Specialty Classification from Medical Transcription

Authored by: Mumuksha Pant | Pragya Prashar | Pratham Sachinbhai Shah

Abstract

In order to solve the issue of categorizing medical transcriptions into the appropriate medical specialties, this project explores the use of several Natural Language Processing (NLP) techniques. The study specifically looks at the application of five models: LSTM, Random Forests, RoBERTa, Support Vector Machines (SVM), and Logistic Regression. The project offers a thorough explanation of the methods used, beginning with preprocessing procedures to organize and clean the medical transcription data before training the chosen models to identify the relevant medical specialty for every transcription. The report also describes how the performance of various models was evaluated, using common metrics to guarantee a reliable and equitable comparison. In addition to outlining each model's advantages, disadvantages, and general task fit, the study compares the models' outputs using a number of criteria. The project's conclusion discusses the results and challenges faced for medical transcription classification, along with future scope of development.

Introduction

Accurate documentation of patient visits is made possible by the profession of medical transcribing, which is vital to healthcare. However, effectively classifying transcriptions into the appropriate specialty is difficult due to the wide variety of medical specialties and the intricacy of the language used in these fields.

Machine learning models have been used in previous text categorization efforts to categorize documents according to their content. In particular, methods ranging from basic statistical techniques to sophisticated deep learning approaches have been used in the field of medical text. These techniques, which are frequently impacted by elements like feature engineering, computational resources, and the caliber of labeled data, have shown differing degrees of effectiveness. Notwithstanding these developments, problems still exist, including managing unbalanced datasets, domain-specific language, and the requirement for interpretable models.

In this study, we use a variety of machine learning and deep learning techniques to tackle the problem of categorizing medical transcriptions into their corresponding specialties. We concentrate on well-known yet potent models like Random Forests, Logistic Regression, Support Vector Machine (SVM), RoBERTa, and LSTM. Using these models, we investigate how to strike a balance between interpretability, performance, and model simplicity. Preprocessing the medical transcriptions to extract relevant features, training the models on labeled datasets, and methodically assessing the models' performance using accepted metrics are all part of the suggested methodology.

Our findings emphasize the advantages and disadvantages of various approaches and offer a comparative evaluation of their efficacy. By providing insights into the possible uses and constraints of both deep learning and classical machine learning techniques in practical settings, this study advances our understanding of how these approaches might be applied to the classification of medical transcriptions.

Team Member	**Contribution**
Mumuksha Pant	- Conducted Exploratory Data Analysis (EDA) for deciding final categories - Applied Word2Vec vectorization using gensim - Trained Logistic Regression with different vectorizers and hyperparameters - Trained and Fine-tuned LSTM model
Pragya Prashar	- Worked on EDA and developed a data cleaning pipeline with test cases - Applied TF-IDF vectorization - Trained Random Forest with different vectorizers and hyperparameters - Trained LSTM model, experimented with different hyperparameters - Fine-tuned the LSTM model using

	various techniques to enhance its accuracy and generalization
Pratham Sachinbhai Shah	- Performed feature engineering by combining and dropping categories aided by Named Entity Recognition (NER) and POS (Parts of Speech) - Created a comprehensive data-cleaning function and its test cases - Medical entity filtering using scispacy - Wrote code for utilizing ClinicalBERT vectorization - Trained 3 SVM models with 3 different embeddings. - Visualized coefficients generated by SVM and Random Forest for understanding the models better - Performed vectorization using RoBERTa embeddings and fine-tuned RoBERTa for classification task - Trained LSTM model using ClinicalBERT embeddings
Team Work	- Model comparison, evaluated model performance using appropriate metrics - Cleaned and optimized code - Worked on Unit tests, Demo, PPT, and Report

Table 1: Highlighting contributions per member

Methods

This study's approach is divided into a number of important phases, including evaluation, comparative analysis, model training, and data preprocessing. Below is a breakdown of each step:

1. Data Collection and Preprocessing

Medical transcriptions are each annotated with the corresponding medical specialty, make up the dataset used in this study. The following preparation actions were conducted in order to get the data ready for analysis:

- **Text Cleaning:** To improve the quality of the data, noise was removed by removing extraneous elements using a thorough cleaning function that included multiple steps. These included removing URLs, brackets, special characters, hyphens, HTML entities, HTML tags, punctuation, and extraneous whitespace, leaving only meaningful textual content. To make sure that words that add no meaning to a phrase are not considered, stopwords were eliminated. Numbers were also removed from the data.
- **Word Tokenization:** To make text analysis and numerical representation easier, the transcriptions were divided into discrete words known as tokens.
- **Lemmatization:** To cut down on repetition and enhance text coherence, words were reduced to their base or root form.
- **Managing Imbalanced Data:** The data is highly imbalanced with 40 target variables. Since it is medical data it would not be appropriate to use oversample and hence we undersampled to 6 categories to address the issue of imbalanced data and also avoid target variables with very less data which hinders the model performance.
- **Feature Engineering:** To further simplify the task, using some domain knowledge, we combined 4 categories into. We combined "Neurosurgery" and "Neurology" into “Neurology”, and combined "Consult - History and Physical" and "General Medicine" into “General Medicine”
- **Vectorization:**. Three vectorizers were employed in this project: Word2Vec, ClinicalBERT, and TfidfVectorizer, which is based on the notion that uncommon phrases in a document frequently contain more meaning.
  - **TF-IDF Vectorization:** Term Frequency-Inverse Document Frequency (TF-IDF) vectors were created from the text data. This approach downweights common words across all classes while emphasizing keywords that are important in specific transcriptions.
  - **Word2Vec:** To capture the semantic links between words, pre-trained Word2Vec embeddings were used. The average vector of the word embeddings that made up each transcription served as its representation. Because skip-gram predicts context words given a target word, we employ it in Word2Vec. It is more appropriate for catching significant unusual word embeddings. Many of the

