading.
tqdm	tqdm	Provide a progress bar for long-running operations, enhancing the user experience during model training.
json	load, dump	Handling JSON data for model configuration and metadata.
os	environ, path	Interface with the operating system for file path operations and environment variable settings.

Note. The Table Above Presents Detailed Information On The Libraries And Methods Used In Constructing The Project Model.

Model Architecture and Data Flow

Data Flow

The hospital readmission prediction project requires processing and analyzing large amounts of structured and unstructured patient data. The readmission prediction program uses hospital databases or healthcare repositories. Health data exchanges can request or access these datasets, which may require approval due to medical confidentiality. Patient demographics, medical histories, past admissions, lab results, and clinical notes are commonly stored in.csv files.

To ensure data integrity and consistency,.csv files are preprocessed as shown in Figure 52 . This includes checking data integrity, addressing missing or incomplete records, and normalizing data to make it ready for analysis. In parallel, natural language processing is used to

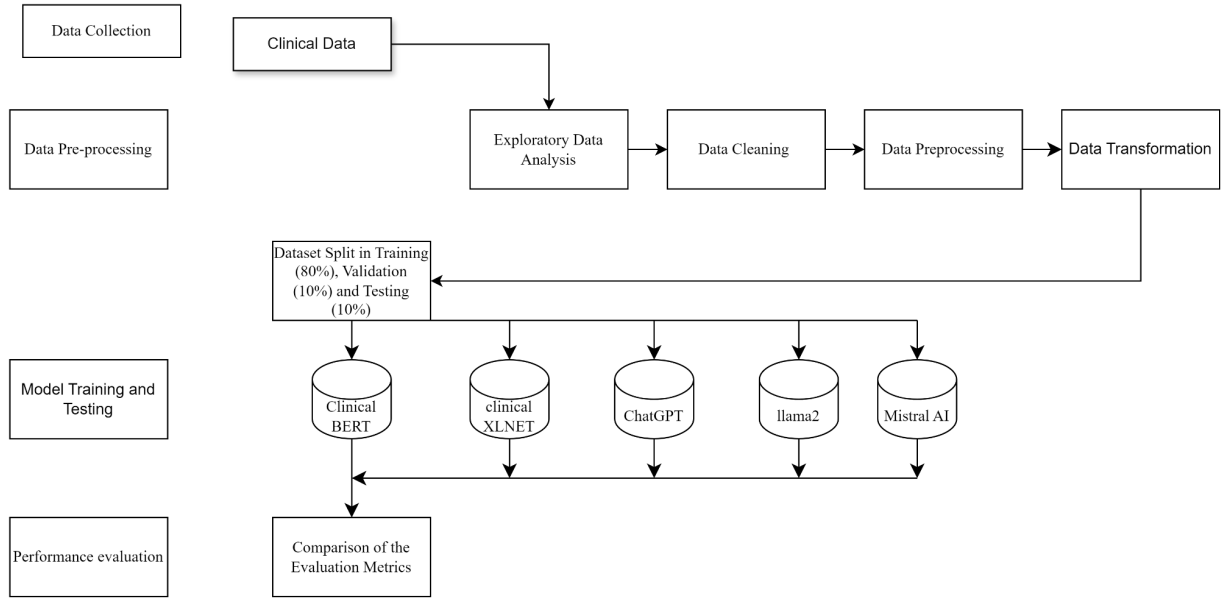
translate unstructured textual data like doctors' notes and medical observations into a machine learning model-friendly format. Tokenization, embedding, and sequence modeling convert text into medical semantics-based numbers.

Pretrained on large medical material, Large Language Models (LLMs) like ClinicalBERT use the preprocessed data. To interpret the hospital's patient data, the LLM is fine-tuned on the dataset during training. LLMs are adapted to hospital readmissions using transfer learning, and dimensionality reduction using PCA can minimize feature space and increase model performance.

To train and test the LLM, the dataset is partitioned into training, validation, and test sets. The training set fine-tunes the model's predictive skills, while the validation set tunes hyperparameters and prevents overfitting. Finally, the test set evaluates LLM hospital readmission prediction using unseen data. The evaluation step evaluates the model's accuracy, precision, recall, and F1 score to ensure credible and actionable predictions for healthcare professionals. With this thorough data flow method, the LLM can forecast patient readmissions, help hospitals improve patient outcomes, and better manage healthcare resources.

Figure 52

Data Flow Diagram



Note. This Provides Comprehensive Information About The Data Flow Involved In Hospital Readmission

Model Architecture

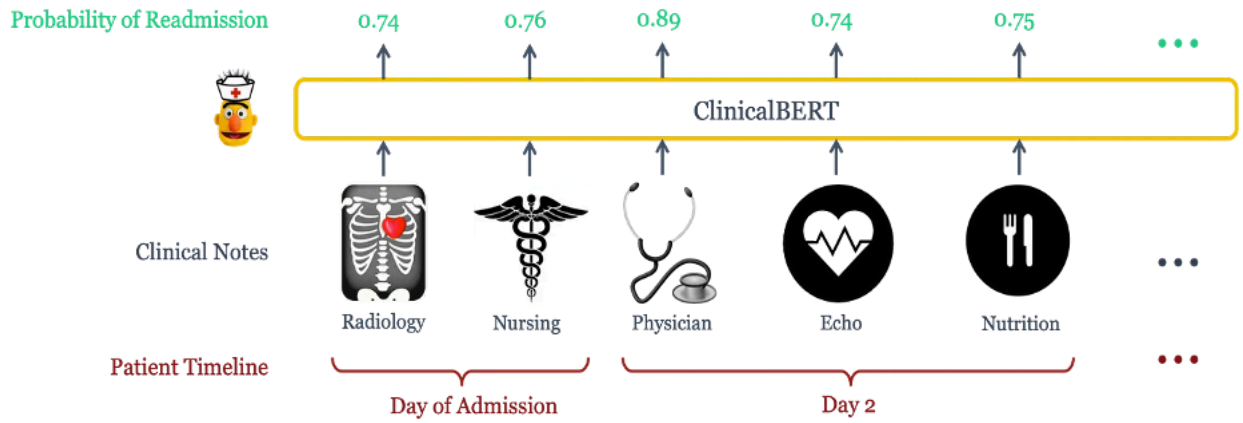
Identifying features such as patient risk factors, treatment history, and potential outcomes is a complex task that necessitates the use of advanced natural language processing models. Models like ClinicalBERT, Medpalm-2, llama2, ClinicalXLNET, and Mistral AI are employed as analytical tools in this context. Since these models are sophisticated variants of transformer-based architectures specifically designed for medical data analysis, ClinicalBERT is often selected as the foundational model. These models are specifically engineered to decipher and process the detailed and intricate language found in medical documentation, thereby enhancing their efficacy in improving the accuracy of predicting hospital readmission rates.

