Reproduction Final Project: Query-Focused Electronic Health Record Summarization for Diagnostic Support

Liam Shen, Abhitej Bokka

Abstract

GitHub: https://github.com/liamshen10/CS598_Final_Project.git
Video Presentation: https://www.youtube.com/watch?v=ruyZwse6_4Y

Introduction

Medical professionals face the challenge of making high-stakes decisions under significant time pressure while navigating the overwhelming volume of unstructured information in Electronic Health Records (EHRs).

The electronic health record captures the extensive medical history of the patient, including clinical notes, diagnoses, medications, and observations. However, much of this information is stored in free text form and makes it difficult to retrieve key insights quickly—especially in time-constrained environments such as radiological interpretation or emergency settings.

Radiologists in particular are often given less than 10 minutes to interpret medical imaging studies, making it extremely difficult to properly review extensive patient histories, even though these records contain vital information that could properly influence diagnostic precision. The burden of extracting clinically relevant context from long and fragmented records hinders the diagnostic process and becomes a bottleneck to proper care.

The paper "Query-Focused EHR Summarization to Aid Imaging Diagnosis" by McInerney et al. (PMLR 2020) directly tackles this problem by showcasing a transformer-based approach to generate query-focused extractive summaries of EHR data. The core hypothesis of the paper is that summarization models trained to highlight clinically relevant sentences before radiology imaging can streamline diagnostic decision making. The novel contribution of the paper lies in its use of distant supervision through future ICD codes and disease codes recorded after a radiology event, to assign weak labels to the historical text.

This method allows the model to learn relevance signals without manually annotating summaries. The authors then finetune certain transformer architectures, including ClinicalBERT and then use these labels to classify which sentences are most relevant to the query (in this case, the future ICD diagnosis). The paper also gives us a clinician-in-the-loop evaluation framework, which reported strong performance on metrics like AUROC, F1, and NDCG, and outperformed traditional extractive summarization baselines. This work contributes to the broader field of clinical Natural Language Processing (NLP) in several meaningful ways.

First, it shows a scalable and low-cost method for supervision using future labels, which is highly valuable given the annotation bottlenecks in healthcare.

Second, we use the summarization as a tool in aiding diagnostics rather than using a summarization after certain results and work is done. In this context, creating a gist of relevant information before helps in guiding the user which in case is the doctor.

Third, it applies transformer-based models pre-trained on biomedical text (e.g., ClinicalBERT), showcasing how domain-specific embeddings can enhance summarization and diagnosis alignment. This paper opens up the ability for EHRs not to be just readable but actionable too.

In our reproduction study, we aimed to implement as many components of the original study as possible using the publicly available MIMIC-III dataset. While the original paper used a combination of proprietary hospital data and MIMIC-III, we confined our efforts to the MIMIC-III subset for accessibility and reproducibility. Here is what we were able to successfully reproduce.

Data Preprocessing

Our data preprocessing pipeline began with loading and aligning data from the `NOTEEVENTS`, `DIAGNOSES_ICD`, and `ADMISSIONS` tables in the MIMIC-III dataset. Since our sample data lacked valid `CHARTTIME` entries and empty or null values, we simulated realistic note timestamps by sampling uniformly between each patient's `ADMITTIME` and `DISCHTIME`. This allowed us to replicate the times constraint from the original paper, which filters clinical notes to include only those written *prior to imaging or discharge*. This ensures that summaries are based on information available to the clinician only at the time of diagnosis.

Next, we extracted note text from the `TEXT` column and segmented it into individual sentences using a spaCy-based sentence tokenizer. This step was needed for fine-grained labeling, as the original model operates at the sentence level.

Labeling via Distant Supervision

To reproduce the distant supervision technique used by McInerney et al., we mapped each hospital admission (HADM_ID) to a set of *future* ICD diagnostic codes using the DIAGNOSES_ICD table. These ICD codes acted as weak labels under the assumption that diagnoses documented post-discharge were brought up in pre-discharge clinical text.

