Lambton College | In Toronto

AML 2304

Natural Language Processing

Learning Lab

[Medical Transcription Prediction]

Group Name: Group 05

Group Members:

First name	Last Name	Student number
Bibek	Shiwakoti	C0898100
Jumana	Lightwala	C0892358
Prajina	Rajkarnikar	C0906293
Ronak	Shah	C0891297
Umesh	Khatiwada	C0893975

Submission date: ***[18-04-2024]***

Contents

1.ABSTRACT……………………………………………………………………………………………01

2.INTRODUCTION…………………………………………………………………………….

3. DATA VISULISATION…………………………………………………………………………….

4. METHOD…………………………………………………………………………………………..

5.PROJECT PROPASAL………………………………………………………………………..

3.1 MOTIVATION……………………………………………………………………………..

3.2 INTENDED EXPERIEMENTS…………………………………………………………………………

3.3 PROJECT MILESTONE……………………………………………………………………..

1. ABSTRACT 1

2. INTRODUCTION 2

3. DATASET 3

4. DATA VISULISATION 3

4.1. BAR CHART 4

4.2. WORD CLOUD 4

4.3. HISTOGRAM 5

4.5. BAR CHART VARIATIONS FOR MEDICAL SPECIALITIES 6

5. METHODS 10

5.1 DATA COLLECTION AND PRE-PROCESSING 10

5.2 CLEANING TEXT AND NORMALIZATION 10

5.3. TOKENIZATION 11

5.4. REMOVING PUNCTUATION AND STOP WORDS 11

5. IMPLEMENTATION OF MODEL 12

5.1. LOGISTIC REGRESSION 12

5.2. RANDOM FOREST: 12

5.3. NAÏVE BAYES 12

4.6 WORD2VEC 12

4.7 HYPERPARAMETER TUNING 13

6. PROJECT PROPOSAL 14

6.1. MOTIVATION 14

6.2. INTENDED EXPERIMENTS 14

6.2.1. DATASET ADJUSTMENT 14

6.2.3. HYPERPARAMETER TUNNING 14

6.2.4. ALGORITHM EVALUATION 14

6.2.5. METRICS ANALYSIS 15

6.2.6. MODEL INTERPRETATION 15

6.3. VERSIONING 15

6.4. PROJECT MILESTONE 16

7. RESULT 16

7.1. PERFORMANCE METRICS 16

7.1.1. LOGISTIC REGRESSION (BEST MODEL) 17

7.1.2. MULTINOMIAL NAÏVE BAYES: 17

7.1.3. SUPPORT VECTOR MACHINE (SVM) 17

7.1.4. RANDOM FOREST 18

7.2. PERFORMANCE EVALUATION 18

7.2.1. K-FOLD CROSS VALIDATION 18

7.2.2. HYPERPARAMETER TUNING 19

7.2.3. PCA ANALYSIS 20

8. DISCUSION 22

8.1. LIMITATIONS 22

8.2. APPROACH 22

12. CONCLUSION AND FUTURE WORK 23

12.1. SUMMARY 23

12.2. CONCLUSION 23

12.3. FUTURE WORK 24

13. REFERENCE 25

TABLE OF FIGURES:

Figure 1:Bar chart for the count of medical specialties 4

Figure 2: word cloud 5

Figure 3: histogram of distribution of token count in Transcriptions. 5

Figure 4: Bar graph for the frequency of Medical Specialties. 6

Figure 5: Bar graph for the frequency of Medical Specialities. 7

Figure 6:Error graph for Frequency of patient by medical specialty. 8

Figure 7:Bar Graph for Percentage of top 15 most repeated words in a dataset 9

Figure 8: histogram graph for the frequency of history across transcriptions. 9

Figure 9: Scatterplot for the correlation between transcription length and medical specialities 10

Figure 10: Word Cosine similarity using Word2Vec model. 13

Figure 11: Model accuracy comparison on TFIDF vectorizer. 17

Figure 12: Best parameter after grid search. 19

Figure 13: confusion matrix for optimal model obtained on Grid Search 20

1. ABSTRACT

The main aim of the project is to predict diagnostics needed for a patient with certain diseases. The diseases comprise a raw text file containing a doctor's notes detailing the patient's age, free-form complaints, patient's medical history, and other relevant information. It lacks structure, meaning that sections present in one patient's diseases may be absent in another's.

The target labels consist of the necessary diagnostic procedures.

The potential impact of this solution lies in its ability to assist doctors in determining the most suitable diagnostic tests for patients. This can lead to significant saving of time and money for patients while enabling doctors to allocate their time more efficiently by avoiding unnecessary tests. Furthermore, in complex cases, the algorithm can expedite the diagnostic process, potentially saving lives by providing faster and more accurate diagnoses. Beyond immediate clinical benefits, the algorithm's identification of patterns and regularities in patient data holds promise for medical research. These insights could uncover novel connections between symptoms and diseases, paving the way for innovative treatment approaches. By increasing unobvious correlations, researchers may gain a deeper understanding of diseases and develop more targeted therapies, ultimately improving patient outcomes and advancing medical science.

In essence, this solution not only enhances clinical decision-making and patient care but also contributes to broader medical knowledge and the development of more effective treatments

2. INTRODUCTION

The major component of healthcare documentation is medical transcriptions are benefiting more and more from cutting-edge Natural Language Processing (NLP) techniques. Medical transcription plays a crucial role in healthcare by converting voice-recorded medical reports into written text. The accuracy and efficiency of this process directly impacts patient care, clinical decision-making, and medical research. In this report, we delve into the realm of Medical Transcription Prediction, where machine learning techniques are harnessed to automate and enhance the transcription process.

In the healthcare and medical data, it is necessary to accurately classify medical specialties based on transcription text. With the huge amount of textual data generated in healthcare, such as patient consultations, medical transcriptions ,medicines, treatments and diagnoses automated classification systems can streamline processes, help in decision-making, and improve overall healthcare delivery. We have develop a robust and reliable classification model that can effectively categorize medical specialties based on the content of transcription texts.

Our objective is to develop and evaluate machine learning models that can accurately predict and transcribe medical reports from voice recordings. By leveraging advanced algorithms and text representation methods, we aim to improve the accuracy, speed, and reliability of medical transcription, ultimately contributing to streamlined healthcare workflows and improved patient outcomes.

The classification process in healthcare can help providers save time and effort by categorizing transcriptions accurately. This enables them to quickly access relevant information tailored to each medical specialty, ensuring appropriate treatments are identified and administered.

Our main goal is to investigate how class imbalance, a common problem in medical data, affects machine learning algorithms' ability to predict outcomes. To do this, we concentrate on three very useful methods for analyzing medical data : Logistic Regression, Random Forest and Naive Bayes.

This project has important ramifications for enhancing the precision and dependability of healthcare decision support systems in addition to pushing the boundaries of medical data analysis. The knowledge gathered from this research may help to improve patient care and clinical results by facilitating the more efficient application of machine learning methods in the processing of medical transcriptions.

3. DATASET

The dataset used in this project has 6 columns and 4999 rows. The columns that were useful for the further analysis includes the following:

description	Short description of transcription
medical_specialty	Medical specialty classification of transcription
sample_name	Transcription title
transcription	Sample medical transcriptions
keywords	Relevant keywords from transcription

4. DATA VISULISATION

We have done Univariate analysis such as Histogram and pie chart and Bivariate analysis such as Heatmap to better understand the data and uncover the patterns that exists between various features of the dataset.