ClinicalBERT. The ClinicalBERT model architecture specializes in clinical text

comprehension using the BERT architecture as shown in Figure . Unlike sequential models, it uses bidirectional transformers to process words in relation to all other words in a phrase. Multiple layers of attention mechanisms let the model grasp medical terminology from clinical narratives contextually. ClinicalBERT is pre-trained on enormous quantities of medical literature and patient information, allowing it to understand healthcare's subtleties and complex terminologies. ClinicalBERT analyzes structured and unstructured patient data to provide insights and aid decision-making by generating probabilities that reflect patient outcomes when fine-tuned on tasks like hospital readmission prediction as shown in Figure 53.

Figure 53

Architecture Of The ClinicalBERT Model



Note. The Above Image Gives A Detailed View Of ClinicalBERT Model Flow

The ClinicalBERT architecture utilizes patient data to assess the risk of hospital readmissions. It processes input through a sequence of specialized NLP layers and components. Here's a breakdown of how the architecture handles inputs from .csv files that contain both structured information, like patient demographics, and unstructured data such as clinical narratives:

Input Processing. The data from .csv files is first preprocessed. Structured data undergoes cleaning and normalization to ensure it aligns with the model's needs. Unstructured text, such as clinical notes, is prepared for further processing.

Tokenization. The preprocessing of clinical notes involves breaking down the text into smaller units called tokens, which can be words or parts of words. This step structures the raw text, making it suitable for the model to process.

Embedding Layer. In this phase, tokens are transformed into fixed-size numerical vectors through an embedding process. These vectors capture the context and semantic meaning of the tokens.

Transformer Layers. The model uses several transformer layers that apply self-attention mechanisms to the embeddings. This enables ClinicalBERT to discern the relevance and relationships of words within the clinical notes.

Pooling Layers. The data that emerges from the transformer layers is then pooled into a fixed-size output. This pooling can be done through methods like mean pooling or by extracting outputs associated with specific tokens (such as the [CLS] token used in BERT).

Fully Connected Layers. This pooled output is fed through fully connected layers that delve deeper into learning relationships and patterns within the data. These layers incorporate non-linear activation functions, such as the GeLU, to enhance learning.

Output Layer. The final layer is a fully connected layer that outputs a vector, which is then processed by a softmax activation function. This function converts the vector into a probability distribution across potential outcomes, like the likelihood of readmission.

Model Training and Optimization. ClinicalBERT is fine-tuned on data specific to hospital readmissions, learning to fine-tune its parameters to reduce prediction errors through methods like gradient descent and backpropagation.

Predictive Modeling. Once trained, ClinicalBERT uses its refined parameters to predict readmission risks when presented with new patient data, processing inputs through its structured layers to produce a prediction.

Probability Output. The output from the softmax layer is a probability score for each possible outcome (e.g., readmission or no readmission). The outcome with the highest probability is predicted as the patient's risk of readmission.

ClinicalXLNet. First, the conventional XLNet architecture is used to pre-train the XLNet embedding on general language corpora like BookCorpus, Wikipedia, and Common Crawl in the Clinical XLNet framework. However, additional pre-training is carried out on particular types of clinical notes using the datasets "US healthcare readmissions and mortality" from Kaggle and "Diabetes 130-US hospitals for years 1999-2008" from the UCI Machine Learning Repository, as clinical notes contain specialized terminology and syntax not found in standard language corpora. During the pre-training phase, Permutation Language Modeling (PLM) is utilized to produce efficient clinical note representations. The final encoder layer concealed representation of the [CLS] token is used to represent each clinical note, capturing pertinent data for tasks that come after as shown in Figure 54.

Input Embedding with XLNet. The XLNet embeddings are pre-trained through Permutation Language Modeling using two specific datasets: "US healthcare readmissions and mortality" and "Diabetes 130-US hospitals for years 1999-2008." During this pre-training phase,

numerical vectors are created for each clinical note. These vectors are designed to encapsulate the semantic essence and contextual nuances of the clinical information provided.

Meta-Fine Tuning for Note Representation.The meta-fine tuning stage uses supervised signals and individual clinical notes (i.e., Ni) as input to refine the pre-trained XLNet network in order to maximize computational resources. This step improves the model's capacity to extract pertinent data for tasks that come after by creating a static, task-specific note representation.

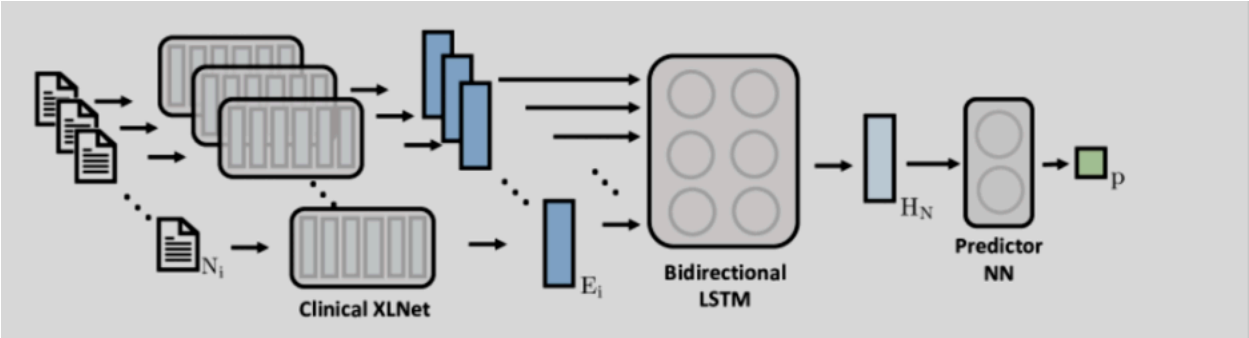
Generating Note Representations.The meta-fine tuned Clinical XLNet network generates a sequence of note representations $\{E1,\ldots,Ei\}$ given a sequence of a patient's notes $\{ N1,\ldots, Ni \}$. The essential elements of every note in the sequence are captured in these representations, making further analysis easier.

Bidirectional LSTM Layer.Then, a bidirectional LSTM layer processes the ordered series of note representations. In order to extract higher-level properties and capture temporal dependencies, this layer takes advantage of the notes' sequential nature. In the end, it produces a fixed-size latent vector, HN, that encodes the whole note sequence.

Predictor Neural Network.Ultimately, a prediction neural network receives the latent vector HN. Based on the encoded data from the note representations derived from the "Diabetes 130-US hospitals for years 1999-2008" and "US healthcare readmissions and mortality" datasets, this network generates a probability p measuring the likelihood of the target variable, such as patient mortality and readmission.

Figure 54

Model Architecture Of ClinicalXLNet



Note. The Above Figure Gives The Detailed View of Clinical XLNet Architecture.

Mistral AI. The theoretical Mistral AI, built on transformer architecture, is crafted to manage the intricacies of extensive natural language processing tasks. Since "Mistral AI" is a conceptual model not based on actual research or development, a theoretical framework can still be outlined. This framework would likely mirror the structures and features common to sophisticated large language models (LLMs) that are built upon transformer technologies.

