Master's Project Report

Enhancing Privacy and Accessibility in Educational Data through Automated PII Detection

By

Lovedeep Singh

Under the guidance of

Professor Desmund Lun

Graduate School - Camden

Rutgers, The State University of New Jersey

May 2024

1. Abstract

Personal Identifiable Information (PII) poses a significant barrier in educational contexts, where data analysis and dataset sharing are crucial for progress. The automated detection and removal of Personally Identifiable Information (PII) from educational texts are crucial for preserving student privacy and enhancing the accessibility of educational datasets for research. This project develops automated techniques to efficiently detect and remove PII from educational data, thereby safeguarding student privacy and enabling the broader use of educational texts and documents. It is expected to lower the costs and barriers associated with the manual screening of educational content, thus facilitating the creation of safer, open-access educational resources.

2. Introduction

In today's digital age, educational institutions and platforms generate vast amounts of data that are crucial for advancing learning science and developing educational tools. However, these data often contain sensitive Personal Identifiable Information (PII) which can include names, email addresses, social security numbers, and other details that can identify individuals. The presence of such information poses significant privacy risks and regulatory challenges, particularly with laws such as GDPR and FERPA imposing strict guidelines on data handling.

In an era where digital education is prevalent, the vast accumulation of educational data containing sensitive information can jeopardize student privacy if mishandled. The presence of PII also hinders anonymous reviews and peer feedback, crucial for unbiased educational assessments. Traditionally, PII detection and removal have relied heavily on manual review, a time-consuming and error-prone process. Current methods of manually screening for PII are costly and scale poorly, creating a bottleneck in educational research and tool development.

This project proposes an automated system to detect and remove PII from educational texts, thereby safeguarding student data, supporting the advancement of learning technologies, and enhancing data privacy and utility. In this project, we use advanced machine learning and deep learning models to handle large volumes of data, significantly reducing the reliance on manual screening processes and accelerating the accessibility of data for educational research. This initiative will not only safeguard student privacy but also facilitate the broader sharing and collaboration of educational data, thereby enhancing the resources available for educational research and development.

2.1 Problem Definition

There is a critical need for automated solutions that can process large datasets quickly and accurately. Current automated methods, such as those based on Named Entity Recognition (NER), are often tailored to detect standard PII like emails and phone numbers but struggle with

less structured data like student names or ambiguous identifiers. This limitation underscores the need for a more sophisticated approach that can adapt to the diverse nature of PII found in educational texts.

2.2 Project Objectives

To create a model / automated system that uses state-of-the-art machine learning techniques, including deep learning models like LSTM and DeBERTa, to identify and remove a wide array of PII types from textual data. We want a solution that can be scaled up to handle data from multiple sources and formats, facilitating faster processing times and reducing operational costs associated with data cleaning.

3. Related Work and Existing Tools

The current landscape of PII detection has seen various approaches, ranging from manual review processes to automated systems leveraging artificial intelligence and natural language processing capabilities. Notable tools in the market include:

- **Varonis**: Focuses on data security and protection, offering solutions for data governance and threat detection, but less specialized in unstructured data, long text formats, and educational texts.
- **Presidio (Microsoft)**: An open-source framework to identify and anonymize PII in text, which can be customized for specific use cases such as educational data.
- **Amazon Comprehend** and **Google Cloud Data Loss Prevention (DLP)**: Both offer robust services for detecting PII across various data types with the power of cloud computing, but can be costly and require significant adaptation for educational contexts.

State-of-the-Art in NLP for PII Detection:

- **Named Entity Recognition (NER) Technologies**: spaCy, NLTK, and Stanford NLP. These libraries provide foundational NER capabilities but often require additional training or customization to accurately detect and classify educational PII.
- **BERT and RoBERTa**: Transformer-based models that have been adapted for PII detection, offering superior performance due to their deep contextual understanding.

Many of the existing solutions are designed for general-purpose PII detection and are often not optimized for the specific challenges presented by educational datasets, such as essays and informal text submissions. These systems generally perform well with structured data and clear PII indicators but struggle with the ambiguity and less structured nature of educational texts. Challenges such as distinguishing between different types of names (student vs. cited author) or detecting less obvious PII like student IDs within narrative text remain significant hurdles for current technologies.

This project has focused on enhancing these models' capabilities to address the nuances of PII in educational texts, such as distinguishing sensitive personal names from non-sensitive ones.

Work by Devlin et al. on BERT and its derivatives has set a high standard for context-aware processing that can be crucial for accurate PII detection.

4. Methodology and Technical Approach

This section illustrates the comprehensive approach employed in developing an automated system for the detection and removal of Personally Identifiable Information (PII) from educational texts. It encapsulates the entire workflow from data procurement and preprocessing to the deployment of advanced machine learning models and continuous system refinement.

4.1 Data Collection and Augmentation

The project leverages a dataset consisting of approximately 6,000 student essays from a massively open online course. These essays are tokenized using the SpaCy English tokenizer to ensure that the text data is broken down into manageable and analyzable pieces. The tokenization process involves dividing the full text of each essay into discrete tokens (words or punctuation), which are then labeled according to a BIO (Beginning, Inner, Outer) format to indicate the presence and structure of PII.

Dataset Structure:

- **Document ID**: An integer identifier for each essay.
- **Full Text**: The complete text of the essay stored in UTF-8 format.
- **Tokens**: A list of strings, each representing a token derived from the essay.
- **Trailing Whitespace**: A list indicating whether each token is followed by whitespace.
- **Labels**: Each token is labeled with one of the following PII categories using the BIO tagging format: NAME_STUDENT, EMAIL, USERNAME, ID_NUM, PHONE_NUM, URL_PERSONAL, STREET_ADDRESS, and O (indicating other words containing no PII data).

