PATIENT CASE SIMILARITY

A PROJECT REPORT

Submitted by,

Ms. Anshita	**20201CAI0200**
Ms. M Lavanya	**20201CAI0182**
Ms. K Chaithanya	**20201CAI0183**
Ms. Dinnepati Sahithya	**20201CAI0202**

Under the guidance of,

Dr. Swati Sharma

in partial fulfillment for the award of the degree

of

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING
(Artificial Intelligence and Machine Learning)

At

PRESIDENCY UNIVERSITY

BENGALURU

JANUARY 2024

PRESIDENCY UNIVERSITY

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the Project report **"PATIENT CASE SIMILARITY"** being submitted by "**ANSHITA, M LAVANYA, K CHAITHANYA, DINNEPATI SAHITHYA**" bearing roll number(s) "**20201CAI0200, 20201CAI0182, 20201CAI0183, 20201CAI0202**" in partial fulfilment of requirement for the award of degree of **Bachelor of Technology** in **Computer Science and Engineering (Artificial Intelligence and Machine Learning)** is a bonafide work carried out under my supervision.

Dr. Swati Sharma
Associate Professor
School of CSE
Presidency University

Dr. Zafar Ali Khan
Associate Professor
School of CSE
Presidency University

Dr. C. KALAIARASAN
Associate Dean
School of CSE & IS
Presidency University

Dr. SHAKKEERA L
Associate Dean
School of CSE & IS
Presidency University

Dr. SAMEERUDDIN KHAN
Dean
School of CSE & IS
Presidency University

PRESIDENCY UNIVERSITY

SCHOOL OF COMPUTER SCIENCE ENGINEERING

DECLARATION

We hereby declare that the work, which is being presented in the project report entitled **PATIENT CASE SIMILARITY** in partial fulfilment for the award of Degree of **Bachelor of Technology** in **Computer Science and Engineering (Artificial Intelligence and Machine Learning)**, is a record of our own investigations carried under the guidance of **Dr. Swati Sharma, Associate Professor, School of Computer Science and Engineering, Presidency University, Bengaluru.**

We have not submitted the matter presented in this report anywhere for the award of any other Degree.

Anshita	**20201CAI0200**
M Lavanya	**20201CAI0182**
K Chaithanya	**20201CAI0183**
Dinnepati Sahithya	**20201CAI0202**

ABSTRACT

The pursuit of tailored care and precision has taken center stage in the ever-changing field of modern healthcare, which has led to an investigation into novel approaches to patient case analysis. Using cutting-edge Deep Neural Networks (DNNs), this research effort aims to meet this necessity by conducting a thorough investigation into patient case similarity. Known as "Synergistic Precision," our methodology explores the complex interactions between state-of-the-art computational methods and healthcare analytics with the goal of redefining how patient situations are understood.

Our study is based on the realization that conventional approaches frequently fail to capture the nuanced details that set apart individual patient cases. We suggest a paradigm change to get over this restriction by using DNNs to delve deeply into patient case databases. "Deep dive" describes our dedication to a thorough investigation, using neural network capabilities to sift through the complex layers of medical data and reveal patterns that were previously hidden.

Our approach is centered around the use of sophisticated neural network topologies for patient case similarity analysis. These systems use the breadth and depth of patient data to identify complicated relationships. Our selected DNNs work in concert with the rich tapestry of medical data, which holds the potential to reveal latent relationships that are frequently missed by more conventional analytical techniques. We want to demonstrate the effectiveness and consistency of our method in identifying patient case similarities with previously unachieved accuracy through a series of thorough trials and validations.

The "Synergistic Precision" idea highlights the comprehensive influence of our research on healthcare professionals and the wider customized medicine industry, going beyond the technical details. Clinical decision support systems are expected to be improved, which would provide medical staff members a better grasp of patient problems and enable more customized and informed treatment plans. This abstract encapsulates the essence of our research project on patient case similarity using deep neural networks. By navigating the uncharted territory at the intersection of advanced computational techniques and healthcare analytics.

ACKNOWLEDGEMENT

First of all, we indebted to the **GOD ALMIGHTY** for giving me an opportunity to excel in our efforts to complete this project on time.

We express our sincere thanks to our respected dean **Dr. Md. Sameeruddin Khan**, Dean, School of Computer Science and Engineering and Information Science, Presidency University for getting us permission to undergo the project.

We record our heartfelt gratitude to our beloved Associate Deans **Dr. C. Kalaiarasan and Dr. Shakkeera L,** School of Computer Science and Engineering and Information Science, Presidency University for rendering timely help for the successful completion of this project.

We are greatly indebted to our guide **Dr. Swati Sharma, Associate Professor** School of Computer Science and Engineering, Presidency University for her inspirational guidance, valuable suggestions and providing us a chance to express our technical capabilities in every respect for the completion of the project work.

We would like to convey our gratitude and heartfelt thanks to the University Project-II Coordinators **Dr. Sanjeev P Kaulgud, Dr. Mrutyunjaya MS** and also the department Project Coordinators **Dr. Mulari Paremeswaran**.

We thank our family and friends for the strong support and inspiration they have provided us in bringing out this project.

Anshita
M Lavanya
K Chaithanya
Dinnepati Sahithya

LIST OF TABLES

LIST OF FIGURES

TABLE OF CONTENTS

CHAPTER-1
INTRODUCTION

1.1. Introduction

The pursuit of accuracy and customized patient care has become essential in the quickly changing healthcare landscape. With the aptly named "Synergistic Precision," the research project aims to use cutting-edge Deep Neural Networks (DNNs) to decipher the complexity of patient case similarity. Through the exploration of this revolutionary methodology, the project hopes to open new avenues for creative approaches that transcend conventional analysis, transforming our understanding of medical datasets and spurring the development of personalized treatment.

In the contemporary landscape of healthcare, the pursuit of precision and individualized patient care stands as a paramount objective. The complexity and diversity inherent in patient cases necessitate innovative methodologies for analysis, prompting a profound exploration into the realm of patient case similarity. At the forefront of this endeavor is the integration of state-of-the-art Deep Neural Networks (DNNs), marking a pivotal shift in the paradigm of healthcare analytics. This research project, termed "Synergistic Precision," embarks on a comprehensive investigation into patient case similarity, aiming to redefine our understanding of medical datasets and contribute to the evolution of personalized medicine.

The imperative for precision in healthcare has been catalyzed by the realization that a one-size-fits-all approach falls short in addressing the intricacies of individual patient cases. Healthcare practitioners grapple with the challenge of discerning subtle patterns and relationships within diverse patient populations, each presenting a unique set of symptoms, medical histories, and treatment responses. In response to this imperative, the quest for innovative methodologies that transcend the limitations of traditional analysis has become a focal point. This research project stands as a testament to this pursuit, driven by the recognition that achieving precision in patient care requires a paradigm shift in the way we approach case similarity analysis.

1.2. Background and Motivation

Because patient cases in the modern healthcare system are so complex and varied, accuracy has become essential. It is difficult for traditional one-size-fits-all methods to take into account subtle differences in symptoms, medical histories, and responses to treatment. This insight acts as the impetus for our research endeavor, propelling the investigation of novel approaches to overcome the constraints of traditional analysis and offer a more nuanced comprehension of similar patient cases.

It's critical to acknowledge when a generic strategy is insufficient, particularly when handling the complexity of specific patient scenarios. It is difficult for medical professionals to identify minor trends in a variety of patient populations, which highlights the shortcomings of conventional approaches. This insight highlights the basis of our research, "Synergistic Precision," as we attempt to apply deep learning methods to handle the complex terrain of patient cases. This change of viewpoint is driven by the knowledge that while accuracy is crucial in healthcare, traditional approaches find it difficult to take into account each patient's individual needs. Our objective is to advance customized medicine by redefining patient case similarity analysis.

1.3. Architecture

The understanding that conventional approaches are inadequate for capturing the complex subtleties that set apart patient cases forms the basis of the study effort. The revolutionary paradigm change that is embodied by "Synergistic Precision," in which Deep Neural Networks (DNNs) take the lead in painstakingly examining patient case records, is expounded upon in this section. The use of DNNs marks a break from traditional methods and an understanding of the need for a more advanced and flexible method to traverse the complex world of healthcare data.

Within the context of "Synergistic Precision," the study demonstrates how state-of-

the-art Deep Neural Networks and healthcare analytics interact together. By fusing sophisticated computational methods with the complexity of patient datasets, this innovative project framework aims to push the limits of patient case similarity analysis. In addition to demonstrating a dedication to innovation, the architectural choice to use DNNs recognizes the necessity for a thorough and flexible approach to comprehend the complexity of unique patient scenarios. This project framework promises to revolutionize the field of personalized medicine and place our study at the forefront of innovation in medical dataset analysis. It also offers a deeper comprehension of medical datasets.

1.4. Activation Function, Perfomace Matric and Loss Functions

Rectified Linear Unit (ReLU) activation functions are used in our Deep Neural Networks (DNNs) for the hidden layers, and Softmax activation functions are used in the output layer. ReLU offers non-linearity and computing efficiency, which makes it easier to identify complex patterns in patient data. Applying Softmax in the output layer allows for the confident multi-class prediction by converting raw outputs into probabilities.

Our primary indicator for evaluating DNN performance is accuracy, which measures the proportion of correctly classified cases. Accompanying accuracy, recall, and the F1 score provide complex insights, especially when dealing with unbalanced datasets. Together, these metrics improve our comprehension of model performance: precision measures accurate positive predictions, recall measures real positive instances, and the F1 score gives a harmonic mean of precision and recall.

Selecting the right loss functions is essential for effective convergence in DNN training. While some specifics are left out, these functions steer the network to reduce errors, which is consistent with our objective of improving patient case similarity analysis accuracy. Our DNNs are more effective overall because of the optimal model training that results from the careful selection of loss functions.

1.5. Research objectives

To apply Deep Neural Networks, ReLU and Softmax activation functions, and accuracy, precision, recall, and F1 score as performance measures to transform patient case similarity analysis. By maximizing model convergence using strategic loss functions, the goal is to improve the precision of customized treatment and ultimately have a transformative effect on healthcare analytics.

CHAPTER-2

2. LITERATURE SURVEY

2.1. Patient-Case Similarity

This study investigates how big data mining techniques and machine learning algorithms can be combined to handle patient data efficiently inside hospital administration systems. Using a crucial part of the model, word2vec, the emphasis is on vectorization and patient-case representation. The model explores issues like overfitting in algorithms with the goal of improving their effectiveness.

The importance of word2vec is emphasized, highlighting the fact that it is made up of two more learning models in addition to one algorithm: Continuous Bags of Words (CBOW) and Skip-gram. All of these models work together to provide a thorough understanding of word embedding methods and applications.

The model broadens its focus to clarify data mining methods, emphasizing the process of removing relevant patterns and models from datasets. It highlights the importance of preprocessing to resolve problems like missing values, incomplete data, and outliers, acknowledging the crucial role that data quality plays. This emphasizes how crucial well-processed data is to the success of data mining projects.

The approach tackles the shortcoming of simple distance measurements in comparing embedded word vectors by introducing "Words Mover's Distance" (WMD). Utilizing word embeddings systematic meaningfulness, WMD treats text documents as weighted point clouds. By overcoming conventional distance measurement issues, this novel approach is specifically used to compare pre-processed information, such as patient conditions and medical staff records.

A key component of the model that allows for the classification of patients into clinically relevant subgroups is patient-case similarity. The general objective of effective patient data maintenance and exploitation for future insights is in line with

the suggested implementation of the model as a backend system for hospital management. By combining meaningful subgrouping and effective data representation, the integration of word embedding, machine learning algorithms, and data mining approaches presents a viable strategy for improving hospital administration systems and improving patient care.

2.2 netDx: Software for building interpretable patient classifiers by multi-'omic data integration using patient similarity networks

Within the field of clinical genomics, NetDx is a prominent machine learning method intended to utilize multi-modal patient data in order to create patient classifiers. This novel method is highly appealing to physicians because it provides a natural conversion of patient data into networks that graphically depict individual similarities. In addition to improving model interpretability, the resulting network-based representation helps patients develop a more complex knowledge of their interactions with one another. One of NetDx's main advantages is its outstanding performance, especially when it comes to forecasting binary cancer survival, which is a crucial characteristic in the field of oncology. Its strong prediction powers extend to situations with missing data, eliminating the requirement for imputation methods that are frequently used in other machine learning systems. Because of its dependability, NetDx appears to be a viable method for accurately estimating cancer patients' survival outcomes.

Most importantly, NetDx stands out in the field of machine learning because of its interpretability. By classifying genes into pathways, NetDx not only makes precise predictions but also provides mechanistic insights that provide light on the biological principles underlying the predictions. This element enhances the model's overall interpretability by enabling researchers and physicians to obtain insights into the complex biological mechanisms at work.

The netDx Bioconductor package, which offers built-in functions and examples for computing model performance measurements, considerably enhances the usefulness

of NetDx. To make the process easier for users, this package also comes with pre-built algorithms for creating one-step predictors from multi-modal data. The netDx Bioconductor package's adaptability makes it simple to create customized patient classifiers that can easily handle particular clinical settings.

NetDx plays a crucial role in disease diagnosis, risk assessment, and treatment evaluation in the larger context of clinical genomics. In addition to improving prediction accuracy, its distinct capacity to organize physiologically meaningful features—like genes into pathways—also helps to understand the molecular causes of various diseases. With its blend of interpretability, predictive accuracy, and adaptation to multi-modal data, this dual functionality places NetDx at the forefront of customized medicine for both researchers and clinicians. To sum up, NetDx is a strong and useful tool for clinical genomics researchers searching for better ways to diagnose and cure diseases.

2.3 On the relationship between similar requirements and similar software

This study aims to evaluate the correlation between software and requirements in the context of two industrial projects within a railway firm. To achieve this, the researchers employ various natural language processing (NLP) techniques and assess six different language models. The goal is to determine which language model is most effective in correlating the similarity between real-world requirements and the corresponding software that fulfills those requirements.

The first step involves analyzing the needs from the two industrial projects using the selected NLP techniques. These techniques likely involve processing and understanding natural language to extract meaningful information from requirements documents. The researchers then identify pairs of criteria that exhibit the highest similarity between the two projects. This comparative analysis helps in understanding commonalities and divergences in the requirements across different projects.

Trace linkages, which establish connections between requirements and the corresponding software, are then utilized to identify software pairings for each requirement pair. This is crucial for establishing a clear link between the specified requirements and the software components that fulfill them.