technical terminologies used in medical transcribing are uncommon, such as "hemangioblastoma" and "pneumothorax." Skip-Gram is an effective way for learning these uncommon words since it determines their meaning by looking at the words that surround them, or their context. The model was fine tuned to suit requirements for the project.

- **ClinicalBERT:** This BERT model was created especially for clinical text processing. A sizable corpus of clinical notes from the MIMIC-III dataset is used to pre-train the model. By utilizing its domain-specific expertise and superior handling of medical language peculiarities, it offers a more contextual method of vectorizing medical transcription data. The model was adjusted to meet the project's specifications.

- **Word Frequency Analysis:** To confirm and comprehend the vector representations, domain-specific keywords were identified by analyzing the frequency of phrases within each specialization.
- **Medical entity tagging:** One more approach that we tried was filtering the text for just medical entities using the scispacy library. However, this approach did not improve results for any of the models, hence, we decided to drop this pre-processing step. This approach can be helpful for other medical applications in the future.

By combining statistical and semantic variables for increased classification accuracy, these preprocessing procedures made sure the models received high-quality input data.

3. Model Selection and Training

For this analysis, five machine learning models were implemented and evaluated, commonly used for text classification. Below are the models and their hyperparameters used for training:

- **Logistic Regression** - It is a linear model which works well for multiclass classification and it is chosen for its simplicity.
  - **Hyperparameters**:
    - Random State: 42
    - Maximum Iterations (max_iter): 1000
    - Solver: 'liblinear'
- **Support Vector Machine (SVM) -** It is chosen because it is effective for multi class classification and suitable for high dimensional spaces.
  - **Hyperparameters**:
    - Kernel: 'sigmoid'
    - C: 1
- **Random Forest -** It is an ensemble method which is known for its accuracy and robustness.
  - **Hyperparameters**:
    - Number of Estimators (n_estimators): 1000
    - Random State: 42
- **Long Short-Term Memory (LSTM) -** It is capable of capturing long-term dependencies in sequential data using memory cells and it is useful for tasks such as multi class classification.
  - **Hyperparameters**:
    - Embedding Dimension: From ClinicalBERT embeddings
    - LSTM Hidden Units: 256
    - Dropout Rate: 0.5
    - Dense Layer Activation: Softmax
    - Optimizer: Adam
      - Learning Rate: 1×10−41 \times 10^{-4}1×10−4
    - Loss Function: Categorical Crossentropy
    - Metrics: Accuracy
- **RoBERTa -** It is a pre-trained transformer-based model optimized for robust natural language understanding tasks.
  - **Hyperparameters**: Fine-tuned with modified parameters (details such as learning rate, batch size, and epochs should be specified if available).

Each model was trained using the cleaned and pre-processed dataset and the hyperparameters were tuned to achieve good performance and results.

4. Model Evaluation

Standard classification metrics were used to evaluate each model's performance:

Accuracy is defined as the proportion of correctly classified instances (transcriptions) to all classified occurrences. In Healthcare, accuracy is a critical metric because it ensures the medical transcriptions are correctly interpreted and classified.

Confusion matrix is a table used to compare a model's actual and predicted outputs and evaluate the performance of a classification model.

Logistic Regression

Overall, the Logistic Regression model with TF-IDF fared the best, according to the confusion matrices. Only 22 cases were incorrectly classified as Neurology and 8 as Cardiovascular/Pulmonary for the Surgery class, out of 234 total genuine labels. In a similar vein, just two cases in the General Medicine class were incorrectly classified as Cardiovascular/Pulmonary, out of 131 cases. In comparison, 161 cases of surgery were accurately identified by the Logistic Regression model with BERT, while 35 cases were incorrectly labeled as orthopedic and 25 as cardiovascular/pulmonary. BERT accurately identified 116 cases in general medicine, but incorrectly identified 6 as neurology and 6 as cardiovascular/pulmonary. Out of 228 occurrences, the Word2Vec model correctly classified 184 instances of surgery, 30 of which were misclassified as orthopedic, and 14 of which were misclassified as cardiovascular/pulmonary. In contrast, it correctly classified 124 instances of general medicine, 6 of which were misclassified as cardiovascular/pulmonary. Neurology and Cardiovascular/Pulmonary were the most incorrectly categorized characteristics in all models. For instance, 27 cases in Neurology and 28 cases in Cardiovascular/Pulmonary were incorrectly classified as Surgery using TF-IDF. Smaller classes like Cardiovascular/Pulmonary and Neurology were commonly misclassified, but overall, the TF-IDF model performed the best in classification, especially for larger classes like Surgery and General Medicine.

