applied system innovation

MDPI

Article

Detecting Personally Identifiable Information Through Natural Language Processing: A Step Forward

Luca Mainetti [1,*] **and Andrea Elia** [2]

[1] Department of Engineering for Innovation, University of Salento, 73100 Lecce, Italy
[2] Faculty of Engineering, University of Salento, 73100 Lecce, Italy; andrea.elia3@studenti.unisalento.it
* Correspondence: luca.mainetti@unisalento.it

Abstract: The protection of personally identifiable information (PII) is being increasingly demanded by customers and governments via data protection regulations. Private and public organizations store and exchange through the Internet a large amount of data that include the personal information of users, employees, and customers. While discovering PII from a large unstructured text corpus is still challenging, a lot of research work has focused on identifying methods and tools for the detection of PII in real-time scenarios and the ability to discover data exfiltration attacks. In those research attempts, natural language processing (NLP)-based schemas are widely adopted. Our work combines NLP with deep learning to identify PII in unstructured texts. NLP is used to extract semantic information and the syntactic structure of the text. This information is then processed by a pre-trained Bidirectional Encoder Representations from Transformers (BERT) algorithm. We achieved high performance in detecting PII, reaching an accuracy of 99.558%. This represents an improvement of 7.47 percentage points over the current state-of-the-art model that we analyzed. However, the experimental results show that there is still room for improvement to obtain better accuracy in detecting PII, including working on a new, balanced, and higher-quality training dataset for pre-trained models. Our study contributions encourage researchers to enhance NLP-based PII detection models and practitioners to transform those models into privacy detection tools to be deployed in security operation centers.

Keywords: personally identifiable information (PII); data privacy; natural language processing (NLP); generative AI

Academic Editor: Patrícia Ramos

Received: 14 February 2025
Revised: 7 April 2025
Accepted: 15 April 2025
Published: 18 April 2025

Citation: Mainetti, L.; Elia, A. Detecting Personally Identifiable Information Through Natural Language Processing: A Step Forward. *Appl. Syst. Innov.* **2025**, *8*, 55. https://doi.org/10.3390/asi8020055

1. Introduction

Research Context

In recent years, the digital transformation introduced by all private sectors and driven by governments to obtain an even more efficient public administration has started the progressive digitization of the functional domains of many business processes. The increasingly widespread adoption of data-driven methods for making decisions and the availability of a high number of very advanced data analytics tools due to the adoption of artificial intelligence algorithms have increasingly pushed private and public companies to store on their own servers and/or on servers hosted by public clouds huge databases and information stores on which to base decisions for the governance of business processes. These data contain the personal information of users, employees, and customers in various forms, such as emails, unstructured texts, structured databases, text documents, spreadsheets, PDF files, images, audio, video, and so on. The privacy risks deriving from the consequences of violating these repositories are therefore increasingly high.

The financial effects of cyber-attacks on data (data breaches) have reached enormous proportions. According to the IBM Data Breach Report 2024 [1], the global average cost of data breaches in 2024 grew by an additional 10 percent over 2023 and reached $4.88 million. IBM estimates that the average cost savings for organizations that used AI-enabled security and automation extensively in prevention versus those that did not is about $2.22 million. In particular, in the United States, we are witnessing a rapid surge in the number of annual data breach incidents, with increasing consequences for both organizations and individuals, as stressed by the Data Breaches and Cyber Attacks USA Report 2024 [2]. These incidents not only lead to financial losses but also trigger a large number of lawsuits and penalties that introduce inefficiency into business processes. The situation is similar in the rest of the world; in particular, the European Union is paying more and more attention to updating the GDPR data privacy regulation [3], making it increasingly binding for private and public organizations.

In summary, the rapid expansion of cyberspace in response to the digital transition makes it challenging to continuously identify vulnerabilities in data protection systems and to adopt effective countermeasures for cybersecurity breaches. In particular, data threats constitute a significant risk for the security and confidentiality of personally identifiable information (PII), which involves a wide range of sensitive data, such as user accounts, names, addresses, social security numbers, financial records, passwords, medical records, and so on.

The general focus of our research is data protection to ensure the privacy of information. The specific research focus is the protection of PII, that is, any data that can be used to identify a specific individual, such as usernames, passwords, addresses, ID numbers, and other personal details.

This article is organized as follows: Section 1 provides an introduction outlining the research context. Section 2 presents a literature review on PII detection, highlighting the growing focus on privacy in the storage and communication of personal information. It also points out the current gaps in research and explains the main motivation behind our study. Section 3 describes in detail the process we followed to develop the artificial intelligence-enabled PII detection system. Section 4 presents the quantitative results from our experiments, including both the training and testing phases of the AI model. Section 5 discusses the results, compares them with a state-of-the-art model, addresses the limitations of our work, and outlines potential future research directions. Finally, Section 6 offers conclusions based on the findings.

2. Related Work and Reasons to Continue Research in PII Detection

2.1. Essential Literature Review

Over the years, corporate companies, healthcare industries, and other public companies have become prime targets for attackers seeking PII, in particular, exploiting the vulnerabilities of social networks, mobile apps, and mobile networks [4]. As an example, more than ten years ago, Liu et al. [5] proposed a method to automatically detect PII in internet traffic using a combination of systematic expressions (such as regular expressions) and dictionary-based methods. As an improvement to the previous work, in 2019, Go et al. [6] introduced a Software-Defined Networking (SDN) and Network Function Virtualization (NFV)-enabled architecture to address the unintended disclosure of PII over network traffic, particularly on social media platforms. This research is highly relevant in today's environment, where both privacy breaches and data leaks on the internet, especially on social networks, are a significant concern. A study by Ren et al. [7] introduced a system called ReCon, which aims to tackle the issue of PII leakage in mobile network traffic. By using a supervised learning model, ReCon effectively identifies and blocks unintended