4.1. BAR CHART

Bar chart for the distribution of Surgery, Speech-Language, Radiology. are plotted as below

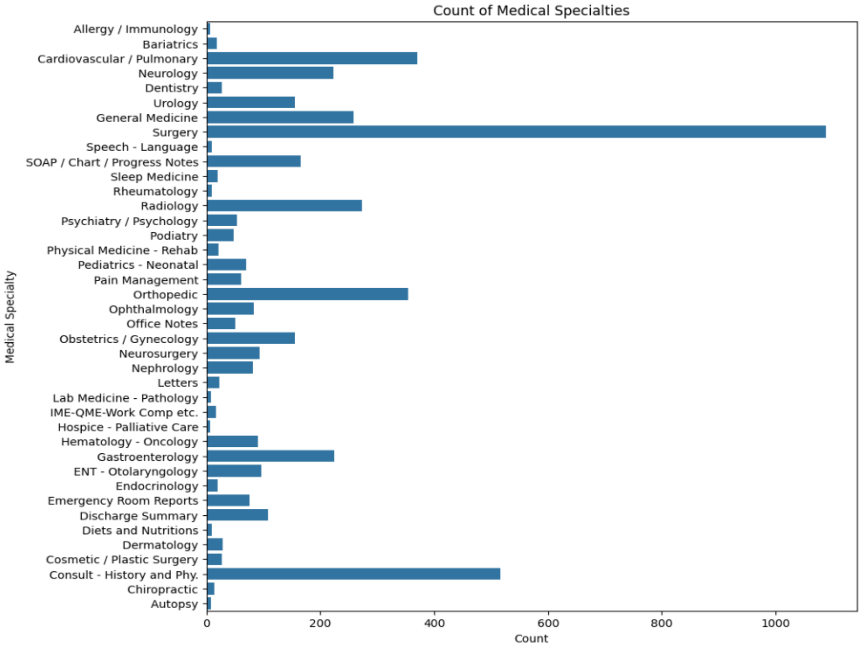


Figure 1:Bar chart for the count of medical specialties

Over 1100 data points are associated with "Surgery," comprising over 25% of the total data. And the least amount of the Specialties were Speech-Language around 5%. Additionally, concerning. this reveals that most medical specialties were included in Surgery.

4.2. WORD CLOUD

The word cloud, it is made up of a collection of words in varying font sizes, with each word's size corresponding to how frequently that term appears in the data. In the word cloud, a word's font size will increase with increasing frequency of occurrence. We can clearly see that patient, procedure, and incision are mostly used.

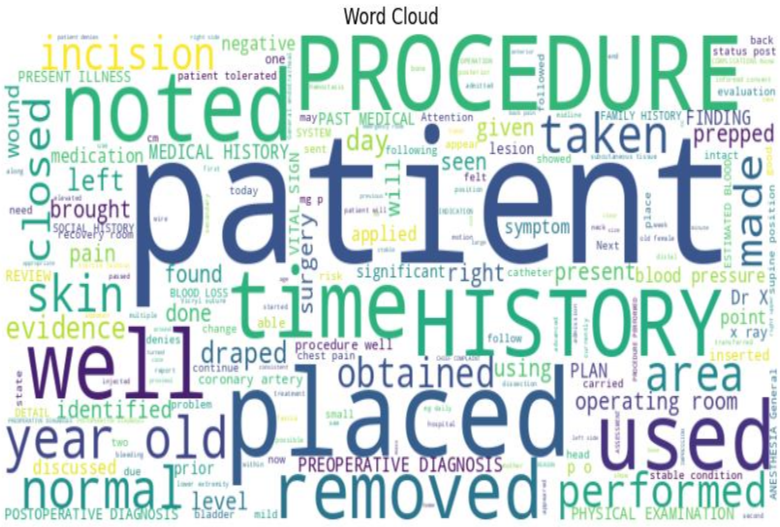


Figure 2: word cloud

4.3. HISTOGRAM

The histogram helps to identify the distribution of token count in Transcriptions, it is right skewed.

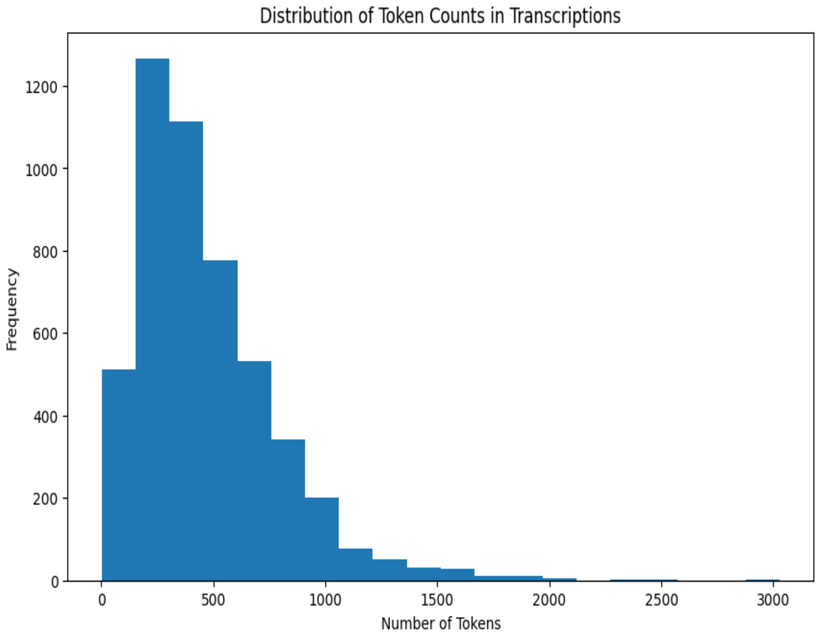


Figure 3: histogram of distribution of token count in Transcriptions.

4.5. BAR CHART VARIATIONS FOR MEDICAL SPECIALITIES

The bar plot helps us to visualizes the top 10 most frequent medical specialties in the dataset, with each bar representing a specialty and its height indicating the frequency of occurrences. The graph helps identify the most common specialties and their relative prevalence within the dataset.

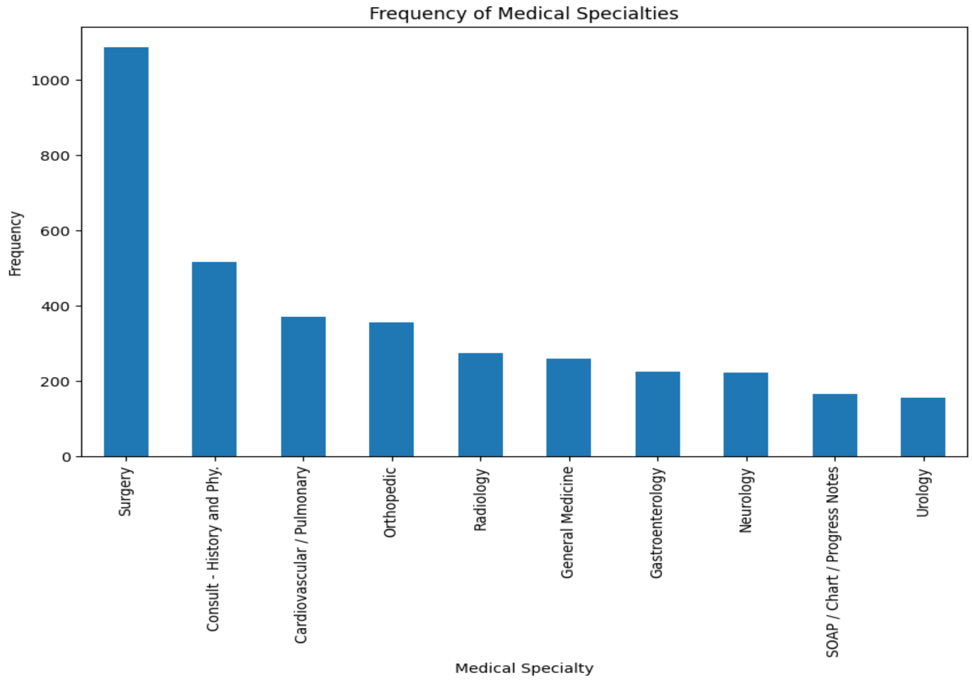


Figure 4: Bar graph for the frequency of Medical Specialties.