Observations from the initial dataset indicated a significant imbalance in the representation of PII labels, with a predominance of non-PII tokens ("O" labels) overwhelming the PII-labeled tokens. The original dataset was also characterized by a significant prevalence of essays without any Personally Identifiable Information (PII), reflecting real-world scenarios where such data may not always be present. Specifically, the dataset included approximately 5,862 essays that did not contain any PII. The distribution of essays with various types of PII was as follows:

- NAME_STUDENT: 891
- EMAIL: 24
- USERNAME: 5
- ID_NUM: 33
- PHONE_NUM: 4
- URL_PERSONAL: 72

- STREET_ADDRESS: 2
- OTHER (non-PII): 5862

To rectify this imbalance and enhance the training data's diversity, the training data was augmented with an additional 4,000 essays. These additional essays were not only sourced but also synthetically generated using advanced techniques. Utilizing Large Language Models (LLMs), the project generated realistic text that mimics student writing while incorporating varied and contextually appropriate PII. Additionally, the Faker package was employed to systematically inject synthetic PII into essays, ensuring a controlled distribution of PII types such as names, email addresses, and phone numbers. This strategic augmentation was aimed at balancing the dataset by increasing the presence of underrepresented PII types and reducing the proportion of non-PII texts. The enriched dataset provided a more robust foundation for training the detection algorithms, significantly enhancing their ability to identify and categorize diverse PII entities accurately.

4.2 Data Preprocessing and Structuring

The expanded dataset, comprising approximately 10,000 essays, was preprocessed with the following:

- **Tokenization**: Each essay is tokenized using SpaCy's English tokenizer, converting the continuous text into a structured format of discrete tokens.
- **Labeling and Annotation**: Tokens are annotated with PII labels using the BIO (Beginning, Inside, Outside) tagging format. This method labels the start of a PII entity with "B-", any continuation with "I-", and non-PII tokens as "O".

4.3 Feature Extraction

This crucial step involves extracting meaningful features from the preprocessed text data, which are instrumental in training the machine learning models effectively. The feature extraction techniques differ based on the model being used, and they significantly impact the model's ability to discern and classify PII accurately from the text.

In this project, we focused on three distinct models for PII detection starting from simpler binary classification models to more advanced deep learning and transformer-based models. The specific models that were explored are: Logistic Regression, Long Short-Term Memory (LSTM), and Decoding-enhanced BERT with Disentangled Attention (DeBERTa). Each model was chosen for its unique strengths and suitability to handle different aspects of text data processing and PII detection:

- **Logistic Regression**: This model was selected for its efficiency and effectiveness in handling binary classification problems. It serves as a strong baseline that can quickly

process text data when transformed into a suitable format like TF-IDF, providing a quick and interpretable model for initial evaluations.

- **LSTM (Long Short-Term Memory)**: Given the sequential nature of text, LSTMs are ideal for capturing the context within sentences or phrases that traditional models might miss. This capability makes LSTMs particularly useful for detecting PII, which often depends on the context in which terms appear.
- **DeBERTa (Decoding-enhanced BERT with disentangled attention)**: A more advanced transformer-based model, DeBERTa was chosen for its sophisticated architecture that enhances the BERT model with disentangled attention. This feature allows it to more accurately understand the relationships between words regardless of their position in the text, making it highly effective for complex PII detection tasks that require understanding nuanced language patterns.

Each of these models requires specific feature extraction techniques that maximize their performance and effectiveness in identifying PII within texts:

4.3.1 Feature Extraction for Logistic Regression

1. **TF-IDF Vectorization**: TF-IDF stands for "Term Frequency-Inverse Document Frequency". It is a statistical measure designed to gauge the significance of a word within a document within a collection or corpus. It serves as a weighting parameter in various applications such as information retrieval, text mining, and user modeling. The TF-IDF score increases with the frequency of the word within the document but is balanced by its prevalence across the entire corpus. This adjustment accounts for the common occurrence of certain words. It transforms text documents into a TF-IDF feature matrix, effectively highlighting words that are important within specific documents but less common across the entire dataset. This feature is particularly useful for identifying unique identifiers or rare names that may indicate Personally Identifiable Information (PII). By focusing on distinctive words within documents, TF-IDF aids logistic regression models in effectively discerning between PII and non-PII elements.
2. **N-Grams**: By considering sequences of words, n-grams capture more contextual information than single words, enhancing the model's ability to recognize patterns indicative of PII. Including bi-grams or tri-grams (sequences of 2-3 words) can help capture more context around each word, which is useful for identifying PII patterns like full names, addresses, or complex identifiers.
3. **Part-of-Speech (POS) Tagging**: Incorporating grammatical information, POS tagging helps differentiate the role of each word in a sentence, aiding the logistic regression in understanding context that might indicate PII. By incorporating POS tags as features, the model can use grammatical structures to better identify PII. For example, proper nouns (tagged as NNP in POS tagging) are more likely to be names or other PII than other parts of speech.