disclosure of PII that might be transmitted over mobile networks. As another application of learning techniques, in 2020, Noever [8] proposed a novel approach to identifying persons of interest (PoI) and PII within the context of corporate email datasets—specifically, the Enron Corpus. The study leverages ensemble learning techniques, including decision trees, support vector classifiers (SVCs), random forests, and neural networks, to analyze textual data from corporate emails and financial records, extracting both PII and PoI while also analyzing employee sentiment during a corporate crisis. Furthermore, the authors use techniques with multiple classifiers (as in [9]) to optimize performance. In their work, Barder et al. [10] focused on the critical issue of PII in the context of public health research, specifically when utilizing online geographic tools and big data tools. Their study addresses the potential risks associated with the handling and sharing of sensitive health data, particularly when such data are geographically tagged or integrated into public health analysis. Similarly, Alnemari et al. [11] explored the issue of protecting PII in the context of healthcare data, particularly focusing on critical infrastructure data (see [12]) and emphasizing innovative methods to balance privacy protection with the need for efficient data analysis. The study examines differential privacy, a prominent privacy-preserving technique, and compares it with other traditional anonymization techniques to protect PII in healthcare data. The authors provide a novel approach for protecting PII in large datasets, demonstrating that multiple attribute workload distribution can be more effective than traditional anonymization and differential privacy techniques. Onik et al. [13] addressed the protection of PII on mobile devices. The work introduces an intelligent risk classification model based on the permissions requested by Android applications to classify vulnerable PII associated with mobile device owners. By analyzing app permissions, the model helps to assess the risks of exposing sensitive data, such as contact numbers, social graph data, emails, location data, biometric identifiers, and unique device IDs. Majeed et al. [14] proposed an innovative approach to anonymizing PII, with a dual focus on privacy and data utility. The goal of their work is to develop a method that protects user privacy by anonymizing PII while maintaining the utility of the data for subsequent analysis or publication. The proposed scheme considers both vulnerability and diversity as important factors in the anonymization process. Venkatanathan et al. [15] investigated the implications of online PII disclosure on social media platforms, specifically examining how public and private sharing behaviors can lead to unintended privacy leaks. In the study by Tesfay et al. [16], the authors delved into the challenges associated with the discovery of PII and other privacy-sensitive information within unstructured texts. The focus of their work is on improving the detection of privacy-revealing information by leveraging ontologies, natural language processing (NLP), and learning approaches. Liu et al. [17] presented a novel approach for identifying PII based on user behaviors in HTTP network traffic. The research focuses on addressing the challenges of detecting PII in massive, complex datasets, specifically from HTTP traffic that contains large volumes of user interactions with web services and applications. By employing a decision tree-based classification system and optimizing it for PII detection, the authors focused on user behavior as a key indicator for identifying sensitive information. The work by Vishwamitra et al. [18] presented a collaborative approach to protect PII when sharing photos in online social networks. The work focuses on addressing the privacy risks associated with photo sharing, which often involves multiple parties and metadata, including the photo owner and various viewers, each with different access rights. Observing that the volume of personal information users store on their mobile devices has driven significant growth over the years and the increasing risk that mobile applications may transmit PII to external servers due to the fact that users may have agreed to allow their PII to be transmitted by applications directly or through applications' use of third-party libraries to the cloud, in 2018 Conti et al. [19] defined the

"PII transmission problem". Wongwiwatchai et al. [20] pointed out that there is no easy way to know whether or not an application transmits PII. If the detection of PII transmission is tackled using heavyweight techniques, such as static code analysis and dynamic behavior analysis, it requires anywhere from several minutes to hours of testing and analysis per application. The authors proposed a significant advancement in the privacy protection landscape for Android applications by focusing on lightweight static analysis, offering a practical and efficient method for identifying when sensitive PII is being transmitted by Android apps. This method balances performance with accuracy, making it suitable for large-scale use in app stores or security audits. It also provides an additional layer of security and helps developers and users mitigate the risks of unintended privacy leaks through mobile applications. A number of studies have investigated the social, economic, and financial consequences of data breaches in private and public institutions, highlighting that repercussions may affect not only the institutions themselves but also their customers, employees, and society at large. For example, Lee et al. [21] argued that adopting a routine activity approach to understanding these risks can help institutions better design security strategies and responses to minimize the damage caused by breaches. Tavana et al. [22] offered a robust decision-making framework to evaluate and select the most appropriate blockchain–IoT technologies for supply chain security, emphasizing the importance of preventing data breaches and cyberattacks. The use of an interval-based multi-criteria decision-making model allows for a nuanced and adaptable approach that accounts for both uncertainty in data and expert judgment. The authors suggested that the integration of blockchain and IoT provides an effective solution for securing sensitive supply chain data. In a previous study, Tavana [23] highlighted the growing interest and potential for big data-driven analytical models in the field of cybersecurity. The author discussed how these models can significantly enhance decision-making processes by providing advanced insights into potential risks and vulnerabilities, ultimately helping organizations anticipate and mitigate cyber threats, including data breaches, before they occur. Ayyagari [24] discussed the various causes and types of data breaches and their respective categories. From a technical point of view, data breaches can be due to unintended disclosure (DISC), hacking or malware (HACK), fraud involvement around account information (CARD), insiders accessing sensitive information (INSD), loss or stolen assets or records (PHYS), loss of portable media (PORT), and loss of digital equipment like a desktop computer (STAT). At the same time, data breaches can also be classified according to the type of industry, including Business, Financial, and Insurance services (BSF); Business Retail/Merchant entities (BSR); Educational Institutions (EDU); Government and Military (GOV); Healthcare and Medical providers (MED); Nonprofit (NGO); and Other Businesses (BSO). Stiennon [25] proposed the Breach Level Index (BLI) metric, which is widely used to categorize the severity of data breaches. The BLI provides a standardized way of evaluating breaches based on specific criteria, offering insight into the scale and impact of data security incidents. Ayaburi [26] explored the broader impact of data breaches on organizations, emphasizing how different measures, such as the BLI, can assist organizations in understanding and managing the aftermath of breaches. The cybersecurity risk quantification and classification framework proposed by Zadeh et al. [27] is a significant contribution to improving how organizations assess and respond to data breaches. By providing a structured, dynamic approach to quantifying breach severity, the framework fills a critical gap in the existing literature and offers practical tools for organizations to make more informed risk mitigation decisions. This framework not only enhances cybersecurity preparedness but also helps organizations respond in a more agile and effective manner to the increasing threat of data breaches in the digital age.