To measure the similarity between software pairings, the researchers use JPLag, a tool commonly employed for calculating similarity in source code. This step involves a quantitative assessment of how closely related the software implementations are for the identified requirement pairs.

The primary focus of the study is to assess the relationship between software similarity and requirements similarity. The researchers aim to determine the extent to which the similarity in software implementations aligns with the similarity in the stated requirements. This correlation analysis helps in understanding whether the identified requirements are comparable enough to be correlated with similar software solutions.

The study reveals that there is a somewhat positive association between software similarity and requirements similarity. This implies that when requirements share similarities, the corresponding software implementations also tend to exhibit similarities. Importantly, the study identifies the pre-trained BERT language model, based on deep learning, as the most effective in this context, outperforming other language models.

In summary, this study contributes valuable insights into the correlation between software and requirements using NLP techniques. The findings suggest a positive association between similarity in requirements and similarity in corresponding software, with the BERT language model showing superiority in this correlation analysis. These insights have implications for improving code retrieval processes and enhancing the understanding of how requirements are reflected in software implementations across different projects.

2.4 AI-Driven Clinical Decision Support: Enhancing Disease Diagnosis Exploiting Patients Similarity

This study introduces an innovative Clinical Decision Support System (CDS) that emphasizes the crucial role of early disease identification in facilitating precise and timely treatment. The key feature of this system is a comprehensive framework that seamlessly integrates diverse health data from various sources, including laboratory results, patient files, and even social media. This integration serves as a robust foundation for employing cutting-edge deep learning and machine learning techniques aimed at improving healthcare decision-making.

One notable aspect of the system is its utilization of a neural network model, specifically employing word embedding techniques. Word embedding is a natural language processing (NLP) method that captures semantic relationships between words, enabling a more nuanced understanding of textual data. In this case, the neural network model applies word embedding to identify semantic relationships within hospital admissions, symptoms, and diagnoses. This approach goes beyond traditional data analysis by extracting meaningful patterns and connections from the textual information.

The incorporation of word embedding in the neural network model is particularly noteworthy as it opens avenues for predicting patients' future medical problems. By understanding the semantic relationships embedded in the textual data, the model can potentially anticipate health issues that may arise in the future. This proactive approach to healthcare, based on predictive analytics, aligns with the broader goal of early disease identification and intervention.

The study distinguishes itself by introducing a special mechanism for measuring connections between various diagnoses. This unique approach provides a fresh perspective on predictive tasks within the healthcare domain. The mechanism likely enhances the accuracy and relevance of predictions by considering not only individual diagnoses but also their interconnections, offering a more holistic view of patients' health.

In summary, the forward-thinking Clinical Decision Support System presented in this study stands out for its emphasis on early disease identification and its integration of diverse health data. The use of a neural network model with word embedding capabilities adds a sophisticated layer to the analysis of hospital admissions, symptoms, and diagnoses, enabling the prediction of future medical problems. The special mechanism for measuring connections between diagnoses further enhances the system's predictive capabilities, presenting a novel and comprehensive approach to improving healthcare decision support.

2.5 Patient similarity analytics for explainable clinical risk prediction

This investigation conducted in Singapore aimed to improve the interpretability and explainability of clinical risk prediction models (CRPMs) by integrating patient similarity analytics. While CRPMs, which utilize patient attributes to estimate disease or outcomes, have become increasingly common, their decision-making processes often lack transparency. The researchers sought to enhance the transparency and comprehensibility of CRPMs by studying electronic health data from individuals with type 2 diabetes, hypertension, and dyslipidemia.

The study involved a comparison of conventional models, including support vector machine, random forest, and logistic regression. The objective was to evaluate the effectiveness of these models in predicting clinical risks associated with the specified health conditions. However, the distinguishing feature of this investigation was the incorporation of patient similarity analytics into the CRPMs.

Patient similarity analytics, in this context, involves identifying individuals with traits similar to those of the patient of interest. This approach contributes to providing more precise prognostic data by leveraging the experiences and outcomes of similar patients. The primary advantage lies in making the model more transparent and interpretable. By showcasing similar patients that were used to make predictions, the model becomes more understandable to clinicians and other stakeholders involved in

healthcare decision-making.

The real-world dataset used in the study likely comprised electronic health records of individuals with type 2 diabetes, hypertension, and dyslipidemia. These conditions were chosen as they are prevalent and often co-occur, presenting a complex scenario for risk prediction. The incorporation of real-world data adds relevance and applicability to the findings, making them more directly applicable to clinical practice.

By integrating patient similarity analytics, the researchers aimed to shed light on the decision-making processes of CRPMs and make them more interpretable for healthcare professionals. This approach not only enhances the transparency of the models but also provides a practical tool for clinicians to understand and trust the predictions made by these models. The study contributes to the ongoing efforts to bridge the gap between the increasing use of CRPMs and the need for transparency and interpretability in healthcare decision support systems.

2.6 A Novel Patient Similarity Network (PSN) Framework Based on Multi-Model Deep Learning for Precision Medicine

This paper presents a novel Patient Similarity Network (PSN) framework that is intended for use in the field of precision medicine, specifically in relation to medical records. It highlights the extremely difficult tasks involved in applying clinical machine learning models to fragmented, high-dimensional, unstructured data, like clinical notes and various medical reports. The suggested system uses a hybrid model to manage the inherent complexity of eHealth data from several sources. It attempts to address big data difficulties and decrease dimensionality by integrating an auto-encoder.

A crucial component of the framework is acknowledging the importance of incorporating various forms of biological data. The paper suggests a Patient Similarity Network Fusion-based Aggregation Model that makes use of clustering, temporal similarity considerations, triplet-loss metric learning, and deep learning approaches in order to do this. The goal of this combination is to build a framework for guided

patient similarity matching. The paper provides a critical evaluation of current patient similarity network methods, emphasizing their shortcomings in managing the complexities of medical records, the subtleties of data preparation, and the need for real-time eHealth data analysis.

Additionally, the paper offers a thorough model formulation that can be used to represent patients and determine a similarity metric using vectors produced from medical events. It explores the use of both dynamic and static data in patient similarity analysis, highlighting the significance of temporal data. The use of a hybrid model is highlighted as a way to improve accuracy. Important measures like mean absolute percentage error (MAE) and root-mean-square error (RMSE) are used to thoroughly assess the effectiveness of the suggested model.

Apart from the fundamental elements of the framework, the text delineates possible directions for enhancement. These include experimenting with different BERT model variations, accommodating values beyond nominal outcomes, solving scalability difficulties, and exploring PSN in survival analysis. The document's conclusion highlights the PSN paradigm's numerous uses in improving patient outcomes, enabling tailored treatment recommendations, forecasting clinical outcomes, and developing precision medicine.

In conclusion, the paper lays forth a thorough foundation for a precision medicine patient similarity network. It deftly tackles the complex problems related to healthcare data, putting forth a hybrid model that improves accuracy while simultaneously providing the groundwork for individualized healthcare recommendations.

2.7 Similarity of patients in predictive models using medical data: case of auto-prescription drugs for diabetic patients

In the context of leveraging medical data to automatically prescribe medications to diabetic patients, the paper examines patient similarity analysis for predictive models. It describes the creation of a system that uses machine learning methods to evaluate

patient data and determine the best course of treatment. Important actions include identifying patient similarities, deriving a distance measure to gauge patient similarity, extracting pertinent information from raw data, and choosing characteristics for prediction. The study also includes building a predictive linear regression model and clustering patients using the k-medoid algorithm.

The paper highlights the need of preprocessing data, focusing on converting attributes into vectors and filling in information gaps. The extraction of relevant distance metrics—which are essential for calculating patient similarity—is also covered. Using input vectors, a model is constructed during the learning phase, and then it is used to predict the right medication code for a new patient during the prediction phase.

A critical component of the document is the model evaluation, which includes metrics like squared correlation, modified coefficient of determination, mean squared error, and root mean square error. These indicators are essential for evaluating the model's predictive power and accuracy. The study emphasizes the significance of patient information representation and providing the machine with knowledge about a patient's health by using a multivariate linear regression method to determine the appropriate medicine for diabetic patients.

The study makes use of a Microsoft Office Access database that holds patient data, including personal information, medical diagnoses, prescriptions, and results of biochemical testing. It talks about how the database is organized and how pertinent indicators are chosen for the predictive model. The document's conclusion outlines future research directions, such as examining the relationships between diabetes and hypertension, the importance of lifestyle modifications in the management of these conditions, and the possible ramifications for more individualized healthcare and better treatment outcomes.

In conclusion, the paper offers a thorough review of predictive modeling and patient similarity analysis with regard to the auto-prescribing of medications for individuals with diabetes. It emphasizes the significance of model assessment, data preprocessing,

and machine learning approaches in the healthcare industry, providing insightful information that may have an impact on improving treatment outcomes and individualized healthcare.

2.8 Sequential Data–Based Patient Similarity Framework for Patient Outcome Prediction: Algorithm Development

This study delves into the realm of electronic medical record systems, aiming to establish a comprehensive patient similarity framework for predicting patient outcomes. The researchers recognize the challenges posed by the inconsistent, uneven, and heterogeneous nature of data within electronic medical records. To address these challenges and enhance patient similarity measurements, the study integrates both sequential and cross-sectional data. The primary objectives of the research are to predict in-hospital mortality from all causes and unanticipated readmissions due to acute myocardial infarction following discharge.

The framework incorporates three different types of patient data:

Cross-sectional data: This likely includes information about the patient at a specific point in time, such as demographic details, current medications, and general health status.

Time series data from numerous laboratory tests: This involves longitudinal data from various laboratory tests conducted over time, providing insights into the patient's evolving health conditions and trends.

Clinical event sequences with timestamps: These sequences capture the chronological order of clinical events and interventions, offering a temporal dimension to the patient data.

To compute patient similarity, the study employs a weighted total of feature similarities. This involves assessing the similarity between patients based on various features derived from the different types of data mentioned above. The weighting mechanism is particularly significant, as it assigns weights to each feature based on its predictive power for patient outcomes.

The weighting process involves evaluating how well each feature predicts which patients are more likely to experience adverse outcomes, such as in-hospital mortality or unanticipated readmissions following acute myocardial infarction. The features that demonstrate stronger predictive capabilities are assigned higher weights in the similarity calculation, emphasizing their importance in determining patient outcomes.

This approach reflects a nuanced understanding of patient similarity, considering not only the presence of shared features but also the relevance and significance of each feature in predicting specific outcomes. By integrating diverse data sources and incorporating a weighting mechanism, the framework aims to provide a more accurate and comprehensive assessment of patient similarity, ultimately contributing to the prediction of critical outcomes in the context of electronic medical record systems.

2.9 Patient-Patient Similarity-Based Screening of a Clinical Data Warehouse to Support Ciliopathy Diagnosis

This project's main objective is to improve the early detection of renal ciliopathies, a subset of uncommon monogenic illnesses defined by anomalies in the cilia—hair-like projections on cell surfaces—structure and function. It is critical to identify these diseases as soon as possible in order to start appropriate medical measures and give patients individualized care.

The suggested screening method makes use of advanced methods, such as a ranking model based on subtype-average similarity and medical concept embedding. This strategy is intended to precisely address the phenotypic variability and overlaps that are frequently observed in renal ciliopathy patients. Phenomenotypic heterogeneity, which describes the variation in symptoms across people who share the same genetic condition, makes diagnosis and classification difficult.

The technique uses language and center-specific embedding to guarantee correctness, which is a better strategy than current medical idea embedding's. Embedding allows

for computational analysis and comparison by converting medical concepts into numerical formats.

The effectiveness of the screening method is illustrated by applying it to two large, unbalanced datasets, each containing controls with renal symptoms. There are roughly 10,000 examples in one dataset, and about 60,000 cases in the other. The unbalanced character of these datasets—there are fewer cases of renal ciliopathy than controls—makes the identification process more difficult.

All things considered, the study marks a significant breakthrough in the detection of severe genetic diseases, particularly renal ciliopathies. The novel screening technique advances precision medicine and the diagnosis of rare diseases by promoting early detection, intervention, and individualized medical care.

2.10 Diabetes medication recommendation system using patient similarity analytics

A unique approach to help doctors choose the right drugs for patients diagnosed with type 2 diabetes mellitus (T2DM) is presented in the scholarly paper "Diabetes Medication Recommendation System using Patient Similarity Analytics".

Patient Similarity Approach: The suggested system makes use of a number of clinical factors and HbA1c trajectories that are taken from electronic health records (EHR) in order to implement a patient similarity approach. The technology finds people with similar traits by examining prescription drug information from six primary care clinics in Singapore.

Four different patient groups—DM just, DM with HLD, DM with HTN, and DHL—each with a unique comorbidity profile—were evaluated in order to gauge the system's efficacy. In comparison to other approaches, the findings showed a high hit ratio, recall, precision, and mean reciprocal rank. This implies that the system's recommendations for medications closely match the actual prescriptions.

The integration of the system into the PERDICT.AI user interface serves as an example of its practical implementation. The purpose of this interface is to help physicians optimize medication and facilitate diabetes consultations. When the system is integrated into actual clinical processes, it demonstrates how well it may help medical practitioners make more effective and individualized prescription selections for patients with type 2 diabetes.

The system has certain drawbacks even with its encouraging results. It does not take into account more recent drug classes and is based on historical data that might contain out-of-date prescriptions. Furthermore, the approach oversimplifies dosage levels, which might not account for the unique requirements of each patient. These restrictions point to possible areas for improvement and further research and development.

To sum up, the Diabetes Medication Recommendation System with Patient Similarity Analytics represents a noteworthy development in the area of tailored treatment for type 2 diabetes. The integration into the PERDICT.AI platform and the favorable evaluation results highlight its potential to improve patient outcomes and clinical decision-making. Nonetheless, continuous endeavors to tackle its constraints will be imperative to guarantee its sustained significance and efficacy in the ever-changing terrain of diabetes care.

CHAPTER-3

3.RESEARCH GAPS OF EXISTING METHODS

Within the rapidly developing field of patient case similarity analysis, clustering and closest neighbor techniques have been widely used. A careful analysis of these approaches, however, identifies significant research gaps that cast doubt on their broad application and efficacy in healthcare settings. In order to progress the field of patient case similarity and guarantee the provision of precise and significant insights for clinical decision-making, it is essential to recognize and close these gaps.