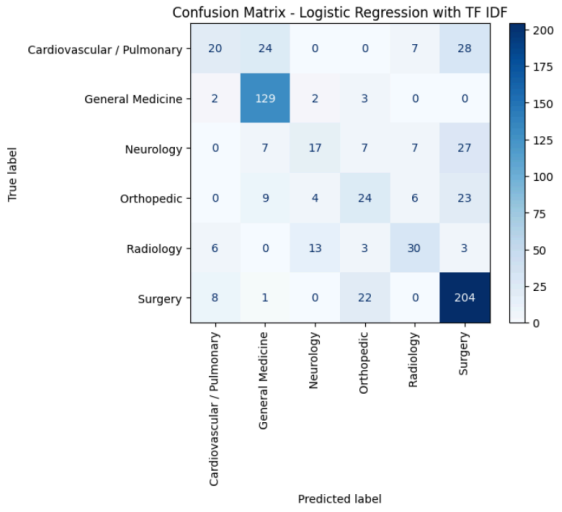


Figure 1: Confusion Matrix - Logistic Regression with TF-IDF

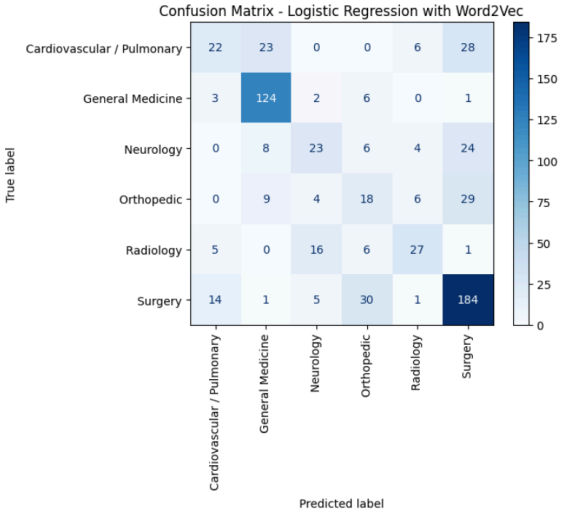


Figure 3: Confusion Matrix - Logistic Regression with Word2Vec

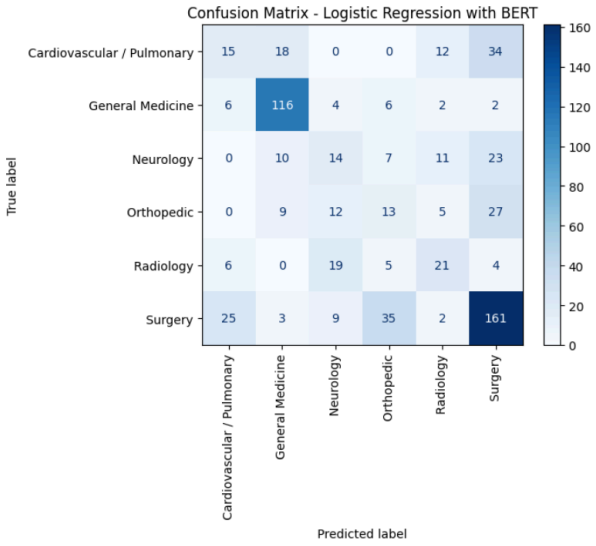


Figure 2: Confusion Matrix - Logistic Regression with BERT

Random Forest

The Random Forest model with TF-IDF performs well based on the confusion matrices, especially when it comes to predicting the high classification rates of general medicine (114 correct predictions) and surgery (143 correct predictions). The BERT and Word2Vec models show similar patterns, with 113 right predictions in general medicine and 142 and 143 correct predictions in surgery, respectively. All models do, however, show notable misclassifications in several areas. Examples of instances that are commonly misclassified as surgery include cardiovascular/pulmonary cases (37 for TF-IDF, 39 for BERT, and 37 for Word2Vec). Likewise, neurology instances are frequently mislabeled as orthopaedic and radiology, or as surgery (27, 29, and 29 for TF-IDF, BERT, and Word2Vec, respectively). Furthermore, orthopedic instances—36 for TF-IDF, 35 for both BERT and Word2Vec—are frequently mistaken for surgical cases. More concentrated diagonal values and comparatively fewer misclassifications show that the TF-IDF model has superior classification accuracy overall.

