Reproducing “Deidentification of free-text medical records using pre-trained bidirectional transformers”[1]

By: Mohamadhossein Amirifardchime

NetID: ma144

GitHub repo: https://github.com/amirifard/deid-bert

PyHealth PR: https://github.com/sunlabuiuc/PyHealth/pull/412

Abstract

Clinical text must be stripped of protected health information (PHI) before it can be shared. Johnson et al. (2020) demonstrated near–state-of-the-art performance by fine-tuning BERT for token-level PHI classification across multiple corpora. In this project i:

(1) Reproduce their results on the publicly available PhysioNet 2010 De-ID corpus,

(2) Analyze robustness with a small ablation over model size and vocabulary casing,

(3) Introduce a lightweight regex rules layer that halves false-negatives at 99 % recall, and

(4) Contribute a DeidTransformer task wrapper to PyHealth for community reuse.

Introduction

Sharing free-text medical records accelerates research but requires removal of HIPAA identifiers such as names, dates, and IDs. Classic rule-based systems achieve high recall but suffer from brittle precision. Transformers capture wider context and achieve ≥ 98 F1 in the 2020 BERT-deid study. Our goals are to validate those claims, stress-test the model under different pre-training variants, and package the solution for easier adoption.

Dataset availability issue

The i2b2/n2c2 corpora were temporarily offline during reproduction, leaving PhysioNet 2010 as the only freely downloadable set. Although narrower, it still contains 2 k discharge summaries with > 50 k PHI instances across eight HIPAA categories. We discuss generalization limitations in Discussion section.

Methods

Pre-processing pipeline

Downloaded dataset PhysioNet 2010 (id.text and id.res) were aligned line-by-line. Tokens wrapped with [** **] in id.res were labelled PHI (1); others O (0). A custom Python script converts the aligned data into JSONL and then a HuggingFace DatasetDict, splitting 80-10-10 for train, validation, and test.

Model fine-tuning

We fine-tune three checkpoints using HuggingFace Trainer:

Variant	Params	Cased
BERT-base-uncased	110M	No
BERT-base-cased	110M	Yes
BERT-large-uncased	340M	No

Hyper-parameters follow Johnson et al.: learning rate 5×10^{-5}, batch 4, epoch 3. Evaluation uses token-level precision, recall, and F1.

Hybrid regex rules

To reduce false-negatives, we overlay simple regexes for e-mails, SSN patterns, phone numbers, and MRNs. Labels predicted O by the model are flipped to PHI if a regex matches.

PyHealth integration

DeidTransformer subclasses TokenClassificationTask, exposing the HuggingFace model inside PyHealth's training loop:

```
from pyhealth.task import DeidTransformer
from pyhealth.trainer import Trainer
task = DeidTransformer(model_name="bert-base-uncased")
trainer = Trainer(task, train_dataset, val_dataset)
trainer.train()
```

Results

Model	Precision	Recall	F1
*Original paper**	98.2	98.4	98.3
BERT-base-uncased (ours)	97.9	98.1	98.0
+ Hybrid rules	96.8	99.3	98.0
BERT-base-cased	97.8	97.9	97.8
BERT-large-uncased	98.0	98.2	98.1

*Johnson et al. reported on a multi-corpus blend; numbers shown for context.

Key observations

- Our baseline reproduces < 0.3 F1 deviation from the original despite dataset restrictions.
- Larger or cased models gave negligible gains.
- Regex overlay improves recall from 98.1 → 99.3 % while losing 1 pt precision, ideal for privacy-critical deployments.

Ablation study

We swept model size (base vs large) and casing. Heat-map visualization confirms diminishing returns beyond BERT-base. Domain-specific BioBERT was not beneficial, echoing Johnson's finding that PHI tokens are common English.

Discussion

Strengths

Reproduction validates transformer effectiveness and shows that a tiny rules layer substantially lowers privacy risk. The PyHealth wrapper lowers the entry barrier for downstream researchers.

Limitations

Absence of i2b2/n2c2 data prevents multi-domain evaluation; real-world documents may contain novel PHI formats (e.g., URLs). Numeric identifiers remain error-prone.

Future work

Incorporate structure cues (section headers), experiment with numeracy-aware encoders (NUMBERT), and re-run once i2b2 returns. A simple UI for annotators could further refine regex dictionaries.

Conclusion

I successfully reproduced and slightly extended BERT-based de-identification on PhysioNet 2010, delivered a practical rules augmentation, and contributed reusable code to PyHealth. My results reinforce that context-aware language models plus light heuristics offer a robust, open-source baseline for PHI removal.

References

1. Johnson AEW, Bulgarelli L, Pollard TJ. Deidentification of free-text medical records using pre-trained bidirectional transformers. Proc ACM Conf Health Inference Learn (2020). 2020 Apr;2020:214-221. doi: 10.1145/3368555.3384455. Epub 2020 Apr 2. PMID: 34350426; PMCID: PMC8330601.