CPT Code Classification and Semantic Search using NLP

Course Code: CIE 553 – NLP

Submitted to: Dr. Samar Elbedwehy
and Eng. Abdullah sabry
Submitted by:
1. Amar Sherief
2. Khaled Ashraf 202100751
3. Mohamed Mahmoud 202100071
4. Mohamed Elsyaed 202100438
5. Ahmed Amin 202101027

Department of Communications and Information Engineering
University of Science and Technology, Zewail City

May 2025

Contents

1 Introduction

Current Procedural Terminology (CPT) codes are a standardized set of medical codes used to describe medical, surgical, and diagnostic procedures and services. Developed and maintained by the American Medical Association (AMA), these codes are essential in the healthcare system for ensuring accurate communication between healthcare providers, insurance companies, and government agencies. CPT codes help in documenting the services provided to patients and play a critical role in billing and reimbursement processes.

Usage of CPT Codes

CPT codes are primarily used by:

- **Healthcare providers** to report procedures and services performed on patients.
- **Insurance companies** to process claims and determine reimbursement amounts.
- **Medical billing professionals** to ensure accurate and timely payments.
- **Researchers and policymakers** for analyzing healthcare trends and service utilization.

There are three categories of CPT codes:

1. **Category I** – Codes for common procedures and services (e.g., 99213 for a routine office visit).
2. **Category II** – Supplemental codes for performance measurement (e.g., 0004F for tobacco use screening).
3. **Category III** – Temporary codes for emerging technologies and experimental procedures (e.g., 0469T for radiofrequency spectroscopy).

Examples of CPT Codes

- **99213** – Office or other outpatient visit for the evaluation and management of an established patient.
- **93000** – Electrocardiogram (ECG) with interpretation and report.
- **12001** – Simple repair of superficial wounds, 2.5 cm or less.

Benefits of CPT Codes

- **Standardization**: They create a common language for medical services, reducing confusion and errors.
- **Efficiency**: Help streamline medical billing and insurance claims.
- **Transparency**: Allow patients and payers to understand what services were provided.
- **Data Collection**: Facilitate health data analysis, research, and policy development.
- **Legal Documentation**: Serve as official records of the services rendered in case of disputes or audits.

In summary, CPT codes are essential tools in modern healthcare. They improve communication, support accurate billing, and contribute to better health system management and patient care.

Final presentation: On-campus during tutorial time on Tuesday, 13 May 2025.

2 Data Description

This section provides a description of a subset of manually collected CPT codes. Each item includes a CPT code and its corresponding formal description. These codes cover a range of medical services including office visits, fine needle aspiration biopsies, and immunization administrations.

- **99202**: Office or other outpatient visit for the evaluation and management of a new patient, straightforward medical decision making, 15 minutes or more.
- **99203**: Office or other outpatient visit for a new patient, low level of medical decision making, typically 30 minutes.
- **99204**: Office or other outpatient visit for a new patient, moderate complexity, typically 45 minutes.
- **99205**: Office or other outpatient visit for a new patient, high complexity, typically 60 minutes.
- **99211**: Office or other outpatient visit for an established patient, minimal issues, typically 5 minutes.
- **99212**: Office or other outpatient visit for an established patient, straightforward decision making, typically 10 minutes.
- **99213**: Office or other outpatient visit for an established patient, low complexity, typically 15 minutes.
- **99214**: Office or other outpatient visit for an established patient, moderate complexity, typically 25 minutes.
- **99215**: Office or other outpatient visit for an established patient, high complexity, typically 40 minutes.
- **10004**: Fine needle aspiration biopsy without imaging guidance; each additional lesion.
- **10005**: Fine needle aspiration biopsy, including ultrasound guidance; first lesion.
- **10006**: Fine needle aspiration biopsy, including ultrasound guidance; each additional lesion.
- **10007**: Fine needle aspiration biopsy, including fluoroscopic guidance; first lesion.
- **10008**: Fine needle aspiration biopsy, including fluoroscopic guidance; each additional lesion.
- **10009**: Fine needle aspiration biopsy, including CT guidance; first lesion.
- **10010**: Fine needle aspiration biopsy, including CT guidance; each additional lesion.
- **10011**: Fine needle aspiration biopsy, including MR guidance; first lesion.
- **10012**: Fine needle aspiration biopsy, including MR guidance; each additional lesion.