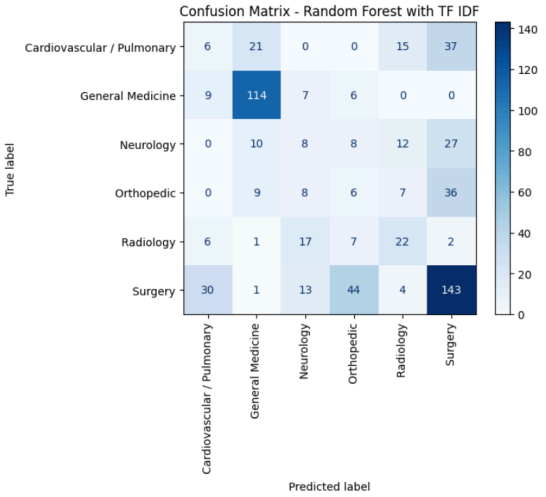


Figure 1: Confusion Matrix - Random Forest with TF-IDF

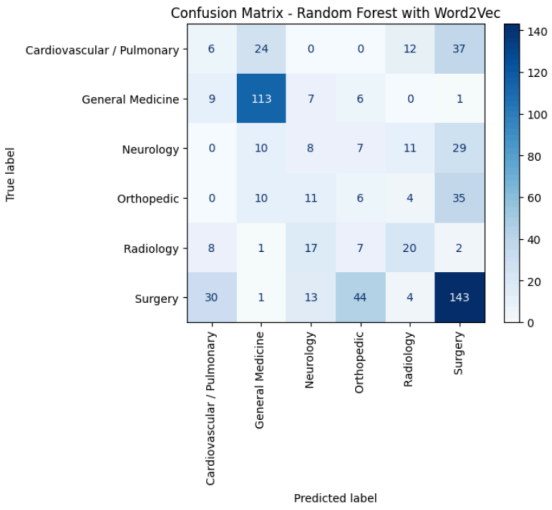


Figure 2: Confusion Matrix - Random Forest with Word2Vec

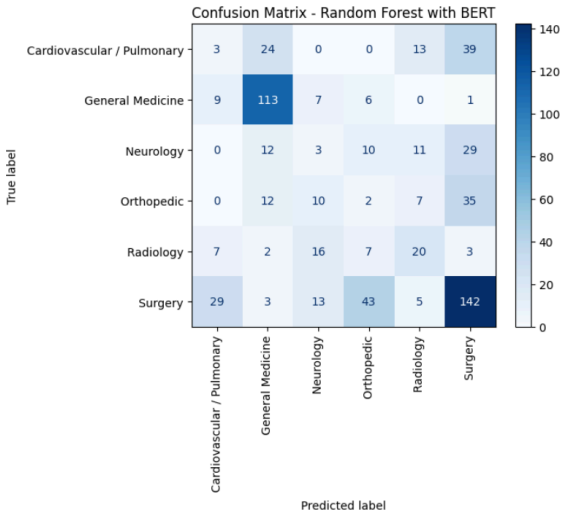


Figure 3: Confusion Matrix - Random Forest with BERT

Support Vector Machine

SVM models were best performing models out of all with the highest accuracy reaching 0.69 for SVM trained using TF-IDF vectors. This was expected as SVM maximizes distance of support vectors instead of minimizing error which is useful for handling imbalanced dataset such as outs.Different kernels were tried for SVM and 'sigmoid' kernels with regularization parameter as 1. SVM models were different with TF-IDF, Word2Vec, and ClinicalBERT embeddings. The model trained with ClinicalBERT excelled in classifying but struggling with Cardiovascular, Orthopedic, and Neurology. It misclassified Cardiovascular/ Pulmonary as Surgery 42 times. SVM with Word2Vec also struggled with these classes but not as much as the one trained with ClinicalBERT. This model was misclassifying Cardiovascular / Pulmonary as General Medicine more than other models. Finally, SVM trained with TF-IDF did perform as well for the 'Surgery' class as it had only 188 correct predictions as compared to 213 and 228.However, it performed the best on minority classes and that's how it got the best accuracy. Cardiovascular/ Pulmonary, Orthopedic , and Neurology are better handled with this model. This shows that for this task statistical embedding such as TF-IDF works the best, as Word2Vec was not able to capture context in an appropriate manner and ClinicalBERT must have struggled since it was trained on organized text and our text was quite unorganized.

