Project Report on

Medical Transcript Analysis

Submitted in partial fulfillment of the requirements

of the degree of Bachelor in Engineering

by

Name of the Student	Class	Roll No.
Rajat Savdekar	BE-9	11
Tushar Chaudhari	BE-4	05
Dhairya Shah	BE-4	59
Darshan Patil	BE-3	42

Under the guidance of

Prof. Pallavi Khodke

DEPARTMENT OF COMPUTER ENGINEERING

SHAH AND ANCHOR KUTCHHI ENGINEERING COLLEGE

CHEMBUR, MUMBAI – 400088.

2022 – 2023

Certificate

This is to certify that the report of the project entitled

Medical Transcript Analysis

is a bonafide work of

Name of the Student	Class	Roll No.
Rajat Savdekar	BE-9	11
Tushar Chaudhari	BE-4	05
Dhairya Shah	BE-4	59
Darshan Patil	BE-3	42

submitted to the

UNIVERSITY OF MUMBAI

during semester VII in partial fulfillment of the requirement for the award of the degree of

BACHELOR OF ENGINEERING

in

COMPUTER ENGINEERING.

(Prof.Pallavi Khodke)

Guide

--	-------------------------------------
(Prof. UdayBhave)	(Dr. Bhavesh Patel)
I/c Head of Department	Principal

Approval for Project Report for B. E. Semester VII

This project report entitled **Medical Transcript Analysis** by Rajat Savdekar, Tushar Chaudhari, Dhairya Shah, Darshan Patil approved for semester VII in partial fulfillment of the requirement for the award of the degree of Bachelor of Engineering.

Examiners

1.____________________________

2.____________________________

Guide

1.____________________________

2.____________________________

Date:

Place:

Declaration

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, we have adequately cited and referenced the original sources. We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Name of the Student	Class	Roll No.	Signature
Rajat Savdekar	BE-9	11	
Tushar Chaudhari	BE-4	05	
Dhairya Shah	BE-4	59	
Darshan Patil	BE-3	42	

Date:

Place:

Attendance Certificate

Date

To,
The Principal
Shah and Anchor Kutchhi Engineering College,
Chembur, Mumbai-88

Subject: Confirmation of Attendance

Respected Sir,

This is to certify that Final year (BE) students **Rajat Savdekar, Tushar Chaudhari, Dhairya Shah, Darshan Patil** have duly attended the sessions on the day allotted to them during the period from **18/07/2022 to 29/10/2022** for performing the Project titled **Medical Transcript Analysis.**

They were punctual and regular in their attendance. Following is the detailed record of the student's attendance.

Attendance Record:

Date	Rajat Savdekar	Tushar Chaudhari	Dhairya Shah	Darshan Patil
	Present/Absent	Present/Absent	Present/Absent	Present/Absent
18/07/2022	Present	Present	Present	Present
25/07/2022	Present	Present	Present	Present
01/08/2022	Present	Present	Present	Present
22/08/2022	Present	Present	Present	Present
29/08/2022	Present	Present	Present	Present
05/09/2022	Present	Present	Present	Present
19/09/2022	Present	Present	Present	Present
26/09/2022	Present	Present	Present	Present
03/10/2022	Present	Present	Present	Present
29/10/2022	Present	Present	Present	Present

Signature and Name of Internal Guide

Abstract

Nowadays, the most well-known and popular platform for healthcare is digital. Making a computerized system that is user-friendly for doctors is one such initiative. Unlike the old ways, this system will let doctors save information about each and every patient, including consultations, procedures, and many other connected items. Physicians spend a lot of time entering free-form text notes into EHR (electronic health record) systems. Much of this documentation work is seen as a burden that reduces time spent with patients and contributes to physician burnout. ML-assisted note-taking aspirations proposes a novel language modeling task that predicts note content based on historical data from patient medical records, including patient demographics, laboratory, medication, and past notes.

Machine learning techniques in healthcare use the growing amount of medical data provided by the Internet of Things to improve patient outcomes. These technologies offer promising applications and significant challenges. The three main areas of machine learning include medical image processing, natural language processing of medical documents, and genetic information. Many of these areas focus on diagnostics, detection, and prediction.

We aim to create a working prototype of a digital medical transcription platform that will enable doctors and surgeons to record patient consultations and summaries of surgeries with the press of a button. We present a system that records relevant medical information from doctor-patient interactions. The system analyzes each interaction, uses context to predict related entities, and extracts entities such as medications and symptoms. It also assigns the main diagnosis to each conversation. It serves as the basis for a system that extracts information from dialogues and automatically creates patient notes for review and editing by clinicians.

Acknowledgement

We are thankful to our college Shah And Anchor Kutchhi Engineering College for considering our project and extending help at all stages needed during our work of collecting information regarding the project.

We are deeply indebted to our Principal, **Dr. Bhavesh Patel,** and Head of Computer Department, **Mr. Uday Bhave, giving** us this valuable opportunity to do this project. We express our hearty thanks to them for their assistance without which it would have been difficult in finishing this project synopsis and project review successfully.

It gives us immense pleasure to express our deep and sincere gratitude to our project guide, **Prof. Pallavi Khodke** for her kind help and valuable advice during the development of project synopsis and for her guidance and suggestions.

We convey our deep sense of gratitude to all teaching and non-teaching staff for their constant encouragement, support, and selfless help throughout the project work. It is a great pleasure to acknowledge the help and suggestion, which we received from the Department of Computer Engineering.

We wish to express our profound thanks to all those who helped us in gathering information about the project. Our families too have provided moral support and encouragement several times.