- **10021**: Fine needle aspiration biopsy, without imaging guidance; first lesion.
- **90460**: Immunization administration through IM, oral, or intranasal route with counseling, live vaccine, for patients 18 years or younger.
- **90461**: Each additional vaccine/toxoid component administered (use with 90460).
- **90471**: Immunization administration (single vaccine) via IM or SC injection, without counseling.
- **90472**: Each additional vaccine administered (use with 90471).
- **90473**: Administration of live vaccine via oral or intranasal route (e.g., oral polio vaccine, flu mist).
- **90474**: Each additional oral or intranasal vaccine administered (use with 90473).
- **90480**: Administration of rotavirus vaccine, live, oral, 3 dose schedule.
- **90593**: Administration of recombinant vaccine for chikungunya virus by intramuscular injection.

3 Methodology

3.1 Collecting the Data Using LLMs and Expert Consultation

To simulate clinical documentation and generate CPT code-related training data, we utilized Large Language Models (LLMs) to create synthetic doctor notes. These generated notes were then reviewed and revised manually to enhance their realism and clinical relevance. This process allowed us to build an initial dataset for model development and experimentation.

In addition, we had the opportunity to meet with Dr. Laila Ramsy, a researcher conducting similar work in the United States. During our meeting, we consulted her on several key points related to fine-tuning language models for clinical coding tasks. We specifically asked the following questions:

- Do you have any recommendations on publicly available datasets (or data generation techniques) for fine-tuning BERT, or any other model, on clinical text-to-code tasks?
- In your experience, what preprocessing steps or input formats work best when adapting BERT or MED-BERT to CPT or ICD classification tasks?
- Are there any specific resources, papers, or benchmarks you recommend for evaluating such systems?

Dr. Ramsy shared valuable insights and suggested additional resources. She recommended several websites and platforms that provide access to real clinical data through formal institutional agreements. Specifically, she directed us to the MIMIC-IV database, which contains comprehensive de-identified clinical data:

`https://physionet.org/content/mimiciv/3.1/`

This resource is restricted-access and requires users to:

- Be a credentialed user.
- Complete the required CITI training titled *"Data or Specimens Only Research"*.
- Submit the training completion for verification.
- Sign the data use agreement for the project.

The meeting with Dr. Ramsy was incredibly insightful and encouraging. Her guidance opened the door for us to access real-world datasets and elevated the direction of our project toward practical, impactful applications in the healthcare domain.

3.2 Preprocessing

The preprocessing stage involved a series of text-cleaning steps to prepare clinical notes and formal descriptions for use in classification models. These steps are summarized below:

- **Lowercasing:** All text was converted to lowercase to ensure consistency across the dataset.
- **Removing special characters:** All characters except lowercase letters, digits, and whitespace were removed. This step eliminates punctuation marks and symbols that do not add value to the classification task.
- **Spacing digits and letters:** A space was added between digits and letters when they appeared together without separation (e.g., "100mg" becomes "100 mg") to improve tokenization accuracy.
- **Tokenization:** The text was split into individual words (tokens) for easier manipulation and analysis.
- **Stopword removal:** Common English stopwords (such as "the", "and", "is") were removed to focus on the most informative words.
- **Reconstruction:** The cleaned tokens were rejoined into full sentences and stored in new columns for use in model training.

These preprocessing steps reduced noise, improved consistency, and ensured that the data was suitable for machine learning tasks related to CPT code classification.

3.3 Embeddings and Models

In this section, we evaluate different embedding techniques and machine learning models for the classification of clinical texts into CPT codes. We explored multiple approaches, including FastText, Word2Vec, TF-IDF, and BERT, along with various machine learning models, such as Logistic Regression, Random Forest, XGBoost, Gradient Boosting, and Support Vector Machines (SVM).

3.3.1 FastText Embedding and Machine Learning Models

We initially utilized the FastText embedding technique, which works well for handling out-of-vocabulary words and is particularly useful for word-level and sentence-level embeddings. The performance of various classifiers with FastText embeddings is as follows:

- **Logistic Regression Results:** Accuracy: 0.08

- **Random Forest Results:** Accuracy: 0.54
- **XGBoost Results:** Accuracy: 0.55
- **SVM Results:** Accuracy: 0.10

3.3.2 Word2Vec Embedding and Machine Learning Models

For the Word2Vec embedding, we leveraged pre-trained word vectors to capture semantic meaning at a word level. The performance of classifiers with Word2Vec embeddings is as follows:

- **Logistic Regression Results (Word2Vec):** Accuracy: 0.11
- **Random Forest Results (Word2Vec):** Accuracy: 0.62
- **XGBoost Results (Word2Vec):** Accuracy: 0.62
- **Gradient Boosting Results (Word2Vec):** Accuracy: 0.60
- **SVM Results (Word2Vec):** Accuracy: 0.11