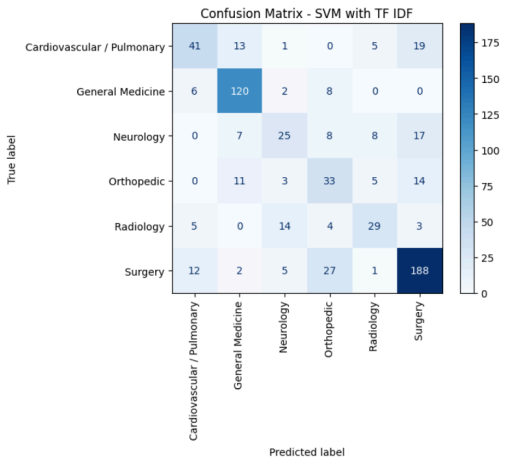


Figure 1: Confusion Matrix - SVM with TF-IDF

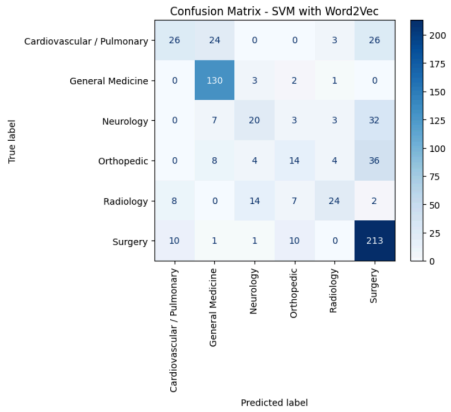


Figure 2: Confusion Matrix - SVM with Word2Vec

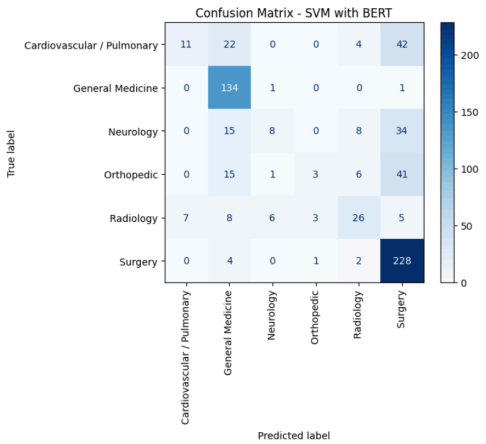
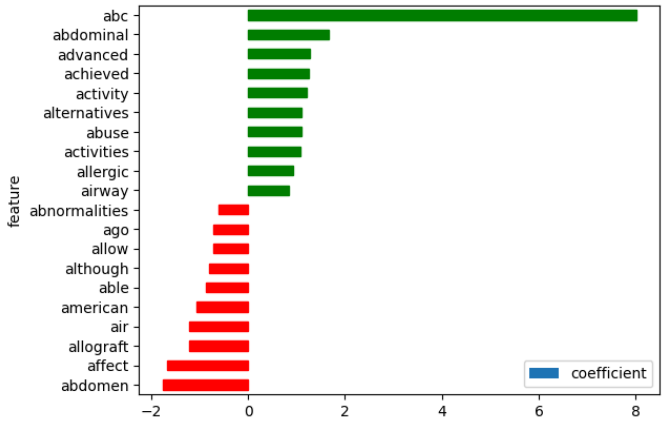


Figure 3: Confusion Matrix - SVM with BERT

Visualization of coefficients in SVM



This chart represents the feature importance for classes 'Cardiovascular / Pulmonary' (green) vs 'General Medicine' (red). Similarly this can be deduced for all combination of classes.This helped in understanding which feature contributes towards which class in the classification

RoBERTa

RoBERTa (Robustly Optimized BERT Approach) is a transformer-based language model that improves upon BERT by training with more data, larger batches, and dynamic masking. The confusion matrix draws attention to a number of important findings about the effectiveness of the RoBERTa categorization model. For the Surgery category, the model shows a high classification rate with 221 accurate predictions and few misclassifications. Similar to this, it does well in general medicine, making 105 accurate predictions, albeit with some misunderstandings about other categories. Misclassifications are noteworthy in some cases, though. Cardiovascular/pulmonary patients, for instance, are frequently mislabeled as General Medicine (16) and Surgery (37) respectively. Additionally, 25 cases of neurology being incorrectly labeled reveal confusion with surgery. 38 cases are wrongly classified as surgery in the orthopedic category, and 8 as radiology. Radiology is frequently confused with neurology (16) and surgery (31). These patterns show that although the model works well for some categories, there is a lot of overlap and misunderstanding in others, especially when it comes to related medical professions. Reducing misclassifications and improving overall model performance may be achieved by efforts to improve class-specific representations or modify feature extraction.

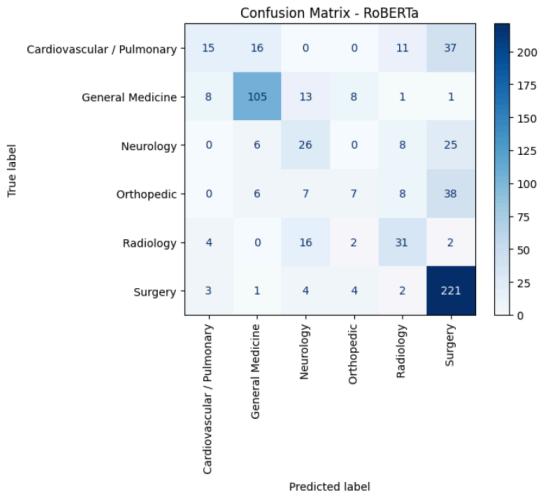


Figure 1: Confusion Matrix - Roberta

LSTM

Long Short-Term Memory (LSTM) is a type of recurrent neural network (RNN) designed to learn long-term dependencies by addressing the vanishing gradient problem. The LSTM model's confusion matrix provides important information about how well it performs in classification. With 210 accurate predictions and comparatively few misclassifications, the model excels in the Surgery area. It also performs well in general medicine, accurately classifying 128 cases and having little misunderstanding with other categories. There are significant misclassifications in a number of domains. Frequently, Cardiovascular/Pulmonary is mistakenly classed under General Medicine (19) and Surgery (27) for example. There is misunderstanding in the Neurology category; 11 cases were incorrectly classed as General Medicine and 28 as Surgery. Furthermore, Radiology is incorrectly classed as Surgery (7) and Neurology (13), while Orthopedic shares substantial overlap with both Surgery (32) and Radiology (6). These results imply that although the LSTM model performs well in some domains, there are significant issues with overlapping characteristics across related classes, especially in the fields of neurology, orthopedics, and cardiovascular/pulmonary. The model's performance could be further enhanced by better feature representation and better differentiation between closely related categories.