4.3.2 Feature Extraction for LSTM

1. **Word Embeddings:** In this implementation, we utilize GloVe embeddings to transform words into dense vector representations. These embeddings capture the semantic meanings of words, enabling the LSTM to process text sequences and comprehend contextual nuances. This is particularly advantageous in PII detection, as the model can discern subtle differences in word usage across various contexts, which is crucial for accurately identifying personal information. We load these embeddings from a pre-trained GloVe model, ensuring each word in our dataset is represented by a 100-dimensional vector, aligning closely with its semantic properties.
2. **Embedding Matrix:** We construct an embedding matrix that maps each word in our vocabulary to its corresponding GloVe vector. This matrix is then used to initialize the weights in the LSTM's embedding layer, making the representations trainable. This step allows the model to fine-tune the embeddings based on the specifics of the task at hand, which enhances its ability to recognize PII within different textual environments.
3. **Tokenization and Padding:** The texts are tokenized, converting them into sequences of integers where each integer represents a unique word. This standardizes the text input for the model. Subsequently, we apply padding to these sequences, ensuring they all share the same length. This uniformity is essential for batch processing by the LSTM, as it requires fixed-length input vectors. Padding the sequences to a consistent length of 300 ensures that the model treats longer and shorter texts equivalently, maintaining the integrity of sequence data processing across varying text lengths.
4. **Contextual Features**: While the primary focus of feature extraction is on word embeddings and sequence handling, contextual information can also be indirectly captured through the model architecture. By using bidirectional GRU layers, the model can gather context from both previous and upcoming words, allowing it to better understand the placement and significance of each word within a sentence. This is particularly useful for identifying specific PII markers that may depend on their context, such as the positioning relative to punctuation or capitalization, which might indicate the start or end of a PII segment.

These feature extraction techniques are integral to optimizing the LSTM model's performance in detecting PII. By leveraging both linguistic content and structural cues, the model enhances its predictive accuracy, making it highly effective for PII detection tasks.

4.3.3 Feature Extraction for DeBERTa

For DeBERTa, advanced tokenization techniques were crucial for preparing text data for processing by the model:

1. **Advanced Tokenization**: DeBERTa employs a tokenizer that converts raw text into a format suitable for the model. This includes breaking down text into tokens, converting these tokens into numerical IDs, and generating attention masks. This process is vital for the model to accurately interpret and analyze the text.

    a. **Token-to-ID Mapping**: Each token is mapped to a unique identifier, allowing DeBERTa to utilize pre-trained embeddings that encapsulate extensive contextual information from vast language corpora. This is crucial for understanding complex language patterns and detecting nuanced instances of PII.
    b. **Attention Masks**: These are used to manage padding and differentiate actual content from padded content, ensuring that the model's attention mechanism focuses on meaningful data. This precision is essential when the model processes variable-length texts, which is common in real-world PII detection scenarios.
2. **Handling of Special Tokens**: DeBERTa processes special tokens (like [CLS], [SEP], etc.) that are used to aggregate sentence-level representations and separate different pieces of text. These tokens play a critical role in tasks like classification and entity recognition, directly impacting the model's ability to identify and delineate PII entities within the text.
3. **Positional Encodings**: Unlike some earlier transformer models, DeBERTa enhances the standard positional encodings with disentangled attention, which allows the model to more accurately assess the relationships between words based on their positions in a sentence. This feature is particularly useful in identifying PII, which may depend heavily on the contextual position of tokens.

These tokenization and encoding steps are integral to DeBERTa's ability to effectively process and analyze text for PII detection, leveraging the model's sophisticated architecture to enhance prediction accuracy and contextual understanding.

4.4 Model Development

This section details the development process of the machine learning models used for the automated detection and removal of Personally Identifiable Information (PII) from educational texts. It involved the implementation of three distinct machine learning approaches: Logistic Regression, Long Short-Term Memory (LSTM), and DeBERTa. Each model was tailored to leverage unique characteristics of text data, allowing for comprehensive analysis and effective identification of PII.

4.4.1 Logistic Regression

Logistic Regression was utilized as an initial approach due to its simplicity and effectiveness in binary classification tasks. This model serves as a baseline for our PII detection system. The model was trained using features derived from TF-IDF vectorization, which helped highlight important terms within the texts. This approach was beneficial for identifying clear-cut instances of PII based on term frequency and distinctiveness. We utilized the TF-IDF vectorization to transform the text data into a high-dimensional space where each dimension corresponds to a

word or n-gram, weighted by its significance across the corpus. This model was trained to classify segments of text as either PII or non-PII based on these features.

The simplicity of Logistic Regression makes it a robust baseline for comparison with more complex models. It provides a quick and interpretable model for identifying potential PII, making it a valuable tool for preliminary data analysis.

The model was trained using a standard logistic regression algorithm with regularization to prevent overfitting. Other training techniques such as cross-validation to tune hyperparameters and validate the model's performance on unseen data were also employed.

4.4.2 Long Short-Term Memory (LSTM)

LSTMs are particularly adept at processing long sequences of data, making them suitable for detecting PII embedded in complex sentence structures. By understanding the context in which words appear, LSTMs provide a deeper analysis of the text, which is critical for accurately identifying nuanced forms of PII.

The LSTM model was explored to capture the sequential nature of text, which is crucial in understanding context and dependencies between words in sentences. The model used word embeddings as primary input features, which encapsulate semantic relationships between terms. Contextual features and sequence padding were also incorporated to enhance the model's ability to process variable-length text data efficiently.

4.4.3 Decoding-enhanced BERT with Disentangled Attention (DeBERTa)

DeBERTa's ability to understand contextual nuances within large blocks of text makes it exceptionally powerful for tasks requiring deep linguistic understanding. DeBERTa utilizes a transformer-based architecture, which is particularly adept at handling the complexities of natural language due to its attention mechanisms. It also includes disentangled attention mechanisms for handling of context and sequence relationships for precise identification of PII across varied and complex text structures.