3.3.3 TF-IDF Embedding and Machine Learning Models

Using TF-IDF (Term Frequency-Inverse Document Frequency) as an embedding method, we transformed the text data into a sparse vector representation based on term importance. The performance results are as follows:

- **Logistic Regression Results (TF-IDF):** Accuracy: 0.89
- **Random Forest Results (TF-IDF):** Accuracy: 0.88
- **XGBoost Results (TF-IDF):** Accuracy: 0.81
- **Gradient Boosting Results (TF-IDF):** Accuracy: 0.82
- **SVM Results (TF-IDF):** Accuracy: 0.91

3.3.4 BERT Embedding and Machine Learning Models

BERT (Bidirectional Encoder Representations from Transformers) is a powerful contextual word embedding model, which has been fine-tuned for various downstream NLP tasks. The classification results using BERT embeddings are as follows:

- **Logistic Regression Results (BERT):** Accuracy: 0.89
- **Random Forest Results (BERT):** Accuracy: 0.87
- **XGBoost Results (BERT):** Accuracy: 0.87
- **Gradient Boosting Results (BERT):** Accuracy: 0.86
- **SVM Results (BERT):** Accuracy: 0.89

3.3.5 LSTM (Long Short-Term Memory) Results

LSTM, a type of Recurrent Neural Network (RNN), is known for its ability to capture long-term dependencies in sequential data. We trained an LSTM model and evaluated its performance in classifying clinical texts. The results are as follows:

- **Accuracy:** 0.86

The LSTM model shows strong performance, capturing long-term dependencies in the text and offering a competitive alternative to traditional machine learning models. However, it still does not outperform TF-IDF + SVM for this task, but it could be more beneficial when dealing with sequential or time-dependent text data.

3.3.6 Insights and Summary

Based on the evaluations above, the TF-IDF embedding technique combined with Support Vector Machines (SVM) provides the highest performance, achieving an accuracy of 0.91 and consistently high precision, recall, and F1-Score. BERT embeddings, although powerful, did not outperform TF-IDF for this particular task. Below are the key insights:

- TF-IDF with SVM emerges as the top-performing combination, suggesting that the traditional TF-IDF approach with sophisticated models like SVM can work very well for text classification tasks.
- BERT, although providing strong performance, did not outperform TF-IDF in this context, which could be due to the nature of the clinical data and the computational resources available for fine-tuning BERT.
- FastText and Word2Vec, while useful in specific scenarios, did not perform as well as TF-IDF or BERT in this case, highlighting the importance of selecting the right embedding method for the task.
- Machine learning models like Logistic Regression, Random Forest, XGBoost, and Gradient Boosting can provide solid performance when combined with appropriate embeddings, but their performance often lags behind SVM, especially in terms of precision and recall.
- In conclusion, TF-IDF with SVM offers the best combination of simplicity and performance for this text classification task, with BERT as a viable alternative depending on the availability of computational resources.

In conclusion, **TF-IDF with SVM** offers the best combination of simplicity and performance for this text classification task, with BERT as a viable alternative depending on the availability of computational resources.

3.4 Training and Evaluation

- Train-test split: 80/20 stratified
- Metrics: Accuracy, Precision, Recall, F1-Score
- Visuals: Confusion matrices