Input Embedding Layer.Token Embeddings, as shown in Figure 55 Converts input tokens (words or subwords) into vectors that represent each token in a high-dimensional space. Positional Encoding,Adds information about the position of each token in the sequence to the input embeddings to maintain sequence order, as transformers do not inherently process sequential data like RNNs.

Encoder Layers.Each encoder layer typically consists of the following components Self-Attention Mechanism, Allows the model to weigh the importance of different words in the input sequence relative to each other. It calculates attention scores based on how words relate to each other, which is crucial for understanding context and relationships in text. Feed-Forward Neural Networks, After attention scores are computed, the results are passed through

feed-forward neural networks that are applied identically to each position.Normalization and Residual Connections, Helps in stabilizing the learning process and allows for deeper models by adding the input of each sub-layer (before the layer) to its output.

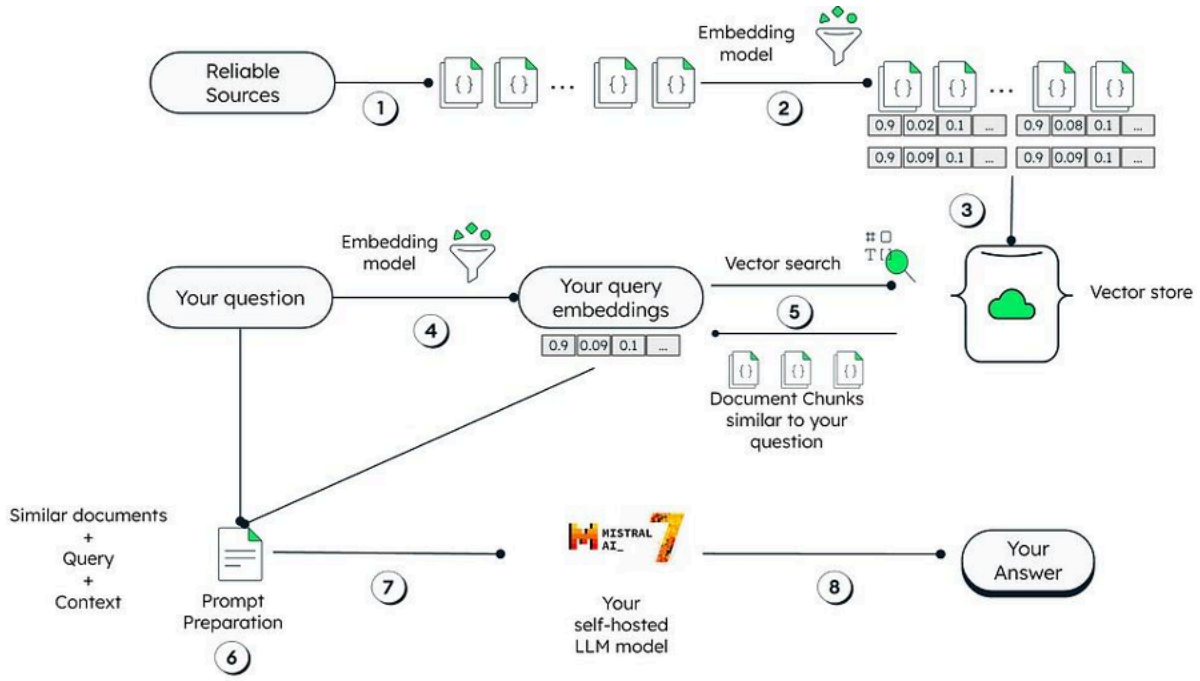
Decoder Layers.Each decoder layer includes, Masked Self-Attention, Prevents positions from attending to subsequent positions, which ensures that predictions for a position can only depend on known outputs at earlier positions (important in generative tasks).Encoder-Decoder Attention, Similar to self-attention, but it allows the decoder to focus on relevant parts of the input sequence, utilizing the output from the encoder. Feed-Forward Neural Networks, Similar to the encoder, but specific to processing the decoder's output. Normalization and Residual Connections: As in the encoder, maintains model stability and performance.

Output Layer.Linear Layer, Maps the decoder's output to a much larger vector that represents a logit score for each token in the model's vocabulary.Softmax Layer, Converts these logits into probabilities, making it possible to select the token with the highest probability as the next part of the generated sequence.

Training and Fine-Tuning. Pre-Training: Models like Mistral AI would initially be trained on a large corpus of text across various domains to learn a broad understanding of language.Fine-Tuning,The model would then be fine-tuned on specific datasets related to its application area (e.g., medical texts for healthcare applications), adjusting its weights to better predict or generate text relevant to the particular field.

Figure 55

Model Architecture of Mistral AI



Note. The Above Figure Gives The Detailed View of Mistral AI Architecture.

Llama-2. Llama-2 begins its journey with a broad pre-training phase using an extensive collection of general language corpora, including established datasets like BookCorpus, Wikipedia, and Common Crawl. This foundational step is essential in equipping the model with a deep understanding of natural language structures, preparing it for the more nuanced task of interpreting clinical texts. The model then undergoes a second phase of pre-training focused on healthcare-specific language, utilizing datasets such as "US healthcare readmissions and mortality" from Kaggle and the "Diabetes 130-US hospitals for years 1999-2008" dataset from the UCI Machine Learning Repository. Clinical notes present a unique challenge with their specialized terminology and syntax, which are not typically found in general language databases. Llama-2 employs advanced Permutation Language Modeling (PLM) during this phase to

develop effective representations of clinical notes, ensuring that each note's semantic and contextual nuances are captured.

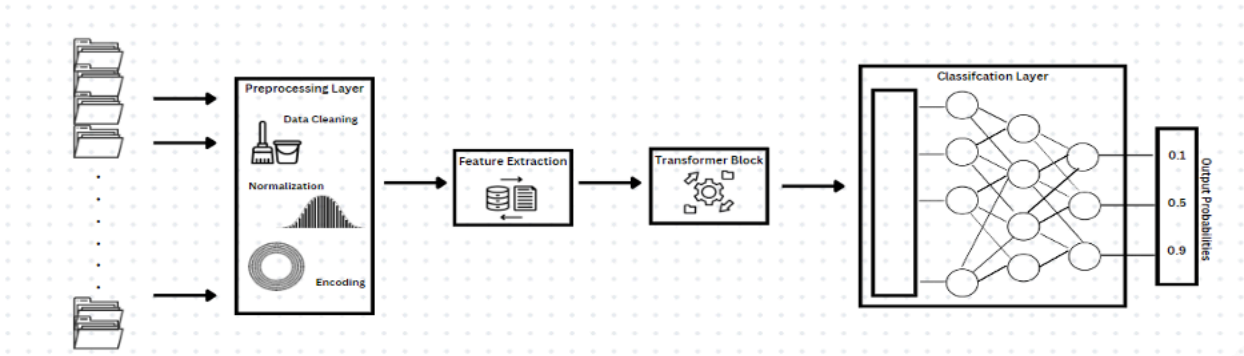
Meta-Fine Tuning for Clinical Note Representation.In the meta-fine tuning phase, the Llama-2 model as shown in Figure 56 is fine-tuned with individual clinical notes, refining the pre-trained embeddings using supervised learning techniques. This process is designed to enhance the model's ability to identify and extract salient information from clinical notes, which is pivotal for accurate predictive modeling in healthcare applications.