This model was particularly effective due to its ability to process words in relation to all other words in a text, regardless of their position. The use of advanced tokenization, attention mechanisms, and positional encodings allowed DeBERTa to achieve a high level of accuracy in PII detection.

This model was fine-tuned on our dataset, leveraging pre-trained weights that have been trained on a vast corpus of text, making it robust in capturing nuanced language patterns. It involved using the Hugging Face Transformers library to implement DeBERTa, which simplifies the usage of transformer models. The model was fine-tuned by adjusting the top layers to better fit the educational context of our data, with special attention to differentiating types of PII.

4.5 Metrics and Performance Evaluation

The evaluation of the models developed for detecting and removing Personally Identifiable Information (PII) from educational texts is critical to ensure their effectiveness and reliability. For this project, several metrics were used to assess the performance of the Logistic Regression, LSTM, and DeBERTa models. These metrics are particularly selected to emphasize the importance of not missing any PII data, which is crucial for maintaining privacy and compliance with data protection regulations.

4.5.1 Key Metrics

1. **Recall**: This metric is critical for PII detection systems because it measures the ability of the model to identify all relevant instances of PII. A high recall rate is imperative in this context as failing to detect PII can have serious privacy implications. It is defined as the proportion of true positives (correctly identified PII) to the total actual positives (all real PII in the text).
2. **Precision**: Precision measures the accuracy of the PII that the model detects. While it is secondary to recall in this application, high precision ensures that the system does not over-flag content as PII, which can be disruptive and lead to unnecessary data sanitization.
3. **F1 Score**: The F1 score is the harmonic mean of precision and recall, providing a single metric that balances both. It is particularly useful when you want to find a balance between recall and precision.
4. **F-beta Score (F_β)**: Given the higher importance of recall in PII detection, the F-beta score is calculated, which adjusts the F1 score to weigh recall more heavily than precision. The formula for the F-beta score is:

$$F_\beta = (1 + \beta^2) \cdot \frac{precision \cdot recall}{(\beta^2 \cdot precision) + recall}$$

   In this project, β is set to 5, emphasizing greater importance of recall over precision. This adjustment is crucial because the cost of missing PII is generally much higher than the cost of false positives in educational contexts.
5. **Accuracy**: While not as critical for this specific application, accuracy measures the overall effectiveness of the model across all classifications. It provides a general sense of how often the model is correct, both in identifying PII and non-PII.

4.5.2 Performance Evaluation Strategy

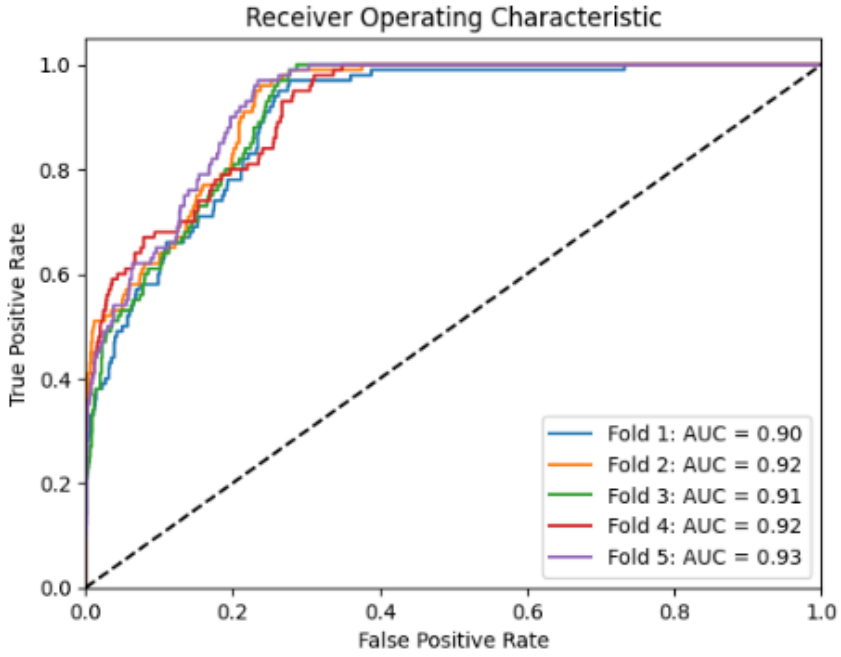
- **Data Splitting**: The dataset was split into training, validation, and test sets. The models were trained on the training set, parameters were tuned using the validation set, and the final model performance was evaluated on the test set.

- **Cross-Validation**: To ensure the models are robust and not just tuned to a specific subset of data, k-fold cross-validation was used during the training process. This technique provides a more comprehensive assessment of model performance.