4 Confusion Matrices for Best Models

4.0.1 Confusion Matrix for TF-IDF with SVM

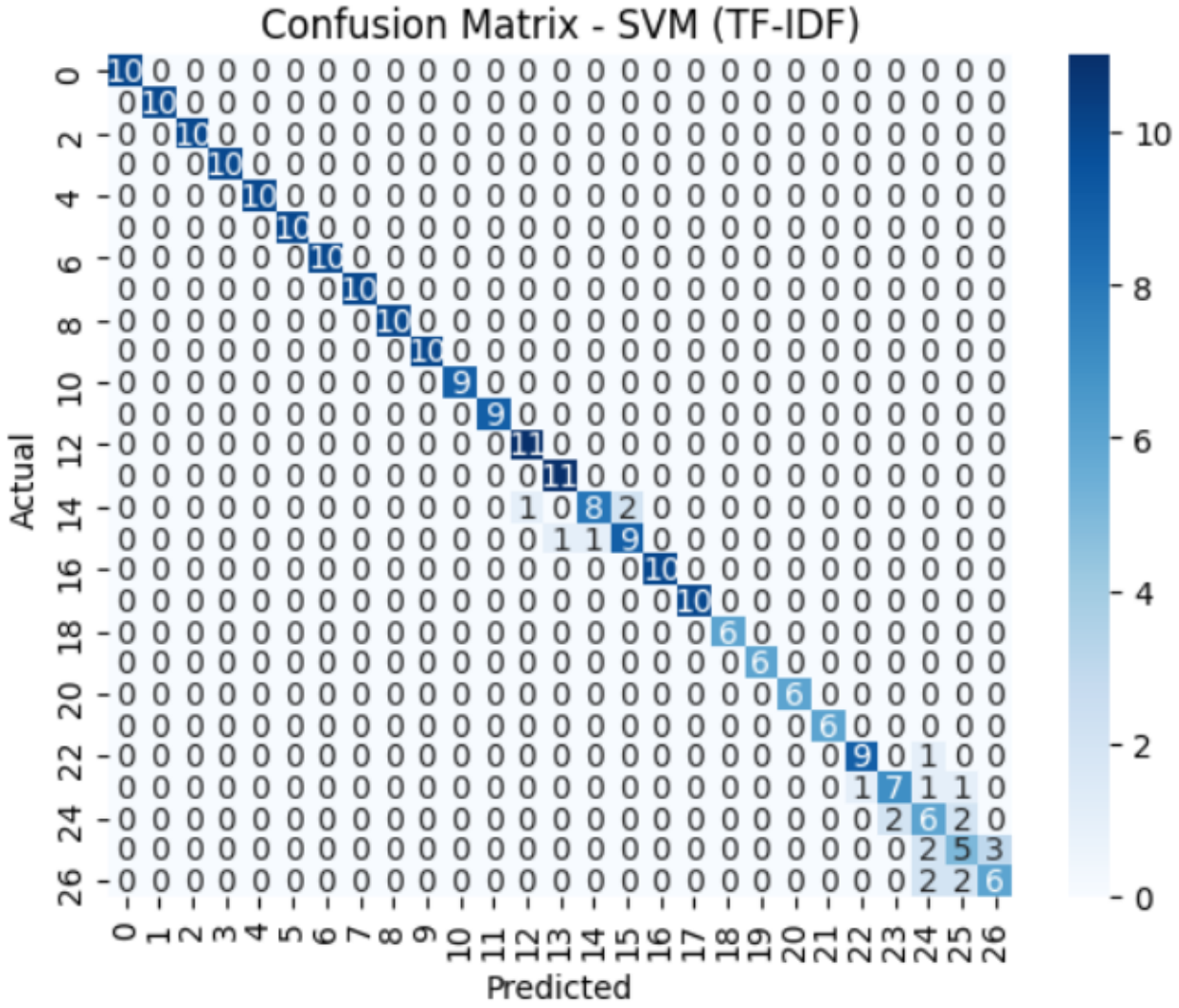

Figure 1: Confusion Matrix for TF-IDF with SVM

4.0.2 Confusion Matrix for Random Forest With TF-IDF

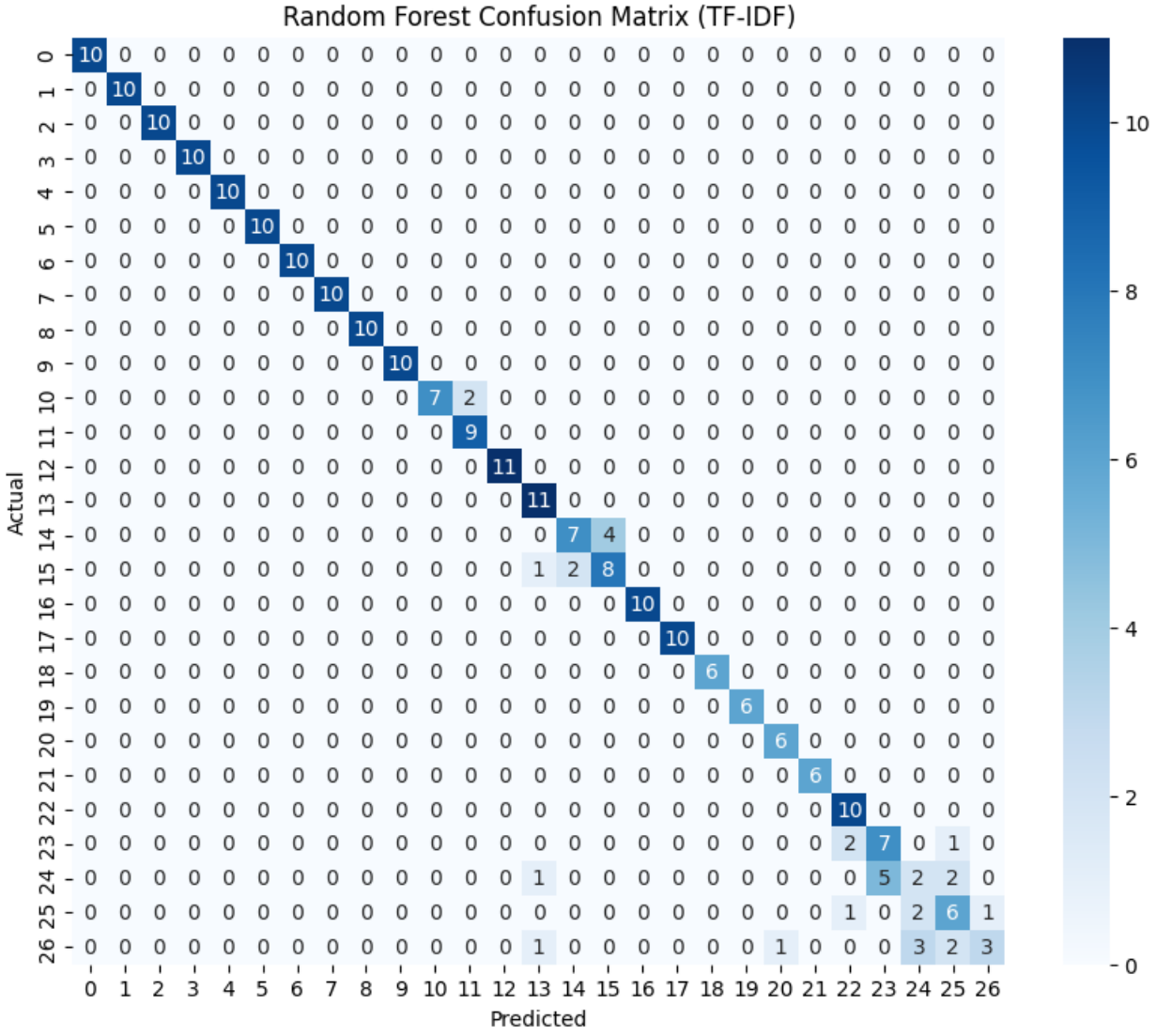

Figure 2: Confusion Matrix for Random Forest With TF-IDF

4.0.3 Confusion Matrix for Gradient Boosting With TF-IDF

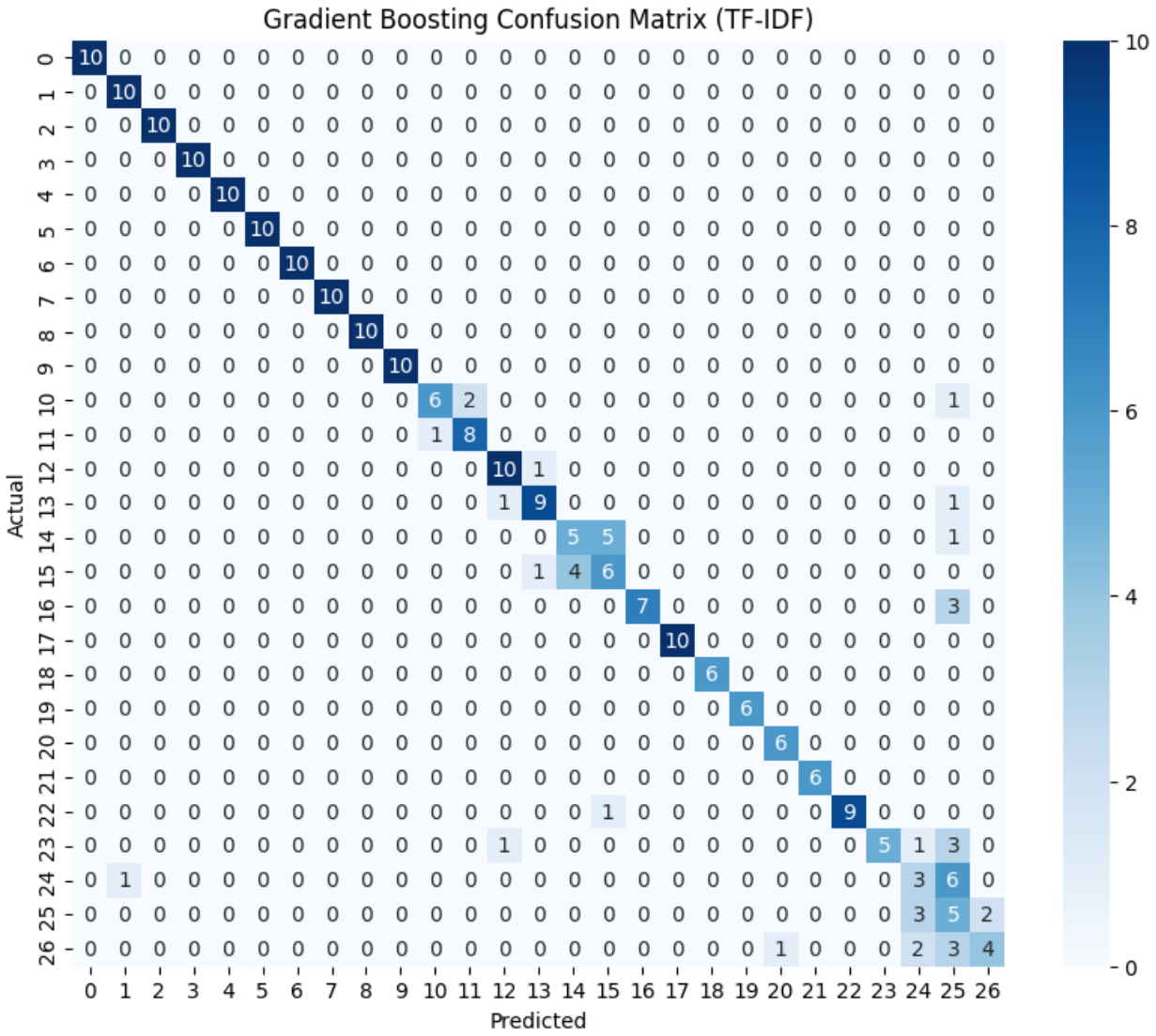

Figure 3: Confusion Matrix for Gradient Boosting With TF-IDF

5 Results and Benchmarking

5.1 Baselines and Comparisons

To evaluate the effectiveness of different embedding techniques and machine learning models, we experimented with a range of combinations. Below, we summarize the results of these models across various embedding methods, including FastText, Word2Vec, TF-IDF, BERT, Sentence-BERT, and LSTM. Each combination was tested on a held-out test set, and classification metrics such as accuracy, precision, recall, and F1-score were reported.

- **FastText:**