Each sentence was then weakly labeled by assigning it all ICD codes associated with its corresponding HADM_ID. To improve label quality and reduce ambiguity, we filtered the dataset to retain only sentences associated with a *single* ICD code. Furthermore, we excluded rare codes that appeared fewer than two times in the dataset. This helped reduce noise, improved class balance, and enabled effective train-test splitting. The resulting dataset was a temporally constrained, sentence-level corpus with weak supervision suitable for transformer-based classification.

Model and Baselines

Model architecture: We implemented a ClinicalBERT-based classifier using the emilyalsentzer/Bio_ClinicalBERT model from HuggingFace. The architecture mirrors that of the paper—sentence embeddings are passed into a linear classifier trained using cross-entropy loss.

Baselines, Training, and Evaluation: We performed supervised sentence-level classification using ICD codes, measuring F1, accuracy, and support metrics across classes. We implemented class weighting to address extreme imbalance in label frequency.

However, some components could not be fully reproduced:

- `Radiologist evaluation`: We were unable to conduct a human evaluation with clinical experts as described in the paper.
- `Query-based summarization interface`: The interactive diagnostic interface shown in the paper was outside our scope.
- `Baselines`: (e.g., TextRank, tf-idf): While the original paper compared against several extractive summarization baselines, we focused primarily on replicating the ClinicalBERT finetuning as a starting point.

In summary, we were able to faithfully reproduce the data pipeline, sentence labeling mechanism, and transformer-based training setup. While our evaluation focuses on standard classification metrics rather than clinician judgment, the implementation validates the feasibility of training summarization models via distant supervision on public clinical data.

Methodology

Environment

Our implementation was conducted using **Python 3.11** within a **Google Colab** environment. The core benefit of Colab was its access to free GPU instances, which provided sufficient resources for lightweight transformer fine-tuning and batch-level inference. However, we encountered limitations around memory capacity and model size, which restricted the scale and depth of experimentation compared to the original paper's setup.

The primary dependencies used in our pipeline included:

- `transformers` (v4.39.3): For accessing pre-trained ClinicalBERT and handling tokenization and model APIs
- `torch` (v2.2.2+cu121): For model definition, GPU acceleration, and gradient optimization
- `pandas` (v2.2.2), `numpy` (v2.0.2), and `sklearn` (v1.6.1): For data manipulation, preprocessing, and classification metrics
- `nltk` (v3.9.1) and `spacy` (v3.8.5): For sentence tokenization and linguistic preprocessing
- `matplotlib` (v3.10.0) and `seaborn` (v0.13.2): For visualizations of data statistics and label distributions

Data

We used the **MIMIC-III** database and it is a publicly available. The records are de-identified dataset of clinical records from the Beth Israel Deaconess Medical Center. To gain access to the data, one will have to be credentialed through an organization, in our case through UIUC, and complete the CITI Data or Specimens Only Research Training as mandated by the MIMIC datasets. Here's a link to the training: https://physionet.org/content/mimiciv/view-required-training/3.1/#1

We specifically leveraged three core tables:

- `NOTEEVENTS.csv`: Contains unstructured clinical notes, including discharge summaries and imaging-related text
- `ADMISSIONS.csv`: Provides admission and discharge timestamps used to enforce the temporal filtering of relevant notes
- `DIAGNOSES_ICD.csv`: Contains ICD-9 and ICD-10 codes, which were used for distant supervision labeling

Data Filtering and Preprocessing We generated a `simulated_noteevents.csv` file containing discharge summaries and clinical notes, filtered by admission windows. Only notes with a `CHARTTIME` within the range of `ADMITTIME` to `DISCHTIME` were retained. Sentence splitting was performed using SpaCy's `en_core_web_sm` pipeline. Each sentence was labeled with one or more **future** ICD codes using the `HADM_ID` mapping.

To address label imbalance, we retained only those ICD codes that appeared at least **twice**. Visualizations including a histogram and pie chart of the top 10 labels supported this filtering.

Model

Although the original article did not release a public codebase, we replicated the summarization model using the **ClinicalBERT** implementation from HuggingFace: https://huggingface.co/emilyalsentzer/Bio_ClinicalBERT. This model was used as our sentence encoder in our classifier pipeline.