2.2. The Need to Continue Research in PII Detection

From the literature review that we presented in the previous section, two aspects emerge with certainty: (i) In recent years, attention has intensified regarding aspects of privacy in the storage and communication of personal information; (ii) however, many research challenges need to be addressed before a universal framework for the protection of PII can be agreed upon, regardless of the technical nature of the information being protected or the industry to which it pertains. All the scientific works mentioned in the previous section agree on these two elements.

A good and updated synthesis of the research gap that still needs to address the protection of PII is well represented by Pool et al. [28]. Although the systematic analysis conducted by the authors is focused on the domain of public healthcare, many of the issues highlighted are common to other sectors, such as finance, education, and supply chains. The study developed a useful model to explain the multifaceted nature of data breaches while providing an important tool for better understanding and mitigating these risks. The study outlines six key research directions: (1) Multilevel Analysis: examining health data breaches from different organizational, technological, and social levels; (2) Novel Methods: the introduction of advanced methodologies for better prediction and detection of breaches; (3) Information Systems Theory: contributions to the theoretical foundations of information systems, particularly in healthcare contexts; (4) Stakeholder Analysis: understanding the roles and responsibilities of various stakeholders involved in the protection of personal health data; (5) Under-explored Themes: topics that have received limited attention, such as the emotional and psychological impacts of breaches; (6) Boundary-Breaching Opportunities: exploring opportunities that may emerge from a better understanding of the dynamics that lead to breaches, such as the role of cross-sector collaboration.

It is clear that the common ground for the six research directions listed above is the ability to detect PII, particularly in unstructured contexts.

2.3. The Main Aim of the Research

The main objective of our research is to improve techniques for recognizing PII, with a focus on increasing their dependability. Specifically, we aim to enhance the performance of PII detection algorithms in unstructured contexts where information does not follow predefined patterns and may escape traditional data security methods. These unshaped contexts, which include free texts, informal communications, and documents with varied language, pose significant challenges because there are no fixed patterns to help security systems automatically and accurately identify PII.

The challenges outlined by Pool et al. [28] highlight the difficulty of operating in such environments and the need to develop more sophisticated approaches to effectively detect PII, even when personal information is disguised, partially altered, or hidden in various data formats. To address these challenges, our research employs NLP techniques, which allow us to analyze, understand, and process natural language texts. NLP methods, such as Named Entity Recognition (NER), semantic analysis, and contextual understanding, are particularly useful in improving the ability of algorithms to recognize PII, even in complex situations and contexts where information may not follow standardized formats and/or semantic contexts.

3. Materials and Methods

3.1. Research Methodology

This chapter provides a detailed description of the activities carried out to analyze the performance of a PII detection system aimed at ensuring both data protection and compliance with the General Data Protection Regulation (GDPR). The entire process required

an in-depth analysis of the data and the methodologies employed, involving a series of systematic and rigorous steps. The work has been divided into six main phases, the first four of which are presented in the following paragraphs and the remaining two in the next chapter (Results). These phases are crucial in determining the system's effectiveness. The research methodology is illustrated in Figure 1, along with the model names, techniques, validation strategies, and performance metrics used during the research.

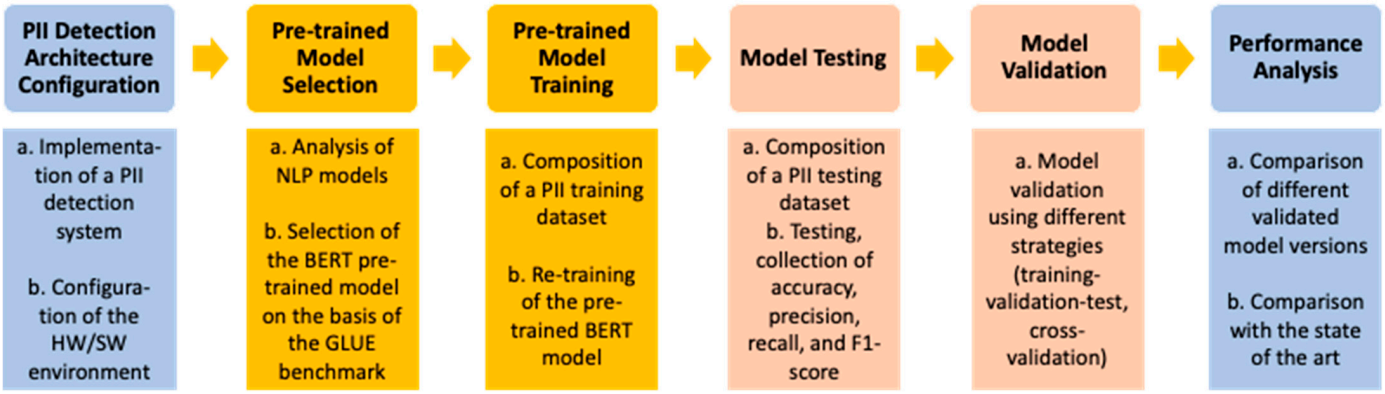


Figure 1. Research process.

3.2. Configuration of the PII Detection Architecture

In this initial phase, the primary goals of the experiment were defined, namely the analysis of the system's ability to recognize personal information while respecting privacy and GDPR guidelines. A controlled testing environment was designed in which the necessary technological resources (hardware and software infrastructure) and tools for data collection and processing were selected or developed. The architecture was designed to execute performance evaluations in a reproducible and reliable manner. Furthermore, evaluation metrics were established to monitor the effectiveness of the system, such as accuracy, precision, recall, and F1-score, which measure the correctness, the ability to identify all PII, and the balance between the two aspects, respectively.