  - **Logistic Regression:** Accuracy = 0.08

  - **Random Forest:** Accuracy = 0.54
  - **XGBoost:** Accuracy = 0.55
  - **SVM:** Accuracy = 0.10
- **Word2Vec:**
  - **Logistic Regression:** Accuracy = 0.11
  - **Random Forest:** Accuracy = 0.62
  - **XGBoost:** Accuracy = 0.62
  - **Gradient Boosting:** Accuracy = 0.60
  - **SVM:** Accuracy = 0.11
- **TF-IDF:**
  - **Logistic Regression:** Accuracy = 0.89
  - **Random Forest:** Accuracy = 0.88
  - **XGBoost:** Accuracy = 0.81
  - **Gradient Boosting:** Accuracy = 0.82
  - **SVM:** Accuracy = **0.91**
- **BERT:**
  - **Logistic Regression:** Accuracy = 0.58
  - **Random Forest:** Accuracy = 0.37
  - **XGBoost:** Accuracy = 0.42
  - **SVM:** Accuracy = 0.62
- **Sentence-BERT:**
  - **Logistic Regression:** Accuracy = 0.64
  - **Random Forest:** Accuracy = 0.49
- **LSTM:** Accuracy = 0.86

6 Error Analysis and Refinement

To further understand the limitations and misclassifications made by our models, an error analysis was conducted, particularly focusing on the best-performing combination: TF-IDF with Support Vector Machine (SVM).

6.1 Misclassified Examples

Several misclassified instances were examined manually. These errors generally fell into the following categories:

- **Ambiguity in Text:** Some sentences contained ambiguous or vague language, making it difficult for the model to assign the correct label even for a human.

- **Lack of Context:** In many cases, the model misclassified due to insufficient contextual information, which limited its ability to grasp the full meaning of the sentence.
- **Rare Categories:** Certain labels had very few training examples, resulting in lower classification performance for those classes.

6.2 Confusion Matrix Analysis

The confusion matrix for TF-IDF + SVM reveals that most of the errors occur between semantically similar classes. This indicates that while the model performs well overall, distinguishing between nuanced or overlapping categories remains a challenge.