4.5.3 Visualizations and Analysis

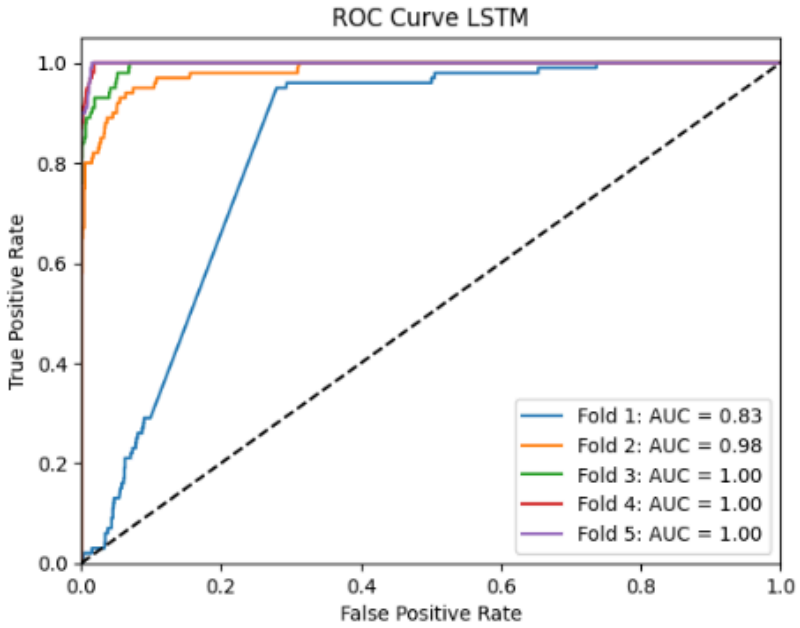
- **ROC Curves**: These plots are crucial for visualizing the trade-off between the true positive rate (sensitivity) and the false positive rate (1-specificity) for Logistic Regression, LSTM and DeBERTa models. The area under the curve (AUC) provides a useful measure of the model's ability to distinguish between classes.

    - **ROC Curve for Logistic Regression (along with performance metrics)**



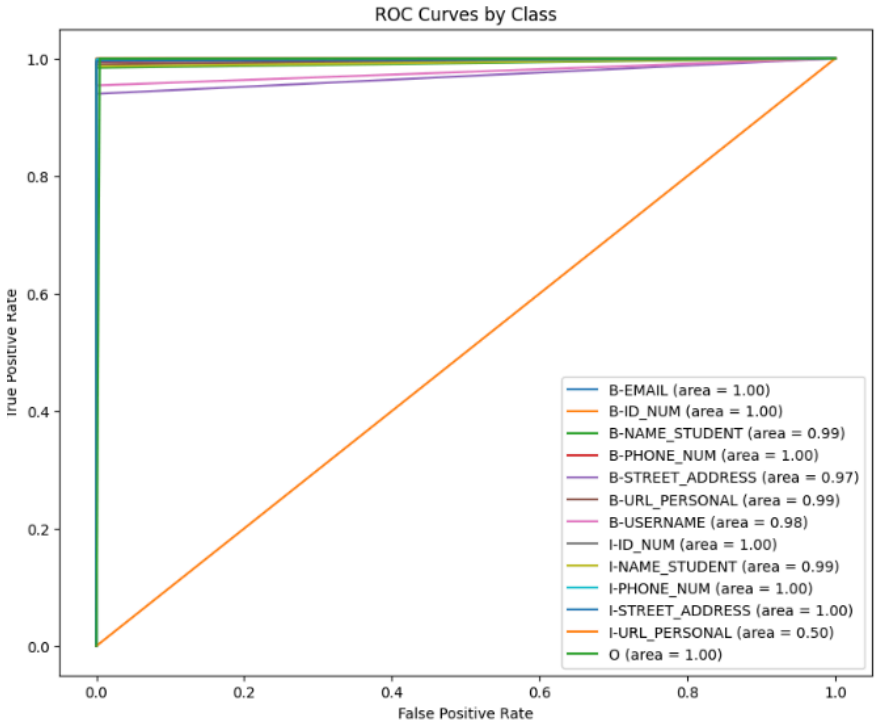

```
Average MCC: 0.6002239412968219
Average Recall: 0.584
Average Precision: 0.597736979518266
Average Accuracy: 0.6319262511412426
Average F1 Score: 0.5821714158558122
Average ROC AUC: 0.914925242849678
```

- **ROC Curve for LSTM**

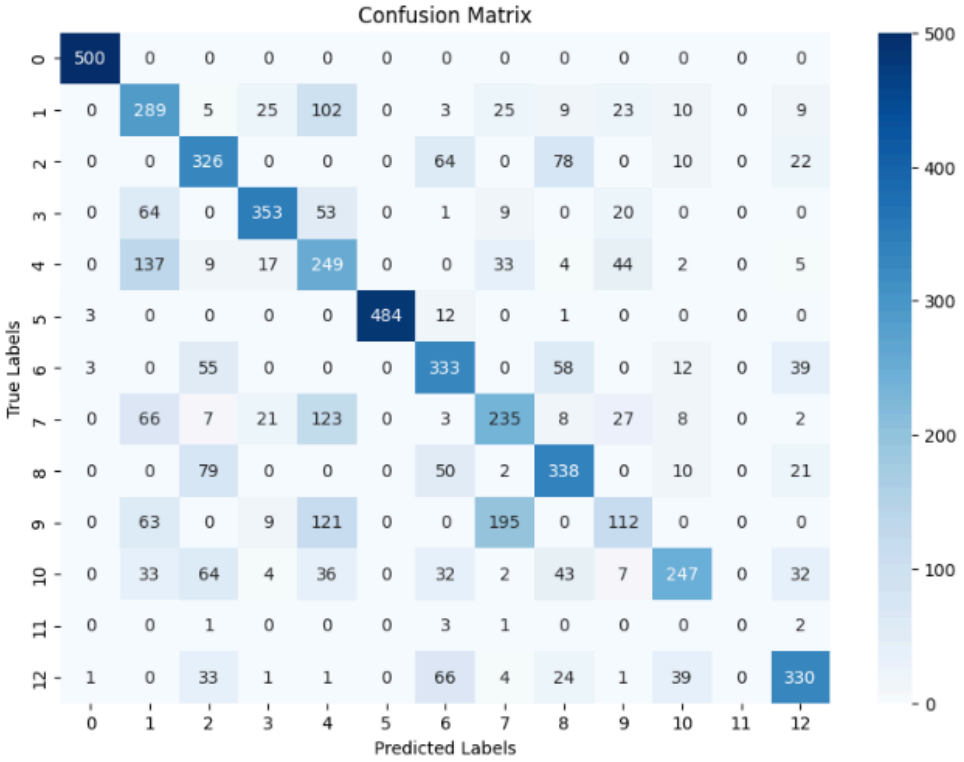



```
LSTM Average MCC: 0.7922225445540583
LSTM Average Recall: 0.758153846153846
LSTM Average Precision: 0.7980083157046504
LSTM Average Accuracy: 0.803925874306076
LSTM Average F1 Score: 0.7593539434446891
LSTM Average ROC AUC: 0.9617019975241119
```