We based the PII detection architecture on NLP models that, in turn, rely on transformers. Transformers were introduced by Google in 2017 in the paper "Attention Is All You Need" as a new neural network architecture based on the self-attention mechanism, which is particularly suited for text understanding [29]. Previously, neural networks processed language by generating representations in fixed- or variable-length vector spaces. These networks typically began with representations of individual words and then aggregated information from neighboring words to infer the sentence's context. In contrast, transformers perform a small and constant number of steps, applying a self-attention mechanism that models the relationships between words, regardless of their position in the sentence. The transformer model can be trained to focus on specific words within a sentence, allowing it to better understand the discourse context. Notably, it can distinguish homographs by considering their different meanings based on context. This self-attention mechanism enables the model to capture long-range dependencies between words, which is especially beneficial for tasks like PII recognition, where the meaning of words often depends on their relationships with other words, independent of their position in the text [30].

In the context of our experimental work, the detection software system was implemented in Python 3.11 using the PyTorch, TensorFlow, and CUDA libraries. The software architecture was structured into four distinct classes: DetectionModelTrainer, DataHandler, PIIDataLoader, and PIIDataSet, as illustrated in Figure 2. The responsibilities of the different modules are explained as follows:

- The *PiiDataLoader* class handles the loading of data from CSV files, performs "tokenization," and divides the data into batches.
- The *DataHandler* class is responsible for managing and preparing the data by splitting them into training (80%), test (10%), and validation (10%) sets, ensuring that the data are appropriately structured and formatted.
- The *PiiDataset* class associates the "tokenized" data with corresponding labels, indicating which portions correspond to personal information.
- The *DetectionModelTrainer* class is responsible for training, evaluating, testing, and managing the pre-trained model, which is provided by the Hugging Face Transformer library [31].
- Finally, the *Main* class orchestrates the other classes to control the execution of the PII detection flow.

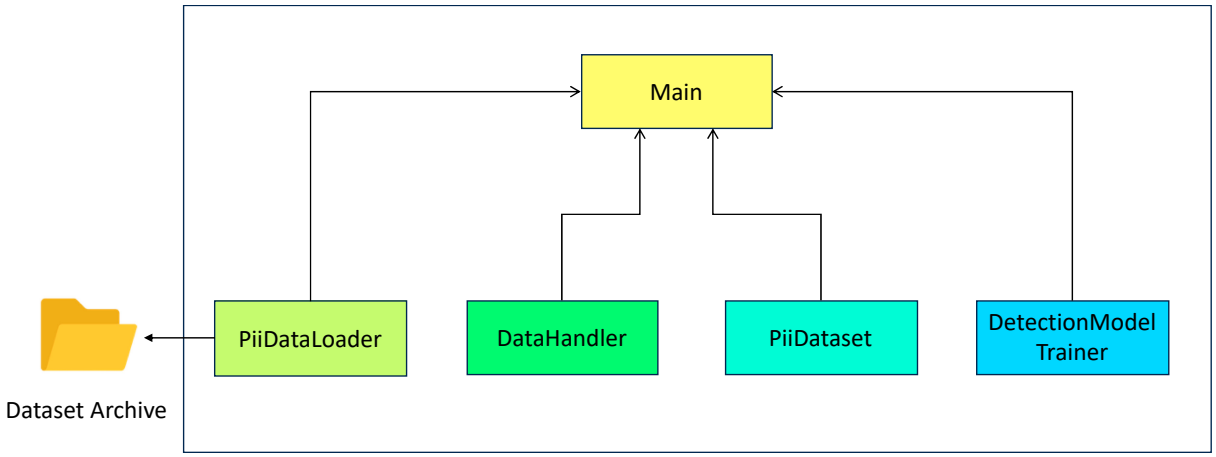


Figure 2. PII detection software architecture.

3.3. Pre-Trained Model Selection

As we know, the selection of machine learning and deep learning models for personal information recognition is a critical phase, as the models need to be not only accurate but also efficient in operation. After a thorough analysis of available solutions, traditional machine learning models, such as neural networks, as well as more advanced deep learning models using NLP techniques, were considered. The models' accuracy was balanced with computational efficiency to prevent the system from being too slow or resource-intensive in a production environment. Special attention was given to the models' ability to handle complex and unstructured data, such as free text containing sensitive information.

Driven by the increasing demands of NLP, several pre-trained transformer models have been proposed, including BERT (Bidirectional Encoder Representations From Transformers), RoBERTa, DeBERTa, GPT, and ELECTRA, which are thoroughly described by Joshi et al. in [32] and by Clark et al. in [33]. To maintain full control over the data generated by the PII recognition experiment and to evaluate its performance, we decided to use the pre-trained BERT model, also based on the high performance claimed by the scientific community in its use for linguistic tasks, as quantitatively reported by Devlin et al. in [34], where the performance of models such as ELMo, OpenAI GPT, BiLSTM + ELMO + Attn, Pre-OpenAi SOTA, CVT, and CSE were compared with BERT ($BERT_{BASE}$ and $BERT_{LARGE}$), using key benchmarks such as GLUE and SQuAD v1.1.

For example, the GLUE benchmark requires the evaluation of performance across the following eight specific tasks:

- MNLI (Multi-Genre Natural Language Inference), which evaluates the model's ability to determine the relationships between a pair of sentences;
- QQP (Quora Question Pairs), which assesses whether two questions are semantically equivalent;
- QNLI (Question Natural Language Inference), which determines if a paragraph of text contains sufficient information to answer a specific question;
- SST-2 (Stanford Sentiment TreeBank), which classifies the sentiment of sentences as positive or negative;
- CoLA (Corpus of Linguistic Acceptability), which determines the grammatical acceptability of sentences;
- MRPC (Microsoft Research Paraphrase Corpus), which recognizes paraphrased sentences;
- RTE (Recognizing Textual Entailment), which determines the entailment relationship between sentences.