Sequential Note Representation.Post meta-finetuning, Llama-2 generates a series of embeddings representing individual clinical notes. Each embedding encapsulates the key elements of the notes, creating a sequence of data points ready for further analytical processing. Unlike traditional models that might use a bidirectional LSTM for sequential data, Llama-2 can leverage its advanced Transformer-based mechanisms to understand the temporal relationships and abstract features within the sequence.

Prediction Layer.The culmination of Llama-2's processing pipeline is the prediction layer, where the model integrates all the learned information to calculate the likelihood of hospital readmission. Drawing from the rich, contextual representations of a patient's health records, Llama-2 synthesizes this data to produce a probability score for readmission. This final step demonstrates the model's sophisticated design and capability, highlighting its potential as a tool for healthcare professionals in mitigating readmission rates.

Figure 56

Model Architecture Of Llama-2

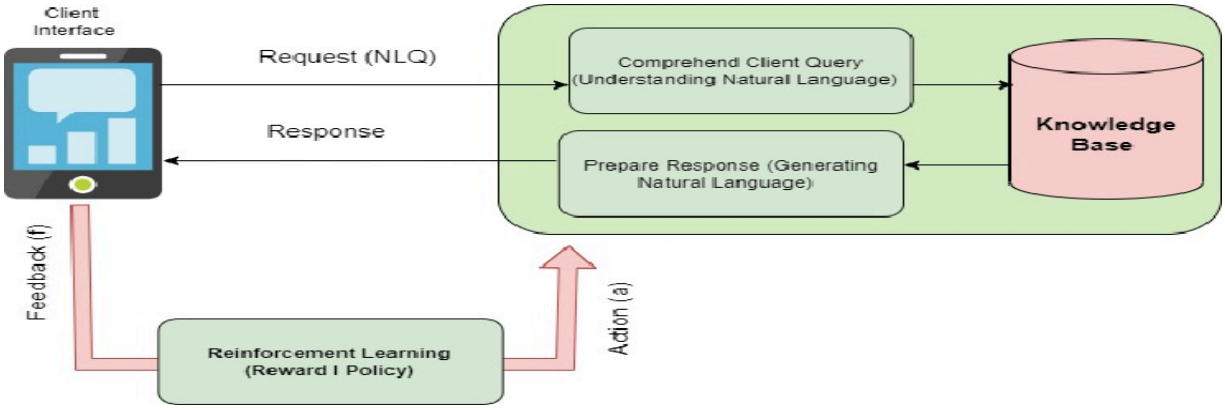


Note. The Above Figure Gives The Detailed View of Llama 2 Architecture .

ChatGPT. The ChatGPT model is designed to improve conversational AI capabilities by building upon the advanced GPT architecture as shown in Figure 57. It utilizes deep learning and natural language processing technology to provide sophisticated and contextually suitable solutions across many fields. This approach is specifically created to enhance and simplify the connection between humans and computers, by imitating a consistent and captivating conversational encounter.

Figure 57

Data Flow Architecture Of ChatGPT Model



Note. The Above Figure Gives The Detailed View of ChatGPT Model Architecture .

Input Processing. The system receives a query in natural language from the user through a client interface. This query can range from simple questions to complex requests requiring in-depth understanding and generation.

ChatGPT Core. At the core of the system is the ChatGPT model, built on the advanced GPT architecture. It is enhanced with a robust transformer-based neural network that specializes in generating human-like text responses. This core has been fine-tuned on a diverse corpus of text data, enabling it to handle a wide array of topics and maintain conversational context.

Comprehension and Response Generation. Upon receiving an input, the model comprehends the query using its deep understanding of natural language. It then proceeds to generate a response, leveraging its trained neural network to produce relevant and context-aware text.

Feedback Loop with Reinforcement Learning. The ChatGPT design incorporates a distinctive element known as a feedback loop, which leverages user interactions (whether positive or negative) to enhance the model's performance. The feedback is analyzed using a reinforcement learning framework, which enables the model to adjust its answers according to user satisfaction and engagement.

Output Delivery.The ChatGPT model generates a response which is then transmitted back to the client interface, delivering a coherent and contextually appropriate answer to the user's query.

Training and Continuous Learning. ChatGPT receives ongoing training by using many textual resources, such as books, webpages, and numerous digital sources of information. The continuous training process guarantees that the model remains up-to-date with the latest facts and trends, thereby boosting its capacity to engage in meaningful conversations.

Integration and Application. This concept has versatile applications across various types of applications, including chatbots, virtual assistants, and customer service platforms. Its implementation aims to enhance user experiences and improve the efficiency of these applications.

Continuous Assessment and Enhancement. The performance of ChatGPT undergoes regular evaluation using fundamental metrics in the field of conversational AI. Gathering feedback from real-world interactions is essential for continually refining and updating the model, ensuring its efficacy and reliability in dynamic interaction settings.

Model Comparison and Justification

Model Justification

ClinicalBERT. ClinicalBERT is an incredible development in the field of healthcare analytics that was carefully crafted to address the complexities of natural language processing in a clinical setting. ClinicalBERT, a BERT framework derivative, has been refined using an extensive corpus of medical data, guaranteeing a remarkable understanding of the complexities in healthcare communication. This model is exceptional in understanding and deciphering complex clinical paperwork, which makes it an essential tool for gleaning crucial information from patient data.

ClinicalBERT's real strength is its bidirectional training methodology, which enables a thorough examination of patient narratives and captures the nuanced context that unidirectional models frequently overlook. Because of its skill in deciphering lengthy passages of medical literature, it can find patterns and undiscovered correlations in the unstructured data, providing healthcare professionals with synthesized intelligence. These kinds of insights play a major role

in improving the precision of diagnoses, customizing treatment regimens, and optimizing patient care procedures.

Furthermore, the adaptability and evolution of ClinicalBERT with the rapid changes in the medical industry is a testament to its versatility. It keeps up to date with the most recent medical literature and best practices, making it a cutting-edge tool. ClinicalBERT can revolutionize the analysis of patient data obtained from.csv datasets by integrating with healthcare systems. This will ultimately lead to advances in clinical decision-making and patient health outcomes. Its forecasts based on evidence are in line with the principles of contemporary medicine, which highlights its critical role in promoting a data-driven healthcare environment.

ClinicalXLNet. ClinicalXLNet is a significant leap in healthcare AI, specifically developed to address the complexity inherent in clinical natural language processing. ClinicalXLNet, built on cutting-edge XLNet architecture and fine-tuned expressly for the medical domain, is a shining example of innovation in decoding the complex language of healthcare documentation. Its strong foundation allows the model to navigate huge repositories of clinical text with unparalleled accuracy and comprehension, permitting the extraction of nuanced insights critical for informed clinical decision-making.