Each sentence was tokenized using the ClinicalBERT tokenizer, truncated to a maximum length of 128 tokens, and passed through the ClinicalBERT encoder. We then passed the resulting `[CLS]` token embedding (a 768-dimensional vector) through a single-layer feedforward classifier.

$$\text{logits} = W \cdot \text{BERT}_{[\text{CLS}]}(x) + b \tag{1}$$

$$\hat{y} = \arg\max(\text{softmax}(\text{logits})) \tag{2}$$

Where:

- $\text{BERT}_{[\text{CLS}]}(x) \in R^{768}$: The sentence embedding produced by ClinicalBERT.
- $W \in R^{C \times 768}$: Learnable weight matrix for the linear classification layer.
- $b \in R^{C}$: Learnable bias vector.
- $\hat{y}$: The predicted ICD code class label.

We framed the task as a single-label classification problem for tractability, even though the original task has a multi-label nature. This simplification allows for a more straightforward model training and evaluation under limited computational constraints.

Feasibility of Computation

We feel like the resources and intense computation required to implement and train transformer-based neural models are quite high. They need high-performance GPUs and significant memory resources. However, using a standard Google Colab environment for this reproduction project is not really feasible due to limitations in available GPU resources and memory capacity. Given this problem, without access to dedicated, high-capacity platforms or cloud-based services, it may be necessary to simplify the complexity of the models used. We can even reduce dataset size, or even utilize pre-trained and partially fine-tuned models to make the large computational heavy tasks manageable within our given resources.

Code Implementation

Seeing the computational constraints shown, our implementation strategy involved utilizing pre-trained Clinical BERT models and using existing publicly available transformer-based architectures rather than developing complex models entirely from scratch. Going from scratch would be really difficult and this approach would still allow a multiple of benefits. We would get our methodological understanding, we can facilitate transparent analysis and debugging all the while significantly reducing the computational burden. This would then enhance the project's feasibility within the constrained computational environments given.

Training

We used cross-entropy loss with class weights to address label imbalance:

$$\mathcal{L} = -\sum_{i=1}^{C} w_i \cdot y_i \cdot \log(p_i) \tag{3}$$

Hyperparameters

- **Learning Rate:** 2e-5
- **Batch Size:** 16
- **Epochs:** 3
- **Max Sequence Length:** 128

Computational Requirements

- **GPU:** NVIDIA T4 (via Google Colab)
- **Epoch Runtime:** $\sim$15 minutes per epoch
- **Total Runtime:** $\sim$45 minutes
- **Total Trials:** 2 (due to time/memory constraints)

To simplify the task, we restricted our classification to the top 300 most frequent ICD codes and implemented early stopping logic manually via epoch limitation.

Evaluation

We evaluated our trained model on a held-out test set. The following metrics were computed using `sklearn.metrics`:

- **Accuracy:** Overall correct predictions
- **Macro-F1:** Unweighted average of F1 scores per class
- **Precision / Recall:** Calculated per class and averaged
- **Support:** Count of instances per label

Since we were facing a label imbalance and a long-tailed distribution, macro-F1 was chosen as our primary evaluation metric. We also reported full per-label precision and recall to evaluate underrepresented ICD codes. This basically helps us to see if the model is missing rare, but important classes.

Novelty, Relevance, and Hypothesis

McInerney et al.'s main innovation lies in their unique use of future ICD codes for distant supervision, enabling efficient training of transformer-based summarization models without having to use laborious and manual annotation. This approach addresses the fundamental challenge in clinical NLP research: the lack of properly annotated data. The relevance of their work is highlighted by the simple improvement it offers to clinicians, significantly outperforming baseline unsupervised summarization methods. The main theme driven by McInerney et al. is that leveraging distant supervision with ICD codes generates clinically meaningful and accurate summarizations that help with diagnostic capabilities [1].