According to Devlin et al. [34], in Table 1, we provide a summary of the GLUE benchmark results, demonstrating BERT's superior performance in text classification. We observe that the authors in [34] focused not explicitly on the detection of PII but on text classification models that form the basis of PII recognition.

Table 1. GLUE benchmark results, as reported by Devlin et al. [34].

Corpus **Model**	**MNLI-(m/mm)** [1] **392k** [2]	**QQP** **363k**	**QNLI** **108k**	**SST-2** **67k**	**CoLA** **8.5k**	**STS-B** **5.7k**	**MRPC** **3.5k**	**RTE** **2.5k**	**Avg** **-**
Pre-OpenAI SOTA	80.6/80.1	66.1	82.3	93.2	35.0	81.0	86.0	61.7	74.0
BiLSTM + ELMo + Attn	76.4/76.1	64.8	79.8	90.4	36.0	73.3	84.9	56.8	71.0
OpenAI GPT	82.1/81.4	70.3	87.4	91.3	45.4	80.0	82.3	56.0	75.1
$BERT_{BASE}$	84.6/83.4	71.2	90.5	93.5	52.1	85.8	88.9	66.4	79.6
$BERT_{LARGE}$	86.7/85.9	72.1	92.7	94.9	60.5	86.5	89.3	70.1	82.1

[1] MNLI-m = MNLI-Matched (the corpus contains sentence pairs where the premise and hypothesis come from the same genre); MNLI-mm = MNLI-Mismatched (the corpus contains sentence pairs where the premise and hypothesis come from different genres). [2] Number of examples.

During the pre-training process, BERT utilizes two key tasks: (i) the Masked Language Model (MLM), where certain words in a sentence are masked, and the model must predict the missing words; and (ii) the Next Sentence Prediction (NSP), where the model determines whether two sentences are logically connected. To effectively perform the NSP task, a bidirectional approach to text analysis is essential. This is achieved using 12 Encoders in $BERT_{BASE}$ or 24 Encoders in $BERT_{LARGE}$. These Encoders enable bidirectional understanding via the multi-headed self-attention mechanism, which compares each word in the sentence with both preceding and succeeding words. In addition, the Encoders ensure computational efficiency and scalability by supporting parallel computation, particularly benefiting from batch processing.

For input data processing, words are represented as a sequence using three types of embeddings: (i) Token Embedding, where the token [CLS] is added at the beginning and [SEP] at the end of the sentence; (ii) Segment Embedding, to help the model differentiate between distinct sentences within a single input, assigning a different parameter to each sentence; and (iii) Position Embedding, to represent the position of each word in the sequence. Once these embeddings are summed, the resulting final embedding is used as input for the BERT model.

During pre-training, 15% of the tokens in the input are randomly masked as [MASK]. The model's objective is to predict which word corresponds to the original word replaced by [MASK]. The masked input passes through the Encoder layers, where the self-attention mechanism is applied at each layer to enrich the representation of each token. It then

proceeds through the classification layers, generating a vector of values (logits), which represents the words that the model believes are most appropriate to replace the [MASK] token. The attention score for each token analyzed is associated with the logits. Finally, a softmax function is applied to convert the logits into probabilities.

For the NSP task, a sample set is used containing 50% related sentences and 50% unrelated sentences. The corresponding embedding vectors are input into the Encoders and then passed to the classification layer, where a softmax function generates the probability that one sentence is related to the other.

3.4. Pre-Trained Model Training

Once the BERT models were selected, the (re-)training process was initiated. During this phase, the data used for training were preprocessed to ensure the quality and diversity of the information, including different types of PII (e.g., phone numbers, email addresses, names, credit card numbers). (Re-)training was performed using a large and diverse dataset to prevent overfitting, created by combining different publicly available datasets.

The adopted training methodology followed the classic training-validation-test framework, with 80% of the dataset used for training, 10% for validation, and 10% for testing. To ensure the model performs well on new, unseen data, we applied a cross-validation technique. Specifically, we used both k-fold cross-validation and stratified k-fold cross-validation methods. The entire process is summarized in Figure 3.

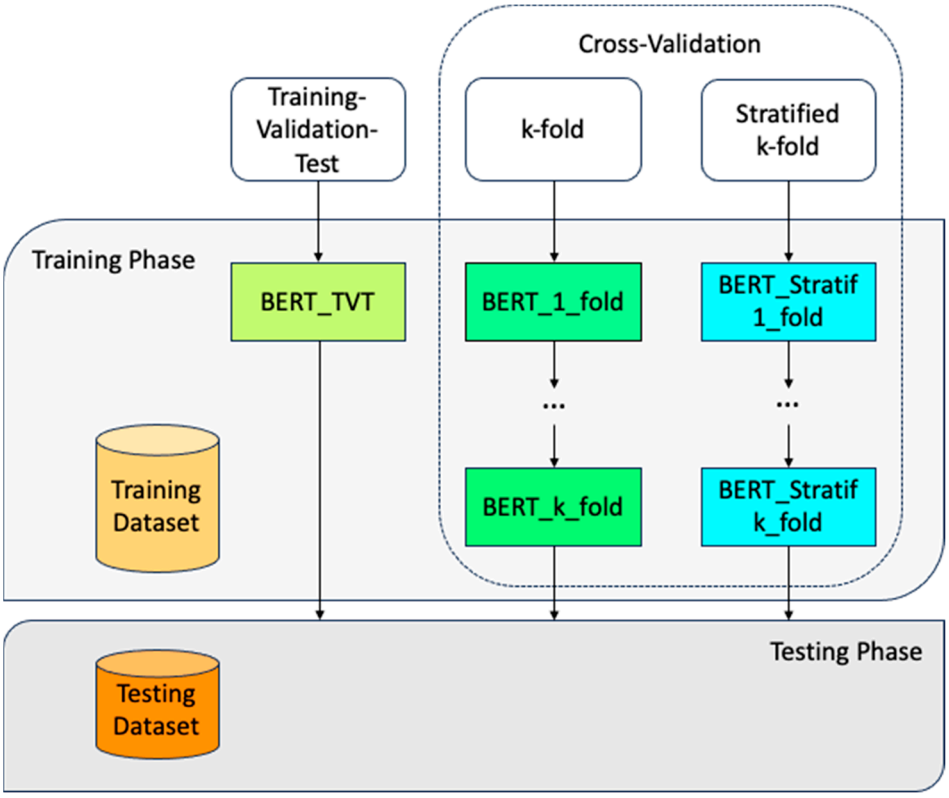


Figure 3. (Re-)training process of the BERT pre-trained model.