Table of Contents

Chapter 1

Introduction

The medical professionals are using conventional ways to record the procedures/surgeries they perform on patients. They either use paper documentation or have to type the information into a text editor of their choosing. While some of these documents are fully integrated with patient electronic record systems (EMR) or electronic health record (EHR), others documents are not. In smaller hospitals and clinics, the medical professionals like doctors and surgeons continue to record each patient's medical history, procedures, and other clinical information using old-fashioned paper-based methods. During clinical visits and after surgeries, doctors and other medical personnel take a lot of time to capture and document patient information.

Data from electronic medical records have grown to be essential to clinical care in recent years. However, physicians can spend a significant amount of their time on data entry because it is currently slow and error-prone . Additionally, this leads to clinical documentation that is inconsistent and highly variable, which poses difficulties for machine learning models.

The majority of the research done so far on information extraction from clinical conversations does not distinguish between things that are relevant to the patient (such past symptoms and current drugs) and entities that are not (such as medications that the patient says were taken by someone else).
In this study, we take information that is clinically significant from a transcript of a conversation between a doctor and patient and use it to automatically classify the transcript.

Chapter 2

Literature Survey

2.1. Review of the Topic

[1]The paper offers a technology for automatically extracting important medical data from patient and doctor conversations. The algorithm analyzes each discussion to extract items like drugs and symptoms while predicting which entities are relevant based on the context. Additionally, it categorizes the primary diagnosis for every chat. Also, it recognises pertinent utterances and extracts topic information. This operates as the starting point for a system that automatically creates a patient note from information gleaned from conversations that can then be reviewed and modified by the doctor.

[2]It is projected that personalized treatment and higher-quality healthcare would be driven by predictive modeling using electronic health record (EHR) data. We suggest using the Fast Healthcare Interoperability Resources (FHIR) format to express the full raw EHR records for patients. Multiple medical occurrences from several centers can be effectively predicted by deep learning techniques utilizing this representation.

[3]The approaches utilized to evaluate EHRs for clinical research lack consistency and have little potential for generalization. Researchers should use systematic, statistically supported, experimentally driven, and verified approaches for evaluating data quality. There are five criteria for measuring the quality of data: completeness, accuracy, concordance, plausibility, and currency.

[4]They offer a method for automatically extracting important medical data from conversations between patients and doctors. Each discussion is parsed by the system, which uses the context to determine which entities are pertinent. Entities like drugs and symptoms are extracted. Additionally, we categorize the primary diagnosis for every chat. We also identify pertinent utterances and extract topic information. This operates as the starting point for a system that automatically creates a patient note from information gleaned from conversations that can then be reviewed and modified by the doctor.

[5]The techniques used to evaluate EHRs for clinical research lack consistency and possible generalizability. Researchers should use systematic, statistically supported, empirically driven, and verified procedures to evaluate the quality of their data. The completeness, accuracy, concordance, plausibility, and currency of the data were determined to be the five dimensions of data quality.

[6]The paper introduces an open web platform (EMIM) that implements a number of features for the healthcare industry, including the storage of medical data, doctor-patient communication, online appointments, access to medical histories, disease self-management, and others. Although there are other other programmes that incorporate some of these services, our platform is made to combine them all into a single cloud database. In this approach, medical data are no longer lost in multiple isolated computers because, once uploaded, they are available anywhere, at any time.

[7]Multiple adverse health conditions are typically associated with poor prognosis and increased office or hospital visits. Developing methods to identify patterns of co-occurring conditions can assist in diagnosis. This task stems from a realistic application of medical providers in capturing symptoms mentioned by patients from their clinical conversations. We propose two novel deep learning approaches to infer the symptoms and their status.

[8]The iterative participatory design of SOPHIE, an online virtual patient for feedback-based practice of private patient-physician discussions, is described in this study along with a preliminary qualitative assessment of the system by professional end users. A computational linguistic study of the transcripts of 383 patient-physician interactions from a crucial office visit of late stage cancer patients with their oncologists served as the inspiration for the design of SOPHIE. We created tools for the automatic recognition of two behavioral patterns that have been demonstrated to significantly correlate with patient prognosis comprehension: lecturing and positive language usage patterns (conversational sentiment trajectory). These automatic indicators for effective communication were implemented into SOPHIE, and a pilot user research revealed that a set of users had given SOPHIE positive feedback.

[9]In this research, they provide a user satisfaction detection system that makes use of speech and image as two multimedia components. Indifferent, unsatisfied, and satisfied are the three categories of contentment. The user's voice and facial image are recorded, sent to the cloud, and evaluated there. It has been demonstrated that the suggested technique detects pleasure with up to 93% accuracy.

[10]The most popular and popular platform for therapy nowadays is digital healthcare. This system will enable doctors to keep records of consultations with patients, surgeries conducted, and a variety of other patient-related data. The fundamental concept is to use Speech Recognition and Synthesis technologies to convert voice recordings into written documents.

2.2. Limitation of Existing system

[1]Only six diagnosis classification categories are currently supported, and they do not take into account conditions that are discussed in addition to the core diagnosis. In order to anticipate various diagnoses and to handle more classes, we will also expand diagnosis classification.

[2]A custom dataset with specific characteristics must be created for each result to be predicted using standard predictive modeling techniques, which makes scaling the construction of predictive models challenging despite the richness and potential of the available data. It is often believed that preprocessing, merging, modifying, and cleaning data sets takes about 80% of the time spent on an analytical model rather than really studying the data for insights. Predictive model scalability is severely constrained by this.

[3]This review had a number of restrictions. It's probable that any pertinent papers were missed in our initial search because there aren't any MeSH terms for data quality. It is also crucial to remember that only one author completed the classification, which was primarily subjective. It might be wise to be more open about the data cleaning or verifying processes.

[4]According to the writers of MIMIKYU notes, the EHR's maximum context is insufficient to accurately forecast the note. Despite the possibility that additional prior occurrences could be instructive, they limited their background data within the previous 24 hours in order to limit input sequence length for performance concerns. To provide a more accurate depiction of patient-provider interactions in non-ICU datasets, a larger window of context should be used.