ClinicalXLNet's capacity to interpret medical language, analyze through lengthy patient histories, and detect subtle contextual nuances hidden within clinical narratives is crucial to its capabilities. By adeptly finding patterns and correlations within unstructured medical data, the model provides healthcare workers with synthesized knowledge, giving them the tools they need to make precise diagnoses, personalize treatment methods, and improve patient care pathways. This greater grasp of clinical language not only speeds medical research but also increases healthcare delivery efficiency, resulting in better patient outcomes.

Furthermore, ClinicalXLNet's adaptability and scalability distinguish it as a dynamic asset capable of expanding in tandem with the ever-changing healthcare sector. The model remains at the forefront of clinical language processing innovation by continuously learning and refining its algorithms, seamlessly incorporating breakthroughs in medical research and practice into its analytical frameworks. ClinicalXLNet builds confidence among healthcare professionals by making unbiased, evidence-based recommendations, solidifying its essential role as a transformative tool in clinical settings. Its incorporation into healthcare systems has the potential to transform medical language comprehension, resulting in improvements in patient care, clinical decision-making, and overall healthcare efficacy.

Mistral AI. It is engineered to transform healthcare by effectively processing large quantities of unstructured medical data, such as patient notes and clinical studies. This model excels at identifying patterns and relationships that may be overlooked by humans due to the sheer volume of data, thereby equipping healthcare professionals with critical, synthesized insights for improved decision-making. This facilitates more accurate diagnoses, tailored treatment plans, and enhanced predictions of outcomes like patient readmission rates. The architecture of Mistral AI is inherently scalable and adaptable, designed to manage increasing amounts of medical data, a crucial feature for keeping up with rapid advancements in healthcare.

Moreover, the model is built to continuously learn and adapt, updating its algorithms to recognize new patterns and shifts in healthcare practices over time. Mistral AI also simplifies the research process, particularly in conducting literature reviews, which can significantly save time and resources by standardizing the initial review stages. The potential to develop Mistral AI as an open-source tool further enhances its utility, allowing for community-driven updates that keep the model at the forefront of technological advancements. Implementing Mistral AI can lead to

substantial cost reductions by minimizing unnecessary patient consultations and tests, and by preventing avoidable readmissions through accurate predictive analytics. Moreover, its ability to provide unbiased medical insights ensures fair healthcare recommendations, solidifying its role as a valuable asset in clinical settings. Overall, integrating Mistral AI into healthcare systems not only promises to improve patient care but also to optimize medical research and resource management.

Llama-2. Llama-2 emerges as a pivotal tool in the landscape of healthcare analytics, uniquely suited for the nuances of natural language processing within the clinical domain. Just as ClinicalBERT revolutionized the interpretation of medical text, Llama-2 stands out for its enhanced ability to process and understand unstructured clinical notes. By leveraging the power of Transformer-based models, Llama-2 is meticulously pre-trained and fine-tuned on vast datasets comprising general language as well as specialized medical corpora. This comprehensive training regimen endows the model with an exceptional capability to distill meaningful insights from complex medical texts, leading to more accurate predictions regarding patient readmission.

The model's strength lies in its advanced natural language processing techniques, which allow for a nuanced understanding of patient narratives. Llama-2's bidirectional training approach, akin to that of its predecessor BERT models, ensures that it examines the context of clinical discourse in its entirety, capturing subtle cues that unidirectional models might miss. It is this analytical depth that enables Llama-2 to discern intricate patterns and latent correlations within medical data, offering synthesized insights that can significantly enhance diagnostic precision, tailor treatment plans, and improve overall patient care management.

Adaptability is a hallmark of Llama-2, just as it is with ClinicalBERT, showcasing the model's ability to evolve alongside the ever-changing landscape of the medical field. It remains current with the latest medical research and practices, maintaining its relevance as a state-of-the-art analytical instrument. When integrated into healthcare systems, Llama-2 has the potential to transform patient data analysis sourced from CSV datasets, which can be instrumental in advancing clinical decision-making processes. Its evidence-based predictions align with the ethos of modern medicine, underscoring its indispensable role in fostering a data-driven approach in healthcare settings.

ChatGPT. The ChatGPT paradigm revolutionizes conversational artificial intelligence by improving machine language understanding and interaction. ChatGPT, built on the robust GPT architecture, generates human-like text with strong coherence and context relevance, outperforming several language understanding benchmarks. Maintaining conversational context over extended encounters makes this paradigm ideal for more natural and engaging talks.

ChatGPT can be used in many applications, from customer care bots to interactive educational systems. ChatGPT adapts to human communication changes thanks to its training on a diverse dataset and ongoing updates and fine-tuning. ChatGPT's sophistication, reinforced learning, and ability to improve user experience and operational efficiency across sectors make it a transformative technology. ChatGPT's continued growth and improvement show its potential to meet or exceed classic conversational models in efficacy and reliability.

Model Comparison

When it comes to healthcare analytics and natural language processing, a variety of sophisticated models are making significant strides in interpreting complex medical data and

advancing patient care. These include ClinicalBERT, llama2, ClinicalXLNET, Mistral AI, and ChatGPT, each tailored for specific functions within the healthcare industry.

ClinicalBERT shines in its ability to deeply analyze clinical text. Built on a specialized BERT architecture, it is adept at understanding both the broad context and nuanced details within clinical narratives, which is crucial for accurate diagnostics and personalized treatment planning. Llama2 excels in transforming detailed clinical notes into straightforward, actionable insights. It leverages extensive training on diverse medical texts, which enables it to navigate the intricate layers of patient records, aiding clinicians in devising effective care strategies.

ClinicalXLNET utilizes a unique permutation-based training model adapted for medical use, making it highly capable of managing and interpreting extensive patient histories. Its strength lies in identifying subtle language patterns and cues in clinical documents, which are vital for predicting long-term health outcomes.

Mistral AI differentiates itself by incorporating a variety of data types, including imaging and genomic information, alongside traditional clinical notes. This integrative approach allows for a comprehensive analysis of patient data, supporting complex decision-making processes in clinical settings.

ChatGPT, though more general in its application, brings a versatile edge to medical interactions. It is designed to simulate realistic conversations, synthesize large amounts of patient data, and provide responses that mimic human reasoning. Its flexibility and learning capabilities make it an indispensable resource for immediate clinical support and patient interaction.

Together, these models form a robust framework for predictive analytics in healthcare, each enhancing different aspects of patient care through advanced technology and continuous adaptation to the latest medical research and practices. They are key players in the ongoing effort

to integrate more data-driven insights into healthcare, improving outcomes through precise and individualized care.