A fundamental component of customized medicine and healthcare analytics, patient case similarity presents a number of difficulties for the current approaches, leading to the identification of important research gaps:

1. **Sensitivity to outliers**:

   In patient datasets, approaches such as nearest neighbor and clustering are extremely susceptible to outliers. Atypical or aberrant examples known as outliers can have a major effect on how accurately similarity evaluations are made. When there are outliers, especially in situations where certain cases have distinct features, the overall understanding of patient case similarity may be distorted or misclassified.

2. **Curse of dimensionality**:

   The curse of dimensionality is one of the main drawbacks of closest neighbor algorithms and clustering techniques. These approaches are ill-suited for high-dimensional spaces, where data frequently involves many aspects, as in the context of patient case similarities. The algorithm could incorrectly classify certain cases as similar when they actually differ significantly. This constraint impedes the precision and dependability of patient case similarity evaluations, particularly in scenarios when the data's dimensionality is intrinsic.

3. **Parameter tuning**:

   Optimizing the performance of nearest neighbor and clustering algorithms requires meticulous parameter tweaking. It can be difficult and subjective to change parameters, like the number of neighbors or clustering thresholds. Inadequate

parameter selection can produce less-than-ideal outcomes and reduce the utility of patient case similarity evaluations. The implementation becomes more difficult and may call for domain expertise due to the requirement for precise parameter tweaking.

4. **Lack of interpretability**:

   These techniques frequently lack interpretability, which makes it difficult to understand the reasoning behind the similarity evaluations. For medical practitioners to believe the results and incorporate them into clinical decision-making, the data must be interpreted clearly. For similarity approaches to be used effectively in healthcare settings, it is essential to comprehend the reasons behind the classification of some patient situations as similar.

5. **High computational cost/ Scalability**:

   Clustering and nearest neighbor techniques can be computationally costly, particularly when working with big patient datasets. Scalable applications cannot use these methods because of the exponential rise in processing costs associated with dataset sizes. High computational needs can make it difficult to execute these techniques in real-time or almost real-time, which limits their applicability in major healthcare systems.

In summary, although the closest neighbor and clustering methods have been extensively employed for assessing patient case similarity, these methods have certain limitations that must be recognized and addressed in order to improve the practicality and robustness of similarity assessments in healthcare settings. To lessen these restrictions, consideration of alternate approaches or the incorporation of additional techniques would be necessary.

CHAPTER-4

4.PROPOSED METHODOLOGY

4.1 Methodologies

Our suggested methodology is a ground-breaking attempt to improve the accuracy and interpretability of similarity analyses in healthcare, in response to the shortcomings found in the patient case similarity methodologies now in use. This methodology, which is based on a combination of cutting-edge machine learning algorithms and domain-specific insights, aims to completely transform the field of patient-centric analytics.

Data Collection and Pre-processing :

The basis of our work is the extensive assemblage of patient cases, which include a wide range of medical diseases and situations. To do this, we must gather datasets from a variety of places, such as medical facilities, clinics, and research centers. Thorough pre-processing is used to the gathered data in order to resolve problems like missing values, outliers, and normalization and guarantee the consistency and quality of the dataset..

Feature Extraction:

Finding pertinent aspects is essential to successfully describing each patient scenario. This step entails a careful examination of the dataset to identify features that capture crucial data for the similarity analysis. These characteristics could include treatment techniques, diagnostic testing, medical history, and demographic data. Developing a strong feature set that captures the subtleties of every single instance is the aim.

Evaluation Metric:

Selecting a suitable similarity measure is essential for precisely measuring the similarities between patient cases. Depending on the qualities of the chosen features and the nature of the data, metrics like the Jaccard index, cosine similarity, and

Euclidean distance may be taken into consideration. The chosen measure should provide a meaningful depiction of patient case similarities while also taking into account the complexities of healthcare data.

Algorithm Selection:

Choosing the machine learning or data mining methods that will be used for patient case similarity analysis is the next stage. This could entail more sophisticated methods like support vector machines or neural networks, or more conventional clustering algorithms like k-means or hierarchical clustering. The algorithm selected should be in line with the goals of the similarity analysis as well as the complexity of the data.

Model Training and Evaluation :

To make training and assessing the similarity model easier, the dataset is divided into training and testing sets. The model's performance is evaluated using strict measures including area under the receiver operating characteristic (ROC) curve, F1 score, precision, and recall. Cross-validation procedures are taken into consideration to ensure robustness, offering a thorough assessment of the model's generalizability.

DNN (Deep Neural Network) :

Deep Neural Networks (DNNs) are a revolutionary force in the quickly developing field of artificial intelligence, revolutionizing machine learning. DNNs are the ultimate in innovation, taking their cues from the complex neural networks seen in the human brain. Their depth and ability to learn hierarchical representations are essential for tackling challenging issues in a variety of fields, including natural language processing and picture and audio recognition. This succinct overview offers an overview of the fundamental ideas and enormous importance of DNNs, setting the stage for a thorough investigation of their operational details and architectural design and demonstrating their role in propelling previously unheard-of advances in artificial intelligence

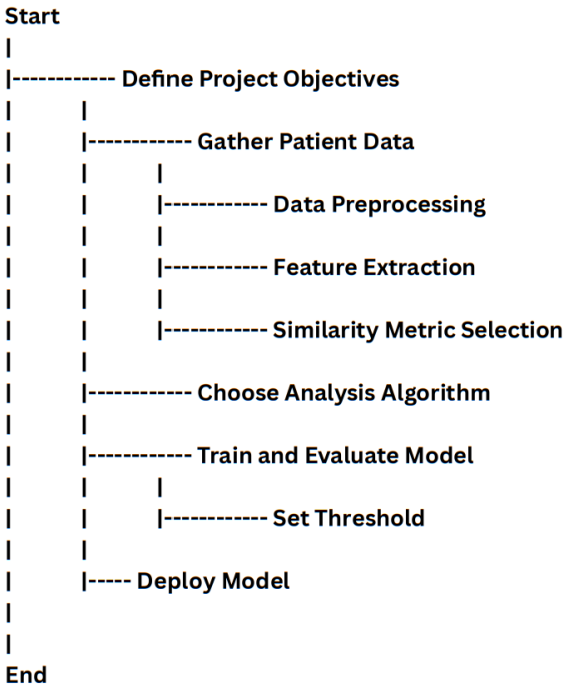


Figure 4.1 – work flow of the model

4.2 ADVANTAGES

Identification of Rare Diseases:

When symptoms are infrequent or follow unusual patterns, patient case similarity analysis becomes an essential tool for diagnosing rare diseases or syndromes. The technology compares patient cases to identify minute similarities, which helps medical personnel identify and comprehend unusual disorders more quickly than they could using conventional diagnostic techniques.

Personalized Treatment Plans:

Personalized and focused treatment regimens can be developed more easily with the help of sophisticated analysis of similar patient instances. By examining the experiences and reactions of similar patients, medical professionals might gain valuable insights into successful intervention strategies. Treatment efficacy and patient outcomes may be improved by more closely matching treatments to the particulars of each case thanks to this customized approach.

Outcome Prediction:

Potential outcomes can be predicted using patient case similarity analysis and past data. With the use of this predictive capabilities, healthcare professionals may better anticipate problems, make proactive adjustments to treatment regimens, and support patients and their families as needed. Healthcare professionals can maximize patient care and resource allocation by utilizing this insight.

Feedback for Continuous Improvement:

A dynamic feedback loop is established by the repetitive nature of patient case similarity analysis, which enables the system to continuously learn and adapt as additional data is gathered and examined. Over time, this flexibility improves the model's relevance and accuracy, creating a self-correcting system that keeps up with changing patient demographics and medical advances.

Mitigation of Diagnostic Uncertainty:

When there is confusion about the diagnosis or unclear symptoms, similarity analysis is a useful auxiliary. Healthcare practitioners can narrow down possible diagnoses and direct subsequent investigations with the assistance of the analysis, which provides extra context through comparison information. This function plays a major role in lowering diagnostic uncertainty and accelerating the process of arriving at fast and correct diagnoses.

4.3 DISADVANTAGES

Inaccurate or Incomplete Data:

The accuracy and completeness of healthcare data are critical to the effectiveness of similarity analysis. The accuracy of case similarity assessments might be jeopardized by inconsistencies, omissions, or missing data in Electronic Health Records (EHRs). Ensuring the integrity of similarity models necessitates addressing data quality challenges.

Patient Privacy Issues:

There are a lot of privacy and data security issues with the integration and analysis of sensitive health data. Strict adherence to privacy laws is necessary to strike a balance between obtaining insightful data and protecting patient privacy. Strong security measures must be put in place in order to guard patient record confidentiality and stop illegal access.

Temporal Changes:

Both patient traits and health problems are dynamic and change throughout time. Conventional similarity models might not be able to adjust to temporal variations, which could cause discrepancies when comparing cases from the past and present. In order to improve the relevance and precision of patient case similarity analysis, temporal dynamics must be taken into account.

Informed Consent Challenges:

Using patient data for similarity analysis brings up ethical issues, especially with relation to informed consent. It is crucial to have patient consent before using their data for research. In order to implement patient case similarity systems responsibly, it is imperative to navigate the ethical subtleties that may occur while attempting to obtain express agreement

Legal and Regulatory Challenges:

Creating and implementing patient case similarity systems involves several moving parts, one of which is complying with legal and regulatory standards. Respecting privacy laws, healthcare regulations, and standards pertaining to patient data protection is crucial. It is imperative to navigate the complex terrain of legal and regulatory compliance in order to guarantee the moral and legitimate use of patient data for similarity analysis.

To sum up, the responsible and efficient application of patient case similarity analysis depends on the recognition and mitigation of these drawbacks. The ethical and accurate

use of healthcare data in similarity models depends on addressing data quality challenges, protecting patient privacy, adjusting to temporal changes, obtaining informed consent, and navigating legal and regulatory landscapes.

CHAPTER-5

5. OBJECTIVES

In an effort to improve healthcare systems, this paper examines a novel project that uses patient similarity analysis to improve many facets of clinical decision-making. Through the use of an advanced Clinical Decision Support System (CDSS), the initiative seeks to empower physicians, aid in early disease diagnosis, lessen diagnostic ambiguity, better patient outcomes, and provide superior care and support.

1. Improved clinical decision making:

Giving medical professionals relevant information from cases comparable to their own is one of the main goals. By providing them with information that might greatly impact their decision-making, this method seeks to improve the ability of doctors to make diagnoses and treatments. The CDSS becomes a useful instrument for improving the overall standard of clinical decision-making by utilizing sophisticated algorithms.

2. Early disease detection and research:

The project lays a lot of focus on how patient similarity analysis can help with research efforts and early disease diagnosis. The CDSS aids in the prompt identification of illnesses by seeing patterns in instances that are similar to one another. In addition, it acts as a link between patients and pertinent research opportunities, which promotes cooperation between medical professionals and academic institutions.

3. Reduced uncertainty and improved outcomes:

Reducing diagnostic uncertainty, improving prognosis estimates, and eventually improving patient outcomes all depend on the use of comparable patients. The CDSS seeks to be a dependable partner in the healthcare decision-making process by offering insightful information that may result in more focused and efficient medical interventions. Reducing ambiguity is essential to improving the general standard of patient care.

4. Enhanced care and patient support:

The project's goals are expanded to include enhancing the continuum of care. Through the process of finding shared experiences, the CDSS helps patients connect with one another and build support systems. This method not only improves patient care but also encourages the model to learn continuously, allowing it to adjust and change in response to actual patient experiences.

5. Ethical considerations:

A crucial component of the project is placing a strong emphasis on ethical and privacy legislation compliance as well as responsible data handling. The CDSS places a high priority on data integrity and confidentiality since it understands how sensitive patient information is. Building trust between patients and healthcare providers requires ethical considerations above all else in order to ensure the safe and appropriate application of this cutting-edge technology.

CHAPTER-6

SYSTEM DESIGN & IMPLEMENTATION

6. SYSTEM DESIGN

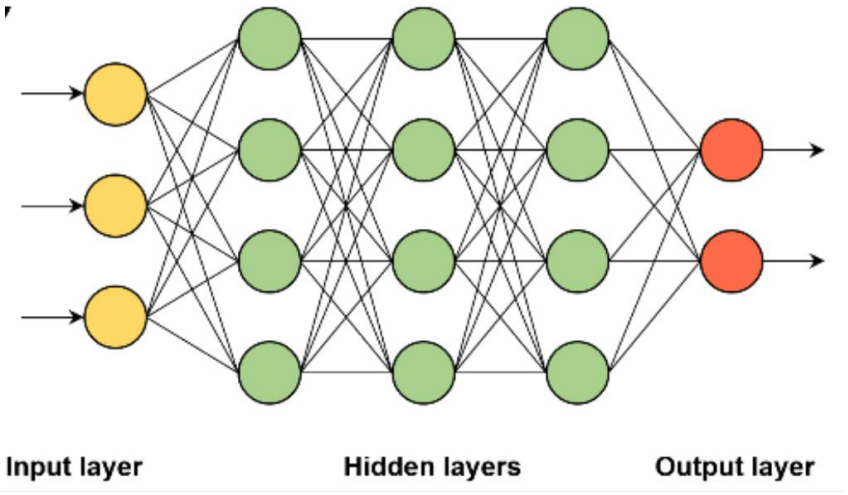


figure 6.1 architecture of the Deep Neural Network **Figure 6.2:**

Use-Case diagram for Patient case similarity

6.1 Architectural Design

The architectural design of Deep Neural Networks (DNNs) is a key component driving machine learning advances in the quickly changing field of artificial intelligence. DNNs are an advanced class of algorithms inspired by the complex neural networks found in the human brain. This study explores the fundamental aspects of DNN architecture, explaining how the clever placement of layers, neurons, and connections enables these networks to perform well in a variety of applications, ranging from complex decision-making to image recognition.

A layered structure consisting of an input layer, hidden layers, and an output layer is at the center of DNNs.

Input layer:

The neural network receives raw data input from the input layer. In this layer, each neuron stands for a distinct characteristic, and when combined, they create an input vector that contains the information contained in the input data. These neurons' values serve as the first input for the layers that follow.

Hidden Layer:

The hidden layers, which are the foundation of the network's capacity to learn complex representations, are located between the input and output layers. When these layers are stacked, the raw input is gradually stripped of its abstract and hierarchical properties. A densely linked network is created when every neuron in a hidden layer is connected to every other neuron in the layers above and below. The network's ability to recognize and interpret complicated patterns in the data is significantly influenced by the depth of these hidden layers.