The bar chart given below helps to get the frequency of medical specialities such as the surgery as the highest number of frequencies followed by cardiovascular diseases.

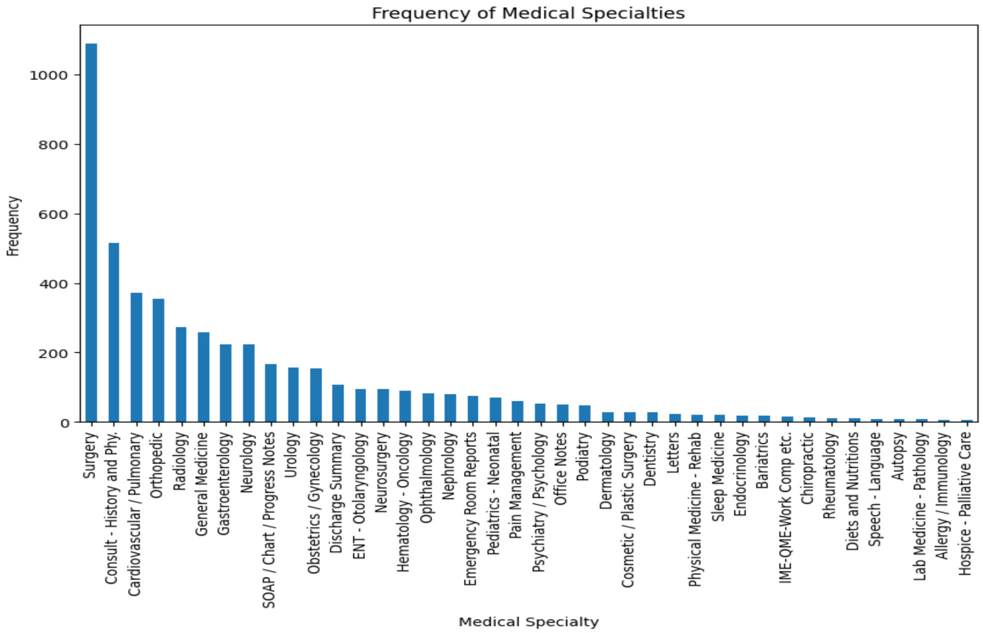


Figure 5: Bar graph for the frequency of Medical Specialities.

The below chart helps us to identify the medical speciality with the frequency of patient, Surgery has the highest number of patients compared to other medical specialities.

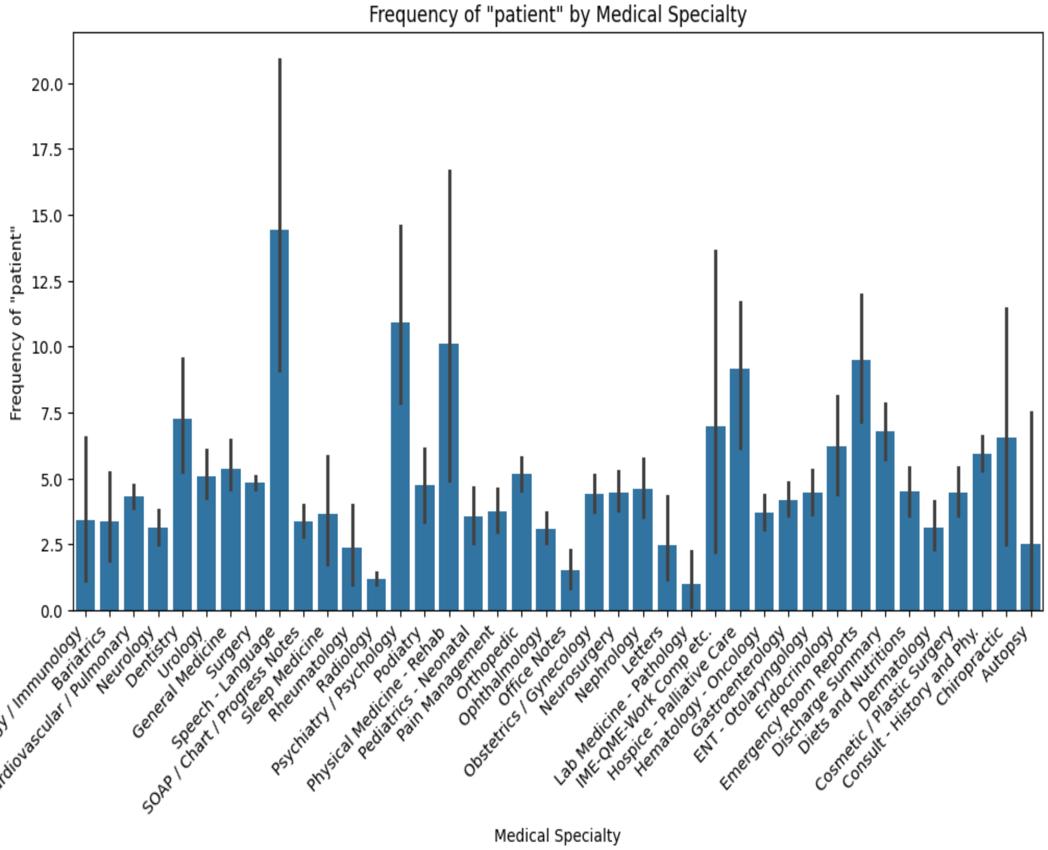


Figure 6:Error graph for Frequency of patient by medical specialty.

The below bar chart shows, the words that occurred more on our dataset, this graph is made by splitting the sentence into tokens and identifying the frequency of each word and then plotting the words with higher frequency count.

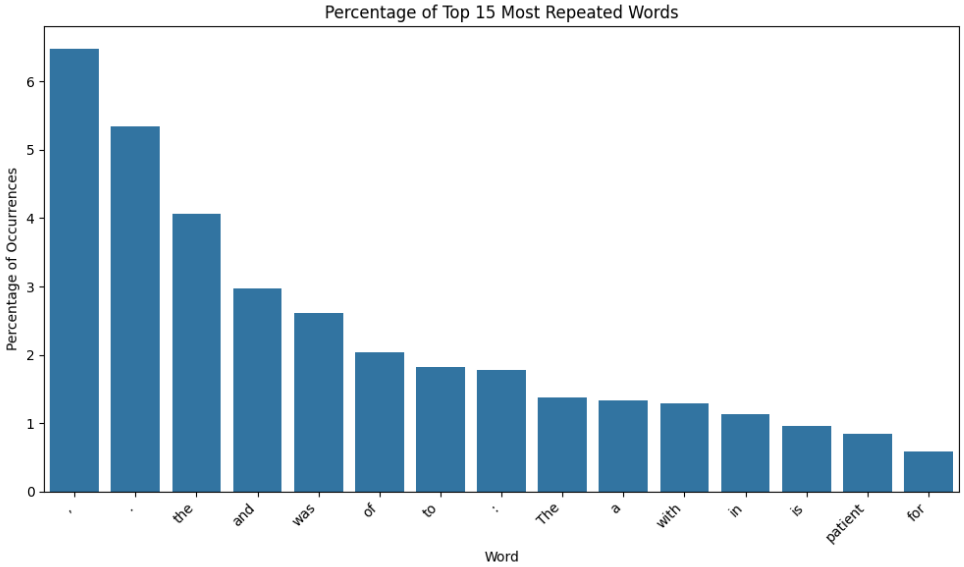


Figure 7:Bar Graph for Percentage of top 15 most repeated words in a dataset

The plot generates a histogram graph that visualizes the frequency of the word 'history' across all transcriptions.

