CS 598 DLH Project Proposal

Amit Jangid, Constantin Kappel, Daniel Sanchez

March 24th 2024

ajangid2@illinois.edu, normank2@illinois.edu, daniel43@illinois.edu

1 Citation to original Paper

We picked a publication by Hur et al. [5] titled "*Unifying Heterogeneous Electronic Health Records Systems via Text-Based Code Embedding*".

2 General Problem

While EHR (Electronic Health Records) provide an attractive data source to do research on exposures and disease, there are many heterogenous medical code formats used by different healthcare providers. Thus, clinical studies using EHRs are difficult to scale up due to data incompatibilities. The technical hurdle is that previous systems learned hidden representations (latent embeddings) for each code system which projected the same medical concept with different encodings into different, incompatible semantic spaces. The paper [5] learns from the unstructured textual descriptions instead of the medcodes themselves, a strategy which the authors termed *description embedding* (*DescEmb*). Using *DescEmb* the authors were able to pool differently structured EHRs, namely *MIMIC-III* and *eICU*, into one larger dataset and achieved higher accuracy.

3 Scientific Approach

1. First, we will try to replicate the results of [5] as a baseline.
2. We plan on using `pyhealth`
   applicable. Especially for handling EHR data this should be very helpful.
3. Thirdly, we intend to test a hypothesis described in more detail in section 4 on the following page. In short, we would like to replace the BERT[1]-derived encoders by one which was pretrained from scratch on biomedical data.

As we outlined in section 7 we will not do comprehensive pre-training from scratch, but work with pre-trained architectures, which we may fine-tune for our experiments.

[1] Bidirectional Encoder Representations from Transformers **(author?)** [2]

4 Hypotheses to be tested

The authors of [5] conducted experiments with several BERT-based architectures, such as BioBERT **(author?)** [8], ClinicalBERT **(author?)** [4] and BlueBERT, which were partially trained or fine-tuned on medical literature. In contrast, Gu et al. [3] found that pre-training on generic NLP corpi, such as derived from Wikipedia, actually perform worse than ones which have been initialized with random values and only pre-trained on PubMed. They called their model PubMedBERT and were able to demonstrate superior performance to the models employed by [5].

We would like to test if PubMedBERT is capable of matching or even superceding the performance of the other models utilized, which were not fully pre-trained on pure corpi from the biomedical domain.

5 Ablations planned

A true ablation study with PubMedBERT and e.g. ClinicalBERT or BioBERT would mean we would have to pre-train all these architectures from scratch on both, generic NLP corpi as well as PubMed. This will not be computationally feasible. Instead, we will cite the work of [3] where applicable and rather try to replicate their results with different, already pre-trained architectures.

6 Data access

The two datasets we plan on utilizing, namely MIMIC-III **(author?)** [6, 7] and eICU **(author?)** [9], are both publicly available. We will need to take into account some regulations and license conditions concerning data safety and privacy.

7 Feasibility of the computation

As far as computational cost is concernced let's consider the size of the data and the size of the model architecture we need to train.

7.1 Model architecture

According to **(author?)** [5] p. 4, bottom right paragraph, the largest architecture the authors trained was BERT-base, which has 110 M (10^6) parameters. Other models the authors experimented with include BioBERT and ClinicalBERT, which are based on BERT-base, too. Let's do a quick estimate[2] of how much GPU memory is needed to load and train such a model:

	Bytes per parameter
Model parameters	4
Adam optimizer	8
Gradients	4
Activations and temp memory	8
TOTAL	24

Thus, a model with 110 M parameters would consume $110 \cdot 10^6 \times 24 bytes = 2.46 GB$ to just load the model. This managable on modern-day GPUs. One of the authors has a GTX 3080 with 12 GB

[2] Assuming 32-bit single precision computation

of VRAM available for full training and a GTX 1070 with 8 GB is available to do some code testing on small batches. As mentioned in sections 3 and 4 we are not planning to do pretraining. While Hur et al. **(author?)** [5] did not explicitely provide information on training time, the authors of BioBERT mentioned that pre-training took 10 days on a V100 GPU. Given the iterative nature of the doing model training and the amount of time we can spend, the cost for pre-training from scratch would be prohibitive for us. So, we will only do inferencing with available pre-trained architectures and fine-tuning where applicable.