Results and Extensions

Results

We report evaluation metrics for our ClinicalBERT-based sentence classifier trained using weak labels derived from future ICD codes. After filtering to the 20 most frequent labels and sampling 1,000 sentences, we trained for 3 epochs and evaluated on a stratified test set of 300 samples. Despite this setup, the model showed limited classification ability.

As shown, the model's accuracy was only 5%, with precision and recall near zero for most classes. The multiclass AUROC score of 0.4902 suggests the classifier performed close to random chance. This underperformance is likely due to multiple compounding factors:

Metric	Macro Avg	Weighted Avg	Accuracy
Precision	0.05	0.07	-
Recall	0.06	0.05	-
F1 Score	0.05	0.05	0.05
AUROC (OVR)		0.4902	

Table 1: Evaluation metrics on the top 20 ICD code sentence classification task.

- The small training sample size (700 sentences) was insufficient for a high-capacity transformer model like ClinicalBERT.
- Each sentence was weakly labeled using entire-hospital-stay ICD codes, introducing significant noise; many sentences may not actually relate to the target diagnosis.
- The classifier was trained in a single-label setup, while most admissions had multiple relevant ICD codes.

Compared to the original study [1], which achieved an AUROC of 0.70–0.82 on similar tasks, our results fell significantly short. The original paper's model operated over full patient histories and used sentence-level attention pooling, allowing more holistic summarization. In contrast, our simplified sentence-level approach ignored inter-sentence relationships and lacked the richer context used in their architecture.

Additional Extensions or Ablations

To extend the original study and evaluate architectural robustness, we implemented an ablation study by replacing the `[CLS]` token-based classification with average pooling over the token embeddings from ClinicalBERT. The goal of this ablation was to determine whether distributing attention across the entire sentence, rather than focusing solely on the beginning token, could improve generalization, especially for short or noisy clinical sentences.

We used a large language model (LLM) to brainstorm and implement this idea. Our prompt to the LLM asked for a potential way to modify the ClinicalBERT summarization architecture with a simple ablation. The model suggested average pooling as a widely used alternative to `[CLS]` classification and generated initial PyTorch code using masked mean pooling over attention-weighted token embeddings.

The following equation was used in the ablation model:

$$\text{avg\_pooled} = \frac{\sum_{i=1}^{T} h_i \cdot m_i}{\sum_{i=1}^{T} m_i} \tag{4}$$

where h_i are the hidden states at token position i, and m_i are the corresponding elements of the attention mask. This average-pooled embedding is passed through a dropout layer and a linear classifier to produce logits.

Despite these theoretical benefits, the change in performance was minimal due to the limitations of our training setup. Specifically, the AUROC for the ablation model remained approximately at **0.4882**, closely matching the original implementation's AUROC of 0.4902. These outcomes suggest that the representational bottleneck is not merely an issue of pooling strategy, but is deeply rooted in the small size of our training dataset and the sparsity of ICD code labels, many of which occur infrequently even in filtered subsets.

Nonetheless, this ablation experiment was highly valuable in validating the flexibility and correctness of our model implementation pipeline. The ease with which we modified the architecture confirmed that our codebase supports rapid experimentation. This extension demonstrated how nuanced changes to input representation can affect downstream diagnostic classification in small, yet subtle ways.

With this said, future directions may involve more aggressive experimentation. This may be with pooling strategies such as max pooling or attention-based pooling and combining them in hybrid formulations. Additionally, incorporating clinical concept-aware embeddings (e.g., via cTAKES or MetaMap) may enhance semantic grounding and reduce noise introduced by domain-agnostic tokenization. Hyperparameter tuning along with architectural changes is another promising area of extension. Overall, this ablation provided key insights into model dynamics under our constrained data settings and served as a practical exercise in rapid prototyping.

Discussion

Implications of the Experimental Results

Our experimental results showed that even when trying our hand at the reproduction of the paper's sentence-level encoding with ClinicalBERT, the classification using future ICD codes, and the use of weak supervision — our model performed significantly worse than what was reported in the original study. We observed an overall macro F1-score close to 0.05 and a multiclass AUROC of approximately 0.49. This stark performance gap can be attributed to several practical limitations in our experimental pipeline.