Our initial focus was on English-language datasets to maximize the model's applicability across various applications. After reviewing publicly available datasets with diverse categories of PII, we selected the "pii-masking-200k" dataset [35]. This dataset is notable for its extensive coverage, containing 54 different PII classes spread across 229 use cases and domains, including business, education, training, psychology, and law. It also includes

five types of interactions—formal and casual conversations, emails, and others—providing a broad spectrum of real-world scenarios for PII identification.

The original pii-masking-200k dataset spans four languages: English, French, German, and Italian. For model training purposes, we extracted a subset consisting of approximately 43,000 records. These records were carefully selected to ensure a balanced representation of various PII types, enabling the model to generalize across diverse contexts.

The pii-masking-200k dataset is likely already masked; however, additional preprocessing steps were necessary to optimize it for BERT. We applied several data preprocessing techniques to ensure the raw text was appropriately formatted and cleaned for efficient processing by BERT. Specifically, we performed lowercasing, eliminated extraneous characters (such as redundant spaces or malformed punctuation), and applied sentence segmentation. These preprocessing functions were readily available through the Python 3.11 libraries used in our experimental setup, allowing for rapid implementation.

We then utilized the Hugging Face Transformers tokenizer, which tokenizes the sentences into sub-word units compatible with BERT. Additionally, we leveraged a pre-trained NER model from Hugging Face to identify and label specific PII entities, including names, street addresses, phone numbers, and email addresses.

The significance of the pii-masking-200k dataset is widely recognized within the research community. In this regard, the interested reader can refer to previous works [36–39]. Holmes et al. [36] underscored the dataset's contribution to advancing automated PII identification systems, particularly through its use in international competitions. They complemented it with the Cleaned Repository of Annotated Personally Identifiable Information (CRAPII), which contains over 20,000 student essays. Wen et al. [37] also explored realistic experimentation with the dataset to better evaluate PII detection methods. Table 2 presents the PII classes contained in the pii-masking-200k dataset, along with their corresponding sensitivity levels (detailed later).

Another criterion for assembling the training dataset involved selecting PII from the pii-masking-200k dataset across different sensitivity levels: less sensitive (LS), sensitive (SS), and very sensitive (VS). We based these classifications on official documentation, particularly the GDPR [3]. This approach allowed us to train the model to recognize these categories and evaluate its performance when handling varying levels of sensitivity, ensuring data protection and privacy compliance. Table 2 provides the sensitivity level for each PII class considered.

To improve the quality of our training corpus and enhance model robustness, we supplemented the "pii-masking-200k" subset with records from an additional dataset that contains no PII. This balancing strategy was implemented to ensure the model can distinguish between contexts with and without PII effectively.

Specifically, we extracted 43,000 records (out of 50,000 available) from the "Generic-Sentiment" dataset [40], sourced from Kaggle. This dataset contains a diverse set of generic sentences from various domains, making it ideal for balancing the training data. The use of generic sentences ensures that the model is exposed to neutral, non-PII data, thus improving its accuracy in real-world scenarios where PII may or may not be present.

The final (re-)training dataset is perfectly balanced, with 50% of the records containing PII and the other 50% consisting of non-PII content, as shown in Figure 4. This balanced approach is essential for training a robust model capable of accurately identifying PII while avoiding overfitting to PII-containing examples.

Table 2. PII classes present in the pii-masking-200k dataset.

1.	Name {SS}	28.	Family member names {SS}
2.	Email {VS}	29.	Place of birth {SS}
3.	Phone number {VS}	30.	Citizenship {SS}
4.	Credit card number {VS}	31.	Gender {SS}
5.	Passport number {VS}	32.	Sexual orientation {SS}
6.	Social security number {VS}	33.	Race/ethnicity {SS}
7.	Date of birth {SS}	34.	Nationality {SS}
8.	Address {SS}	35.	Religious beliefs {SS}
9.	Zip code {SS}	36.	Political affiliation {SS}
10.	IP address {VS}	37.	Professional license number {SS}
11.	Bank account number {VS}	38.	Education history {SS}
12.	Routing number {SS}	39.	Employment history {SS}
13.	Driver's license number {SS}	40.	Income level {LS}
14.	License plate number {SS}	41.	Financial status {LS}
15.	Vehicle identification number {LS}	42.	Social media handles {LS}
16.	Bank account details {VS}	43.	Account numbers {VS}
17.	Medical record number {VS}	44.	Transaction history {VS}
18.	Health insurance number {VS}	45.	Digital signature {VS}
19.	Employee number {SS}	46.	Purchase history {SS}
20.	Student ID {SS}	47.	Subscription information {SS}
21.	Government ID {SVS}	48.	Health information (e.g., conditions, treatments) {VS}
22.	Fingerprint {VS}	49.	Emergency contact information {LS}
23.	Voiceprint {VS}	50.	Insurance policy number {VS}
24.	Face image {VS}	51.	Academic records {SS}
25.	Geolocation {VS}	52.	Tax identification number {LS}
26.	Biometric data {VS}	53.	License key or serial number {SS}
27.	Mother's maiden name {SS}	54.	Location data (GPS) {VS}