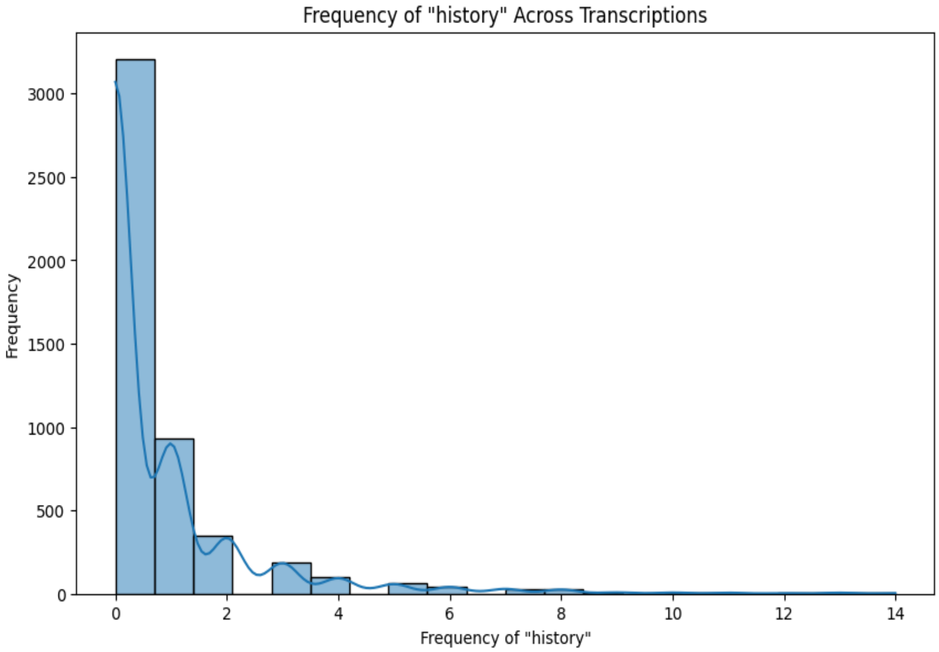


Figure 8: histogram graph for the frequency of history across transcriptions.

The plot provided generates a scatterplot graph showing the correlation between Transcription Length and Medical Specialty.

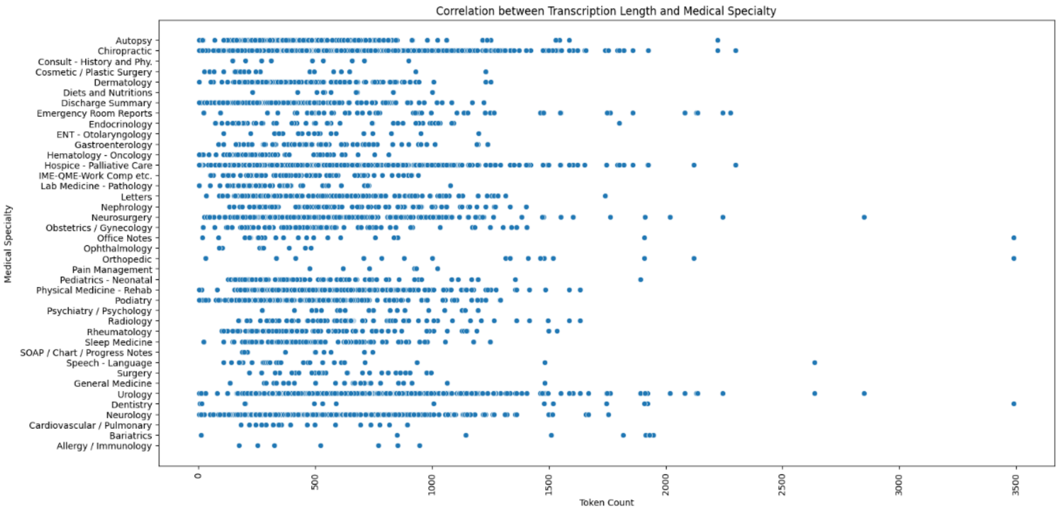


Figure 9: Scatterplot for the correlation between transcription length and medical specialities

5. METHODS

5.1 DATA COLLECTION AND PRE-PROCESSING

In this stage, healthcare providers' voice-recorded reports are gathered, and they are formatted so that transcription may begin. Data cleansing, data normalization, and addressing any incomplete or missing data are all included in the process of preparing data. To prepare the data for additional processing, it could also entail arranging the reports according to patient ID and medical specialization, date.

5.2 CLEANING TEXT AND NORMALIZATION

Cleaning text entails eliminating any extraneous characters, symbols, or background noise from the textual information. This entails eliminating HTML tags, non-alphanumeric characters, special characters, etc. By transforming the text into a consistent format, text normalization seeks to standardize the content. This can involve doing things like handling abbreviations, expanding contractions, changing all text to lowercase, and fixing spelling errors.

5.3. TOKENIZATION

Tokenization is the method of dividing text into smaller pieces known as tokens. Depending on the tokenizer being used, tokens can be sentences, letters, or phrases. This stage is essential for making text data processing and analysis since it facilitates the extraction of valuable information from the text.

5.4. REMOVING PUNCTUATION AND STOP WORDS

Common words like "the," "and" "is," etc. are frequently removed during text analysis because they don’t add any meaning to it. Removing stop words increases text processing systems' performance by cutting down on noise. To concentrate on the text's real content, punctuation—such as commas, periods, exclamation points, etc.—is usually handled or eliminated during text cleaning.

5. IMPLEMENTATION OF MODEL

5.1. LOGISTIC REGRESSION

In medical transcription, tasks like sentiment analysis (e.g., patient satisfaction levels), binary classification (e.g., presence or absence of a medical condition), and outcome prediction based on textual data are frequently performed using logistic regression. It can handle both categorical and numerical input features, making it appropriate for binary classification jobs with a categorical conclusion.

5.2. RANDOM FOREST:

Random Forest works well for managing and minimizing overfitting, noisy data, processing high-dimensional data. It is appropriate for multi-class classification (e.g., classifying medical reports into different disciplines) and feature importance ranking (e.g., finding important medical terms or phrases). To increase accuracy and generality, Random Forest constructs several decision trees and merges their predictions. It is appropriate for complicated medical transcribing datasets since it is resilient.

5.3. NAÏVE BAYES

Because it is an effective probabilistic classifier that relies on feature independence, Naive Bayes is well-suited for text classification applications. It's frequently applied to projects like sentiment analysis, and medical report classification. Using the Bayes theorem, Naive Bayes predicts the likelihood of each class based on the feature values. It performs well on huge datasets and is computationally efficient.

4.6 WORD2VEC

We used this for preprocessed tokens to train a word embedding model using Word2Vec. As it can quickly and effectively identify related keywords in medical contexts by finding semantically similar words based on vector representations and saving the model for later use.