[5]An EMR dataset was analyzed using LDA to find patterns of co-occurring medical disorders. The findings show that the majority of the diseases classified together as subjects do, in fact, commonly co-occur. A system that simplifies diagnosis and data entry in clinical settings can be built on the data-driven strategy for finding related illnesses.

[6]A new web platform called EMIM combines numerous e-health services. Because it uses the cloud, any data can be found there at any moment (not isolated in desktop computers). appointments, conversations, and consults between the patient and the doctor can all take place online. The platform will continue to offer services in the future. The mobile EMIM application will soon be made available for both Android and iOS devices.

[7]When the models are applied to ASR transcripts after being trained on manual transcripts, their performance suffers noticeably in comparison to when they are applied to manual transcripts. The performance gap cannot be closed by training the model on ASR transcripts or on both ASR and manual transcripts. According to analysis, the SA-T model has greater precision while the Seq2Seq model has greater recall; as a result, the two models work well together. The better choice will be to combine two models.

[8]With reference to the wider picture of SOPHIE-like virtual agents, there are several

restrictions. According to prior studies, while speaking with a virtual agent or conversational agent powered by AI, people tend to take shorter rounds. Our LECT-UR scoring approach makes use of a window of subsequent turns that also includes the turn of the virtual agent. As a result, the lecturing feedback might change in response to the conversation's condition and the user's actions.

[9]Several trials were conducted, and SVM was employed as the classifier. The characteristics from the speech and image signals were combined to get the best accuracy (78%) possible. It plans to apply active learning, a highly advanced classification method that has been used to great effect in emotion recognition, in upcoming efforts. Audio features from MPEG-7 were successfully applied to an audio-visual emotion identification system. To improve the suggested system's accuracy, they might employ such attributes and incorporate more input modalities.

[10]In this study, speech engines, open source cross-platform telephones that may perform various functions, and flexible resource-offering developing technologies (the cloud) are examined. Like Asterisk, FreeSwitch is an open source PBX, but it is more flexible and abundant in its ability to scale and add modules as desired by the user, making it a dependable and practical platform that can be summed up as one machine performing several duties. The lengthy transcription procedure, regular spreadsheet data gathering, and sophisticated machine learning algorithms are the limits in this case.

2.3. Problem Statement and Objectives

2.3.1. Problem Statement

1. For most doctors to be able to function and carry out their tasks effectively, they need to save as much time as possible in every case.

2. Data from electronic medical records (EMRs) have grown to be essential to clinical care in recent years. Data entry into EMRs is currently laborious and prone to mistakes.

3. We want to make our project as natural and user-friendly as we can so that doctors and other organizations involved in medical science, such as pharma, can use it.

2.3.2. Objectives

1. To develop a text conversation analysis system for patients and medical professionals.

2. To create an interactive user interface that accepts text files for analysis.

3. Designing a system that is user friendly and easily accessible to medical professionals and organizations.

2.3.3. Scope

1. To make our findings more and more efficient and accurate as more advanced segmentation, pre-processing, testing methods are being investigated.

2. We may make use of different and bigger and bigger dataset to make the result more accurate. But owing to hardware restriction we are confined by our computing equipment power.

3. It may be integrated with various hardwares and softwares to give the best outcome for medical professionals but at the risk of being pricier than the present options.

Chapter 3

Software Requirements Specification (SRS)

3.1. Introduction

3.1.1. Purpose

The purpose of this project is to develop a system which will perform analysis on the medical transcript of an interaction between medical personnel and the patient. After this the system will classify the patient into a category of medical condition depending on the transcript. We can share this information in pdf format to the patient and also for keeping medical records.

3.1.2. Document Conventions

In this report, Every general text in Times New Roman Font and Font size should be kept 12pt, Chapter number and Chapter name has font size of 18pt with bold text. Italic text is used for figures and tables. General line spacing is 1.5.

EHR – Electronic Health Record

EMR – Electronic Medical Record

ML - Machine Learning

NLP - Natural Language Processing

3.1.3. Intended Audience and Reading Suggestions

The different types of users are -:

a) Doctors.

b) Hospitals.

c) Medical database Communities.

d) Patients.

3.1.4. Scope

This Product will automate the clinical note taking process. It will help in automatic development of a brief note of an interaction between the doctor and the patient. It will enable all the users to have editable access to the note for any error detection and correction.

3.2. Overall Description

3.2.1. Product Perspective

The client will have a client interface in which he/she can interact with in our system. He/she can start recording or either insert a recorded file as the input. After the input is inserted, analysis will be performed on it by first converting speech to text and then extracting medical information and other relevant information from it. This will be available for the user to edit and correct if any information is mislabeled or incorrect.

3.2.2. Product Functions

This section provides the functional overview of the product. The project will require the Python tkinter as a front end and at the back end we have used colab and python.

3.2.3 Operating Environment

Python : Python is an easy to learn, powerful programming language. It has efficient high-level data structures and a simple but effective approach to object-oriented programming.

Jupyter Notebook : It is an open source web application that you can use to create and share documents that contain live code, equations, visualizations, and text. Jupyter Notebook is maintained by the people at Project Jupyter. The core supported programming languages that it supports: Julia, Python, and R. Jupyter ships with the IPython kernel, which allows you to write your programs in Python.

Google colab : Colab notebooks integrate executable code and rich text, as well as graphics, HTML, LaTeX, and other elements, in a single document. Colab notebooks are saved in your Google Drive account when you create them.

3.2.4 Design and Implementation Constraints

The scope of this study is limited to using a dataset with limited meme images.

3.3 External Interface Requirements

3.3.1 User Interfaces

The user interface will accept text or an audio file as input and give a category in which the patient's medical condition belongs. This output will be efficient to categorize the number of medical conditions in a general category.

3.3.2 Hardware and Software Interfaces

A minimum of 4 GB RAM is required. Python, Google Colab, Jupyter Notebook.