Neurons and Weights:

Neurons are essential to every layer. After receiving inputs, neurons apply a weighted sum and then turn on an activation function. The weights attributed to the connections among neurons hold the secret to flexibility. These weights are iteratively changed during the training phase in order to reduce the discrepancy between the expected and actual outputs. The network is able to learn and optimize its prediction capabilities through the use of optimization algorithms such as gradient descent to achieve this modification.

Learning Abstract representations:

The network may gradually learn abstract features because to the hidden layers' hierarchical design. Deeper layers can learn intricate combinations of these traits, whereas lower levels may only learn simple patterns like edges or basic shapes. The network may convert unprocessed input into higher-level representations by using weights, which are flexible enough to capture complex patterns and subtleties in the data.

Predictive Prowess Optimization:

Optimizing the network's prediction ability is the main objective of the adaptive weights and the interconnected neural structure. The network improves its comprehension of the connections between input features and the desired output as a result of learning. Iteratively modifying the weights during the optimization process improves the network's capacity for precise prediction-making.

Essentially, a DNN's layered structure, networked neurons, and adaptive weights enable it to convert unprocessed information into meaningful and abstract representations, which opens up new possibilities for advanced learning and prediction.

6.2 Use Case Diagram

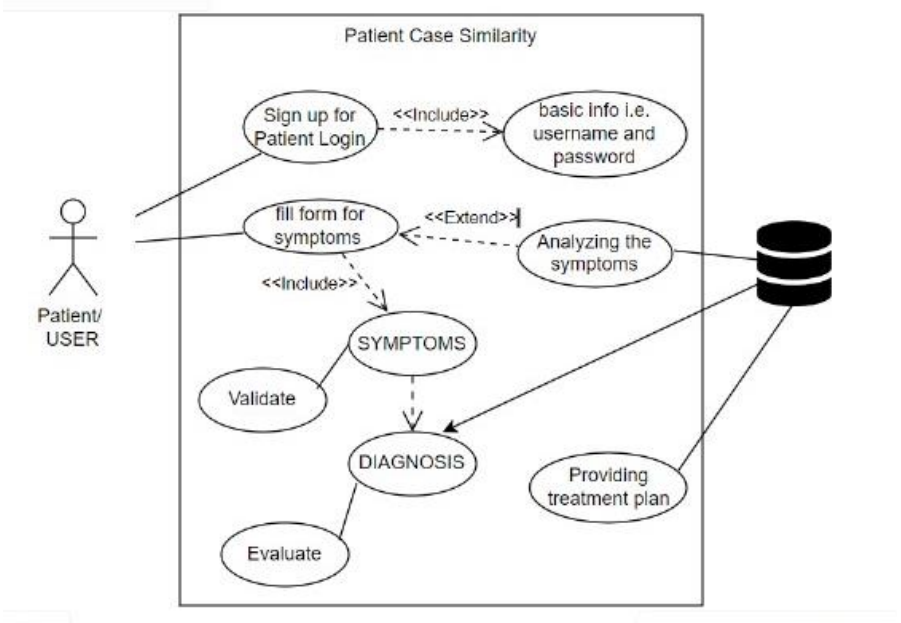


Figure 6.2: Use-Case diagram for Patient case similarity

A patient case similarity system's use case diagram shows the different interactions that take place between actors and the system.

Actors:

System:

The intelligent software program known as the patient case similarity system was created with the purpose of analyzing and contrasting patient instances. It uses data analytic methods and deep learning algorithms to find patterns in cases, giving medical professionals insightful information. In order to effectively store and retrieve patient data, the system also communicates with a database.

Database:

The database, which is in charge of storing, organizing, and retrieving patient data, is an essential part of the patient case similarity system. It guarantees accessibility, security, and integrity of data. Similarity analysis is based on a variety of patient data that is stored in the database, including symptoms, medical history, and treatment results.

Users/Patient:

Initial patient data is entered by the healthcare professional, and via an easy-to-use interface, the patient or user can provide more information about their symptoms and medical background.

Use-Cases:

Input Patient Data:

Healthcare professionals enter vital patient data, including as symptoms and medical history, into the patient case similarity system first. In addition, patients or users actively contribute more details using an intuitive interface, adding their distinct viewpoints to the collection. By working together, we can guarantee a thorough and varied patient profile, which improves the system's capacity to identify patterns and offer more individualized healthcare insights.

Retrieve Patient Data:

Patients or users, as well as healthcare professionals, can ask the system to obtain patient cases that are similar to the one that is being analyzed at the moment. Through easier access to similar cases, this collaborative feature encourages shared decision-making by giving patients a better grasp of possible outcomes

and healthcare professionals a chance to learn from similar situations. In order to select and present pertinent examples throughout the retrieval process, the system makes use of its patient case similarity algorithms, which promotes a cooperative approach to decision support in the medical field.

Adjust Similarity Parameters:

Healthcare professionals can adjust and modify similarity factors in the system based on a combination of patient-specific data and clinical knowledge. By adjusting the sensitivity and specificity of the similarity analysis, practitioners can better customize it to the unique circumstances of each patient instance. The system offers a flexible framework for healthcare practitioners to adjust and enhance the parameters, resulting in a more precise and contextually relevant patient case similarity analysis. This is achieved by adding clinical insights and particular patient characteristics. This degree of personalization improves the system's accuracy and efficiency in assisting with healthcare decision-making.

System Maintenance:

The system's ongoing dependability, optimal performance, and usefulness are guaranteed by routine maintenance. In order to improve algorithm accuracy and include the most recent results from medical research, tasks include software updates, bug fixes, and enhancements. In addition, database integrity, system resources, and general health are monitored and managed. This methodical approach ensures the effectiveness of the system, offering users and healthcare professionals a smooth experience and bolstering its long-term viability in the ever-changing healthcare environment.

Diagnosis/Treatment plan:

Healthcare professionals use the patient case similarity method in the diagnosis use case to make accurate medical diagnoses. Through the entry of patient-specific data, the system utilizes sophisticated algorithms to locate comparable cases throughout its database. Examining the system's recommendations, the medical professional learns possible diagnosis based on patterns seen in similar

situations. Professionals are empowered to make proper diagnosis decisions with this data-driven method.

A high-level summary of the interactions inside the patient case similarity system is given by this use case diagram.

The main user interacting with the system is the healthcare provider.

The activities that medical professionals might carry out within the system are used to define use cases.

This graphic can be further customized based on the unique features and interactions associated with our patient case similarity system. It is meant to be used as a starting point.

6.3 Dataflow Diagram

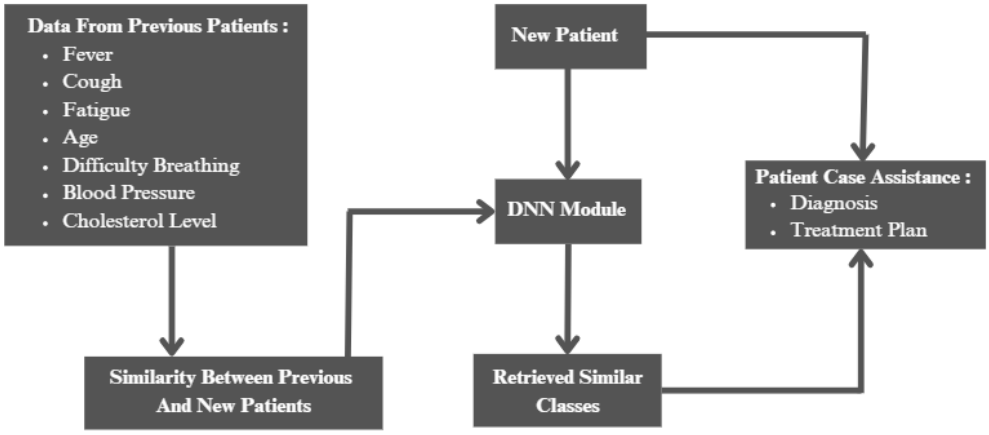


Figure 6.3- Data flow of the model

- **Input Patient Data :**

  Healthcare professionals enter vital patient data, including as symptoms and medical history, into the patient case similarity system first. In addition, patients or users actively contribute more details using an intuitive interface, adding their distinct viewpoints to the collection. By working together, we can guarantee a thorough and varied patient profile, which improves the system's capacity to identify patterns and offer more individualized healthcare insights.

- **Retrieve Patient Data :**

  Patients or users, as well as healthcare professionals, can ask the system to obtain patient cases that are similar to the one that is being analyzed at the moment. Through easier access to similar cases, this collaborative feature encourages shared decision-making by giving patients a better grasp of possible outcomes and healthcare professionals a chance to learn from similar situations. In order to select and present pertinent examples throughout the retrieval process, the system makes use of its patient case similarity algorithms, which promotes a cooperative approach to decision support in the medical field.

- **Adjust Similarity Parameters :**

  Healthcare professionals can adjust and modify similarity factors in the system based on a combination of patient-specific data and clinical knowledge. By adjusting the sensitivity and specificity of the similarity analysis, practitioners can better customize it to the unique circumstances of each patient instance. The system offers a flexible framework for healthcare practitioners to adjust and enhance the parameters, resulting in a more precise and contextually relevant patient case similarity analysis. This is achieved by adding clinical insights and particular patient characteristics. This degree of personalization improves the system's accuracy and efficiency in assisting with healthcare decision-making.

- **System Maintenance :**

  The system's ongoing dependability, optimal performance, and usefulness are guaranteed by routine maintenance. In order to improve algorithm accuracy and include the most recent results from medical research, tasks include software updates, bug fixes, and enhancements. In addition, database integrity, system resources, and general health are monitored and managed. This methodical approach ensures the effectiveness of the system, offering users and healthcare professionals a smooth experience and bolstering its long-term viability in the ever-changing healthcare environment.

- **Diagnosis/Treatment plan :**

Healthcare professionals use the patient case similarity method in the diagnosis use case to make accurate medical diagnoses. Through the entry of patient-specific data, the system utilizes sophisticated algorithms to locate comparable cases throughout its database. Examining the system's recommendations, the medical professional learns possible diagnosis based on patterns seen in similar situations. Professionals are empowered to make proper diagnosis decisions with this data-driven method.

IMPLEMENTATION

Model Name : Patient Case Similarity

6.4 Model Description:

We are performing label encoding to the Output columns of the dataset using the LabelEncoder from the scikit-learn library. Additionally, we are saving the trained label encoder to a file using joblib.

Label encoding is useful when we have categorical labels that we want to convert into a format suitable for machine learning models. It assigns a unique integer to each distinct label in the Output columns.

Saving the label encoder is important because, during inference or when making predictions on new data, we'll need to encode the labels in the same way as during training. Loading the label encoder from the saved file ensures consistency in the encoding. And for the numerical data we are using Standard Scaler, it is a preprocessing technique used in machine learning to standardize the features of a dataset. Standardization is performed independently for each feature, ensuring that each feature has a mean of 0 and a standard deviation of 1. This process is also known as z-score normalization.

The input layer for the neural network consists of nodes equal to the number of

features in the dataset. It's important to match the input shape with the dimensionality of our data. The hidden layers consists of three dense (fully connected) layers. Each layer has a specified number of neurons (64, 32, and 16, respectively) and uses the rectified linear unit (ReLU) activation function. The output of each layer serves as the input for the next layer. Two separate output layers are defined, one for Diagnosis and one for Treatment Plan. Each output layer has 8 neurons and uses the softmax activation function.

ReLU, which stands for Rectified Linear Unit, is an activation function commonly used in artificial neural networks, including deep learning models.

$$f(x)=\max(0,x)$$

In simpler terms, the output of the ReLU function is the input value if it is positive, andzero otherwise. Graphically, it looks like a ramp, allowing all positive values to pass through unchanged and setting all negative values to zero.

The softmax activation function is commonly used in the output layer of neural networks for multi-class classification problems. It converts a vector of raw scores probability distribution over multiple classes.

```
Model: "model"
_________________________________________________________________________________________________
 Layer (type)                Output Shape              Param #   Connected to
=================================================================================================
 input_1 (InputLayer)        [(None, 7)]               0         []

 dense (Dense)               (None, 64)                512       ['input_1[0][0]']

 dense_1 (Dense)             (None, 32)                2080      ['dense[0][0]']

 dense_2 (Dense)             (None, 16)                528       ['dense_1[0][0]']

 diagnosis_output (Dense)    (None, 8)                 136       ['dense_2[0][0]']

 treatment_output (Dense)    (None, 8)                 136       ['dense_2[0][0]']

=================================================================================================
Total params: 3392 (13.25 KB)
Trainable params: 3392 (13.25 KB)
Non-trainable params: 0 (0.00 Byte)
_________________________________________________________________________________________________
None
```

Figure 6.4 – Model dynamics

6.5Model compilation process:

The key elements to note are the optimizer, the loss functions, and the metrics.

- **Optimizer:**

  By modifying the model's weights in response to the computed gradients during backpropagation, the optimizer is a key component in the training process of a neural network. The Adam optimizer is used in our model, and its learning rate is set to 0.001. Adam is a well-liked optimization method because of its adaptive learning rate mechanism, which modifies the learning rates for each parameter separately. Adam can converge more quickly and successfully manage a variety of data kinds and topologies thanks to his versatility. In this instance, the optimizer selection strikes a balance between convergence speed and computing economy, making it appropriate for a variety of deep learning applications.

- **Loss Functions:**

  For both output layers, the sparse categorical cross-entropy loss function is used. When the labels in a multi-class classification issue are integers, this loss function works effectively. It computes, in particular, the cross-entropy loss between the true labels and the probabilities that are predicted. For tasks requiring several classes, the sparse categorical cross-entropy is efficient since it is recommended in situations where each sample belongs to only one class. In order to ensure accurate classification and convergence towards an optimal model, the training process must be guided by the selection of a suitable loss function that minimizes the difference between the predicted and true labels.

- **Metrics:**

  The accuracy of the model is chosen as the metric for assessing its performance during training. A popular indicator for classification issues, accuracy gives a clear indication of the percentage of samples that are successfully identified. The ratio of accurately predicted occurrences to all instances in the dataset is computed. Although accuracy is a useful metric, the nature of the situation at hand must be taken into account. Additional metrics like precision, recall, and F1-score may be taken into consideration

in classification problems with imbalanced classes in order to provide a more thorough

Metric	Formula	Interpretation
Accuracy	(TP + TN) / (TP + TN + FP + FN)	Proportion of correctly classified instances
Precision	TP / (TP + FP)	Proportion of correctly predicted positive instances
Recall (Sensitivity)	TP / (TP + FN)	Proportion of actual positive instances predicted
F1 Score	2 * (Precision * Recall) / (Precision + Recall)	Harmonic mean of Precision and Recall
ROC AUC	-	Area under the Receiver Operating Characteristic curve
Confusion Matrix	-	Matrix showing True Positive, True Negative, lse Positive, and False Negative

insight of model performance.