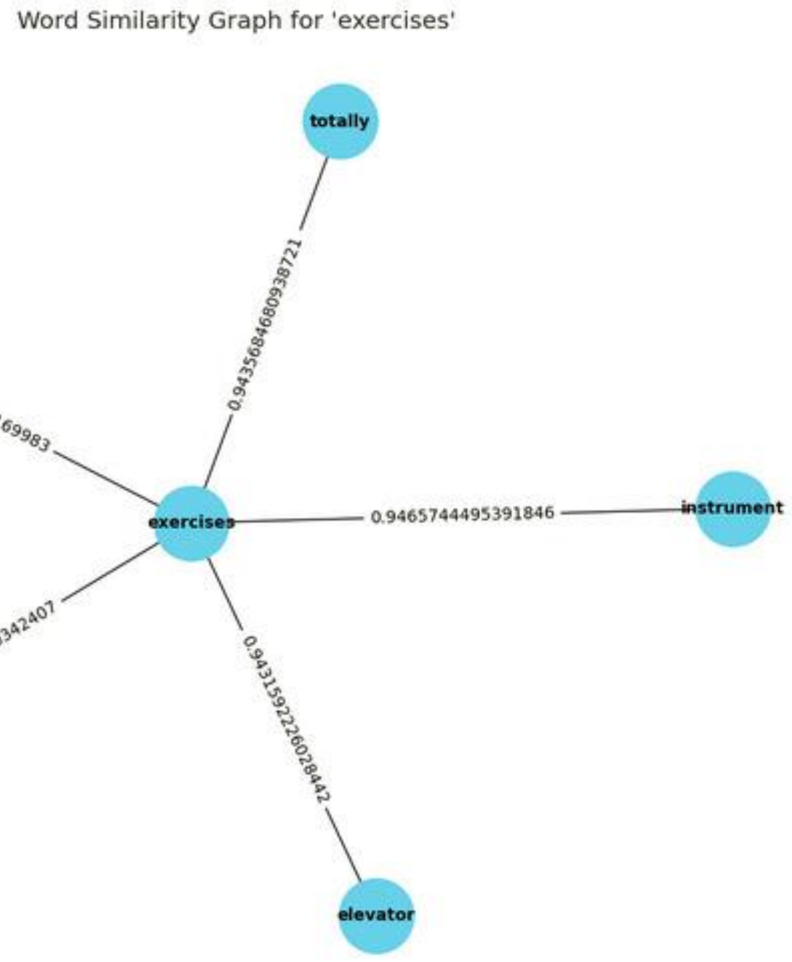


Figure 10: Word Cosine similarity using Word2Vec model.

4.7 HYPERPARAMETER TUNING

Using a Medical Transcription dataset, different machine learning models (e.g., Random Forest, Logistic Regression) were subjected to hyperparameter tuning by cross-validation. An ideal model for transcribing jobs was chosen with the help of metrics such as mean accuracy and standard deviation, which were used to determine which model performed the best.

6. PROJECT PROPOSAL

6.1. MOTIVATION

The primary problem we aim to address in the context of medical transcriptions is the issue of class imbalance within the dataset. Class imbalance refers to a scenario where one class (e.g., a rare medical condition) is significantly underrepresented compared to another class (e.g., common medical conditions) in the data. This imbalance may result in biased machine learning models that are not very good at accurately predicting instances of the minority class.

The main goal is to increase the accuracy of predictive models and decision support systems based on medical transcriptions, leading to more accurate clinical decision-making, patient outcomes better, and overall improvement in healthcare quality.

6.2. INTENDED EXPERIMENTS

6.2.1. DATASET ADJUSTMENT

We started by adjusting the dataset to address class imbalance issues commonly found in medical data. This involved generating synthetic samples for the minority class to create a more balanced dataset.

6.2.3. HYPERPARAMETER TUNNING

Utilized Grid Search and K-Fold Cross Validation for hyperparameter tuning of machine learning algorithms.

6.2.4. ALGORITHM EVALUATION

Next, we evaluated the performance of three widely used machine learning algorithms—Naive Bayes, Logistic Regression, and Random Forest—on both the original dataset and the adjusted dataset. This comparison allowed us to assess the impact of Oversampling techniques on algorithm performance.

6.2.5. METRICS ANALYSIS

We analyzed various performance metrics such as BOW and TF-IDF Train and Test Accuracy to measure the effectiveness of each algorithm in handling class imbalance and improving predictive capabilities.

6.2.6. MODEL INTERPRETATION

To understand how approaches affected the decision boundaries, feature importance rankings, and general model behavior, we also concentrated on reading the models trained on the altered dataset.

6.3. VERSIONING

Below are the specific versions of libraries used in our project, which served as prerequisites for ensuring compatibility and addressing potential version-related issues:

Pandas	Version 2.1.4
Scikit-learn	Version 1.2.2
NumPy	Version 1.26.3
NLTK	Version 3.8.1
Matplotlib	Version 3.8.0
Wordcloud	Version 1.9.3

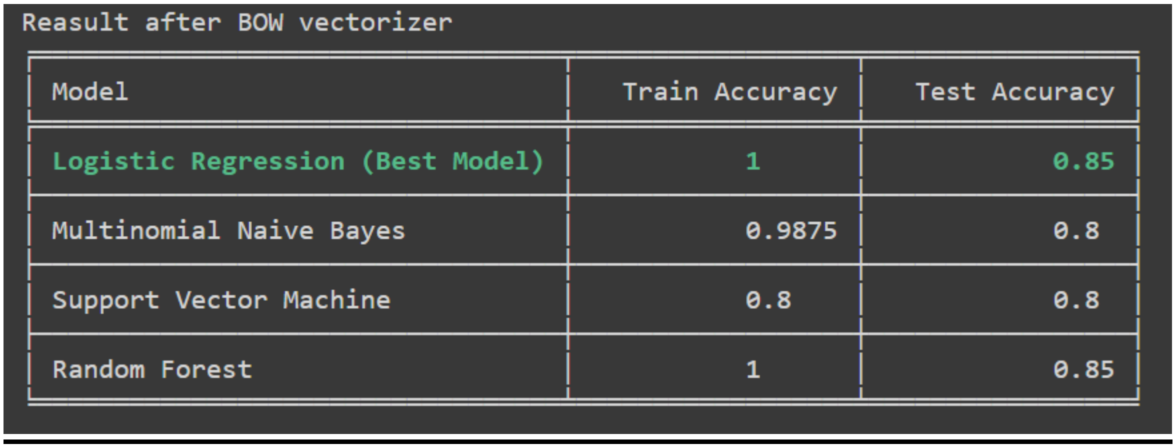
6.4. PROJECT MILESTONE

Task 1: Data cleaning and preprocessing	Ronak
Task 2: PCA and Data Visualization, formatting and finalizing the document.	Prajina
Task 3: Hyperparameter tuning using Grid Search	Umesh
Task 4: Model training and evaluation	Bibek
Task 5: Writing project documentation and final report	Jumana

7. RESULT

7.1. PERFORMANCE METRICS

Performance metrics for the Medical Transcription project include accuracy, F1 score, confusion matrix, precision, recall, , ROC curve with AUC, These metrics assess model accuracy, class-wise performance, and discrimination ability, aiding in evaluating models using Bag-of-Words and TF-IDF representations.



```
Reasult after BOW vectorizer
╒═══════════════════════════════════╤═════════════════╤═════════════════╕
│ Model                             │  Train Accuracy │   Test Accuracy │
╞═══════════════════════════════════╪═════════════════╪═════════════════╡
│ Logistic Regression (Best Model)  │               1 │            0.85 │
├───────────────────────────────────┼─────────────────┼─────────────────┤
│ Multinomial Naive Bayes           │          0.9875 │             0.8 │
├───────────────────────────────────┼─────────────────┼─────────────────┤
│ Support Vector Machine            │             0.8 │             0.8 │
├───────────────────────────────────┼─────────────────┼─────────────────┤
│ Random Forest                     │               1 │            0.85 │
╘═══════════════════════════════════╧═════════════════╧═════════════════╛
```