3.3.3 Communications Interfaces

Users will have to give input in text format or audio document and output will be displayed on the user screen.

3.4 Other Nonfunctional Requirements

3.4.1 Performance Requirements

There are no specific performance requirements.

3.4.2 Safety Requirements

A secure connection between GUI and dataset must be established.

3.4.3 Security Requirements

The audio files or text data uploaded by the user must be protected against any security breach because if the images are hampered, the result will also be hampered.

3.4.4 Software Quality Attributes

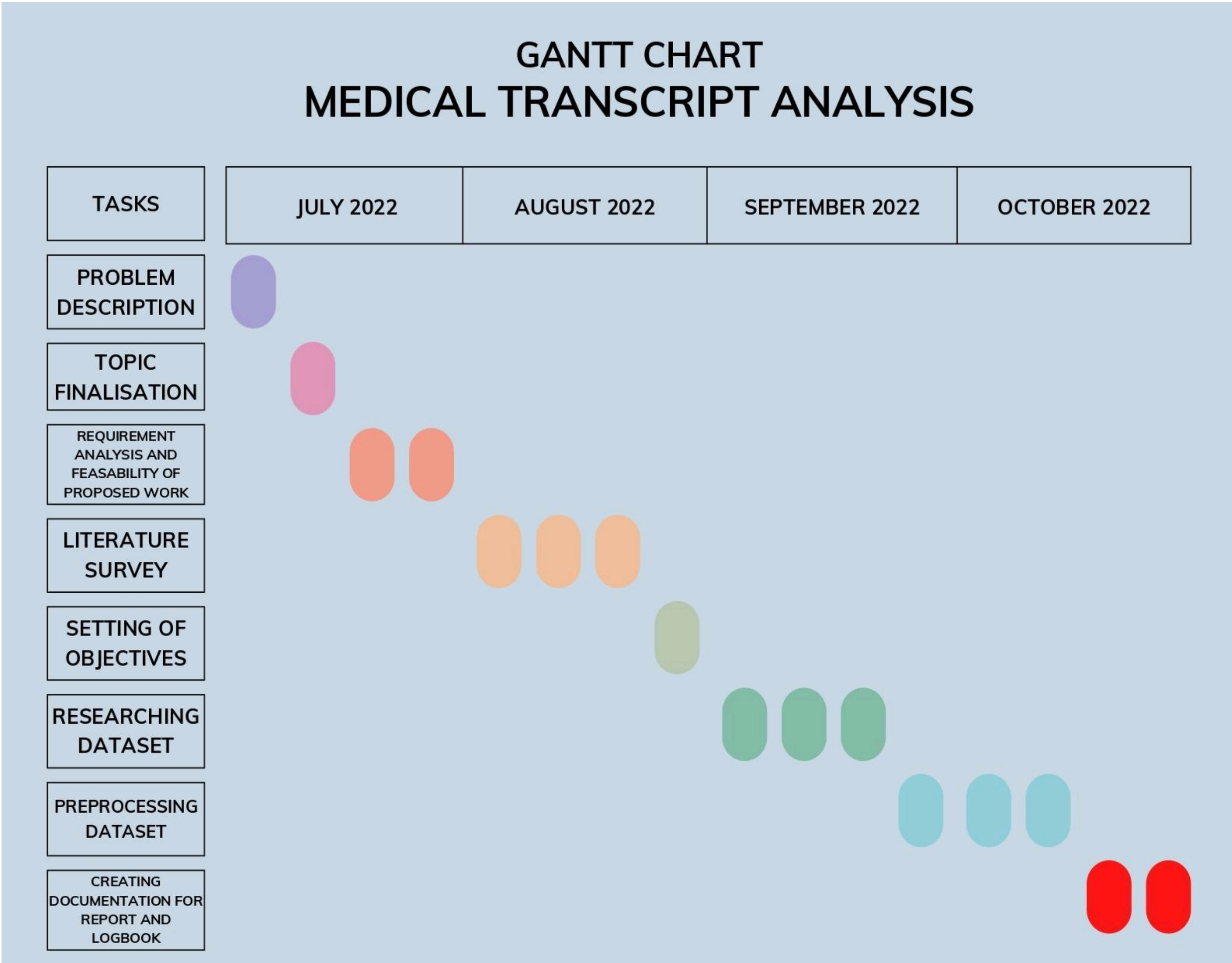
Adaptability, availability, correctness, flexibility, maintainability, reliability, reusability, robustness, testability, and usability.

3.4.5 Business Rules

This system can be used by medical organizations, doctors, patients,etc. So the decorum of the meeting audio must be maintained accordingly.

Chapter 4

Project Scheduling and Planning



Chapter 5

Proposed System

5.1. Algorithm / Analysis

5.1.1 Tf-Idf Feature Extraction

To gauge the significance of a word in a document or corpus, the notions of term frequency (TF) and inverse document frequency (IDF) are crucial in information retrieval (IR) and natural language processing (NLP).

The phrase "frequency of occurrence" (TF) refers to the number of times a term (word) appears in a document. It is derived by dividing the number of times a term appears in the document by the total number of words. A term's importance to the document is indicated by its TF value, which increases as it appears in the document.

Contrarily, DF quantifies a term's rarity within a corpus. It is derived by dividing the whole corpus's document count by the proportion of texts that specifically mention the phrase. A term's rarity and importance to the corpus increase with its IDF value.

When used together, TF-IDF gives each term in a document a score that accounts for both the phrase's relevance in the document (TF) and its rarity in the corpus (IDF). In an IR system, this score can be used to rank and retrieve pertinent documents in answer to a query.