The most important implication is that reproducing advanced transformer-based models in the medical domain is highly sensitive to data scale and compute resources. Our results suggest that while the methodological framework proposed by McInerney et al. is valid, its practical effectiveness depends heavily on access to large volumes of clean, temporally valid clinical notes and the ability to train on high-end GPUs for extended periods.

Is the Original Paper Reproducible?

In principle, the original paper is reproducible; however, in practice, several critical barriers inhibit full reproduction:

- **Data Access and Quality:** We had to restrict our study to a small subset of MIMIC-III due to computational constraints, and we also lacked access to curated note types like radiology reports that were central in the original study.
- **Temporal Filtering:** While we implemented chart time filtering using the `ADMISSIONS` table, the available note timestamps were often sparse or missing, which weakened our ICD label supervision alignment.

- **Label Sparsity:** Even after sampling the top 20 ICD codes, many were still underrepresented. This likely impacted class balance and learning stability.
- **Compute Limitations:** Training was performed on a single-session notebook (Colab) with limited RAM and restricted GPU access. We could only afford 3 epochs on 1,000 samples, which is insufficient for convergence on such a complex task.

What Was Easy?

The initial preprocessing of MIMIC-III tables using pandas and the integration of HuggingFace's ClinicalBERT model was relatively straightforward, thanks to the extensive documentation and community support for these tools. Similarly, the weak labeling pipeline using ICD mappings was made easy by the well-structured relational nature of MIMIC-III.

What Was Difficult?

Several components posed significant challenges:

- **Data alignment and filtering**: Making sure that sentence timestamps fell within the admission-discharge window required careful temporal logic and sometimes manual correction due to missing `CHARTTIME` fields.
- **Class imbalance**: With hundreds of ICD labels and only a few samples per label, achieving a useful stratification split for training/testing required careful label thresholding.
- **Training instability**: The model was prone to overfitting or failing to generalize due to the small dataset size and high variance in sentence structure.

Recommendations for Improving Reproducibility

There are our suggestions to improve the reproducibility of this class of clinical NLP models:

1. **Public Codebase and Pretrained Weights:** Publishing both the training pipeline and pretrained checkpoints would significantly reduce the barrier for reproduction. The absence of an official repo slowed our progress and required some reimplementation from scratch.
2. **Detailed Dataset Construction Scripts:** Sharing exact filtering steps (e.g., which note categories, temporal filters, and ICD mappings) would improve clarity. Additionally, publishing label distribution plots and intermediate data summaries would allow others to verify their reproduction steps.

In summary, although we implemented a reasonable approximation of the original study's pipeline, our limited training data, lack of compute, and reliance on weak supervision with sparse labels resulted in poor performance. The methodology is sound, but reproducing the full benefit requires access to scale—both in data and in hardware.

Author Contributions

Liam Shen was responsible for implementing the data preprocessing pipeline, including temporal filtering and weak labeling using ICD codes. He also set up the model architecture using ClinicalBERT, handled integration with HuggingFace's tokenizer, and ran all training and evaluation experiments. Liam contributed significantly to the results analysis, ablation studies, and writing of the methodology and discussion sections.

Abhitej Bokka focused on reproducing the dataset processing and aligning the MIMIC-III tables (NOTEEVENTS, ADMISSIONS, and DIAGNOSES_ICD). He managed the data exploration, label distribution, and sampling strategy. Abhitej also did the literature review, wrote the introduction and paper framing sections, and handled the LaTeX formatting, visualizations, and documentation for the final report.

We collaborated closely on prompt engineering for LLM support, validation of reproduction steps, and brainstorming extensions for the project.

References

[1] D. J. McInerney, B. Dabiri, A.-S. Touret, G. Young, J.-W. van de Meent, and B. C. Wallace, "Query-Focused EHR Summarization to Aid Imaging Diagnosis," *Machine Learning for Healthcare Conference*, vol. 126, pp. 582–604, 2020. [Online]. Available: https://proceedings.mlr.press/v126/mcinerney20a/mcinerney20a.pdf