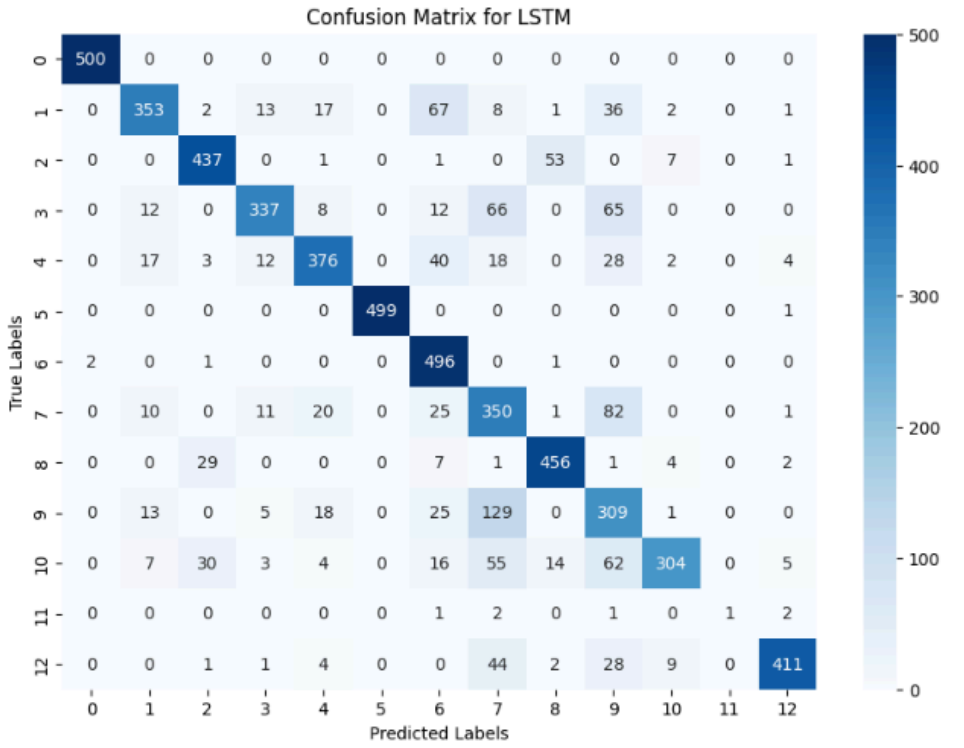
- **ROC Curve for DeBERTa (by Class)**



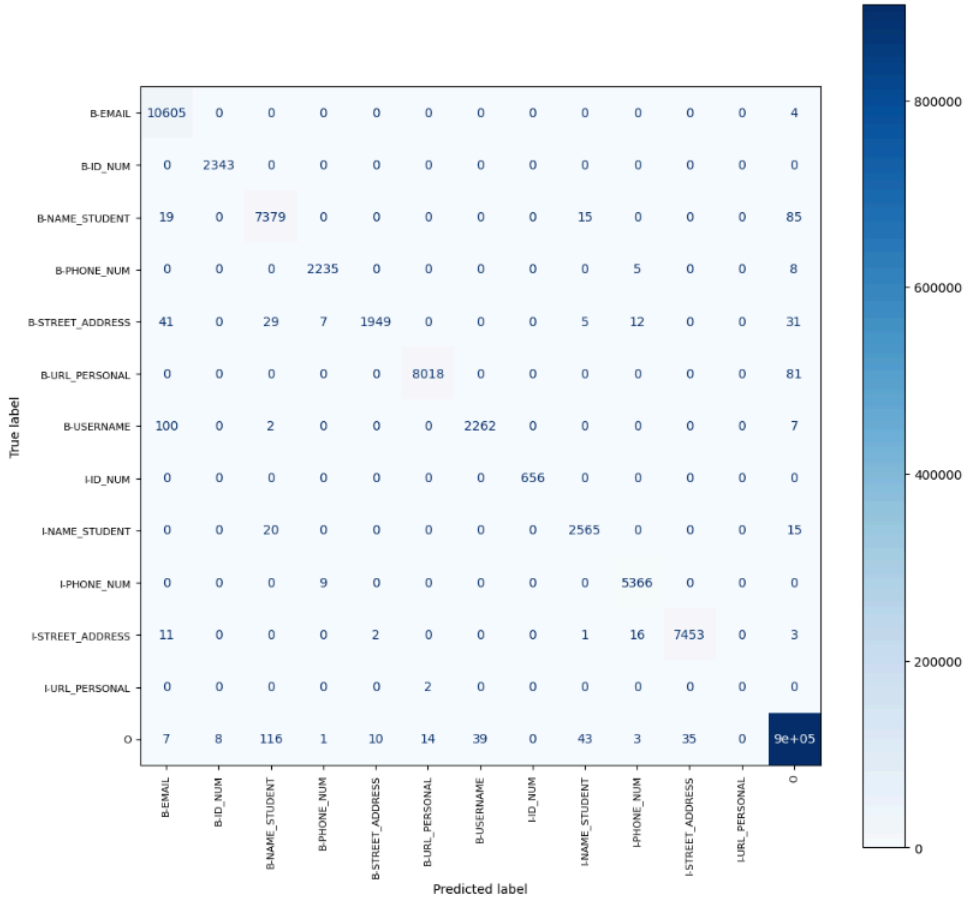
- **Confusion Matrix:**
  - **Confusion Matrix for Logistic Regression**