7.2 Datasets

We will use two datasets : MIMIC-III and eICU. According to **(author?)** [7] the MIMIC-III dataset is a large, de-identified, and publicly-available collection of medical records. It consists of 112,000 clinical reports records with an average length of 709.3 tokens. The dataset includes 1,159 top-level ICD-9 codes, and each report is assigned to 7.6 codes on average. This makes for about 78056000 tokens in MIMIC-III. We can make a quick estimate of the amount of compute necessary by using the calculation summarized by Adam Casson **(author?)** [1][3]: Using the data above we get about 600 M FLOPs per token. At the number of tokens that amounts to about 46800 TFLOPs for the whole MIMIC-III dataset (that is per epoch). On a GTX 3080 with about 30 TFLOPs/s one training epoch would thus take about 26 min. In original BERT paper[4] **(author?)** [2] the pre-training was done for 40 epochs and fine-tuning was done for 2-3 epochs on some more specialized NLP tasks. 40 epochs would then take about 17-18 hours. We conclude that for MIMIC-III fine-tuning or some amount of training over the course of several weeks we have availalbe should be possible on our available hardware.
The eICU Collaborative Research Database holds data associated with over 200,000 patient visits. If we assume the same average length per visit (we didn't find any average number in the paper) and we further assume that each report has the same average number of codes, the eICU dataset would comprise very roughly about twice the number of tokens compared to MIMIC-III. So, fine-tuning would still be possible, while many dozens of training rounds should be avoided or at least attempted only very few times. If possible we might want to limit most experiments to MIMIC-III.

7.3 Pre-processing

On GitHub the authors mention that pre-processing took 1 hour on 128 cores and 60 GB of RAM. It will likely take us significantly longer and if RAM was limiting we might need to rent a cloud node for this part and download the processed data.

8 Existing Code

The authors Hur et al. **(author?)** [5] provide code on GitHub[5] for pre-processing as well as training. We thus have a reference in case we get stuck with reproducing their results. We will try to rely on pyhealth as much as we can to keep the amount of code we have to write managable.

[3] Note: The calculation presented there was originally made for decoder-only architectures like GPT, while BERT is an encoder-only, bidirectional architecture. For the sake of a rough estimate we disregarded that fact here.

[4] According to **(author?)** [2] the size of the token embeddings is 768, the number of transformer blocks is 12 and the number of self-attention heads is 12.

[5] https://github.com/hoon9405/DescEmb

References

[1] Adam Casson. Transformer flops. 2023. 7.2

[2] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv:1810.04805 [cs]*, May 2019. arXiv: 1810.04805. 1, 7.2, 4

[3] Yu Gu, Robert Tinn, Hao Cheng, Michael Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. Domain-Specific Language Model Pretraining for Biomedical Natural Language Processing. *ACM Transactions on Computing for Healthcare*, 3(1):1–23, January 2022. arXiv:2007.15779 [cs]. 4, 5

[4] Kexin Huang, Jaan Altosaar, and Rajesh Ranganath. ClinicalBERT: Modeling Clinical Notes and Predicting Hospital Readmission, November 2020. arXiv:1904.05342 [cs]. 4

[5] Kyunghoon Hur, Jiyoung Lee, Jungwoo Oh, Wesley Price, Young-Hak Kim, and Edward Choi. Unifying Heterogeneous Electronic Health Records Systems via Text-Based Code Embedding, March 2022. arXiv:2108.03625 [cs]. 1, 2, 1, 4, 7.1, 8

[6] Alistair Johnson, Tom Pollard, and Roger Mark. MIMIC-III Clinical Database, 2015. 6

[7] Alistair E. W. Johnson, Tom J. Pollard, Lu Shen, Li-wei H. Lehman, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, Peter Szolovits, Leo Anthony Celi, and Roger G. Mark. MIMIC-III, a freely accessible critical care database. *Scientific Data*, 3(1):160035, May 2016. Publisher: Nature Publishing Group. 6, 7.2

[8] Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. BioBERT: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, page btz682, September 2019. arXiv:1901.08746 [cs]. 4

[9] Tom J. Pollard, Alistair E. W. Johnson, Jesse D. Raffa, Leo A. Celi, Roger G. Mark, and Omar Badawi. The eICU Collaborative Research Database, a freely available multi-center database for critical care research. *Scientific Data*, 5(1):180178, September 2018. 6