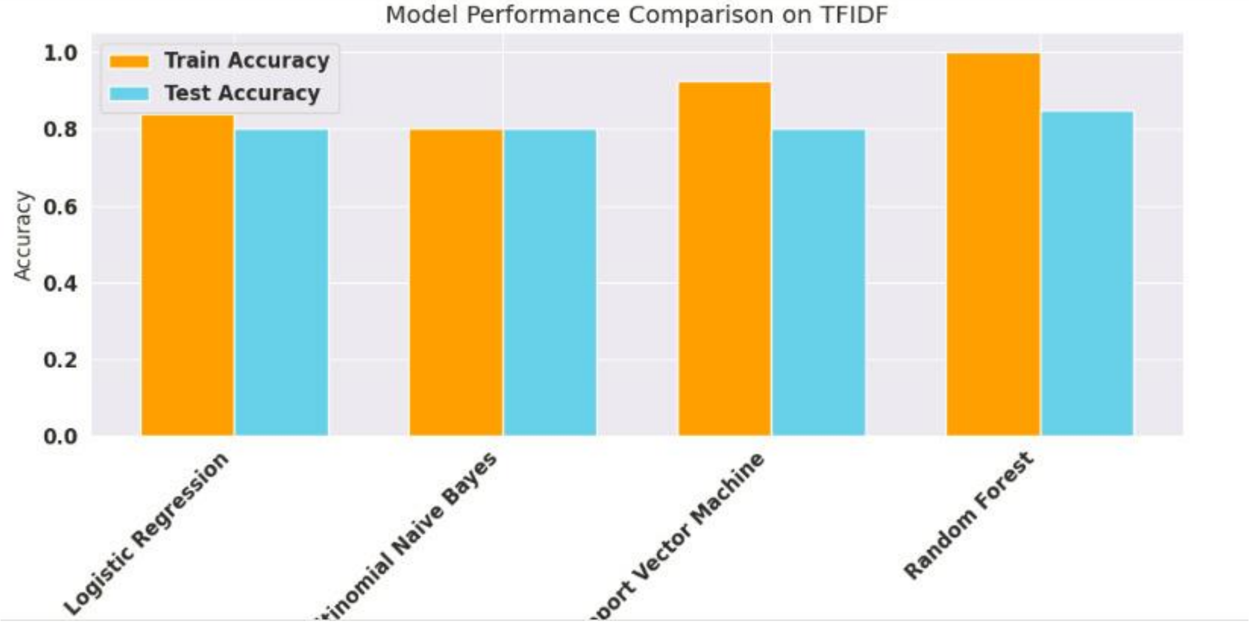


Figure 11: Model accuracy comparison on TFIDF vectorizer.

7.1.1. LOGISTIC REGRESSION (BEST MODEL)

Logistic Regression was trained on medical transcription data.

It achieved a high training accuracy of 1 (100%) and a test accuracy of 0.85 (85%).

Logistic Regression is more effective in tasks like binary classification and is suitable for medical transcription tasks were predicting outcomes (e.g., medical conditions, sentiments) is crucial.

7.1.2. MULTINOMIAL NAÏVE BAYES:

Multinomial Naive Bayes was trained on preprocessed data.

It achieved a training accuracy of 0.9875 (98.75%) and a test accuracy of 0.8 (80%).

Naive Bayes is a probabilistic classifier that assumes independence among features, making it suitable for text classification tasks like medical transcription.

7.1.3. SUPPORT VECTOR MACHINE (SVM)

SVM was trained on preprocessed data.

It achieved a training accuracy of 0.8 (80%) and a test accuracy of 0.8 (80%).

SVM is effective for both linear and non-linear classification tasks and can handle high-dimensional data, making it suitable for text classification in medical transcription.

7.1.4. RANDOM FOREST

It achieved a high training accuracy of 1 (100%) and a test accuracy of 0.85 (85%).

Random Forest is an ensemble learning method that combines multiple decision trees to improve accuracy and reduce overfitting, making it effective for text classification tasks.

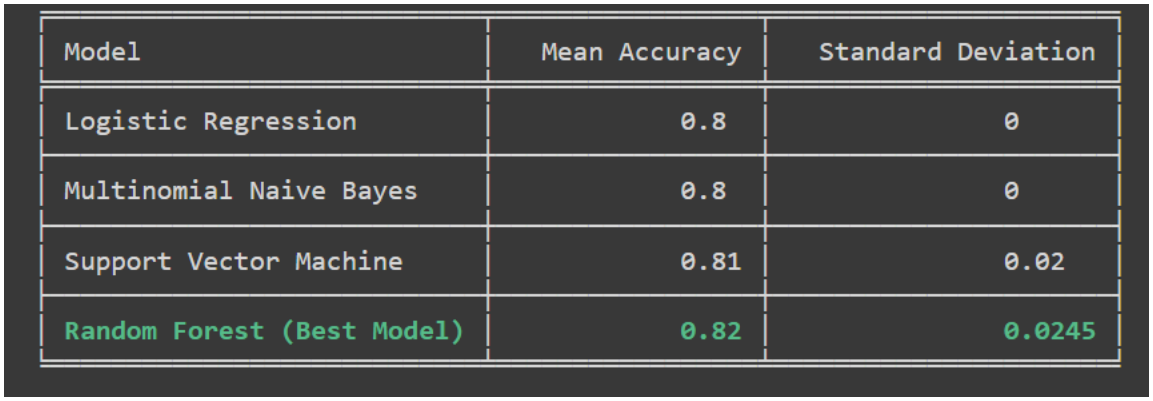
7.2. PERFORMANCE EVALUATION

The best-performing model based on test accuracy is Logistic Regression with an accuracy of 85%.

Random Forest also achieved the same test accuracy of 85%, indicating its effectiveness in the task.

Multinomial Naive Bayes and SVM achieved slightly lower test accuracies of 80%, but they are still viable options depending on specific requirements and trade-offs (e.g., computational complexity, interpretability)

7.2.1. K-FOLD CROSS VALIDATION

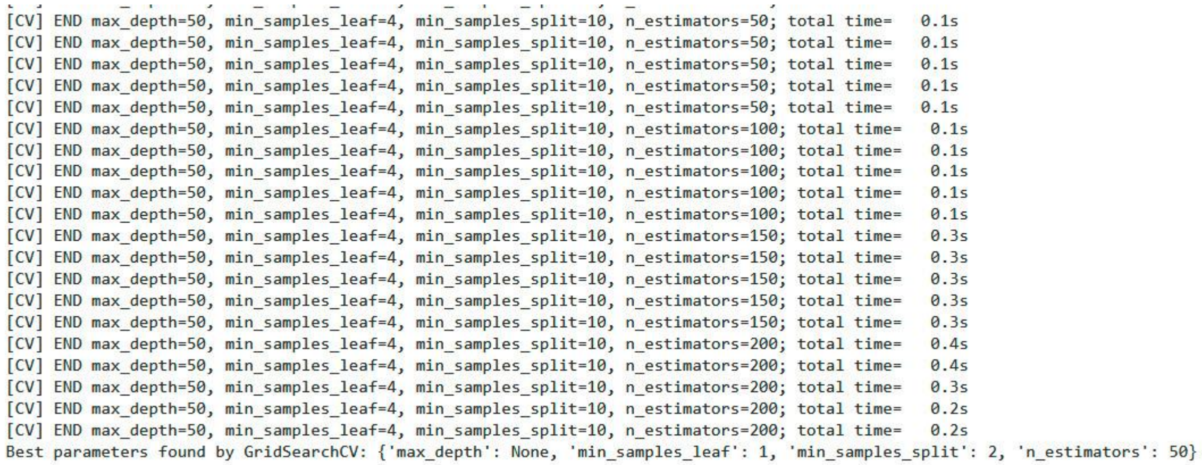


Model	Mean Accuracy	Standard Deviation
Logistic Regression	0.8	0
Multinomial Naive Bayes	0.8	0
Support Vector Machine	0.81	0.02
Random Forest (Best Model)	0.82	0.0245

The random forest model achieved the highest mean accuracy of 82% among all models evaluated. It also had a standard deviation of 0.0245, indicating slightly more variability in performance compared to logistic regression and Naive Bayes but still demonstrating robustness.