5.1.2.Principal Component Analysis (PCA)

PCA (Principal Component Analysis) is a commonly used technique in machine learning and data analysis for reducing the dimensionality of a dataset. The aim of PCA is to identify the most important features (i.e., the principal components) that contribute the most to the variation in the data and transform the data into a new coordinate system that captures this variation.
PCA works by identifying the directions in which the data varies the most, and then projecting the data onto these directions. These directions are called principal components, and they are computed by finding the eigenvectors of the covariance matrix of the dataset. The eigenvectors with the largest eigenvalues represent the directions of maximum variation in the data and are therefore the most important components.
Once the principal components have been identified, the data can be transformed into a new coordinate system that is aligned with these components. This new coordinate system has a lower dimensionality than the original dataset, since it only includes the most important components. This can be useful for reducing the computational complexity of models, improving their accuracy, and visualizing high-dimensional data.

5.1.3.Entity Extraction

SpaCy Models: spaCy is a popular natural language processing library that provides a range of pre-trained models for processing text in various domains. In addition to its general-purpose models, spaCy also provides specialized models for processing biomedical, scientific, or clinical text.

These specialized models are trained on domain-specific text and can help extract relevant information from text in these domains. For example, the "en_core_sci_sm" model is a small English language model trained on biomedical text and can be used to extract entities like genes, diseases, and chemicals. Similarly, the "en_ner_clinical" model is a model trained on clinical text and can be used to extract entities like symptoms, procedures, and medications.

In addition to entity recognition, these models can also be used for other tasks such as sentence classification, dependency parsing, and named entity linking. By leveraging these pre-trained models, researchers and practitioners can speed up their natural language processing pipelines and focus on building domain-specific applications.

During the application of this model in our project, we faced a few limitations and hence opted to use SMOTE Algorithm.

SMOTE Algorithm: The problem of class imbalance is commonly addressed by the oversampling method known as SMOTE (Synthetic Minority Over-sampling Technique) in machine learning and data mining. Class imbalance happens when there is an unbalanced distribution of classes in the training dataset, meaning that some classes contain a lot more examples than others. This may lead to inaccurate model predictions, particularly for the minority class.

How SMOTE functions is it identifies samples from minority classes: Identify the minority class as the one with the fewest examples in the training dataset. Choose a sample from a minority class: Choose a sample of the minority class at random from the data. Identify its k-closest neighbors: Calculate the separations between the sample from the minority class that was chosen and its k-nearest neighbors. This stage typically involves the use of Euclidean distance. Create artificial samples: Choose a neighbor at random and interpolate between the sample from the minority class and the selected neighbor to create a synthetic sample. This is accomplished by averaging the feature values of the two samples using a weighted formula. Repeat: To generate the desired number of synthetic samples, repeat steps 2-4. Mix artificial samples with the original data: Including the produced synthetic samples to the training dataset, effectively oversampling the minority class to balance the class distribution.

5.2. Details of Hardware and Software

5.2.1 Hardware Requirements

System: laptop/desktop

Processor: Intel i3 5th Gen or greater

Ram: Min 4GB,

Recommended: 8GB

Storage: Min 2 GB HDD/SSD

Internet Connectivity (Optional)

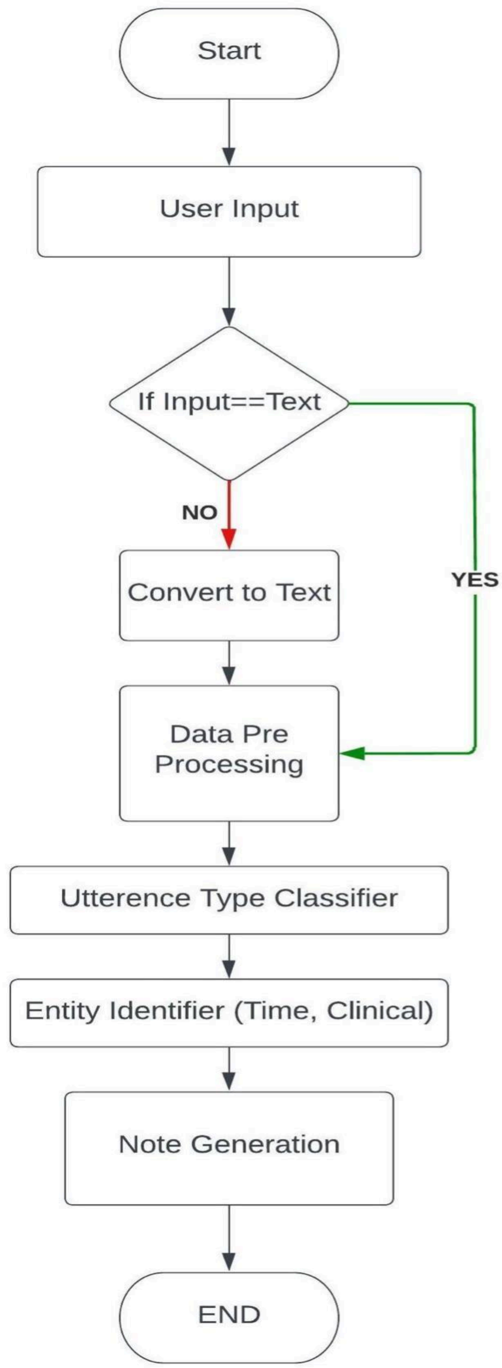
5.2.2 Software Requirements

PYTHON IDLE
GOOGLE COLAB
Python 3.10/3.11

5.3. Methodology CHANGE THIS DIAGRAM

#ADD

1. **dml**
2. **flow**
3. **system**
4. **use case**


Start
User Input
If Input==Text
NO
YES
Convert to Text
Data Pre
Processing
Utterence Type Classifier
Entity Identifier (Time, Clinical)
Note Generation
END

5.3.1 Dataset Collection

We used the medical text dataset from mtsample.com for our model. The dataset comprises five columns, namely description, medical_specialty, sample_name, transcription, and keywords. The dataset consists of text samples for 40 categories of medical specialities, making it a comprehensive source of medical data. This dataset allowed us to train our model to accurately classify medical text into its corresponding medical speciality.