6.3 Compariosn with SOTA

Using BERT embeddings here resulted in lower accuracy compared to the baseline models and traditional feature extraction methods. We reached highest accuracy using TF-IDF with SVM, which outperformed BERT embeddings. This indicates that while BERT contextualized embeddings are powerful, they may not always provide the best results for all datasets. especially when simpler methods like TF-IDF are well-suited for the problem at hand. Classical machine learning models with TF-IDF - and even other traditional feature extraction methods- can still outperform more complex models like BERT in some cases. This can be explained by the fact that BERT requires a large amount of data and fine-tuning to achieve optimal performance, while traditional methods like TF-IDF are simpler and can be effective with smaller datasets. Also the fact that BERT is a transformer model that captures contextual information, but may not always be the best fit for tasks where simpler representations are sufficient. Here, it's more like mapping the text to a vector space, and the simpler methods are more effective in this case.

6.4 Refinement Strategies

Based on these findings, the following improvements are proposed:

- **Data Augmentation:** Increase the number of training examples for underrepresented classes to improve model generalization.
- **Contextual Embeddings:** Incorporate models that capture broader context (e.g., fine-tuned BERT) to reduce errors caused by ambiguity.
- **Class Rebalancing:** Apply techniques such as SMOTE or weighted loss functions to mitigate class imbalance.
- **Model Ensemble:** Combine predictions from multiple models to leverage their complementary strengths.

7 Visualizations

In this section, we show some visualizations for the best three models in performance

- Accuracy Score in SVM with TF-IDF (Best Model)

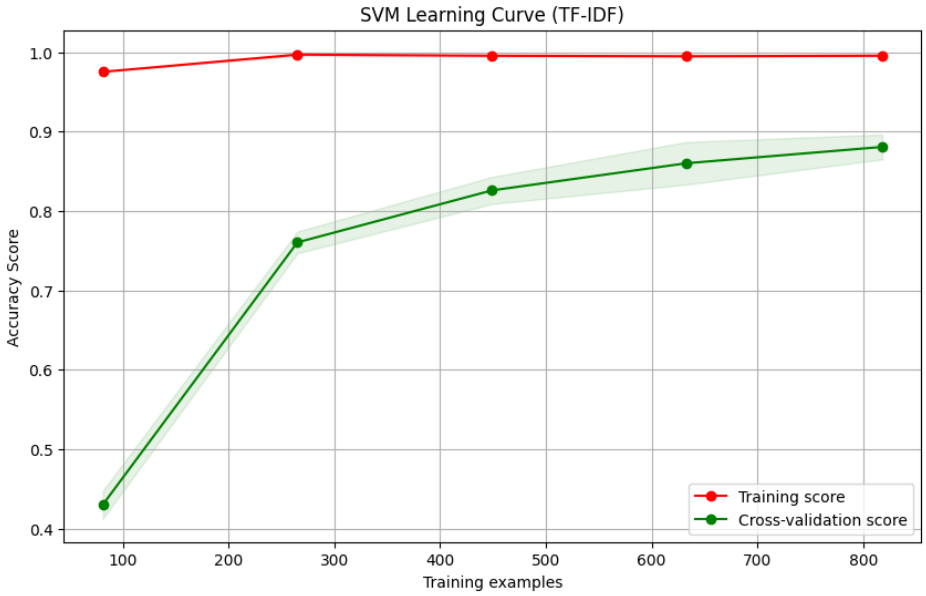

Figure 4: Training and Validation Accuracy for SVM with TF-IDF

- Accuracy Score in Random Forest with TF-IDF (Best Model)