  - **Confusion Matrix for LSTM**



- **Confusion Matrix for DeBERTa**



- **Performance and Loss Metrics for DeBERTa**: The training and validation loss over several epochs show how well DeBERTa learns from the training data and generalizes to new data. These plots help identify issues like overfitting or underfitting and guide further tuning of the model parameters. The below results provide an insight into the model's learning process, illustrating the convergence of loss over time and highlight the effectiveness of the learning rate and other hyperparameters.

Epoch	Training Loss	Validation Loss	Recall	Precision	F1
1	0.005900	0.005283	0.982396	0.985771	0.982525
2	0.002100	0.005096	0.985622	0.985929	0.985634
3	0.001700	0.004880	0.986669	0.986306	0.986655

4.6 Model Comparison and Selection

A detailed analysis of the above three models was done for determining the most suitable machine learning model for detecting and removing Personally Identifiable Information (PII) from educational texts. Each model was compared on criterias such as performance metrics, computational efficiency, scalability, and ease of integration.

4.6.1 Comparison Criteria

1. **Performance Metrics**: Evaluation focused on Recall, Precision, F1 Score, and F-beta Score, with a particular emphasis on the F-beta Score to prioritize recall. This emphasis ensures that the model selected minimizes the risk of missed PII, which is critical in educational contexts.
2. **Robustness**: The ability of the models to handle diverse and complex data structures typical of educational texts was assessed. This includes evaluating how well each model adapts to different styles of writing and varied formats of PII.
3. **Scalability**: The capacity of each model to efficiently process large volumes of data was considered. This is crucial for practical deployment, especially when dealing with extensive educational databases.
4. **Computational Efficiency**: The resource requirements for training and deploying each model were analyzed. Models that require less computational power are preferred for scenarios where real-time data processing is needed.
5. **Ease of Integration**: The complexity of integrating each model into existing educational data processing systems need to be evaluated. This includes considerations of maintenance, adaptability to changes in data privacy laws, and the need for continuous model training.

4.6.2 Model Evaluation Outcomes

- **Logistic Regression** offered a solid baseline with its quick processing times and decent accuracy. However, it struggled with the contextual dependencies crucial for PII detection in complex text data.
- **LSTM** improved on contextual understanding due to its sequential data processing capability, leading to better performance in capturing nuanced language patterns associated with PII. The increased computational complexity, resources, and slower training times posed challenges.
- **DeBERTa** was the top performer across almost all metrics, particularly excelling in recall and F-beta scores. Its sophisticated architecture allows for an in-depth understanding of contextual relationships within texts, making it highly effective for the complex task of PII detection. Despite its higher computational demands, the benefits of its advanced contextual understanding and high accuracy outweigh the operational costs.

4.7 Validation and Testing

The validation and testing phase was essential for ensuring the DeBERTa model's reliability and effectiveness in detecting PII within educational texts. This section describes the methodologies used to validate and test the model, ensuring it meets the necessary standards for accuracy and robustness.

4.7.1 Validation Approach

1. **Holdout Validation**: The dataset was split into training, validation, and testing sets. The DeBERTa model was initially trained on the training set, with hyperparameters tuned based on performance metrics observed on the validation set. This approach helps in optimizing the model settings without leaking test set information.
2. **Cross-Validation**: To further ensure the model's robustness and generalizability, k-fold cross-validation was employed. This method involves dividing the entire dataset into k smaller sets (or folds), then training the model k times, each time using a different fold as the test set and the remaining as the training set. This technique provides a thorough insight into the model's performance across various subsets of data.

4.7.2 Testing Methodology

1. **Performance Metrics**: Upon finalizing the model configuration through validation, the DeBERTa model was subjected to the test set, which had been completely isolated from the training process. The key metrics such as Recall, Precision, F1 Score, and particularly the F-beta Score were calculated to evaluate the final model performance.
2. **Real-World Scenario Testing**: To mimic real-world operational conditions, the model was also tested on a separate set of data collected from actual educational environments that were not part of the initial dataset. This test helps to assess how well the model performs in genuinely practical settings.

4.7.3 Visualization of Results

- **Confusion Matrix**: A confusion matrix was used to visualize the model's performance, showing the distribution of true positives, true negatives, false positives, and false negatives. This helps in understanding the effectiveness of the model in classifying PII accurately.
- **ROC Curve and AUC**: The Receiver Operating Characteristic (ROC) curve and the Area Under the Curve (AUC) were plotted. These visualizations help assess the model's ability to distinguish between classes of PII and non-PII effectively.

This validation and testing strategy ensures that the DeBERTa model is not only theoretically sound but also practical and effective for real-world applications. The rigorous evaluation

demonstrates the model's capability to perform under varied conditions, confirming its suitability for deployment in educational settings.