We employ a dataset of 250 patient-clinician conversations that contains demographic data about the patient as well as the primary diagnosis for testing our models. The information consists of human-generated transcripts and audio recordings. The dataset basically includes audio files and their transcripts in text.

5.3.2 Data Preprocessing

ADD DESCRIPTION HERE

In conclusion, examining medical records is a worthwhile and difficult endeavor that can shed light on patient care, clinical outcomes, and healthcare administration. Medical transcript collections, however, frequently exhibit a class imbalance, with underrepresentation of key clinical outcomes or rare medical disorders. For a medical transcript analysis to be accurate and dependable, class imbalance must be addressed.

In this study, we investigated how the SMOTE method could be used to address class imbalance in medical transcript analysis. A common oversampling approach called SMOTE creates fake samples for the minority class in order to balance the distribution of the classes. We successfully increased the representation of the minority class and enhanced the performance of our machine learning model by applying SMOTE to the medical transcript dataset.

6. Implementation Details

6.1 Modules & Description

6.1.1 Pre Build Model:

a. Vosk:

Vosk is an open-source, offline voice recognition library that allows developers to include real-time speech recognition into their applications. It is intended to be small, portable, and readily integrated with other software projects.

Vosk is built on deep neural network models and utilizes cutting-edge speech recognition techniques to achieve excellent accuracy and low latency. It supports various languages, including English, Chinese, Russian, Spanish, Portuguese, French, and German.

Vosk can be used to create virtual assistants, speech-to-text transcription, voice-controlled systems, and speech-enabled mobile apps, among other things. It can also be utilized for automatic speech recognition in a variety of industries, including healthcare, banking, and legal.

Vosk is free and open-source software, which means that anybody can use, change, and distribute it freely. It also includes a pre-trained model, allowing developers to get started with speech recognition quickly and easily.

Overall, Vosk is a powerful tool for developers wishing to add voice recognition features to their applications, and its open-source nature and support for a variety of languages make it a good choice for a wide range of projects.

Used: larger pre trained model from range for converting audio file to corresponding text and real-time communication into text.

b. Build Model:

The "analyzer specialty" model is a machine learning tool that is designed to automatically identify the medical specialty associated with a given transcription from a CSV file. This model takes a CSV file as input, with a column named "transcription" containing the text to be analyzed. The main objective of this model is to detect the medical specialty associated with the transcription from a list of predefined options. These options could include various specialties, such as cardiology, oncology, neurology, and more. To develop this model, a machine learning approach is generally employed, where the model is trained on a dataset of transcriptions with their corresponding medical specialties. The model is trained to identify patterns and features in the text that are related to different medical specialties. Once the model is trained, it can be used to predict the medical specialty associated with new transcriptions. This can be done by inputting the transcription text into the model, which will then output the predicted specialty. The "analyzer specialty" model can be valuable in a range of contexts, including healthcare organizations that need to automatically categorize patient notes or in medical research where large amounts of text data need to be processed and analyzed. By automating this process, the model can save time and increase efficiency, allowing medical professionals to concentrate on other essential tasks.

6.1.2 List of Packages Used:

Csv
Wave
Pickle
Pyaudio
Pandas
Tkinter
Pillow
Scikit-learn
Vosk

Csv: The csv library is a Python module that provides an easy-to-use way to read and write CSV (Comma Separated Value) files. It provides a way to handle large datasets and to manipulate them in various ways, such as filtering and sorting. The csv library can be used to extract, process, and store data from various sources, making it a valuable tool for data analysis.

Wave: The wave module is a Python module that allows you to read and write WAV (Waveform Audio File Format) files, which are commonly used for digital audio data. The wave module can be used to process and analyze audio data, such as removing noise, amplifying or attenuating signals, or applying various filters. It provides a way to extract valuable information from audio data, making it a useful tool for speech recognition, music analysis, and other applications.

Pickle: The pickle module is a Python module that allows you to serialize and deserialize Python objects, which is useful for storing and retrieving data structures in a more efficient way. The pickle module can be used to save and load trained machine learning models, configuration files, or any other Python objects. It provides a way to store and retrieve data in a compact and efficient format, making it a useful tool for data storage and transfer.

Pyaudio: The Pyaudio library is a Python module that provides a way to play and record audio using Python. Pyaudio can be used to capture speech, music, or other sounds and process them in real-time. It provides a way to extract valuable information from audio data, such as detecting speech or music patterns, making it a useful tool for speech recognition, music analysis, and other applications.

Pandas: The pandas library is a Python module that provides high-performance, easy-to-use data structures and data analysis tools. It provides a way to work with tabular data, such as CSV files, in a powerful and flexible manner. Pandas can be used to clean, transform, and aggregate data, making it a valuable tool for data preprocessing and analysis. It provides a way to work with large datasets and to visualize and communicate the results of data analysis.

Tkinter: The Tkinter library is a Python module that provides a way to create graphical user interfaces (GUIs) using Python. Tkinter can be used to create windows, dialogs, buttons, and other GUI elements in a simple and easy-to-use manner. It provides a way to design and develop interactive applications, such as data visualization tools, image editors, or games.

Pillow: The Pillow library is a Python module that provides a way to work with images in Python. Pillow can be used to read, manipulate, and save various image formats, making it useful for applications such as image recognition or computer vision. It provides a way to preprocess images, such as resizing, cropping, or filtering, and to extract valuable information from them, such as detecting objects, faces, or patterns.

Scikit-learn: The scikit-learn library is a Python module that provides a wide range of machine learning algorithms and tools. Scikit-learn can be used to build and train machine learning models, and it also includes tools for data preprocessing and model evaluation. Scikit-learn provides a way to perform various tasks, such as classification, regression, clustering, and dimensionality reduction, making it a valuable tool for data analysis and prediction.