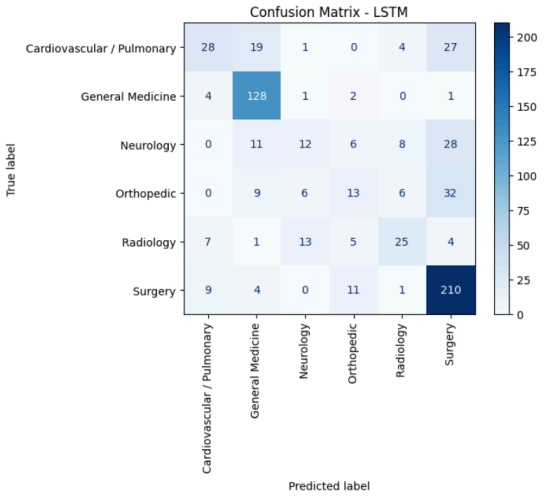


Figure 1: Confusion Matrix - LSTM

Comparative Analysis

The results of all models were compared to identify the best-performing approach.

Across the various models and feature extraction techniques tested, Support Vector Machine with TF-IDF emerged as the most effective overall with an accuracy of 0.69, particularly for larger and more easily distinguishable classes like Surgery and General Medicine.

TF-IDF vectorization provides the best overall performance for classification. It is simple, efficient, and handles structured medical text data well.

All models consistently faced challenges (misclassifications) with smaller classes, such as Cardiovascular/Pulmonary and Neurology.

In contrast to TF-IDF and Word2Vec, which surprisingly, generated accuracy values that were the same for all models, we did not notice any appreciable performance gain even when we used ClinicalBERT which was a domain-specific vectorizer.

Implementation Details

The models were implemented using Python and standard libraries, including scikit-learn for machine learning algorithms, NLTK for text preprocessing, gensim for vectorization and PyTorch for transformer based models . This step-by-step methodology ensures a systematic approach to understanding and addressing the problem of classifying medical transcriptions into their respective specialties.

Data

Medical transcription annotated with the corresponding medical specialty, make up the dataset used in this investigation. A diverse and thorough dataset for classification purposes was ensured by curating the data to cover a range of medical areas.

Dataset Composition

- **The dataset contains 4,998 data points with the following columns -**

  description, medical_speciality, sample_name, transcription, keywords

	description	medical_specialty	sample_name	transcription	keywords
0	A 23-year-old white female presents with comp...	Allergy / Immunology	Allergic Rhinitis	SUBJECTIVE:, This 23-year-old white female pr...	allergy / immunology, allergic rhinitis, aller...
1	Consult for laparoscopic gastric bypass.	Bariatrics	Laparoscopic Gastric Bypass Consult - 2	PAST MEDICAL HISTORY:, He has difficulty climb...	bariatrics, laparoscopic gastric bypass, weigh...
2	Consult for laparoscopic gastric bypass.	Bariatrics	Laparoscopic Gastric Bypass Consult - 1	HISTORY OF PRESENT ILLNESS:, I have seen ABC ...	bariatrics, laparoscopic gastric bypass, heart...
3	2-D M-Mode. Doppler.	Cardiovascular / Pulmonary	2-D Echocardiogram - 1	2-D M-MODE: , ,1. Left atrial enlargement wit...	cardiovascular / pulmonary, 2-d m-mode, dopple...
4	2-D Echocardiogram	Cardiovascular / Pulmonary	2-D Echocardiogram - 2	1. The left ventricular cavity size and wall ...	cardiovascular / pulmonary, 2-d, doppler, echo...

Figure 1: Medical Speciality Dataset

- **Number of Transcriptions** - The dataset includes [total number] transcriptions.
- **Medical Specialties** - 40 fields, including radiology, dermatology, cardiology, and others, are represented in the transcriptions. The vocabulary and background unique to its field are reflected in each transcription.
- **Text Characteristics** - Common medical practice acronyms, specialty-specific jargon, and basic medical terminology are all included in the vocabulary.

We created a graph showing the number of transcriptions by specialty. It made it easier to comprehend how data was distributed among various medical specializations. In order to guarantee that the classifier concentrates on specialties with enough data to efficiently learn patterns, we eliminated the underrepresented specialty based on this.