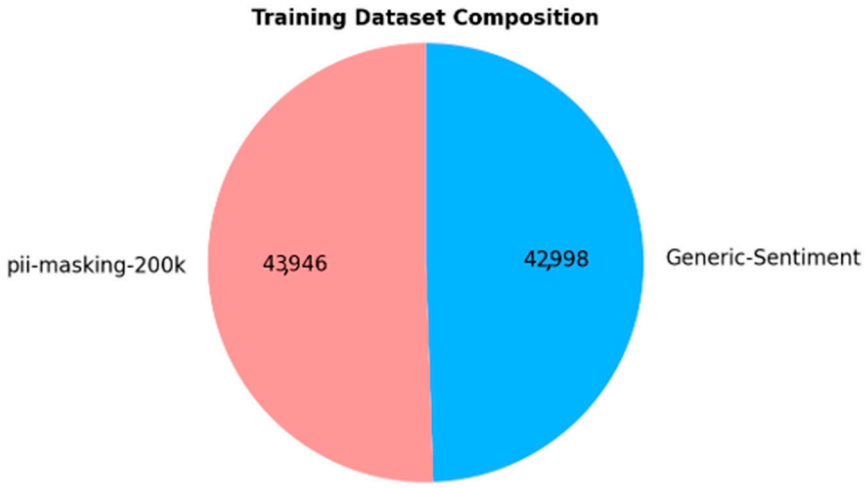


Figure 4. Composition of the (re-)training dataset.

The performances of the training phase are reported in detail in the next chapter (Results) using quantitative metrics.

3.5. Model Testing

After training was completed, the models underwent rigorous testing using a separate dataset, distinct from the training set. The goal of this test was to assess the system's performance on data that it had not encountered before. Various metrics, such as precision, recall, and F1-score, were used to evaluate the model's effectiveness. During this phase, the

impact of data variability on the model's behavior was also analyzed, and response times were considered to ensure the system could operate in real-time or near-real-time scenarios.

We put effort into creating a testing dataset, separate from the training dataset, that is truly representative. To do this, we collected data from the following four publicly available and well-documented sources:

- "pii-masking-43k" [41]: This dataset, released by Ai4Privacy, is similar to the larger pii-masking-200k dataset. We chose it because it is known for its reliable data. It has been used to optimize the Distilled BERT model, achieving impressive results: precision (99.86%), recall (99.89%), and accuracy (99.45%). We included it as the core part of the testing dataset since it provides different types of data compared to the larger pii-masking-200k dataset. In past research, the pii-masking-43k dataset has been used to fine-tune various models, including RoBERTa and GPT-2. The dataset provides pre-trained models, which serve as a starting point to build effective PII detection tools.
- "Dialog Dataset" [42]: This dataset contains chatbot conversations without any PII. We used it to add variety to our test data, selecting 3726 records to make sure the model could handle different formats of conversational data.
- "Movie Review" [43]: Known as the "IMDB Dataset of 50K Movie Reviews," this is a popular dataset for text classification. It is often used for sentiment analysis of movie reviews, with comments labeled as "positive" or "neutral." We checked that no PII was included, and we selected 10,000 records to make sure the model handles sentiment detection correctly while avoiding privacy issues.
- "Chat-GPT 4 Sentences" [44]: We asked GPT-4 to generate 1000 sentences without PII. This portion of the dataset evaluates the model's ability to handle artificially generated content, providing us insight into its performance with AI-generated data.

The composition of the testing dataset is illustrated in Figure 5, which highlights the four distinct data sources used.

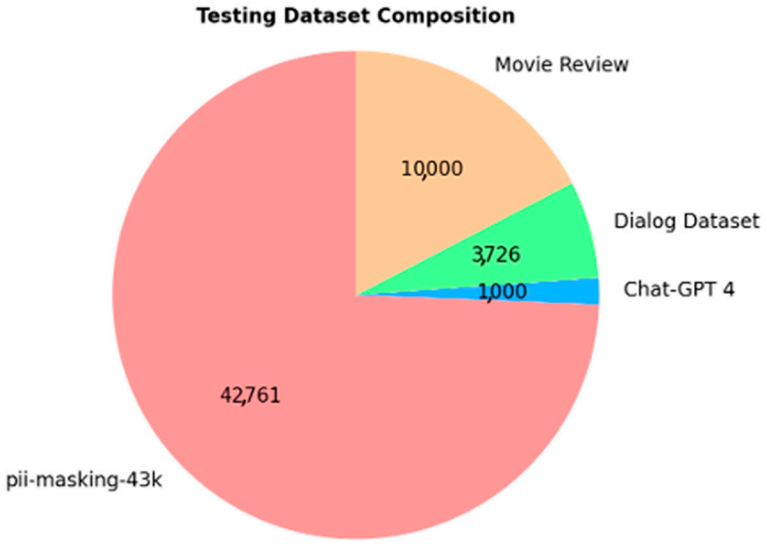


Figure 5. Composition of the testing dataset.

The performances of the testing phase are reported in detail in the next chapter (Results) using quantitative metrics.

4. Results

In this section, we present the results both for training and testing, as well as the evaluation metrics employed to assess model performance. In other words, we provide

the quantitative results we collected in the last three phases of our research process, with reference to Figure 1.

The training dataset was iteratively shaped by comparing the model's results with other models trained on a manually annotated dataset created by experts. In this phase, the model's accuracy in correctly identifying PII was assessed, as well as its ability to recognize personal information in more hidden or implicit forms, such as in unstructured texts or ambiguous contexts. The results from both validation and testing were analyzed in detail. A comprehensive review was conducted to identify potential issues, such as false positives or false negatives, which could compromise data protection. Additionally, patterns were examined that could reveal weaknesses in the models used, such as difficulties in recognizing specific categories of PII or handling ambiguous texts.

4.1. Experimental Results Obtained from Model Training

During the training process, we thoroughly analyzed the different models developed and their respective performances. The key parameters used during training were the following: number of epochs = 3, batch size = 64, and the number of folds = 5. These settings were critical in shaping the learning process and ensuring the models were properly evaluated. The experimental training environment consisted of the following hardware and software configuration:

- GPU: MSI GeForce RTX 4090 VENTUS 3X 24G OC;
- CPU: AMD Ryzen 9 7900X;
- RAM: Corsair DDR5 Vengeance 2 × 32 GB 5600;
- Disk: Crucial P3 Plus 4TB M.2 SSD;
- Motherboard: Asrock X670E Pro RS;
- Operating System: Ubuntu 24.