7.2.2. HYPERPARAMETER TUNING

In our project we have used two hyperparameter tuning techniques which are used for optimizing the RandomForestClassifier: RandomizedSearchCV and GridSearchCV. RandomizedSearchCV performs randomized search across various parameter distributions, while GridSearchCV exhaustively searches through a specified parameter grid. Both approaches aim to find the best combination of hyperparameters to increase model accuracy and performance in transcribing medical reports accurately.



```
[CV] END max_depth=50, min_samples_leaf=4, min_samples_split=10, n_estimators=50; total time=   0.1s
[CV] END max_depth=50, min_samples_leaf=4, min_samples_split=10, n_estimators=50; total time=   0.1s
[CV] END max_depth=50, min_samples_leaf=4, min_samples_split=10, n_estimators=50; total time=   0.1s
[CV] END max_depth=50, min_samples_leaf=4, min_samples_split=10, n_estimators=50; total time=   0.1s
[CV] END max_depth=50, min_samples_leaf=4, min_samples_split=10, n_estimators=50; total time=   0.1s
[CV] END max_depth=50, min_samples_leaf=4, min_samples_split=10, n_estimators=100; total time=   0.1s
[CV] END max_depth=50, min_samples_leaf=4, min_samples_split=10, n_estimators=100; total time=   0.1s
[CV] END max_depth=50, min_samples_leaf=4, min_samples_split=10, n_estimators=100; total time=   0.1s
[CV] END max_depth=50, min_samples_leaf=4, min_samples_split=10, n_estimators=100; total time=   0.1s
[CV] END max_depth=50, min_samples_leaf=4, min_samples_split=10, n_estimators=100; total time=   0.1s
[CV] END max_depth=50, min_samples_leaf=4, min_samples_split=10, n_estimators=150; total time=   0.3s
[CV] END max_depth=50, min_samples_leaf=4, min_samples_split=10, n_estimators=150; total time=   0.3s
[CV] END max_depth=50, min_samples_leaf=4, min_samples_split=10, n_estimators=150; total time=   0.3s
[CV] END max_depth=50, min_samples_leaf=4, min_samples_split=10, n_estimators=150; total time=   0.3s
[CV] END max_depth=50, min_samples_leaf=4, min_samples_split=10, n_estimators=150; total time=   0.3s
[CV] END max_depth=50, min_samples_leaf=4, min_samples_split=10, n_estimators=200; total time=   0.4s
[CV] END max_depth=50, min_samples_leaf=4, min_samples_split=10, n_estimators=200; total time=   0.4s
[CV] END max_depth=50, min_samples_leaf=4, min_samples_split=10, n_estimators=200; total time=   0.3s
[CV] END max_depth=50, min_samples_leaf=4, min_samples_split=10, n_estimators=200; total time=   0.2s
[CV] END max_depth=50, min_samples_leaf=4, min_samples_split=10, n_estimators=200; total time=   0.2s
Best parameters found by GridSearchCV: {'max_depth': None, 'min_samples_leaf': 1, 'min_samples_split': 2, 'n_estimators': 50}
```

Figure 12: Best parameter after grid search.

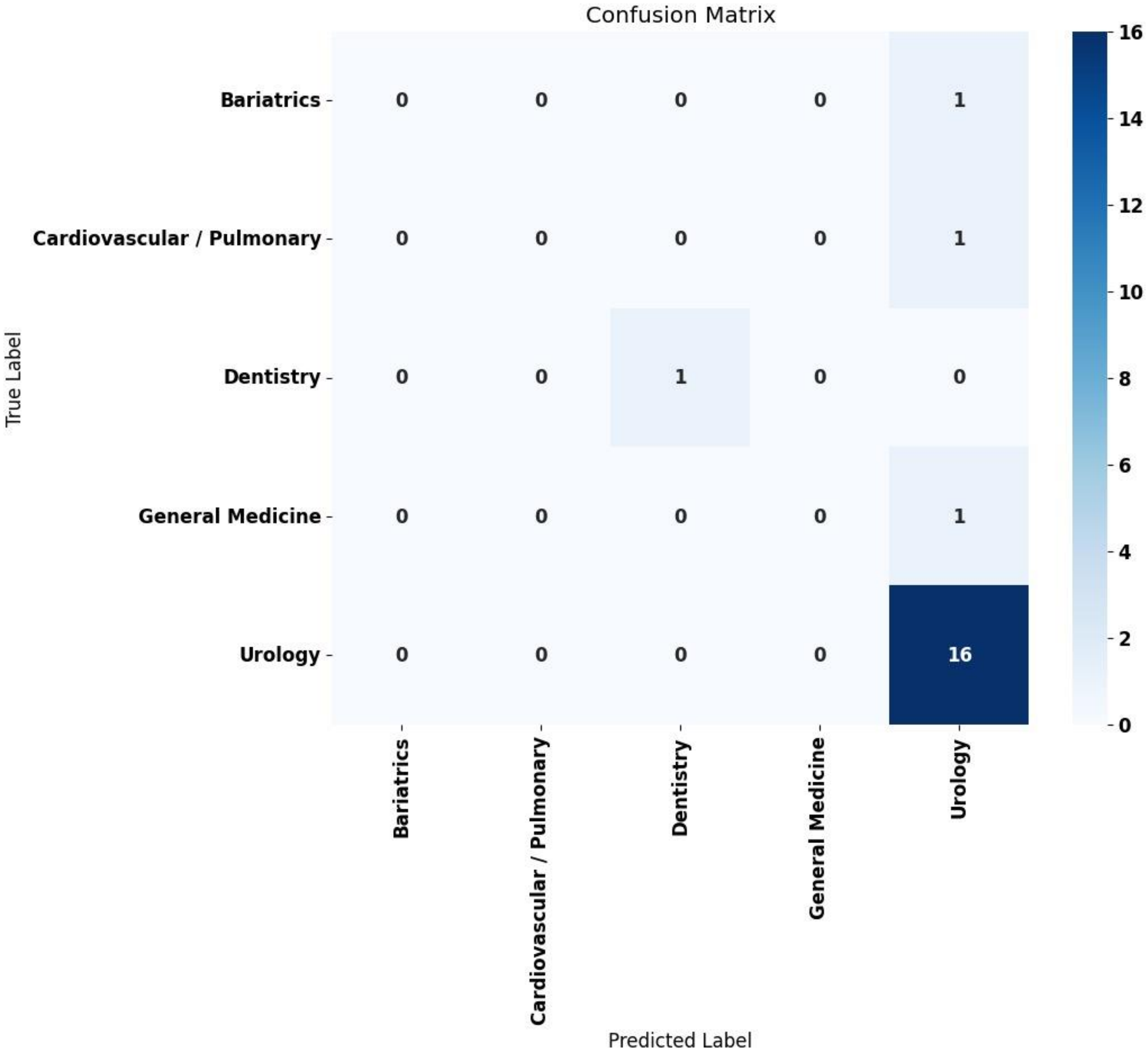


Figure 13: confusion matrix for optimal model obtained on Grid Search

7.2.3. PCA ANALYSIS

A popular dimensionality reduction method in data analysis and machine learning is principal component analysis, or PCA. Preserving the most significant information while converting high-dimensional data into a lower-dimensional space is its main objective.