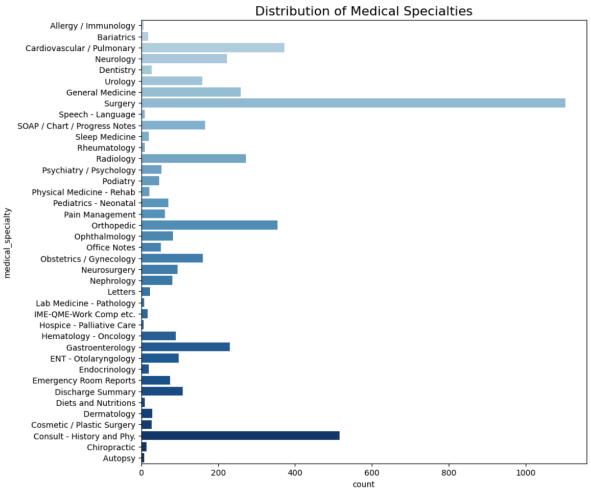


Figure 1: Medical specialty vs count

Class Distribution

The dataset shows a class imbalance, with specialized fields like autopsy, hospice-palliative care, lab medicine-pathology, and allergy/immunology having fewer transcriptions than other disciplines like surgery. In order to successfully handle this imbalance, which presents a barrier for model training, techniques like oversampling, undersampling, or weighted loss functions are required.

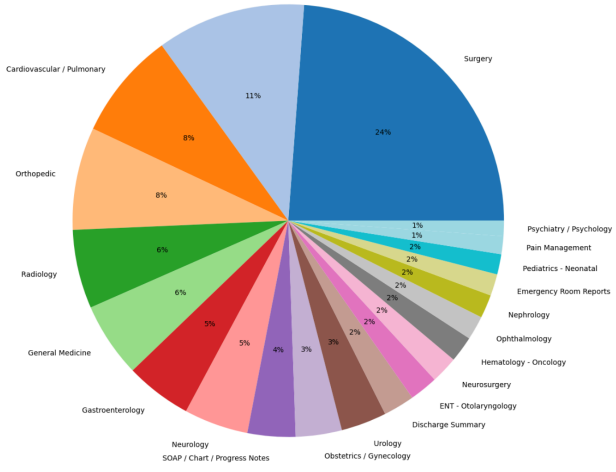
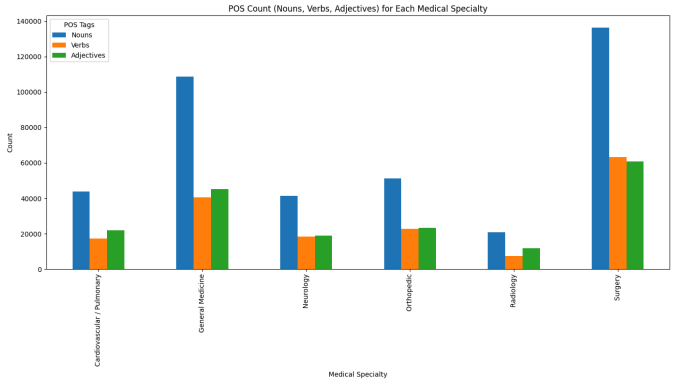


Figure 1: Percentage transcriptions for the top 21 categories

This graph (pie chart - Figure 1) shows the percentage of transcriptions for the top 21 categories.

POS Count



This chart shows the counts of nouns, verbs, and adjectives across various medical specialties. Surgery has the highest counts for all three POS categories, with nouns being the most frequent. General Medicine follows closely, also exhibiting a dominant noun count compared to verbs and adjectives. Cardiovascular/Pulmonary and Orthopedic specialties display similar trends, with nouns leading but overall counts being lower than those of General Medicine and Surgery. Neurology and Radiology have the lowest counts overall, yet maintain the same noun dominance. In summary, nouns consistently appear most frequently across all specialties, followed by verbs and adjectives.

Source of Data

Kaggle was the source of the dataset. To preserve patient privacy and guarantee adherence to data privacy laws like HIPAA, the transcriptions were anonymised.

Data Quality

- To get rid of duplicate or unnecessary entries, the dataset was thoroughly cleaned.
- Transcripts with inadequate material were not included in the study, and missing values were handled correctly.

This dataset offers both opportunities and difficulties for enhancing model performance as it provides a strong basis for training machine learning models to categorize medical transcriptions into their specialized fields.

Results

The classification findings show notable differences in performance across different models and embedding methods. With an accuracy of 69%, a weighted average F1-score of 68%, and a macro average F1-score of 61%, SVM with TF-IDF outperformed the other models in terms of overall performance, demonstrating its efficacy in handling textual features represented by TF-IDF. With weighted average F1-scores of 64% and 61%, respectively, and accuracies of 67% and 63%, respectively, Logistic Regression with TF-IDF and Logistic Regression with Word2Vec also demonstrated the usefulness of these less complex embedding techniques.