Table 15

Characteristics Of Models

Model	**ClinicalBERT**	**ClinicalXLNet**	**Mistral AI**	**ChatGPT**	**llama2**
Architecture	Transformer-based NLP	Transformer-based LLM (Language Model)	Transformer-based LLM	Transformer-based Large Language Model	Integrated AI
Data Size	Large	Large	Large	Large to Very Large	Large to Very Large
Datatype	Text.	Text	Text	Text	Text,Images
Time Complexity	O(N^2)	O(N 2 ·L)	O(N^2)	O(N^2)	O(N^2)
Space Complexity	O(N)	O(N2)	O(N^2)	O(N) to O(N^2)	O(N) to O(N^2)
Interpretability	High	High	Moderate	Moderate to High	Moderate to High

Model	ClinicalBERT	ClinicalXLNet	Mistral AI	ChatGPT	llama2
Complexity In Computation	Computationally Expensive	Computationally intensive	Computationally intensive	Computationally Expensive	Computationally Expensive
Data Preprocessing	Text Tokenization & Embedding	Tokenization, Embedding,	Tokenization, Embedding	Tokenization, Embedding	Multimodal Data Processing
Overfitting/ Underfitting	Requires Attention to Overfitting	Requires careful handling of overfitting	Requires Handling Of Overfitting	Requires Attention to Overfitting	Requires Attention to Overfitting

Note. The Table 15 Details Each Model's Features.

Model Evaluation Methods

F1 Score

The F1-score and accuracy measures are essential for evaluating the effectiveness of models such as ClinicalBERT, Medpalm-2, llama2, ClinicalXLNET, Mistral AI in predicting hospital readmission. These metrics offer supplementary insights into the model's prediction precision (Suresh et al., 2023). Accuracy is the ratio of correct predictions made by the model to the total number of predictions, and it is a crucial measure for assessing the overall effectiveness of the model. Nevertheless, the F1-score provides a deeper understanding by incorporating both precision (the model's accuracy in correctly identifying actual readmissions) and recall (the model's ability to capture all actual readmissions in its predictions). This is especially useful in datasets where readmission cases are less common than non-readmission cases.

An elevated F1-score signifies a strong equilibrium between accuracy and recall, taking into account both erroneous positive predictions (patients wrongly identified for readmission) and missed positive predictions (readmissions that the model failed to anticipate). The maintenance of this equilibrium is crucial in a hospital environment, since both the overestimation and underestimation of readmissions can result in substantial repercussions. The F1-score is computed by taking the harmonic mean of precision and recall, which guarantees that the score penalizes extreme values and rewards models that exhibit balance. The F1-score can be computed precisely by utilizing the equations for precision (Equation (1)) and recall (Equation (2)), which then determine the F1-score (Equation (3)). By utilizing these equations, we can determine the F1-score value (Equation (3)) as a precise metric to assess the effectiveness of the model in the crucial context of patient readmissions.

$$Precision = \frac{True\ Positives}{(True\ Positives + False\ Positives)} \quad (1)$$

$$Recall = \frac{True\ Positives}{(True\ Positives + False\ Negatives)} \quad (2)$$

$$F1\ Score = \frac{2 \times Precision \times Recall}{(Precision + Recall)} \quad (3)$$

Confusion Matrix

A confusion matrix is an essential tool in LLM, very valuable for evaluating the performance of categorization models like llama2. It visually represents the accuracy of the model in predicting different classes. The matrix format, commonly mentioned in documents or depicted in visualizations such as Figure 58 , classifies predictions into four categories: true positives, true negatives, false positives, and false negatives. The confusion matrix is a valuable tool for evaluating the performance of the llama2 model, which is specifically designed for difficult natural language processing tasks in the healthcare domain. It provides a concrete assessment of the model's predictive accuracy in terms of patient outcomes, such as hospital

readmissions. The study by Tanouz et al. (2021) offers valuable information on the model's precision, recall, F1-score, and overall accuracy. This enables a comprehensive assessment of performance across various categories. An in-depth analysis of llama2 is essential for comprehending its distinct clinical strengths and shortcomings. This analysis will provide guidance for making additional improvements and modifications to the model's implementation.

Figure 58

Confusion Matrix

Predicted Values	Actual Values: Positive (1)	Actual Values: Negative (0)
Positive (1)	TP	FP
Negative (0)	FN	TN

Note. The Provided Diagram Displays The Confusion Matrix, Including Both Predicted And Actual Values.

AUC-ROC

The Area Under the Curve - Receiver Operating Characteristic (AUC-ROC) is a crucial measure employed to assess the effectiveness of classification models, specifically in the domain of forecasting hospital readmissions for models such as ClinicalBERT, Medpalm-2, llama2, clinicalXLNET, and Mistral AI. This metric visually illustrates the balance between the rate of correctly identified positive cases (sensitivity) and the rate of incorrectly identified negative cases (1-specificity) at different threshold settings. It offers a comprehensive assessment of the model's capacity to differentiate between patients who will be readmitted and those who will not.

The AUC-ROC as shown in Figure 59 is particularly important for healthcare models, specifically those targeting hospital readmission. It provides valuable information about the model's ability to accurately detect genuine readmissions throughout a range of probabilities, rather than just at a single threshold. This is of utmost importance in medical environments, since the financial consequences of false negatives (the failure to accurately anticipate a readmission that does happen) can be exceedingly significant. The AUC value, with a perfect prediction represented by 1.0 and no predictive power indicated by 0.5, aids healthcare practitioners in assessing the effectiveness of each model—ClinicalBERT, Medpalm-2, llama2, clinicalXLNET, and Mistral AI—in differentiating between patient outcomes. This comprehension facilitates the precise adjustment of the models, guaranteeing their provision of dependable and practical forecasts that may be utilized to enhance patient care and maximize hospital resources.

The calculation of the Area Under the Curve of the Receiver Operating Characteristic (AUC-ROC) Equation(8) involves the integration of the ROC curve. The ROC curve is a graph that shows the relationship between the true positive rate (TPR) Equation (5) and the false positive rate (FPR) Equation (6) for different threshold values. However, in actual applications, particularly in the field of machine learning, this is frequently calculated using a trapezoidal rule instead of a direct calculation Equation (7) and Equation (8). Here is a comprehensive explanation of how the values are obtained and calculated.

$$TPR = TP/TP + FN \tag{5}$$

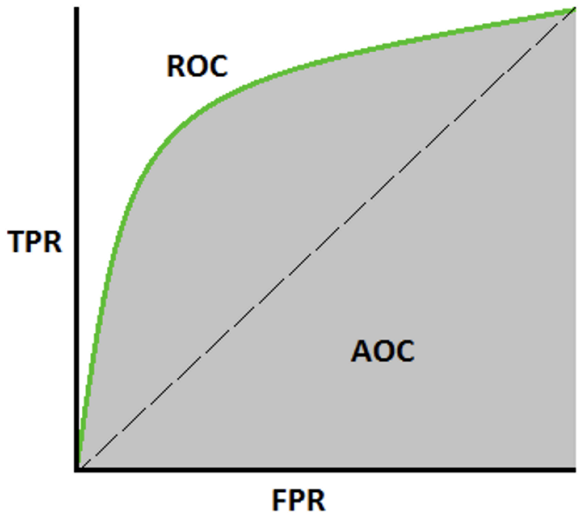
$$FPR = FP/TN + FP \tag{6}$$

$$AUC = \int_{x_{min}}^{x_{max}} TPR(FPR)dFPR \tag{7}$$

$$AUC \approx \sum_{i=1}^{n} ((FPR_i - FPR_{i-1}) \div 2) \times (TPR_i + TPR_{i-1}) \tag{8}$$