The primary components of the data, which are linear combinations of the original features, are found by PCA. The first component, which captures the greatest volatility in the data, is followed by the second component, and so on, in order of their variance.

PCA was used in this study to minimize the dataset's dimensionality while preserving as much variation as possible. The goal was to increase computing efficiency and streamline the modelling procedure without compromising prediction accuracy.

The data can be shown in a reduced-dimensional space following PCA, providing insights into the underlying structure and connections between the data points. Furthermore, significant characteristics and patterns that might not be visible in the original high-dimensional space can be found with the aid of PCA.

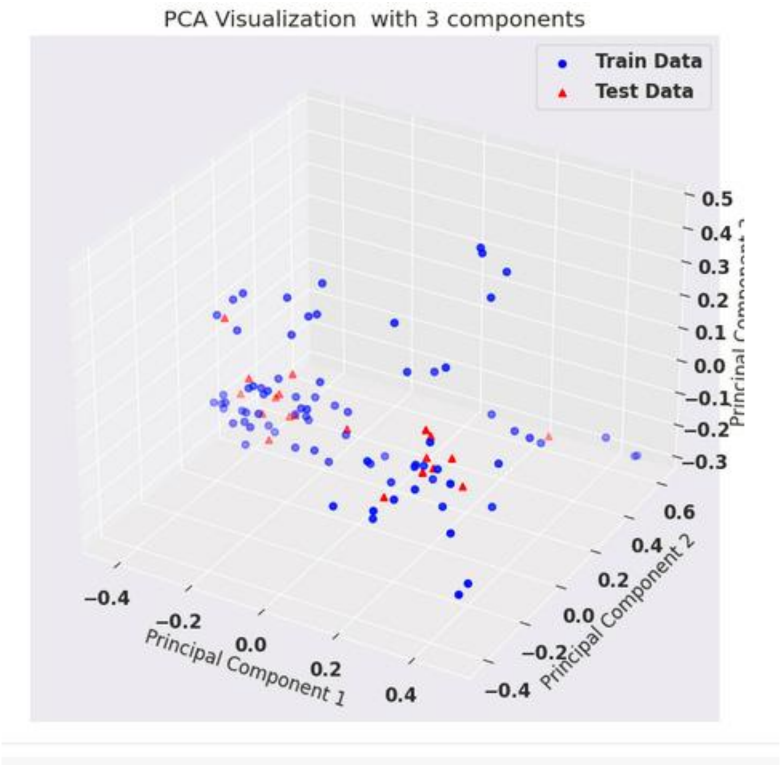


Figure 11: PCA visualization with 3 components after Dimensionality Reduction

8. DISCUSION

8.1. LIMITATIONS

As the transcription column in the dataset has the highest number of noises it is very much difficult to classify and predict the medicine for particular patients. For example, if a prescription has words like Neurology and nephrology and has both the words in the prescription so it is hard to identify whether the person is suffering from a disease that is related to Neurology or Nephrology.

8.2. APPROACH

Making a superset of medical specialties is one way to deal with the transcribing column's noise limit. We can increase classification accuracy and make the work of classification simpler by grouping similar specialties into larger categories.

We may, for instance, construct a superset called "Neuro-Nephrology" that consists of both Nephrology and Neurology. Prescriptions falling within the "Neuro-Nephrology" superset are those that contain terminology associated with either neurology or nephrology.

By using this method, we can identify the similarities and overlaps between different medical specializations, which facilitates the accurate prediction of medications and the classification of prescriptions.

12. CONCLUSION AND FUTURE WORK

12.1. SUMMARY

Our Medical Transcription project employed machine learning models (Logistic Regression, Random Forest, SVM, Naive Bayes) with Bag-of-Words (BoW) and TF-IDF representations. Logistic Regression performed best. Additionally, the model showed best results in metrics like accuracy, indicating its efficiency in text modeling.

12.2. CONCLUSION

The goal in this research was to accurately categorize text data into predetermined categories by tackling the problem of NLP classification. We thoroughly tested and fine-tuned several methods to improve model performance and obtain a better understanding of the dataset.

First, to avoid overfitting and enhance the generalization of the models, use of regularization approaches. The performance of models was improved by optimizing the hyperparameters using grid search and hyperparameter tuning, taking into account variables like the number of estimators, maximum depth, and minimum samples split.

In addition, we used Principal Component Analysis (PCA) in conjunction with dimensionality reduction to shrink the feature space while maintaining crucial data, improving training time and model efficiency. Furthermore, the application of TF-IDF vectorization improved classification accuracy by capturing the significance of words in our documents and providing useful feature representation.

Furthermore, by using Word2Vec embeddings, investigating the semantic connections among words, which improved our models' comprehension of textual input and might have improved performance.

We conducted a thorough examination and comparison of these strategies, and the results showed that these techniques significantly improved the accuracy of the models; the optimized models outperformed the default models.

In summary, the processes of experimentation and fine-tuning have not only resulted in increased classification accuracy but have also yielded important insights into the suitability of various approaches for NLP classification problems. These results can be used as a starting point for more study and practical application, leading to improvements in text classification and natural language processing.

12.3. FUTURE WORK

While the evaluated models provide valuable insights into Medical Diagnosis outcome prediction, there are several areas that offer potential for future improvement and refinement. These areas are crucial to enhance the accuracy and effectiveness of the models in real-world scenarios:

Advanced Text Preprocessing: To enhance the quality of the text data before modelling, try out more complex text preprocessing methods like lemmatization, stemming, or part-of-speech tagging.

Model Ensembles: To combine forecasts and maybe enhance overall performance, look at ensemble techniques like stacking or blending different models.

Deep Learning Models: Examine how to extract complicated patterns from text data using deep learning models including transformer-based models like BERT or GPT, recurrent neural networks (RNNs), and convolutional neural networks (CNNs).

Handling Class Imbalance: If there is class imbalance in the dataset, devise ways to remedy it. Some of these methods include under- or oversampling majority classes, utilizing SMOTE (Synthetic Minority Over-sampling Technique), or oversampling minority classes.

13. REFERENCE

AltexSoft. (n.d.). Medical datasets: 30+ datasets for medical imaging, genomics, healthcare & more. Retrieved April 18, 2024, from https://www.altexsoft.com/blog/medical-datasets/

Chandra, M. (n.d.). TF-IDF vectorizer (Scikit Learn). Medium. Retrieved April 10, 2024, from https://medium.com/@cmukesh8688/tf-idf-vectorizer-scikit-learn-dbc0244a911a

Pathmind. (n.d.). Word2Vec. Pathmind Wiki. Retrieved April 12, 2024, from https://wiki.pathmind.com/word2vec

Grid search for model tuning. Towards Data Science. Retrieved April 12, 2024, from https://towardsdatascience.com/grid-search-for-model-tuning-3319b259367e

A step-by-step explanation of principal component analysis. Builtin. Retrieved April 14, 2024, from https://builtin.com/data-science/step-step-explanation-principal-component-analysis

SMOTE and ADASYN: Handling imbalanced data set. Coinmonks on Medium. Retrieved April 14, 2024, from https://medium.com/coinmonks/smote-and-adasyn-handling-imbalanced-data-set-34f5223e167