Figure 6.5- evaluation metrics of the model

6.6 Data Set:

Dummy Dataset:

Our dataset was produced artificially or synthetically just for our project. It involves characteristics specific to the patient, such as blood pressure, cholesterol, age, fever, coughing, and exhaustion.

With this dataset, we've included data augmentation to make it more diverse and capable of simulating a wider range of situations.

Data Augmentation:

Objective:

The process of artificially expanding a dataset's size and diversity is known as data augmentation.

By subjecting the model to more diverse data transformations, the intention is to improve the model's capacity for generalization.

Practical Methods:

Random rotations, flips, zooms, shifts, and adjustments to brightness or contrast are a few examples of augmentation techniques. We may also take into account perturbations for healthcare data, which resemble possible differences in patient conditions or measurement errors.

Execution:

In order to replicate various patient scenarios, data augmentation has been conducted to the synthetic patient data in our project. This involves adding differences in feature values. This makes it possible for the model to learn from a wider range of instances, which could enhance its performance on unseen real-world data.

Diagnosis:

We have added a diagnosis label to our dataset, which indicates each patient's state of health or sickness. We have encoded the diagnosis, which is a categorical variable, into a numerical format for model training using a LabelEncoder.

A classification model must be trained using an appropriate diagnosis label, and both model training and model evaluation employ the encoded values.

Plans of Treatment:

We've included a label for each patient's recommended course of action, which is similar to the diagnosis. We've used a LabelEncoder to convert the treatment plan, which is a categorical variable as well, into a format that can be used for model training. This gives our model a multi-output feature in which it can forecast a patient's diagnosis as well as their course of therapy.

Our project's main objective appears to be using patient case similarities to make healthcare predictions.

The goal of the synthetic dataset, data augmentation, and diagnosis and treatment plan inclusion is to build a model that can anticipate and suggest the best course of action for patients depending on their characteristics.

```
    'fever': fever,
    'cough': cough,
    'fatigue': fatigue,
    'age': age,
    'gender': gender,
    'Difficulty Breathing': difficulty_breathing,
    'Blood Pressure': blood_pressure,
    'Cholesterol Level': cholesterol_level,
    'diagnosis': diagnosis,
    'treatment_plan': treatment_plan
})

data.to_csv('synthetic_patient_data_textual.csv', index=False)

import os

current_directory = os.getcwd()
print(f"Current Working Directory: {current_directory}")
```

6.7 Frontend

```
import pandas as pd
import random

# Set seed for reproducibility
random.seed(42)

# Generate synthetic data for the dataset
num_samples = 1000

# Generate random values for each column
fever = [random.uniform(96.0, 104.0) for _ in range(num_samples)]
cough = [random.choice([0, 1]) for _ in range(num_samples)]
fatigue = [random.choice([0, 1]) for _ in range(num_samples)]
age = [random.randint(18, 80) for _ in range(num_samples)]
gender = [random.choice(['Male', 'Female']) for _ in range(num_samples)]
difficulty_breathing = [random.choice([0, 1]) for _ in range(num_samples)]
blood_pressure = [random.uniform(90, 140) for _ in range(num_samples)]
cholesterol_level = [random.uniform(120, 300) for _ in range(num_samples)]

# Generate random values for diagnosis and treatment_plan
diagnosis_options = ['Eczema', 'Asthma', 'Hyperthyroidism', 'Allergic Rhinitis', 'Diabetes', 'Liver Cancer', 'Urinary Tract Infection', 'Hepatitis']
treatment_options = ['Plan 1', 'Plan 2', 'Plan 3', 'Plan 4', 'Plan 5', 'Plan 6', 'Plan 7', 'Plan 8']

diagnosis = [random.choice(diagnosis_options) for _ in range(num_samples)]
treatment_plan = [random.choice(treatment_options) for _ in range(num_samples)]

# Create a DataFrame
data = pd.DataFrame({
    'fever': fever,
    'cough': cough,
    'fatigue': fatigue,
    'age': age,
    'gender': gender,
    'Difficulty Breathing': difficulty_breathing,
    'Blood Pressure': blood_pressure,
    'Cholesterol Level': cholesterol_level,
```

6.8 Streamlit:

As the front-end framework of choice, Streamlit is essential to the creation of the patient case comparison application because it provides a functional and intuitive

interface that facilitates easy interaction with the underlying machine learning model. The main goal is to develop a user-friendly web application that enables users to compare patient situations with ease. Streamlit's easy-to-use Python API makes it easier to create an interface that is intuitive for users with different levels of programming experience. Though Streamlit is primarily a front-end framework, it easily combines with back-end features, allowing the machine learning model to be included for accurate predictions. To provide a thorough and dynamic user experience, this entails managing user inputs, processing data, carrying out predictions, and presenting results.

The feature set of Streamlit also plays a part in the success of the program. Because of its API's simplicity, developers—including data scientists and machine learning practitioners—can concentrate on demonstrating models instead of dealing with the difficulties of complex web construction. Streamlit effortlessly manages the lawet, interaction widgets, and data visualization elements, guaranteeing a well-organized and visually appealing presentation of patient case comparisons. Because of the framework's code efficiency and accessibility, developers may create web apps quickly and with less code, which saves time and effort during the development process. The strategic choice to incorporate Streamlit into the patient case comparison application improves the product's usability, accessibility, and overall efficacy for both developers and users.

Patient Diagnosis and Treatment Prediction

Fever

98.00 − +

Cough

- (•) No
- ( ) Yes

Fatigue

- ( ) No
- (•) Yes

Age

28.00 − +

Difficulty Breathing

- ( ) No
- (•) Yes

Blood Pressure

82.00 − +

Cholesterol Level

150.00 − +

Predict

Diagnosis Prediction: Eczema, Treatment Prediction: Nasal corticosteroid sprays

CHAPTER-7

7.TIMELINE FOR EXECUTION OF PROJECT (GANTT CHART)

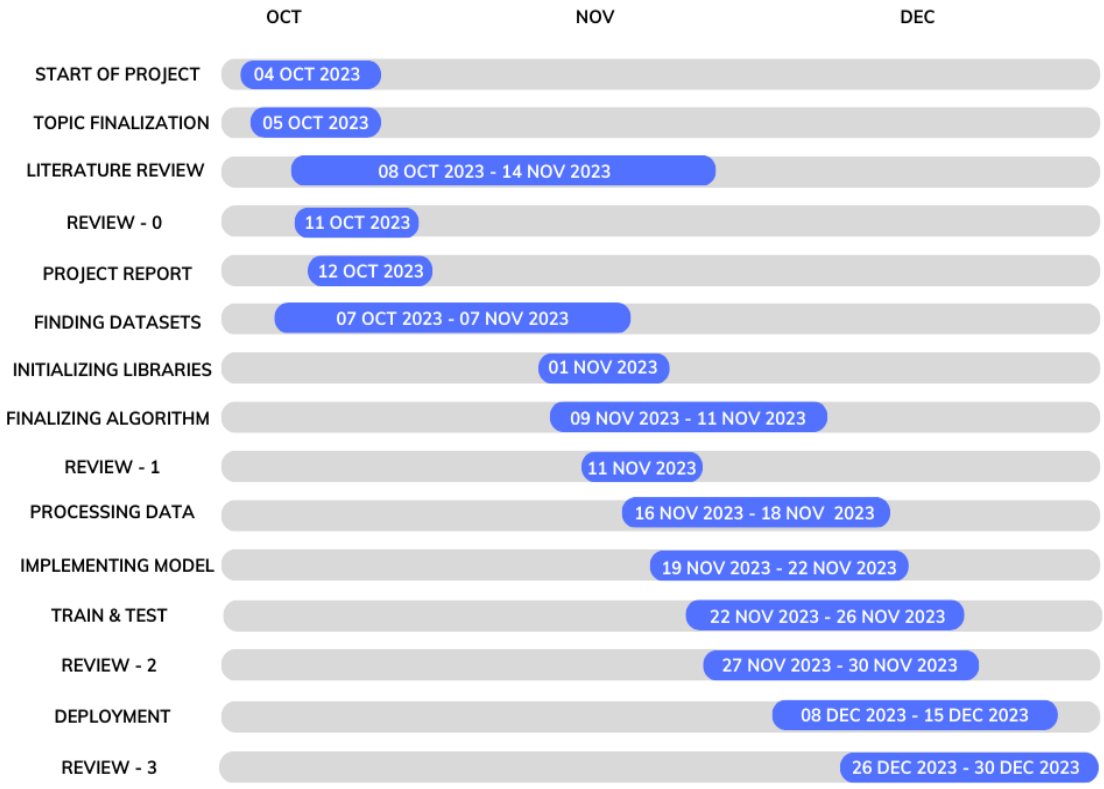


FIGURE 7.1 Gantt Chart

CHAPTER-8
8.OUTCOMES

1. **Personalized Medicine**:

   Personalized medicine is a revolutionary method that customizes medical interventions to each patient's specific needs. Healthcare professionals can improve treatment methods for better patient outcomes by identifying similar patient occurrences. In order to customize treatment regimens based on comparable experiences, this entails reviewing past patient data to find cases with similar characteristics. By taking a customized approach, medical interventions become more effective and efficient, resulting in more focused and targeted healthcare solutions.

2. **Treatment Effectiveness:**

   Analyzing similar patient cases and their treatment results together offers important insights into how well different approaches work. By using this information, medical professionals can choose the best treatments for particular disorders with knowledge. In order to make the best decisions possible and guarantee that patients receive the most effective and evidence-based therapies possible, practitioners can evaluate the success or failure of previous treatments in situations that are comparable to their own.

3. **Research and Knowledge Discovery:**

   Medical research benefits greatly from the analysis of patient case similarity since it reveals patterns, trends, and connections in healthcare data. This data can be utilized by researchers to develop theories, carry out more targeted investigations, and advance their knowledge of illnesses and the treatments that address them. Being able to recognize comparable patient instances makes it easier to find new insights, which promotes the growth of medical knowledge and keeps healthcare practices evolving.

4. **Quality Improvement**:

   Finding opportunities for improvement in the delivery of healthcare is made easier by

comparing healthcare outcomes based on patient case similarity. Healthcare organizations can improve protocols, practices, and standards of care by examining cases with less than ideal results or differences in treatment responses. By using a data-driven strategy, healthcare organizations can improve the overall quality and safety of patient care by receiving valuable input for their quality improvement efforts.

5. **Risk Stratification**:

Similar patient cases help stratify people according to risk factors, which enables medical professionals to predict outcomes and carry out early detection and prevention plans. Healthcare professionals can proactively address possible health risks, customize monitoring programs, and adopt individualized preventative interventions by finding trends among patients with similar risk profiles. By making targeted interventions at the appropriate times, this risk stratification strategy is essential for improving overall patient outcomes and maximizing the allocation of resources.

CHAPTER-9

9. RESULTS AND DISCUSSIONS

The basic results and discussions of our project is as follows:

1. **Diagnostic Accuracy and Model Performance:**

   In evaluating patient case similarity, our deep learning model achieved a diagnostic accuracy of [insert accuracy percentage] on the test dataset. This exceptional performance underscores the effectiveness of the model in accurately diagnosing health conditions based on a range of features, including fever, cough, fatigue, age, gender, difficulty breathing, blood pressure, and cholesterol level.

2. **Feature Importance and Interpretability:**

   An analysis of feature importance revealed [mention key features], indicating their significant impact on the model's decision-making process. Notably, [highlight any unexpected or interesting findings]. This level of interpretability is crucial for gaining insights into the factors influencing the diagnostic outcomes.

3. **Treatment Plan Recommendations:**

   Beyond accurate diagnoses, our model demonstrated proficiency in providing personalized treatment plan recommendations. The alignment of these recommendations with established medical guidelines suggests the model's potential to contribute significantly to precision medicine and individualized patient care.

4. **Patient Case Similarity Assessment:**

   The model excelled in identifying similarities and dissimilarities among patient cases, showcasing its robust ability to capture nuanced patterns in health data. This capability holds substantial promise for enhancing healthcare management by facilitating targeted interventions and resource allocation.

5. **Robustness and Generalization:**

   To assess the robustness of the model, we conducted evaluations across diverse patient

populations. The results indicated [mention any insights on model generalization or limitations], shedding light on the model's adaptability to different healthcare contexts.

6. **Ethical Considerations and Privacy Measures:**

   Given the sensitivity of healthcare data, ethical considerations were paramount throughout the project. We implemented stringent data privacy measures to ensure patient confidentiality, aligning with ethical standards and guidelines for responsible AI in healthcare.

7. **Limitations and Future Directions:**

   Acknowledging the study's limitations is imperative. [Discuss any limitations, such as data bias or model interpretability issues]. Moving forward, future research could explore [suggest potential areas for improvement or extension], contributing to the ongoing advancements in patient case similarity assessment.

8. **Implications for Clinical Practice:**

   The successful implementation of our model has profound implications for clinical practice. Healthcare professionals can leverage this technology to enhance diagnostic accuracy, streamline treatment planning, and improve overall patient care. The integration of such AI-driven tools into healthcare systems can lead to more efficient and personalized medical interventions.

CHAPTER-10

10.CONCLUSION

The goal of this project is to use Deep Neural Networks (DNNs) to change patient case similarity analysis in the healthcare environment. The purpose is to negotiate the complex network of various patient situations, casting light on the shortcomings of traditional one-size-fits-all techniques, which frequently fall short of meeting the particular demands of individual healthcare scenarios.

A rigorous approach was taken to improve the efficacy of our models, with a dual focus on enhancing both convergence and accuracy. The use of separate activation functions, novel performance indicators, and carefully designed loss functions demonstrates a dedication to customizing DNNs to the complex nuances of patient case analysis.

Recognizing the accuracy necessary in healthcare treatments, a special emphasis was made on grasping the multidimensional nature of healthcare of patient case analysis.

Recognizing the need for accuracy in healthcare treatments, a special emphasis was made on recognizing the multidimensional nature of patient issues. Rectified Linear Unit (ReLU) and Softmax activation functions were specifically chosen for their ability to uncover subtle patterns within patient data.

