# Project Status Report for Assessing and Masking of Personally Identifiable Information(PII) Leakage in Documents

Tran Huy Nguyen, Bhuvan Shah, Sriram Gurazada, Payal Rashinkar, Ashley Taylor

## 1 Tasks Performed

**1.1 Data Preprocessing** For this project, we are using the PIImasking43k dataset. This dataset derived from the AI4Privacy Company in relation to Hugginface. The dataset contains over 43,000 annotated sentences with more than 40 unique PII tags, including 'B-NAME', 'I-NAME', 'B-LOCATION', 'I-LOCATION', 'B-URL', 'I-URL', 'B-ZIPCODE', etc... Each sentence is tokenized, and the corresponding PII tags are aligned with the tokens to create input-target pairs for model training. For testing purposes, we only use the first 100 rows in the dataset as training the model will take a lot of time if we do it for the whole dataset. The dataset covers a range of contexts in which PII can appear. The sentences span 54 sensitive data types ( 111 token classes), targeting 125 discussion subjects/use cases split across the business, psychology, and legal fields, and five interaction styles

**1.2 Model Fine-tuning** We employ the pre-trained BERT (Bidirectional Encoder Representations from Transformers) model, specifically the '*bert-base-cased*' variant, as our base model. BERT, originally trained on general corpora, is adapted to the PII detection task by fine-tuning its final layer to predict the 44 PII tags in our dataset. Training parameters include a learning rate of 2e-5, a device train batch size of 4, warmup steps of 500, weight decay of 0.01, and 30 epochs.

**1.3 Custom Evaluation** Due to the label mismatch between the pre-trained BERT model, which was originally trained on the CoNLL-2003 dataset with only 9 NER tags (compared to our 40), a customized evolution metric is needed for our project. This metric currently assesses the model's performance on our specific PII tags, focusing on precision, recall, and F1 score.

**1.4 Other** We also did a few literature reviews on the topics to find possible directions and possible solutions to help with our project. Furthermore, we have also replicated a few PII related models, such as attacks on LMs to see PII leakage and Removing student name from Education papers.

## 2 Preliminary Results

The fine-tuned model achieved an accuracy of 0.91 and an F1 score of 0.51 on the first 100 rows of the PII43k dataset. These results indicate that the model has a promising ability to detect PII entities. However, there is much room for improvement, especially regarding F1 score with more training and fine-tuning.

## 3 Risks and Challenges

**3.1 Label Mismatch** The major challenge is the label discrepancy between the pre-trained BERT model and the PIImasking43k dataset. Since the pre-trained BERT model has its own trained NER labels and tokenized sentences compared to the custom schema from PIImaksing43k, the input data needed to be compatible with the model, resulting in misidentified labels. This necessitates custom modifications to the model's output layer, especially evaluation metrics, as the classification report function of the model does not allow us to change the parameters of the predefined function for custom NER tags for evaluation.

**3.2 Data Imbalance** In PII or in general NER taggings, the dataset would exhibit an imbalance distribution of tags due to limited objects in a sentence. This means the model would be biased toward the O'/Other tag. This may result in PII being misidentified as 'O', leaving it to degrade in performance and vulnerable in data for attacks.

**3.3 Computational Resources** Fine-tuning large pre-trained models like BERT requires significant GPU power and computational resources, which can be a constraint for us with no access to high-performance computing facilities.

## 4 Plan to Mitigate Risks and Address Challenges

### 4.1 Custom Evaluation Metrics

To accurately evaluate the performance of the fine-tuned model on the PII43k dataset, we will develop and refine custom evaluation metrics that align with our specific PII tags. This involves: Precision, Recall, and F1 Score: Calculating these metrics separately for each PII tag to understand the model's performance on individual entity types.

*Macro and Micro Averages*: Computing macro-averages (averaging the metric scores across all tags) and micro-averages (aggregating the contributions of all tags to compute the average) to capture the overall performance.

*Confusion Matrix*: Developing a detailed confusion matrix for PII tags to identify specific areas where the model is struggling, such as confusing similar tags.

*Error Analysis*: Conducting thorough error analysis to pinpoint common mistakes and patterns in the model's predictions, which will inform further model refinement and training strategies.

*Privacy Attacks*: Finally, for our custom evaluation, three different privacy attacks (Extraction, Inference, and Reconstruction) may be introduced to test the effectiveness of masked data. These attacks would be an additional metric to see the effectiveness of our model.

*Biases*: Since most tokens in data would be 'O', Weights may be introduced to ensure that PII would be properly identified and masked in input data. However, it should be noted that proper utilization of weights must be used as this can also lead to misidentifying the 'O' token as PII.

**4.2 Addressing Data Imbalance** The imbalance in the distribution of PII tags in the dataset can lead to a biased model. To mitigate this, we will explore several techniques:

*Oversampling:* Increasing the frequency of underrepresented PII tags in the training dataset to balance their presence. Undersampling: Reducing the frequency of overrepresented PII tags to prevent the model from becoming biased towards them.

*Synthetic Data Generation:* Using techniques like SMOTE (Synthetic Minority Over-sampling Technique) to generate synthetic examples of underrepresented PII tags.

*Weighted Loss Functions:* Modifying the loss function to assign higher weights to underrepresented PII tags, thereby increasing their importance during model training.

*Stratified Sampling:* Ensuring that each batch of data used for training contains a representative distribution of PII tags to maintain balance.

**4.3 Computational Efficiency** To address the challenge of computational resources, the following strategies will be employed:

*Model Pruning:* Reducing the size of the BERT model by pruning less important neurons or layers can decrease the computational load without significantly impacting performance.

*Quantization:* Converting the model's weights from floating-point to lower-precision formats can reduce memory usage and speed up inference.

*Efficient Hardware:* Utilizing more efficient GPUs or specialized hardware like TPUs (Tensor Processing Units) can provide better performance per watt.

*Distributed Training:* Leveraging multiple GPUs or cloud computing resources for parallel training can reduce the time required for fine-tuning the model.

## 5 Individual Contributions

1. Payal and Sriram were responsible for analyzing the PII43k dataset to understand the distribution of PII tags. They preprocessed the data by tokenizing sentences and aligning PII tags with tokens.

2. Huy and Bhuvan focused on fine-tuning the pre-trained BERT model for the PII detection task. They adapted the model to predict the specific PII tags in the dataset and set the training parameters.

3. Ashley and Payal developed custom evaluation metrics to accurately assess the model's performance on the PII43k dataset. They implemented precision, recall, and F1 score calculations for each PII tag.

4. Sriram and Huy evaluated the fine-tuned model on the PII43k dataset using the custom evaluation metrics. They analyzed the model's accuracy and F1 score to identify areas for improvement.

5. Bhuvan and Ashley were responsible for documenting the methodology, results, and insights gained from the preliminary evaluation.

## 6 References

1. Devlin, J., Chang, M.-W., Lee, K., Toutanova, K. (2019). BERT: Pre-training of Deep Bidi- rectional Transformers for Language Under- standing. NAACL HLT 2019.

2. Sang, E. F., De Meulder, F. (2003). Introduction to the CoNLL-2003 Shared Task: Language-Independent Named Entity Recognition. In Proceedings of CoNLL-2003.

3. AI4Privacy. (2021). PII43k: A Dataset for PII Detection in Text. Hugging Face Datasets. Available at: https://huggingface.co/datasets/ai4privacy/piimasking43k

4. Lukas, Salem, Sim, Tople, Wutschitz, Zanella-Béguelin. (2023). Analyzing Leakage of Personally Identifiable Information in Language Models.