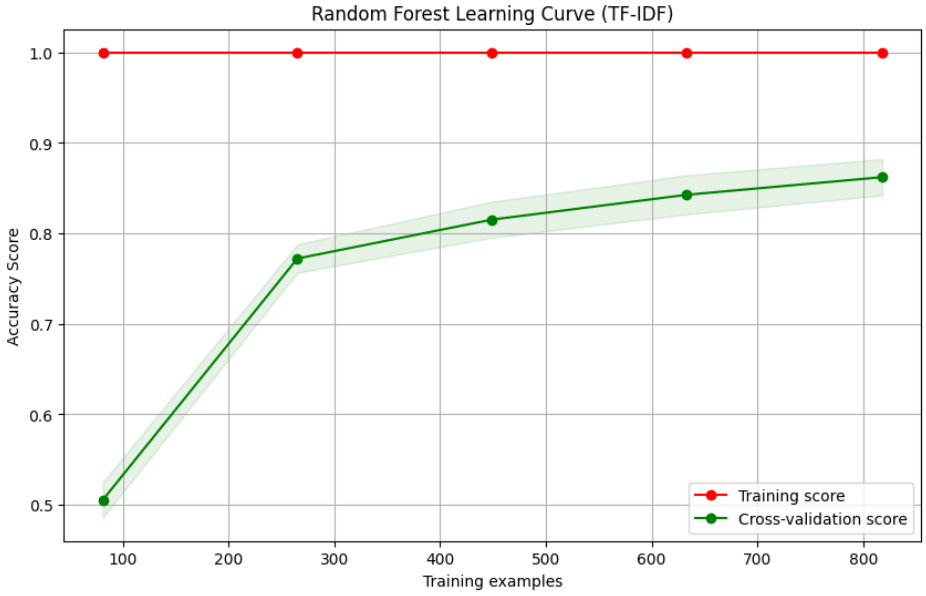

Figure 5: Training and Validation Accuracy for Random Forest with TF-IDF

- Accuracy Score in Gradient Boosting with TF-IDF (Best Model)

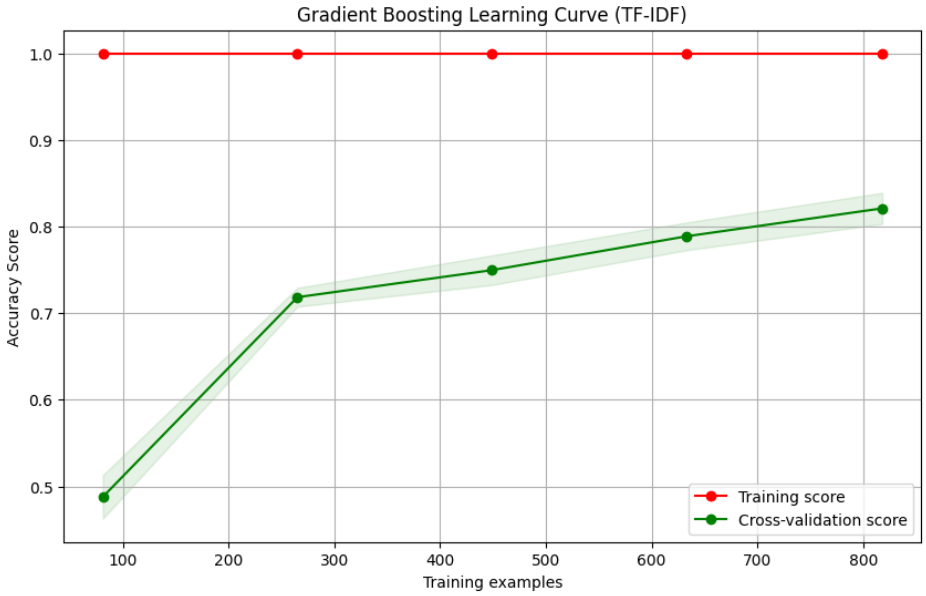

Figure 6: Training and Validation Loss for LSTM Model

8 Innovation and Contribution

This research presents several key innovations and contributions to the field of clinical text classification:

- **Comprehensive Benchmarking:** A thorough comparison was conducted between traditional machine learning models (e.g., Logistic Regression, Random Forest, SVM, XGBoost) and deep learning models (e.g., LSTM), using a range of embedding techniques such as TF-IDF, Word2Vec, FastText, and BERT. This helps identify the most effective combinations for clinical NLP tasks.
- **Evaluation of Embedding Techniques:** The study highlights the strengths and limitations of various embedding methods. While TF-IDF proved most effective in this setting, the findings emphasize the importance of choosing embeddings based on data characteristics and computational constraints.
- **Performance-Resource Trade-off Insights:** By comparing models with different levels of computational complexity, the research provides practical guidance on model selection when computational resources are limited, especially relevant for real-world deployment in healthcare settings.

In essence, this work not only benchmarks models for clinical text classification but also provides actionable insights and methodological contributions that can guide both research and practical applications in medical NLP.

9 Team Participation

All team members contributed in:

- Dataset design and note generation
- Model implementation and evaluation
- Error inspection and final performance tuning

10 References

- AAPC CPT Descriptions: https://www.aapc.com/
- TensorFlow Hub (ELMo): https://tfhub.dev/google/elmo/3
- GPT-Based Augmented Dataset from Internal Chat Generation
- NMT-Based CPT Research: https://www.researchgate.net/publication/350987852_Neural_Machine_Translation-Based_Automated_CPT_Classification_System_Using_Procedure_Text_Development_and_Validation_Study_Preprint