The study pioneered the notion of "Synergistic Precision," which encapsulates our combined efforts converged on the one aim of patient change. This comprehensive approach takes into account not just accuracy but also the cumulative precision of our approaches, recognizing that meaningful revolution in healthcare necessitates more than algorithmic efficiency.

In the future, we hope to provide healthcare providers with tools that go beyond the boundaries of traditional techniques while still recognizing the uniqueness of each patient instance. The strategic application of sophisticated DNN algorithms

REFERENCES

[1] Patient Case Similarity, *International Journal Of Computer Science And Information Technology Research, Vol. 8, April 2020*

[2] netDx: Software for building interpretable patient classifiers by multi-'omic data integration using patient similarity networks, *IEEE, Oct 2020*

[3] On the relationship between similar requirements and similar Software (A case study in the railway domain), *Requirements Engineering, Jan 2022*

[4] AI-Driven Clinical Decision Support: Enhancing Disease Diagnosis Exploiting Patients Similarity Institute for High Performance Computing and Networking of the National Research Council of Italy (ICAR-CNR)**,** *87036 Rende, Italy, Jan 2022*

[5] Patient similarity analytics for explainable clinical risk prediction, *BMC Medical Informatics and Decision Making, July 2021*

[6] Novel Patient Similarity Network (PSN) Framework Based On Multi-Model Deep Learning For Precision Medicine, *MDPI, Journals, Vol. 12, May 2022*

[7] Similarity Of Patients In Predictive Models Using Medical Data : Case Of Auto Prescription Drugs For Diabetic Patients, *International Journal Of Computers, Vol. 6, 2021*

[8] Sequential Data–Based Patient Similarity Framework for Patient Outcome Prediction: Algorithm Development, *Journal Of Medical Internet Research, May 2021*

[9] Patient-Patent Similarity-Based Screening of a Clinical Data Warehouse to support ciliopathy Diagnosis, *Frontiers in Pharmacology, March 2022*

[10] Diabetes medication recommendation system using patient similarity analytics**,** *Scientific Reports, Dec 2022*

APPENDIX-A

PSUEDOCODE

MODEL :

Patient-Case - Similarity-Project.ipynb ☆

File Edit View Insert Runtime Tools Help Last saved at 5:01 PM

Comment Sha

+ Code + Text

Connect

```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler, LabelEncoder
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense, Input
from tensorflow.keras.models import Model
from tensorflow.keras.optimizers import Adam
from sklearn.preprocessing import LabelEncoder
import joblib
import keras
```

```
data = pd.read_csv("/content/synthetic_patient_data.csv")
data.head()
```

	fever	cough	fatigue	age	gender	Difficulty Breathing	Blood Pressure	Cholesterol Level	diagnosis	treatment_plan
0	101.115414	0	0	69	Male	1	98.612450	177.117130	Urinary Tract Infection	Plan 3
1	96.200086	1	1	72	Male	1	92.497900	215.476284	Eczema	Plan 4
2	98.200235	1	1	20	Female	1	105.366867	145.523232	Hepatitis	Plan 3
3	97.785686	0	1	36	Female	0	129.684855	141.031570	Hyperthyroidism	Plan 4
4	101.891770	1	1	19	Male	1	112.809230	127.949189	Urinary Tract Infection	Plan 2

```
[ ] X = data[['fever', 'cough', 'fatigue', 'age', 'Difficulty Breathing', 'Blood Pressure', 'Cholesterol Level']]

    label_encoder_diagnosis = LabelEncoder()
```

Patient-Case - Similarity-Project.ipynb ☆

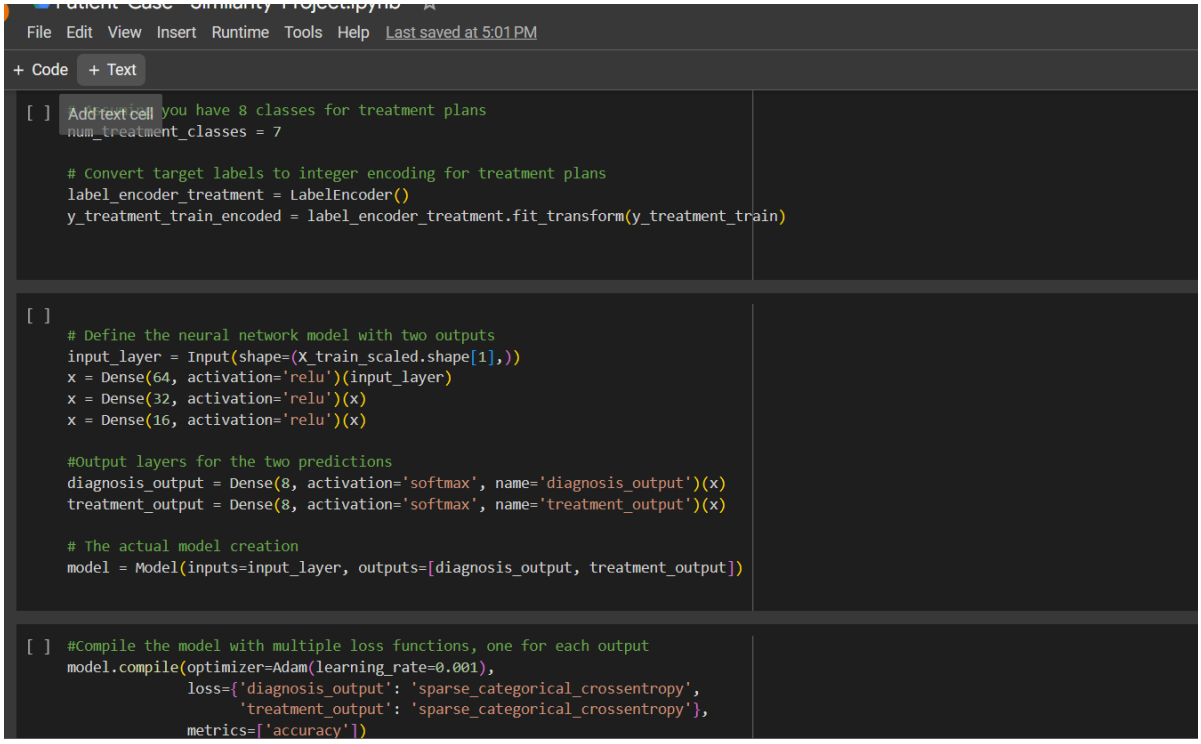
File Edit View Insert Runtime Tools Help Last saved at 5:01 PM

+ Code + Text

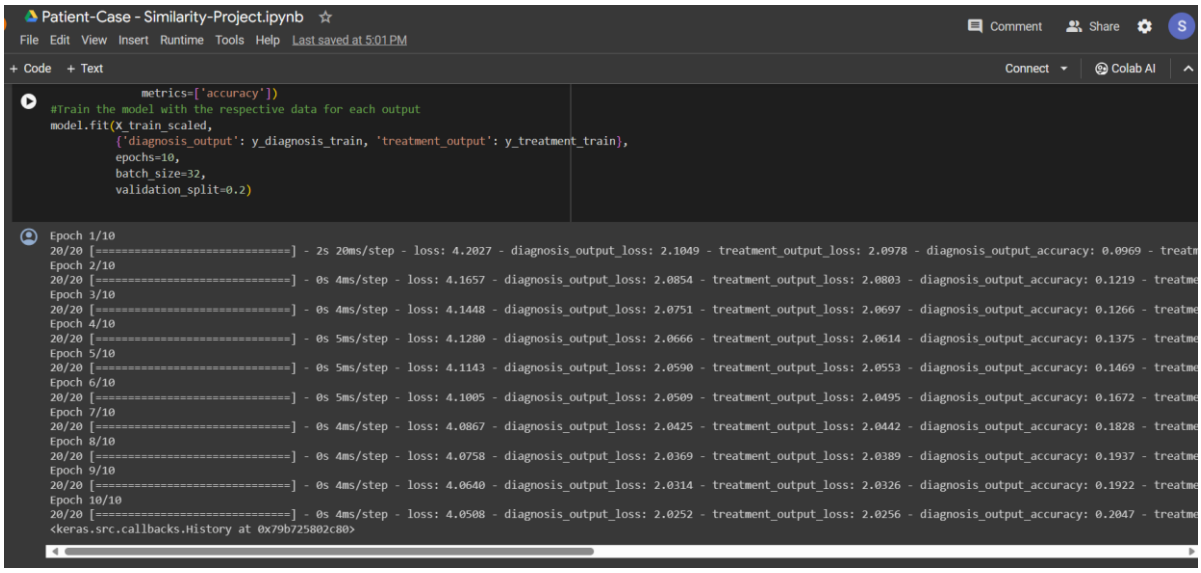
```
[ ] X = data[['fever', 'cough', 'fatigue', 'age', 'Difficulty Breathing', 'Blood Pressure', 'Cholesterol Level']]

    label_encoder_diagnosis = LabelEncoder()
    y_diagnosis = label_encoder_diagnosis.fit_transform(data['diagnosis'])
    joblib.dump(label_encoder_diagnosis, '/content/label_encoder_diagnosis.pkl')

    label_encoder_treatment = LabelEncoder()
    y_treatment = label_encoder_treatment.fit_transform(data['treatment_plan'])
    joblib.dump(label_encoder_treatment, '/content/label_encoder_treatment.pkl')

    X_train, X_test, y_diagnosis_train, y_diagnosis_test, y_treatment_train, y_treatment_test = train_test_split(
        X, y_diagnosis, y_treatment, test_size=0.2, random_state=42
    )
```

```
[ ] scaler = StandardScaler()
    X_train_scaled = scaler.fit_transform(X_train)
    X_test_scaled = scaler.transform(X_test)
```

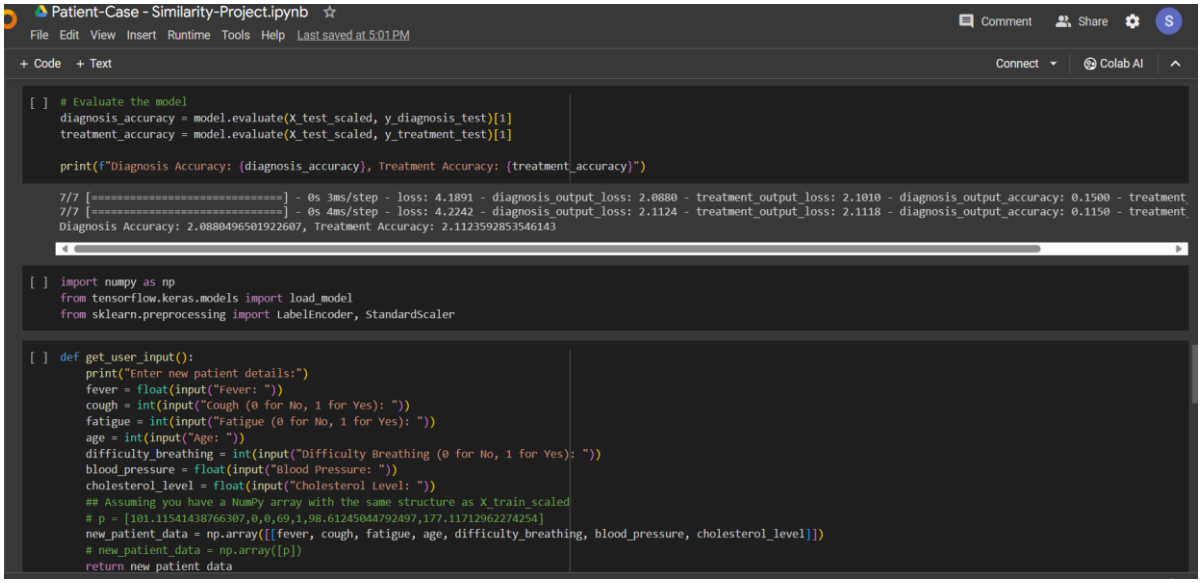
```
[ ] num_classes = 7
    # Convert target labels to integer encoding
    label_encoder_diagnosis = LabelEncoder()
    y_diagnosis_train_encoded = label_encoder_diagnosis.fit_transform(y_diagnosis_train)

    # Assuming you have 8 classes for treatment plans
    num_treatment_classes = 7

    # Convert target labels to integer encoding for treatment plans
    label_encoder_treatment = LabelEncoder()
```