Vosk: The Vosk library is a speech recognition toolkit that provides a way to add real-time speech recognition capabilities to applications. Vosk is based on deep neural network models and supports a wide range of languages, making it useful for applications such as virtual assistants or voice-controlled devices. The library is lightweight, fast, and accurate, making it suitable for real-time applications. Vosk provides pre-trained models for various languages, which can be used to transcribe speech into text. The library can also be used to train custom models using your own data, allowing you to create speech recognition systems tailored to your specific needs. Vosk is open source and can be easily integrated into Python applications, making it a popular choice for developers working on speech recognition projects.

6.2 Snapshot

[#ADD SCREENSHOT FOR EACH ACTION WITH PROPER TITLE OR SUBTITLE DESCRIBING THE SAME PROCESS #everything center align #whole report justified and spacing from all four side as instructed]

Fig 6.2.1 main ui

Fig 6.2.2 Record button

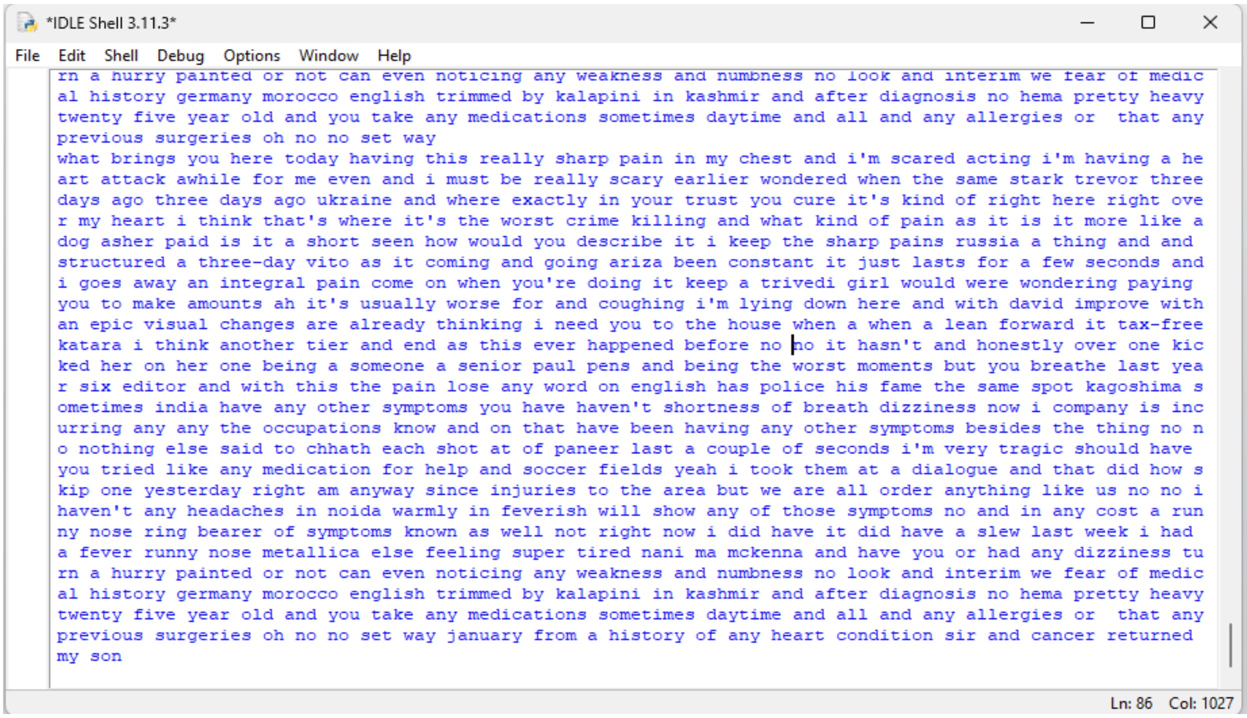


Fig 6.2.3 Transcript recognition after giving input in audio button

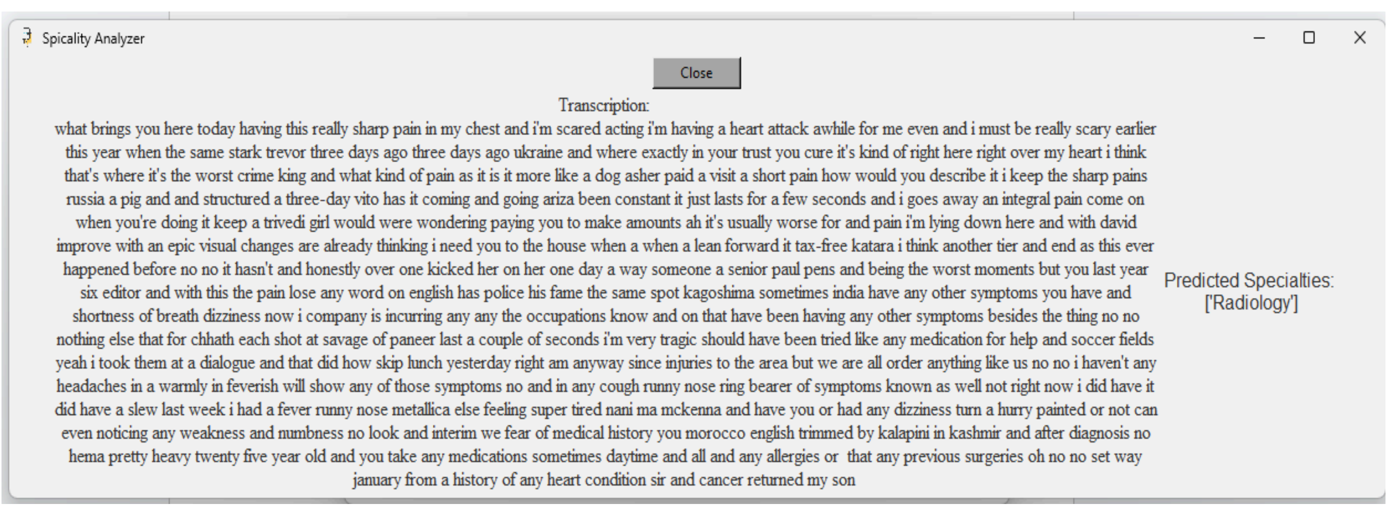


Fig 6.2.4 Final Output

7. Testing
Development of Test cases (Functional)

Test Case. ID	Objective	Description	Input	Expected Output	Actual Output	Result Remark
T01	Providing real time communication between doctor & patient for classification	To Classify into a medical speciality	Transcript 01 (CSV)	Radiology	Radiology	Pass
T02	Classification	Classify into a medical speciality	Transcript 02			Pass
T03	Classification	Classify into a medical speciality	Transcript 03			Pass

Test Case. ID	Objective	Description	Input	Expected Output	Actual Output	Result Remark
T04	**Classificatio n**	**Classify into a medical speciality**	**Transc ript 04**			**Pass**

#module by module test cases#
#ARRANGE CHAPTER SYSTEMATICALLY
#HEADING - 16
#subtext -14
#font - Times New Roman
#align text - center
#font color -balck

Chapter 6

#not needed!
#DELETE

Implementation plan for the next semester

Implementation	Timeline	Milestone
Implementation of clinically relevant entity extraction.	December 2022	Develop a text extraction model to get the clinically relevant utterances.

Implement the clinical notes creation.	January 2022	Create a clinical note in accordance with the National Resource Centre for EHR Standards (NRCeS) 2016.
Implement and test the project with other languages especially native to India.	February 2022	Create an system which can also handle the analysis in Indian native languages
Practical testing of the software with real doctor and patient dialogue.	March 2022	Approach a medical practitioner and gain insights into our project and learn about future scope and real world limitations.

Chapter 7

#not needed!

add CONCLUSION

ADD FUTURE SCOPE (IF ANY)

Summary