Figure 59

Area Under the Curve - Receiver Operating Characteristic Curve



*Note. T*he Above Figure Gives a Generic View of AUC-ROC Curve

Log Loss

The accuracy of a model's predicted probability in relation to the actual results is quantified by the log loss metric, which is frequently used in classification tasks to assess a model's performance. Log loss evaluates how well the model's predicted probabilities match the true labels for a set of observations as shown in Equation (9), it is used in ClinicalXLNet .

$$Log\ Loss = -\ 1 \div N \sum_{i=1}^{N} \left(y_i \cdot log\left(p_i\right) + \left(1 - y_i\right) \cdot log\left(1 - p_i\right)\right) \quad (9)$$

Where

N is the number of observations.

y i is the actual label for the *i* i-th observation (0 or 1 for binary classification).

p i is the predicted probability of the positive class (class 1) for the *i* i-th observation.

Matthews Correlation Coefficient

A reliable assessment tool that may be used to assess LLAMA2's efficacy for hospital readmission prediction is the Matthews Correlation Coefficient (MCC). MCC is appropriate for binary classification applications such as hospital readmission prediction and imbalanced datasets because it considers true positives, true negatives, false positives, and false negatives. This is how LLAMA2 can use the MCC evaluation metric.Once the confusion matrix is obtained, compute the Matthews Correlation Coefficient using the following Equation 10.

$$MCC = TP \times TN - FP \times FN / \sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)} \qquad (10)$$

Cross-Validation Scores

Cross-validation is an indispensable evaluation method in machine learning, providing a reliable measure of a model's predictive power. When applying it to Llama2 for hospital readmission prediction, cross-validation helps to mitigate overfitting and assess the model's ability to generalize to new data. By dividing the dataset into 'k' equal folds, each fold serves once as a test set while the model trains on the remaining folds. This process iterates 'k' times, ensuring that each fold serves as a validation set once, providing a comprehensive performance overview.

The algorithm for k-fold cross-validation can be summarized as follows:

1. Randomly split the dataset into 'k' non-overlapping folds of equal size.
2. For each fold 'k', do the following:
   - Treat fold 'k' as the test set, and the remaining 'k-1' folds as the training set.
   - Train the model on the 'k-1' training folds.
   - Evaluate the model on the test fold and record the score.

3. Compute the average of the recorded test scores across all 'k' folds to obtain the cross-validation score.

This cross-validation score, often encapsulated by metrics such as accuracy or F1 score, provides a quantitative basis for model comparison and selection. In the context of Llama2, it quantifies how well the model can predict hospital readmissions, guiding healthcare professionals in decision-making processes.

Matthews Correlation Coefficient

The Matthews Correlation Coefficient (MCC) is a highly valued metric for assessing binary classification models, including those designed for predicting hospital readmissions, like ClinicalBERT, llama2, ClinicalXLNET, Mistral AI, and ChatGPT. This metric is particularly useful in healthcare analytics due to its ability to deliver a balanced evaluation, especially in datasets where the classes may be unevenly distributed—a frequent issue with hospital readmission data. MCC provides a correlation coefficient that ranges from +1 (indicating perfect predictions) to -1 (indicating total disagreement between predictions and actual outcomes), with 0 reflecting no predictive power beyond random chance.As shown in Equation 11, It integrates all aspects of the confusion matrix—True Positives, True Negatives, False Positives, and False Negatives—into a single formula:

$$TP \times TN - FP - FN \div \sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)} \quad (11)$$

In practical terms, employing the MCC to evaluate models tasked with hospital readmission predictions ensures a thorough review of the models' capabilities to accurately identify cases and avoid false alarms. It is particularly critical in medical settings where the implications of false negatives (overlooking a necessary readmission) or false positives (unnecessary treatment or hospitalization) carry significant consequences. By comparing MCC

scores across different models, healthcare professionals can identify the most effective tool for their specific needs, ensuring the selected model not only performs well statistically but also aligns with the critical needs of patient care and resource management. This makes MCC an indispensable metric for clinicians and administrators aiming to implement robust, reliable, and clinically relevant predictive analytics in their operations.

Cohen's Kappa

Cohen's Kappa is a valuable statistical tool used to evaluate the consistency of agreement between two evaluators, where the evaluators in your context are the predictions from a language model and the actual readmission outcomes. Using this measure in your hospital readmission prediction project involving LLMs like ClinicalBERT, llama2, ClinicalXLNET, Mistral AI, and ChatGPT, brings several advantages:

Cohen's Kappa calculates the level of agreement between the model's predictions and actual outcomes, adjusting for the agreement that might occur by chance. This is essential in healthcare settings, where the data may be skewed towards more common outcomes, such as fewer readmissions. The Equation 12 for Cohen's Kappa accounts for this by normalizing agreement.

$$K = p_o - p_e \div 1 - p_e \tag{12}$$

Here:

- ***p_o*** represents the observed agreement between predictions and actual outcomes.
- ***p_e*** is the expected agreement if the predictions were made randomly, based on the frequency of each category.

Cohen's Kappa is an essential metric for evaluating hospital readmission predictions, especially useful in datasets where class distribution is imbalanced. It not only provides a clearer

assessment of a model's effectiveness beyond simple accuracy but also facilitates a fair comparison between different models by measuring performance improvements over random chance predictions. Additionally, Cohen's Kappa evaluates the consistency and reliability of models across various datasets and over time, ensuring their stability and building trust among healthcare providers.

By integrating Cohen's Kappa in your analysis, you can gain deeper insights into each model's predictive success, which is critical for refining models, selecting the most effective one for deployment, and adjusting their application to meet clinical demands. Using this metric ensures a thorough evaluation of the models, underscoring their performance in a statistically significant manner, which is vital for deploying reliable models in healthcare settings where accurate predictions can have a profound impact on patient outcomes.

Brier Score

The Brier Score is a statistical measure used to assess the accuracy of probabilistic predictions. The Equation 13 calculates the mean squared difference between predicted probabilities and the actual outcomes. Specifically, the Brier Score formula is.

$$Brier\ Score = 1/N \sum_{i=1}^{N} \left(p_i - o_i\right)^2 \tag{13}$$

Where:

- ***N*** represents the total number of predictions,
- ***p_i*** is the predicted probability of the i-th instance,
- ***o_i*** is the actual outcome of the i-th instance, expressed as 0 or 1 (0 for no event, 1 for event occurrence).