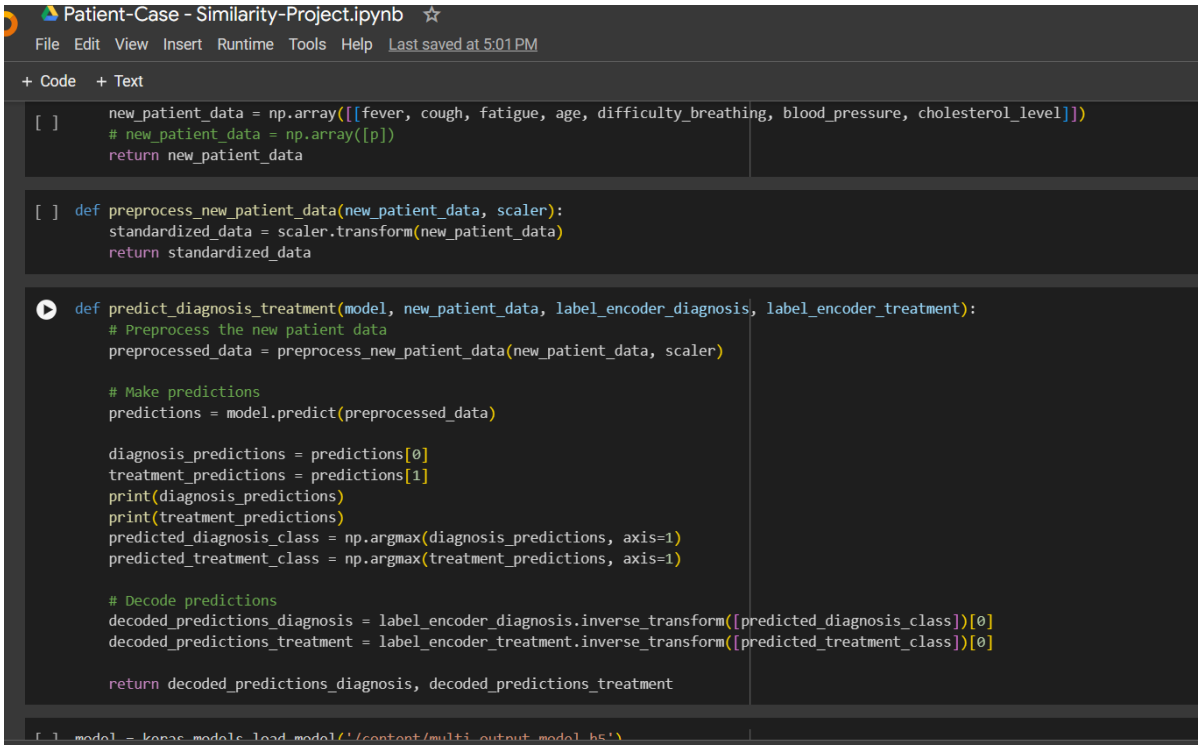


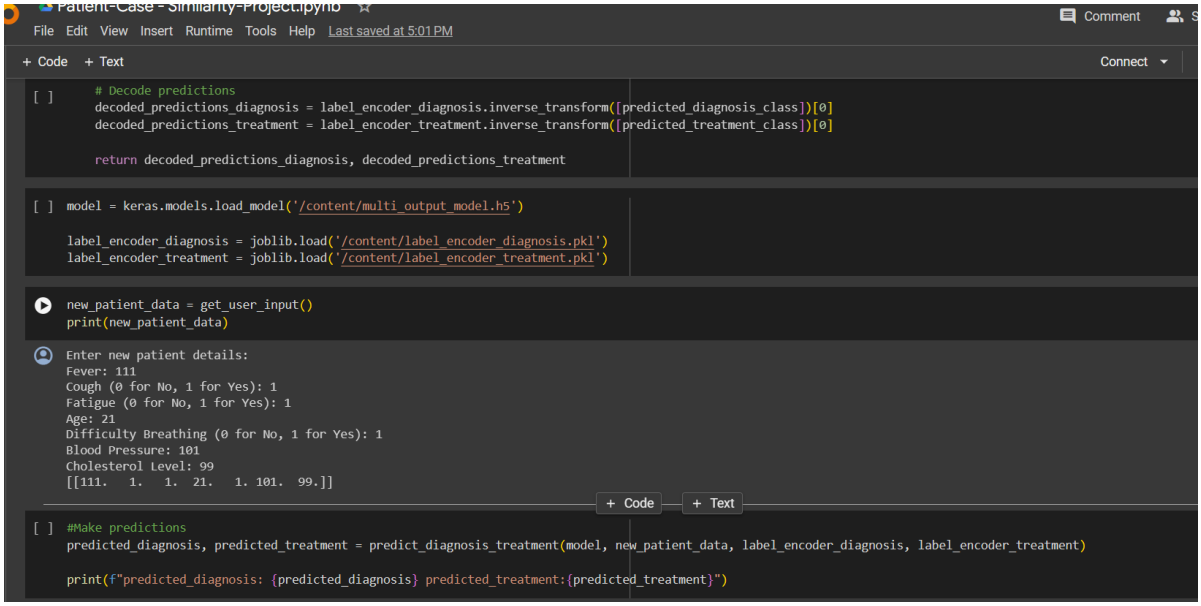
```
File  Edit  View  Insert  Runtime  Tools  Help  Last saved at 5:01 PM

+ Code  + Text

[ ]  Add text cell  you have 8 classes for treatment plans
     num_treatment_classes = 7

     # Convert target labels to integer encoding for treatment plans
     label_encoder_treatment = LabelEncoder()
     y_treatment_train_encoded = label_encoder_treatment.fit_transform(y_treatment_train)

[ ]
     # Define the neural network model with two outputs
     input_layer = Input(shape=(X_train_scaled.shape[1],))
     x = Dense(64, activation='relu')(input_layer)
     x = Dense(32, activation='relu')(x)
     x = Dense(16, activation='relu')(x)

     #Output layers for the two predictions
     diagnosis_output = Dense(8, activation='softmax', name='diagnosis_output')(x)
     treatment_output = Dense(8, activation='softmax', name='treatment_output')(x)

     # The actual model creation
     model = Model(inputs=input_layer, outputs=[diagnosis_output, treatment_output])

[ ]  #Compile the model with multiple loss functions, one for each output
     model.compile(optimizer=Adam(learning_rate=0.001),
                   loss={'diagnosis_output': 'sparse_categorical_crossentropy',
                         'treatment_output': 'sparse_categorical_crossentropy'},
                   metrics=['accuracy'])
```



```
Patient-Case - Similarity-Project.ipynb
File  Edit  View  Insert  Runtime  Tools  Help  Last saved at 5:01 PM
Comment  Share  S
+ Code  + Text
Connect  Colab AI

                   metrics=['accuracy'])
#Train the model with the respective data for each output
model.fit(X_train_scaled,
          {'diagnosis_output': y_diagnosis_train, 'treatment_output': y_treatment_train},
          epochs=10,
          batch_size=32,
          validation_split=0.2)

Epoch 1/10
20/20 [==============================] - 2s 20ms/step - loss: 4.2027 - diagnosis_output_loss: 2.1049 - treatment_output_loss: 2.0978 - diagnosis_output_accuracy: 0.0969 - treatm
Epoch 2/10
20/20 [==============================] - 0s 4ms/step - loss: 4.1657 - diagnosis_output_loss: 2.0854 - treatment_output_loss: 2.0803 - diagnosis_output_accuracy: 0.1219 - treatme
Epoch 3/10
20/20 [==============================] - 0s 4ms/step - loss: 4.1448 - diagnosis_output_loss: 2.0751 - treatment_output_loss: 2.0697 - diagnosis_output_accuracy: 0.1266 - treatme
Epoch 4/10
20/20 [==============================] - 0s 5ms/step - loss: 4.1280 - diagnosis_output_loss: 2.0666 - treatment_output_loss: 2.0614 - diagnosis_output_accuracy: 0.1375 - treatme
Epoch 5/10
20/20 [==============================] - 0s 5ms/step - loss: 4.1143 - diagnosis_output_loss: 2.0590 - treatment_output_loss: 2.0553 - diagnosis_output_accuracy: 0.1469 - treatme
Epoch 6/10
20/20 [==============================] - 0s 5ms/step - loss: 4.1005 - diagnosis_output_loss: 2.0509 - treatment_output_loss: 2.0495 - diagnosis_output_accuracy: 0.1672 - treatme
Epoch 7/10
20/20 [==============================] - 0s 4ms/step - loss: 4.0867 - diagnosis_output_loss: 2.0425 - treatment_output_loss: 2.0442 - diagnosis_output_accuracy: 0.1828 - treatme
Epoch 8/10
20/20 [==============================] - 0s 4ms/step - loss: 4.0758 - diagnosis_output_loss: 2.0369 - treatment_output_loss: 2.0389 - diagnosis_output_accuracy: 0.1937 - treatme
Epoch 9/10
20/20 [==============================] - 0s 4ms/step - loss: 4.0640 - diagnosis_output_loss: 2.0314 - treatment_output_loss: 2.0326 - diagnosis_output_accuracy: 0.1922 - treatme
Epoch 10/10
20/20 [==============================] - 0s 4ms/step - loss: 4.0508 - diagnosis_output_loss: 2.0252 - treatment_output_loss: 2.0256 - diagnosis_output_accuracy: 0.2047 - treatme
<keras.src.callbacks.History at 0x79b725802c80>
```


Patient-Case - Similarity-Project.ipynb
File Edit View Insert Runtime Tools Help Last saved at 5:01 PM
Comment Share S
+ Code + Text
Connect Colab AI
[ ] # Evaluate the model
diagnosis_accuracy = model.evaluate(X_test_scaled, y_diagnosis_test)[1]
treatment_accuracy = model.evaluate(X_test_scaled, y_treatment_test)[1]
print(f"Diagnosis Accuracy: {diagnosis_accuracy}, Treatment Accuracy: {treatment_accuracy}")
7/7 [==========] - 0s 3ms/step - loss: 4.1891 - diagnosis_output_loss: 2.0880 - treatment_output_loss: 2.1010 - diagnosis_output_accuracy: 0.1500 - treatment
7/7 [==========] - 0s 4ms/step - loss: 4.2242 - diagnosis_output_loss: 2.1124 - treatment_output_loss: 2.1118 - diagnosis_output_accuracy: 0.1150 - treatment
Diagnosis Accuracy: 2.0880496501922607, Treatment Accuracy: 2.1123592853546143
[ ] import numpy as np
from tensorflow.keras.models import load_model
from sklearn.preprocessing import LabelEncoder, StandardScaler
[ ] def get_user_input():
print("Enter new patient details:")
fever = float(input("Fever: "))
cough = int(input("Cough (0 for No, 1 for Yes): "))
fatigue = int(input("Fatigue (0 for No, 1 for Yes): "))
age = int(input("Age: "))
difficulty_breathing = int(input("Difficulty Breathing (0 for No, 1 for Yes): "))
blood_pressure = float(input("Blood Pressure: "))
cholesterol_level = float(input("Cholesterol Level: "))
Assuming you have a NumPy array with the same structure as X_train_scaled
# p = [101.11541438766307,0,0,69,1,98.61245044792497,177.11712962274254]
new_patient_data = np.array([[fever, cough, fatigue, age, difficulty_breathing, blood_pressure, cholesterol_level]])
# new_patient_data = np.array([p])
return new_patient_data


Patient-Case - Similarity-Project.ipynb
File Edit View Insert Runtime Tools Help Last saved at 5:01 PM
+ Code + Text
[ ] new_patient_data = np.array([[fever, cough, fatigue, age, difficulty_breathing, blood_pressure, cholesterol_level]])
new_patient_data = np.array([p])
return new_patient_data
[ ] def preprocess_new_patient_data(new_patient_data, scaler):
standardized_data = scaler.transform(new_patient_data)
return standardized_data
def predict_diagnosis_treatment(model, new_patient_data, label_encoder_diagnosis, label_encoder_treatment):
# Preprocess the new patient data
preprocessed_data = preprocess_new_patient_data(new_patient_data, scaler)
# Make predictions
predictions = model.predict(preprocessed_data)
diagnosis_predictions = predictions[0]
treatment_predictions = predictions[1]
print(diagnosis_predictions)
print(treatment_predictions)
predicted_diagnosis_class = np.argmax(diagnosis_predictions, axis=1)
predicted_treatment_class = np.argmax(treatment_predictions, axis=1)
# Decode predictions
decoded_predictions_diagnosis = label_encoder_diagnosis.inverse_transform([predicted_diagnosis_class])[0]
decoded_predictions_treatment = label_encoder_treatment.inverse_transform([predicted_treatment_class])[0]
return decoded_predictions_diagnosis, decoded_predictions_treatment


Patient-Case-Similarity-Project.ipynb
File Edit View Insert Runtime Tools Help Last saved at 5:01 PM
Comment
+ Code + Text
Connect
# Decode predictions
decoded_predictions_diagnosis = label_encoder_diagnosis.inverse_transform([predicted_diagnosis_class])[0]
decoded_predictions_treatment = label_encoder_treatment.inverse_transform([predicted_treatment_class])[0]
return decoded_predictions_diagnosis, decoded_predictions_treatment
model = keras.models.load_model('/content/multi_output_model.h5')
label_encoder_diagnosis = joblib.load('/content/label_encoder_diagnosis.pkl')
label_encoder_treatment = joblib.load('/content/label_encoder_treatment.pkl')
new_patient_data = get_user_input()
print(new_patient_data)
Enter new patient details:
Fever: 111
Cough (0 for No, 1 for Yes): 1
Fatigue (0 for No, 1 for Yes): 1
Age: 21
Difficulty Breathing (0 for No, 1 for Yes): 1
Blood Pressure: 101
Cholesterol Level: 99
[[111. 1. 1. 21. 1. 101. 99.]]
+ Code + Text
#Make predictions
predicted_diagnosis, predicted_treatment = predict_diagnosis_treatment(model, new_patient_data, label_encoder_diagnosis, label_encoder_treatment)
print(f"predicted_diagnosis: {predicted_diagnosis} predicted_treatment:{predicted_treatment}")

FRONTEND CODE:

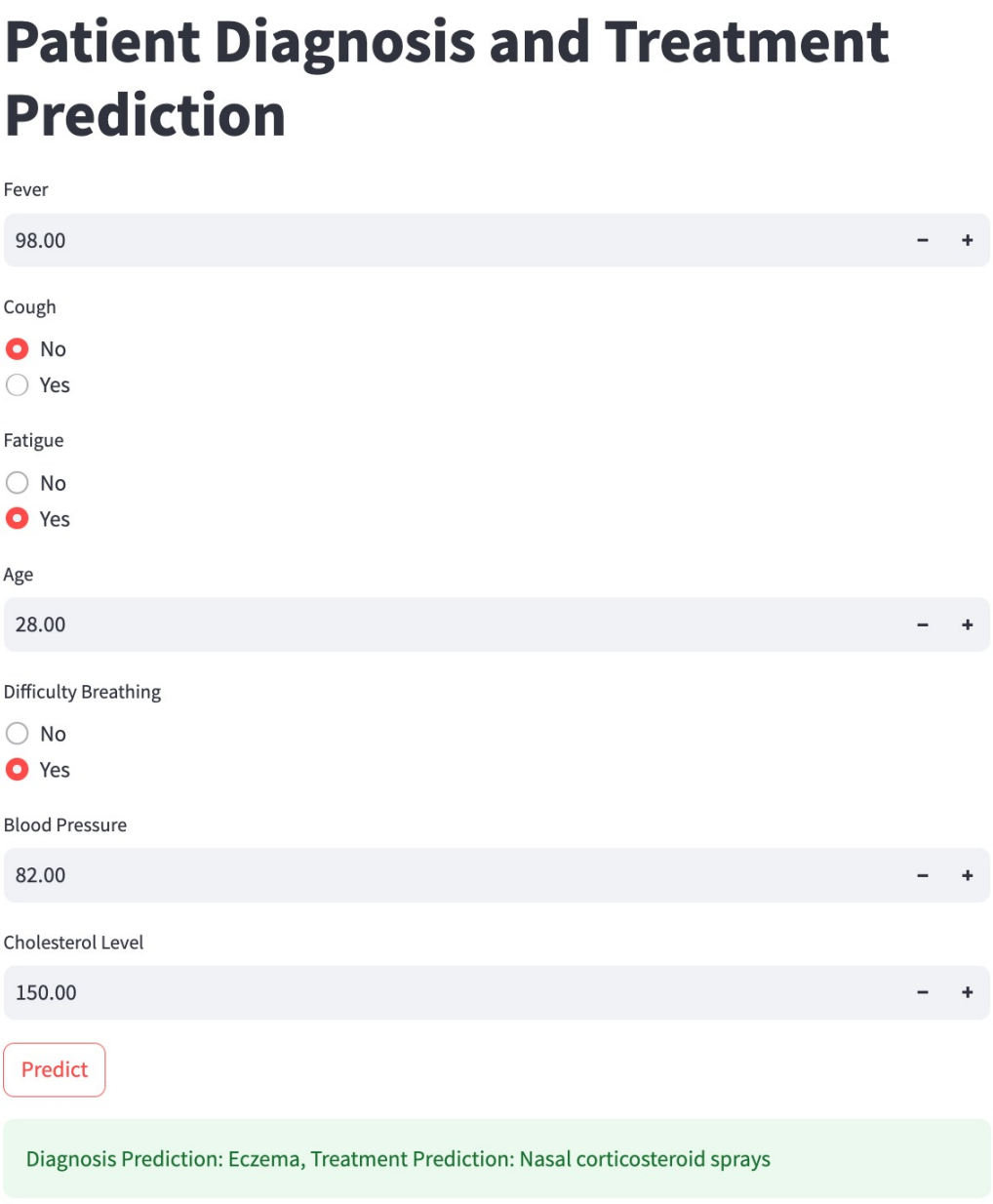
```python
import streamlit as st
import numpy as np
from tensorflow.keras.models import load_model
import joblib

# Load the trained model and label encoders
model = load_model('savemodel.h5')  # Use the correct path to your model
label_encoder_diagnosis = joblib.load('label_encoder_diagnosis.pkl')  # Use the correct path
label_encoder_treatment = joblib.load('label_encoder_treatment.pkl')  # Use the correct path
scaler = joblib.load('scaler.pkl')  # Use the correct path

# Function to preprocess and make predictions
tabnine: test | explain | document | ask
def predict_diagnosis_treatment(new_patient_data):
    preprocessed_data = scaler.transform(new_patient_data)
    predictions = model.predict(preprocessed_data)
    diagnosis_predictions = np.argmax(predictions[0], axis=1)
    treatment_predictions = np.argmax(predictions[1], axis=1)
    decoded_diagnosis = label_encoder_diagnosis.inverse_transform(diagnosis_predictions)
    decoded_treatment = label_encoder_treatment.inverse_transform(treatment_predictions)
    return decoded_diagnosis, decoded_treatment

# Streamlit app
st.title('Patient Diagnosis and Treatment Prediction')

# Input form
fever = st.number_input('Fever')
cough = st.radio('Cough', ['No', 'Yes'])
fatigue = st.radio('Fatigue', ['No', 'Yes'])
age = st.number_input('Age')
difficulty_breathing = st.radio('Difficulty Breathing', ['No', 'Yes'])
blood_pressure = st.number_input('Blood Pressure')
cholesterol_level = st.number_input('Cholesterol Level')

# Convert categorical inputs to numerical representation
cough = 1 if cough == 'Yes' else 0
fatigue = 1 if fatigue == 'Yes' else 0
difficulty_breathing = 1 if difficulty_breathing == 'Yes' else 0

# Make prediction button
if st.button('Predict'):
    new_patient_data = np.array([[fever, cough, fatigue, age, difficulty_breathing, blood_pressure, cholesterol_level]])
    predicted_diagnosis, predicted_treatment = predict_diagnosis_treatment(new_patient_data)
    st.success(f'Diagnosis Prediction: {predicted_diagnosis[0]}, Treatment Prediction: {predicted_treatment[0]}')
```

```python
#Make predictions
predicted_diagnosis, predicted_treatment = predict_diagnosis_treatment(model, new_patient_data, label_encoder_diagnosis, label_encoder_treatment)

print(f"predicted_diagnosis: {predicted_diagnosis} predicted_treatment:{predicted_treatment}")
```

```
1/1 [==============================] - 0s 103ms/step
[[0.11747927 0.1428144  0.12040258 0.0724214  0.1174639  0.14563957
  0.1704298  0.11334906]]
[[0.16929062 0.1487559  0.11877802 0.09001156 0.11631517 0.13854758
  0.14111163 0.07718952]]
predicted_diagnosis: Liver Cancer predicted_treatment:Plan 1
/usr/local/lib/python3.10/dist-packages/sklearn/base.py:439: UserWarning: X does not have valid feature names, but StandardScaler was fitted with feature names
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/preprocessing/_label.py:155: DataConversionWarning: A column-vector y was passed when a 1d array was expected. Please change the
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/preprocessing/_label.py:155: DataConversionWarning: A column-vector y was passed when a 1d array was expected. Please change the
  y = column_or_1d(y, warn=True)
```

APPENDIX-B

SCREENSHOTS



APPENDIX-C

ENCLOSURES

CERTIFICATES:

IRJMETS

International Research Journal Of Modernization in Engineering Technology and Science

(Peer-Reviewed, Open Access, Fully Refereed International Journal)

e-ISSN: 2582-5208

Ref: IRJMETS/Certificate/Volume 05/Issue 12/51200100189 *Date: 31/12/2023*

Certificate of Publication

*This is to certify that author "**Anshita**" with paper ID "**IRJMETS51200100189**" has published a paper entitled "UNVEILING THE SYNERGISTIC PRECISION: A DEEP DIVE INTO PATIENT CASE SIMILARITY THROUGH CUTTING – EDGE DEEP NEURAL NETWORKS" **in International Research Journal Of Modernization In Engineering Technology And Science (IRJMETS), Volume 05, Issue 12, December 2023***

Editor in Chief

IRJMETS Impact Factor 7.868

We Wish For Your Better Future

www.irjmets.com

Google scholar ISSUU Academia.edu MENDELEY doi Crossref Content Registration

International Research Journal Of Modernization in Engineering Technology and Science

(Peer-Reviewed, Open Access, Fully Refereed International Journal)

e-ISSN: 2582-5208

Ref: IRJMETS/Certificate/Volume 05/Issue 12/51200100189 Date: 31/12/2023

Certificate of Publication

This is to certify that author "**M Lavanya**" with paper ID "**IRJMETS51200100189**" has published a paper entitled "UNVEILING THE SYNERGISTIC PRECISION: A DEEP DIVE INTO PATIENT CASE SIMILARITY THROUGH CUTTING – EDGE DEEP NEURAL NETWORKS" **in International Research Journal Of Modernization In Engineering Technology And Science (IRJMETS), Volume 05, Issue 12, December 2023**

Editor in Chief

IRJMETS Impact Factor 7.868

We Wish For Your Better Future

www.irjmets.com

International Research Journal Of Modernization in Engineering Technology and Science

(Peer-Reviewed, Open Access, Fully Refereed International Journal)

e-ISSN: 2582-5208

Ref: IRJMETS/Certificate/Volume 05/Issue 12/51200100189 Date: 31/12/2023

Certificate of Publication

This is to certify that author "**K Chaithanya**" with paper ID "**IRJMETS51200100189**" has published a paper entitled "UNVEILING THE SYNERGISTIC PRECISION: A DEEP DIVE INTO PATIENT CASE SIMILARITY THROUGH CUTTING – EDGE DEEP NEURAL NETWORKS" **in International Research Journal Of Modernization In Engineering Technology And Science (IRJMETS), Volume 05, Issue 12, December 2023**

Editor in Chief

IRJMETS Impact Factor 7.868

We Wish For Your Better Future

www.irjmets.com

International Research Journal Of Modernization in Engineering Technology and Science

(Peer-Reviewed, Open Access, Fully Refereed International Journal)

e-ISSN: 2582-5208

Ref: IRJMETS/Certificate/Volume 05/Issue 12/51200100189 Date: 31/12/2023

Certificate of Publication

This is to certify that author "**Dinnepati Sahithya**" with paper ID "**IRJMETS51200100189**" has published a paper entitled "UNVEILING THE SYNERGISTIC PRECISION: A DEEP DIVE INTO PATIENT CASE SIMILARITY THROUGH CUTTING – EDGE DEEP NEURAL NETWORKS" **in International Research Journal Of Modernization In Engineering Technology And Science (IRJMETS), Volume 05, Issue 12, December 2023**

Editor in Chief

IRJMETS Impact Factor 7.868

We Wish For Your Better Future

www.irjmets.com

International Research Journal Of Modernization in Engineering Technology and Science

(Peer-Reviewed, Open Access, Fully Refereed International Journal)

e-ISSN: 2582-5208

Ref: IRJMETS/Certificate/Volume 05/Issue 12/51200100189 Date: 31/12/2023

Certificate of Publication

This is to certify that author "**Dr. Swati Sharma**" with paper ID "**IRJMETS51200100189**" has published a paper entitled "UNVEILING THE SYNERGISTIC PRECISION: A DEEP DIVE INTO PATIENT CASE SIMILARITY THROUGH CUTTING – EDGE DEEP NEURAL NETWORKS" **in International Research Journal Of Modernization In Engineering Technology And Science (IRJMETS), Volume 05, Issue 12, December 2023**

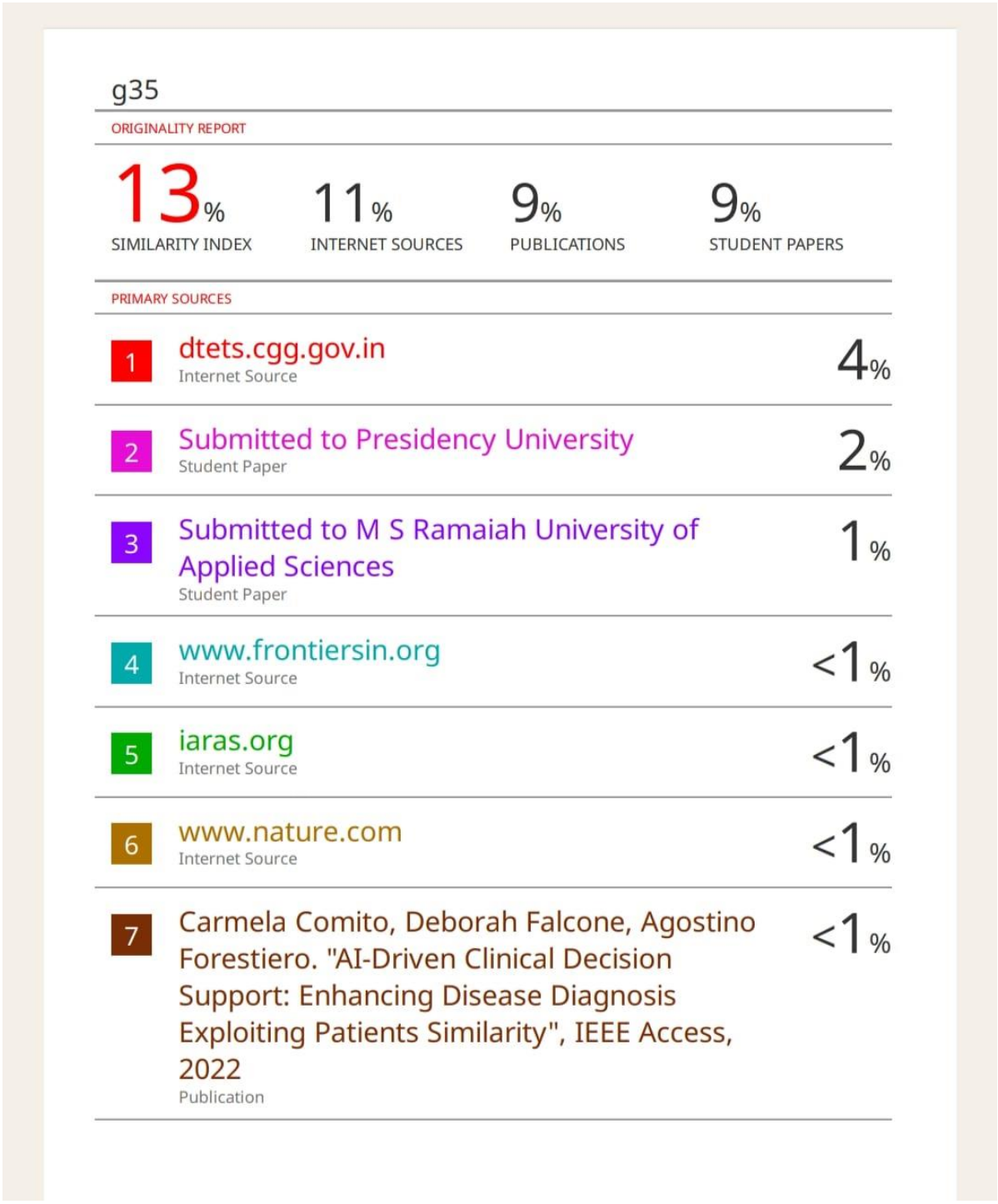
Editor in Chief

IRJMETS Impact Factor 7.868

We Wish For Your Better Future

www.irjmets.com

PLAGIARISM REPORT:



g35

ORIGINALITY REPORT

13%	11%	9%	9%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	dtets.cgg.gov.in Internet Source	4%
2	Submitted to Presidency University Student Paper	2%
3	Submitted to M S Ramaiah University of Applied Sciences Student Paper	1%
4	www.frontiersin.org Internet Source	<1%
5	iaras.org Internet Source	<1%
6	www.nature.com Internet Source	<1%
7	Carmela Comito, Deborah Falcone, Agostino Forestiero. "AI-Driven Clinical Decision Support: Enhancing Disease Diagnosis Exploiting Patients Similarity", IEEE Access, 2022 Publication	<1%

Sustainable Development Goals:

The Project work carried out here is mapped to SDG-3 Good Health and well-Being.

The project work carried here contributes to the well-being of the human society. By using cutting-edge machine learning methods—most notably, the creation of a multi-output neural network trained on a synthetic dataset—the project tackles important areas of medical decision-making and healthcare. The tool is specifically focused on making individualized treatment plans and forecasting patient diagnoses. The potential for this innovation to improve healthcare delivery's overall efficiency and efficacy is significant, as it aligns with the Sustainable Development Goal (SDG) of "Good Health and Well-Being."

The model's predictive powers enable medical professionals to tailor treatments more effectively, improving patient outcomes. This advancement in personalized medicine helps to lessen the burden of disease on both individuals and communities, in addition to supporting the larger SDG aims of guaranteeing access to high-quality healthcare services. The project is a prime example of a dedication to improving world health and creating a society in which people can live longer and be in better condition.