5. Challenges and Future Scope

5.1 Challenges Encountered

1. **Data Diversity and Quality**: One of the major challenges was dealing with the diversity and varying quality of data within educational texts. Collecting a diverse set of educational texts and augmenting the dataset to balance the distribution of PII and non-PII labels posed significant challenges. This required ensuring that the added data maintained a high quality and relevance to the context.
2. **Handling False Positives**: One of the trickier aspects of PII detection was minimizing false positives without compromising on the sensitivity needed to catch all true PII instances. Balancing this precision-recall trade-off required continuous adjustments and testing.
3. **Technical Learning Curve**: The project demanded a deep understanding of advanced computational technologies. Learning to configure and effectively utilize CUDA, CuDNN, and GPUs for training complex language models was a steep learning curve that was both challenging and time-consuming.
4. **Intensive Training Requirements**: The computational demand and time required to train especially complex models like DeBERTa were significant. Each training session consumed considerable resources, and optimizing the training process was a persistent challenge.
5. **Model Fine-Tuning**: Due to time constraints, there were limitations on how extensively the model could be fine-tuned. Additional iterations to adjust learning rates and other parameters might have led to better model performance.
6. **Advanced Metrics and Visualization**: Incorporating more sophisticated visualization tools and metrics, such as those offered by Weights & Biases, for monitoring model performance during training was identified as an area for future improvement. Further learning in ML Ops could enhance the efficiency and effectiveness of the model deployment and maintenance processes.

5.2 Future Scope

1. **Multilingual and Cultural Variability**: With the increasing diversity of student populations, there is a growing need for PII detection systems that can accurately handle multilingual data and understand cultural nuances in names and personal information. Extending the model’s capabilities to include multilingual PII detection would greatly enhance its applicability in diverse educational settings across the globe.

2. **Better Training Techniques**: Future projects could explore advanced model training techniques such as transfer learning or semi-supervised learning approaches to improve efficiency in model training.
3. **Real-Time Detection Capabilities**: Developing the capability for real-time PII detection would significantly enhance the utility of the system, allowing for immediate action to be taken as sensitive data is identified.
4. **User Feedback in Training Loop**: Establishing mechanisms to integrate user feedback directly into the model training and updating process could help in fine-tuning the model based on practical user experiences.
5. **Model Optimization Techniques**: Investigating further model optimization strategies such as model pruning, quantization, and knowledge distillation could help in deploying these advanced models more efficiently.

6. Conclusion

This project tackled the critical issue of detecting and removing Personally Identifiable Information (PII) from educational texts, enhancing data privacy and compliance with regulatory standards. Employing advanced machine learning techniques, especially the DeBERTa model, showcased significant achievements in accurately identifying PII. The integration of this model into educational systems would be a significant step forward in automating data protection measures and reducing reliance on manual processes. However, the project also highlighted the complexities and challenges involved in handling such sensitive data. The insights gained from this initiative not only contribute to the technical field of data privacy but also underscore the importance of continuous development and adaptation of privacy-enhancing technologies in education.

7. References

- GloVe: Global Vectors for Word Representation: https://nlp.stanford.edu/projects/glove/
- CUDA, NVIDIA. "CUDA Toolkit Documentation." https://developer.nvidia.com/cuda-toolkit
- Weights & Biases. "Experiment Tracking with Weights and Biases." https://wandb.ai/site
- Kaggle Competition for Dataset: https://www.kaggle.com/competitions/pii-detection-removal-from-educational-data
- External Additional Datasets: https://www.kaggle.com/competitions/pii-detection-removal-from-educational-data/discussion/470921, https://www.kaggle.com/datasets/alejopaullier/pii-external-dataset/data
- Discussion section for various insights: https://www.kaggle.com/competitions/pii-detection-removal-from-educational-data/discussion
- W&B experiment tracking: https://www.kaggle.com/competitions/pii-detection-removal-from-educational-data/discussion/472740
- Understanding the F-beta score: https://www.kaggle.com/code/eliocordeiropereira/understanding-the-f-beta-score

- Xinyu Wang, Yong Jiang, Nguyen Bach, Tao Wang: Automated Concatenation of Embeddings for Structured Prediction
- Oscar Sainz, Iker García-Ferrero, Rodrigo Agerri, Oier Lopez de Lacalle, German Rigau, Eneko Agirre: GoLLIE: Annotation Guidelines improve Zero-Shot Information-Extraction
- Devlin, J., et al., 2018, BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding
- Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, Luke Zettlemoyer: Deep contextualized word representations (NAACL 2018)

8. Project Links

1. Github Link: https://github.com/lvdpsingh/pii-data-detection
2. Link to Jupyter Notebook (.ipynb): https://drive.google.com/file/d/1kzSqLrJjlQHdTWShFOjAyVjgcSlHzPHK/view
3. Link to Google Drive: https://drive.google.com/drive/folders/1zGdHH9QPz96epgdRQvnecPcCgI1_reUL?usp=drive_link
4. Link to Final Report: https://drive.google.com/file/d/1hB80c7WTrYrBVz185GiQg1hbeP-1zHUa/view?usp=sharing
5. Link to Final Presentation: https://docs.google.com/presentation/d/1xr6lSTaVF8b4qztj3YsqHl1wyvW5ho5MpMEAwTd1UN4

9. Acknowledgments

Special thanks to Dr. Desmond Lun for his guidance throughout this project, and to all references which contributed to this project.