Medical practitioners still document patient procedures and operations in the same antiquated methods. Electronic medical records (EMR) or electronic health records are fully integrated with some papers (EHR). Some documents are still on paper and are not electronic, due to its slowness and error-proneness, data input can take up a substantial amount of time for doctors. This results in very varied clinical documentation, which presents challenges for machine learning algorithms. In this work, we create a clinical note using information from a doctor-patient dialogue that is clinically relevant. The client has the option to begin recording or to add an already recorded file as input. Following the insertion of the input, analysis will be carried out on it by first translating speech to text and then extracting medical data. In the event that any information is mislabelled or inaccurate, the user will have the option to update this. In our project at the preprocessing we have used Colab and Python. A minimum of 4 GB Ram is required to use our programme. Python, Google Colab, and Jupyter Notebook are all available. This system can be used by medical organizations, doctors, patients,etc.

Chapter 8

References

[1] Serena Jeblee, Faiza Khan Khattak, Noah Crampton, Muhammad Mamdani, and Frank Rudzicz. 2019. Extracting relevant information from physician-patient dialogues for automated clinical note taking. In Proceedings of the Tenth International Workshop on Health Text Mining and Information Analysis (LOUHI 2019), pages 65–74, Hong Kong. Association for Computational Linguistics.

[2] Rajkomar, A., Oren, E., Chen, K. et al. Scalable and accurate deep learning with electronic health records. npj Digital Med 1, 18 (2018).

https://doi.org/10.1038/s41746-018-0029-1

[3] Nicole Gray Weiskopf, Chunhua Weng, Methods and dimensions of electronic health record data quality assessment: enabling reuse for clinical research, Journal of the American Medical Informatics Association, Volume 20, Issue 1, January 2013, Pages 144–151, https://doi.org/10.1136/amiajnl-2011-000681.

[4] Liu, Peter J. "Learning to write notes in electronic health records." arXiv preprint arXiv:1808.02622 (2018).

[5] M. Bhattacharya, C. Jurkovitz and H. Shatkay, "Identifying patterns of associated-conditions through topic models of Electronic Medical Records," 2016 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), 2016, pp. 466-469, doi: 10.1109/BIBM.2016.7822561.

[6] L. Nita, "Cloud Platform for Medical Data Acquisition and Physician-Patient Interconnection," 2018 International Conference and Exposition on Electrical And Power Engineering (EPE), 2018, pp. 0996-0999, doi: 10.1109/ICEPE.2018.8559835.

[7] Nan Du, Kai Chen, Anjuli Kannan, Linh Tran, Yuhui Chen, and Izhak Shafran. 2019. Extracting Symptoms and their Status from Clinical Conversations. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 915–925, Florence, Italy. Association for Computational Linguistics.

[8] Ali, M.R., Sen, T.K., Kane, B., Bose, S., Carroll, T.M., Epstein, R.M., Schubert, L.K., & Hoque, E. (2020). Novel Computational Linguistic Measures, Dialogue System and the Development of SOPHIE: Standardized Online Patient for Healthcare Interaction Education. ArXiv, abs/2009.11247.

[9] A. Alamri, "Monitoring System for Patients Using Multimedia for Smart Healthcare," in IEEE Access, vol. 6, pp. 23271-23276, 2018, doi: 10.1109/ACCESS.2018.2826525.

[10] Kulkarni, Sushmita & Torse, Dattaprasad & Kulkarni, Deepak. (2020). A Cloud based Medical Transcription using Speech Recognition Technologies. 7. 2395-0056.

[11] Ngai, Hillary and Frank Rudzicz. “Doctor XAvIer: Explainable Diagnosis on Physician-Patient Dialogues and XAI Evaluation.” BIONLP (2022).