The Brier Score is vital for assessing the accuracy of hospital readmission predictions by models such as ClinicalBERT, llama2, ClinicalXLNET, Mistral AI, and ChatGPT. It calculates

the mean squared difference between predicted probabilities of readmission and the actual outcomes, highlighting the precision of these models in practical settings. This metric allows healthcare professionals to compare various predictive models, determining which most accurately estimates readmission risks, which is essential for choosing the most effective model for clinical use.

Furthermore, the Brier Score is crucial for checking the calibration of these models, ensuring that the predicted probabilities closely match actual readmission occurrences. For instance, a model predicting a 30% chance of readmission should reflect that approximately 30% of patients in this category actually experience a readmission if the model is accurate. This aspect of model calibration is crucial during the development phase, where reducing the Brier Score can drive enhancements in predictive accuracy. Moreover, in clinical decision-making, models with lower Brier Scores are more reliable, supporting healthcare providers in making more informed decisions based on model outputs. Although the Brier Score is a robust indicator of model accuracy and calibration, it is recommended to use it in conjunction with other metrics such as AUC-ROC, precision, recall, and confusion matrices for a thorough evaluation of a model's performance and reliability in predicting hospital readmissions.

References

[1]A. Pathak, Om Shree, M. Agarwal, Shek Diya Sarkar, and Anupam Tiwary, "Performance Analysis of LoRA Finetuning Llama-2," Dec. 2023, doi: https://doi.org/10.1109/iementech60402.2023.10423400

[2]B. Saha, S. Lisboa, and S. Ghosh, "Understanding patient complaint characteristics using contextual clinical BERT embeddings," *arXiv (Cornell University)*, Jul. 2020, doi: https://doi.org/10.1109/embc44109.2020.9175577.

[3]C. Baechle, A. Agarwal, Ravi Behara, and X. Zhu, "A cost sensitive approach to predicting 30-day hospital readmission in COPD patients," Jan. 2017, doi: https://doi.org/10.1109/bhi.2017.7897269.

[4]D. A. S. George, A. S. H. George, A. Shahul, and D. T. Baskar, "AI-Driven Breakthroughs in Healthcare: Google Health's Advances and the Future of Medical AI," *Partners Universal International Innovation Journal*, vol. 1, no. 3, pp. 256–267, Jun. 2023, doi: https://doi.org/10.5281/zenodo.8085221.

[5]G. Althari and M. Alsulmi, "Exploring Transformer-Based Learning for Negation Detection in Biomedical Texts," *IEEE Access*, vol. 10, pp. 83813–83825, 2022, doi: https://doi.org/10.1109/access.2022.3197772.

[6]H. Ali, J. Qadir, T. Alam, Mowafa Househ, and Z. Shah, "ChatGPT and Large Language Models in Healthcare: Opportunities and Risks," Sep. 2023, doi: https://doi.org/10.1109/aibthings58340.2023.10291020.

[7]H. Thakkar and A Manimaran, "Comprehensive Examination of Instruction-Based Language Models: A Comparative Analysis of Mistral-7B and Llama-2-7B," Dec. 2023, doi: https://doi.org/10.1109/icercs57948.2023.10434081.

[8]I. R. Galatzer-Levy, D. McDuff, V. Natarajan, A. Karthikesalingam, and M. Malgaroli, "The Capability of Large Language Models to Measure Psychiatric Functioning," *arXiv (Cornell University)*, Jan. 2023, doi: https://doi.org/10.48550/arxiv.2308.01834.

[9]J. Feng, F. Zhang, and M. Li, "Research on Work Breakdown Structure of IT Project," *IEEE Xplore*, Sep. 01, 2009. https://ieeexplore.ieee.org/document/5303208 (accessed May 03, 2020).

[10]J. Qian, Z. Jin, Q. Zhang, G. Cai, and B. Liu, "A Liver Cancer Question-Answering System Based on Next-Generation Intelligence and the Large Model Med-PaLM 2," *International journal of computer science and information technology*, vol. 2, no. 1, pp. 28–35, Mar. 2024, doi: https://doi.org/10.62051/ijcsit.v2n1.04.

[11]Nikhil Kumar Gupta, A. Chaudhary, R. Singh, and R. Singh, "ChatGPT: Exploring the Capabilities and Limitations of a Large Language Model for Conversational AI," Nov. 2023, doi: https://doi.org/10.1109/icaiccit60255.2023.10465811.

[12]N. S. Pagad *et al.*, "Clinical XLNet-based End-to-End Knowledge Discovery on Clinical Text Data using Natural Language Processing," *Research Square (Research Square)*, Sep. 2022, doi: https://doi.org/10.21203/rs.3.rs-2069629/v1.

[13]R. Assaf and Rashid Jayousi, "30-day Hospital Readmission Prediction using MIMIC Data," Oct. 2020, doi: https://doi.org/10.1109/aict50176.2020.9368625.

[14]R. Kosara and S. Miksch, "A visualization of medical therapy plans compared to Gantt and PERT charts," *Proceedings Seventh International Workshop on Temporal Representation and Reasoning. TIME 2000*, doi: https://doi.org/10.1109/time.2000.856599.

[15]S. Rahman, L. Y. Jiang, S. Gabriel, Y. Aphinyanaphongs, E. K. Oermann, and R. Chunara, "Generalization in Healthcare AI: Evaluation of a Clinical Large Language Model," *arXiv (Cornell University)*, Feb. 2024, doi: https://doi.org/10.48550/arxiv.2402.10965.

[16]Sea Jung Im, Y. Xu, J. Watson, A. Bonner, H. Healy, and W. Hoy, "Hospital Readmission Prediction using Discriminative patterns," *Griffith Research Online (Griffith University, Queensland, Australia)*, Dec. 2020, doi: https://doi.org/10.1109/ssci47803.2020.9308381.

[17]S. Tang *et al.*, "Predicting 30-day all-cause hospital readmission using multimodal spatiotemporal graph neural networks," *IEEE Journal of Biomedical and Health Informatics*, pp. 1–12, Jan. 2023, doi: https://doi.org/10.1109/jbhi.2023.3236888.

[18]Y. Pawar, A. Henriksson, P. Hedberg, and Pontus Naucler, "Leveraging Clinical BERT in Multimodal Mortality Prediction Models for COVID-19," Jul. 2022, doi: https://doi.org/10.1109/cbms55023.2022.00042.

[19]Y. Labrak, A. Bazoge, E. Morin, P.-A. Gourraud, M. Rouvier, and R. Dufour, "BioMistral: A Collection of Open-Source Pretrained Large Language Models for Medical Domains," *arXiv (Cornell University)*, Feb. 2024, doi: https://doi.org/10.48550/arxiv.2402.10373.

[20]Z. Chai *et al.*, "Noise Reduction Learning based on XLNet-CRF for Biomedical Named Entity Recognition," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, pp. 1–1, 2022, doi: https://doi.org/10.1109/tcbb.2022.3157630.

Appendix A

https://github.com/Priyankaakula/Hospital-Readmission-Prediction-System-With-LLM