On the other hand, Random Forest demonstrated relatively lower performance overall, obtaining a lower macro F1-score of about 35% and only 47% accuracy with both TF-IDF and Word2Vec, indicating that it has trouble with these datasets. Models employing BERT embeddings, including Random Forest, SVM, and Logistic Regression, showed varying degrees of effectiveness; Random Forest had the lowest accuracy at 44%, while SVM had the greatest at 64%. With an accuracy of 64%, a weighted F1-score of 60%, and a macro F1-score of 49%, the Roberta model with BERT embeddings demonstrated good performance, successfully striking a balance between precision and recall. Finally, the LSTM with BERT exhibited solid performance with an accuracy of 65%, a macro F1-score of 51%, and a weighted F1-score of 62%, showing its potential in sequence modeling. Overall, Random Forest performed low regardless of the embedding utilized, whereas SVM models consistently outperformed others across various embeddings.

Model	Dataset	Accuracy	Macro Avg Precision	Macro Avg Recall	Macro Avg F1-Score	Wt.Avg Precision	Wt. Avg Recall	Wt, Avg F1-Score
Logistic Regression	TF-IDF	0.67	0.58	0.54	0.54	0.64	0.67	0.64
Logistic Regression	BERT	0.53	0.42	0.42	0.42	0.51	0.53	0.52
Logistic Regression	Word2Vec	0.63	0.55	0.52	0.52	0.61	0.63	0.61
SVM	TF-IDF	0.69	0.62	0.6	0.61	0.68	0.69	0.68
SVM	BERT	0.64	0.57	0.46	0.43	0.61	0.64	0.56
SVM	Word2Vec	0.67	0.6	0.52	0.54	0.64	0.67	0.64
Random Forest	TF-IDF	0.47	0.34	0.36	0.35	0.44	0.47	0.45
Random Forest	BERT	0.44	0.29	0.32	0.3	0.4	0.44	0.42
Random Forest	Word2Vec	0.47	0.34	0.35	0.34	0.44	0.47	0.45
RoBERTa	RoBERTa	0.64	0.53	0.5	0.49	0.6	0.64	0.6
LSTM	ClinicalBERT	0.65	0.55	0.5	0.51	0.61	0.65	0.62

Table: Comparative Analysis of Multiple Models

Discussion

Our results demonstrate the intrinsic overlap in medical language by showing that all models had trouble differentiating between categories like orthopedic and surgery, cardiovascular and surgery, etc. Generally, for all our models, TF-IDF and Word2Vec embeddings gave the best results. We did not notice any appreciable performance gain even when we used ClinicalBERT, a vectorizer with domain-specific expertise. The intricacy of medical terminology and the lack of availability of medical transcriptions as a result of HIPAA regulations resulted in a lack of availability. This combined with the fact that medical terms have a lot of overlap across classes made it difficult to model this task. With many combinations we were able to push the accuracy to 0.69 for classifying 6 different classes. This can be improved in the future if more data is available and using further domain specific models.

References

[1] Medical Transcriptions Dataset. Retrieved from https://www.kaggle.com/datasets/tboyle10/medicaltranscriptions

[2] Oduse Samuel, Temesgen Zewotir and Delia North, “Application of machine learning methods for predicting under-five mortality: analysis of Nigerian demographic health survey 2018 dataset”.

[3] The 7 best arXiv papers to learn how LLMs work article published in DeepGram.

[4] Lee Kah Win, Gan Keng Hoon, “Text Classification of Medical Transcriptions using N-Gram Machine Learning Approach”.

[5] M. A. Clinciu and H. F. Hastie, "A Survey of Explainable AI Terminology," *Edinburgh Centre for Robotics.*

[6] Yinhan Liu, Myle Ott, Naman Goyal, et al., “RoBERTa: A Robustly Optimized BERT Pretraining Approach”

[7] Abdulrezzak Zekiye and Adil Alpkocak, "Classification of Medical Transcriptions with Explanations”

[8] M. T. Ribeiro, S. Singh and C. Guestrin, "Why Should I Trust You?": Explaining the Predictions of Any Classifier," 2016.

[9] Jurgen Schmidhuber and Sepp Hochreiter, “Long Short-Term Memory”, 1991.

[10] Kexin Huang, Jaan Altosaar and Rajesh Ranganath, "ClinicalBERT: Modeling Clinical Notes and Predicting Hospital Readmission”, 2020.

Appendix

This section consists of user manual to run the code successfully.

102main code.ipynb

102maincode.ipynb contains code for replicating all our results. It is written in a way that you only require to run all the cells. It’ll automatically fetch the dataset from Kaggle, performing all necessary steps before modeling, train the models and also generate classification reports.

demo.py

demo.py contains the code for the demo. First, install all dependencies using requirements.txt and then run the command “streamlit run demo.py”. This will open up a Streamlit dashboard. You can choose any of our 11 models based on the combination of vectorization technique and the model. Input is given in the text field and press the ‘predict’ button to get the prediction.

Commands for running demo.py:

1. python -m venv myenv
2. myenv/Scripts/activate OR source myenv/Scripts/activate
3. python -m pip install --upgrade pip
4. pip install -r requirements.txt
5. streamlit run demo.py

unittests.py

unittests.py contains unit test for our cleaning function.

Command for running unittests.py

python unittests.py