This setup provided the necessary computational power for the training of the models, particularly leveraging the GPU for high-performance parallel processing, which is crucial in deep learning tasks.

To compare the performance of the trained models, we utilized the following metrics, which allowed us to extract and quantify various performance indices:

- Training Loss: This metric represents the model's error as calculated on the training set. It is monitored throughout the training process and quantifies the incorrect predictions the model makes. A lower training loss indicates better performance in terms of fitting the training data.
- Validation Accuracy: This measures the percentage of correct predictions made by the model on the validation subset. The validation set is a separate subset of the training set (10%) used to evaluate how well the model can generalize during the training process. It helps to measure the model's performance beyond just the training data.
- Test Accuracy: This measures the percentage of correct predictions made by the model when evaluated on the test subset. The test set is a separate subset of the training set (10%) that is used to assess the model's overall performance and to provide a more detailed analysis of how well it generalizes to new, unseen data just after the training phase.

Tables 3 and 4 illustrate the trends in these performance indices as the training parameters are varied. The collected data provide quantitative evidence of how different configurations of the training process affect the model's ability to learn and generalize.

Table 3. Experimental results obtained during the training phase for training-validation-test.

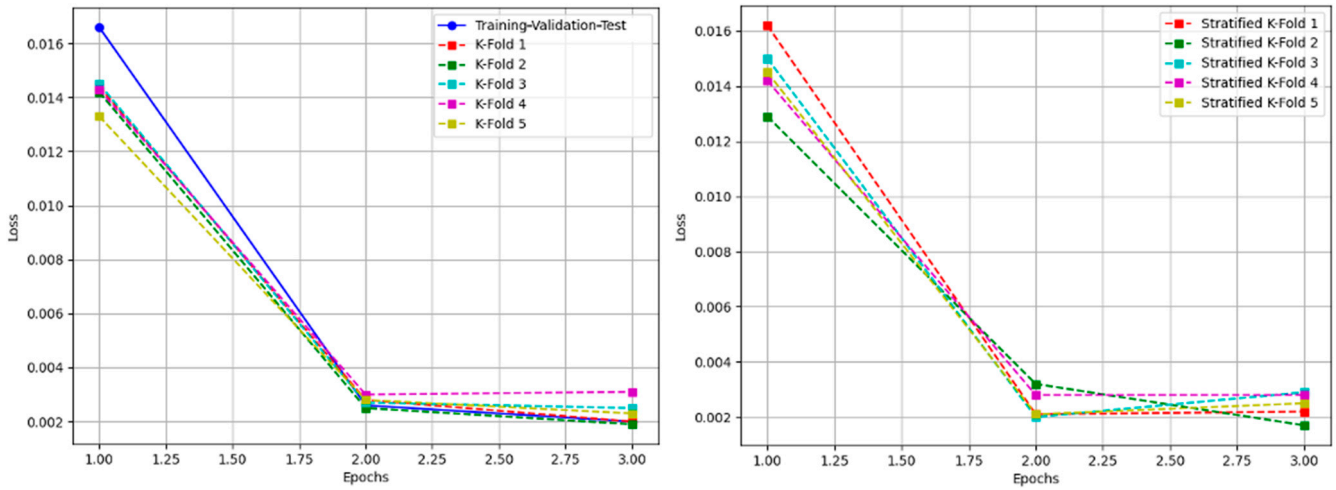
Model	Epoch	Training Loss	Validation Accuracy	Training Time	Allocated Memory	Test Accuracy
Training-Validation-Test	1	1.66%	99.93%	31.92 min	1689.06 MB	99.86%
	2	0.26%	99.92%			
	3	0.20%	99.85%			

Table 4. Experimental results obtained during the training phase for cross-validation.

Model	Epoch	Training Loss	Training Time	Allocated Memory	Test Accuracy
K-fold 1	1	1.44%	31.39 min	1688.71 MB	99.92%
	2	0.28%			
	3	0.20%			
K-fold 2	1	1.42%	30.78 min	1689.42 MB	99.86%
	2	0.25%			
	3	0.19%			
K-fold 3	1	1.45%	30.55 min	1689.58 MB	98.91%
	2	0.27%			
	3	0.25%			
K-fold 4	1	1.43%	30.75 min	1690.58 MB	99.84%
	2	0.30%			
	3	0.31%			
K-fold 5	1	1.33%	30.70 min	1690.91 MB	99.87%
	2	0.28%			
	3	0.23%			
Stratified k-fold 1	1	1.62%	30.59 min	1688.71 MB	99.62%
	2	0.21%			
	3	0.22%			
Stratified k-fold 2	1	1.29%	30.63 min	1689.42 MB	99.93%
	2	0.32%			
	3	0.17%			
Stratified k-fold 3	1	1.50%	30.54 min	1689.58 MB	99.78%
	2	0.20%			
	3	0.29%			
Stratified k-fold 4	1	1.42%	30.60 min	1690.58 MB	99.84%
	2	0.28%			
	3	0.28%			
Stratified k-fold 5	1	1.45%	30.52 min	1690.91 MB	99.91%
	2	0.21%			
	3	0.25%			

Tables 3 and 4 also provide data that help illustrate the computational complexity of the model during training. For each model we generated, we included the training time and memory usage. The hyperparameters used are as follows: optimizer = adamw_torch, learning rate = 5×10^{-5}, weight decay = 0.01, batch size = 64, number of training epochs = 3, and number of parameters = 340 M.

The images presented in Figure 6 show the evolution of the training parameters for the different models trained using both the training-validation-test, k-fold (a), and stratified k-fold (b) cross-validation techniques. These figures provide an in-depth look at the impact of varying the training setup on model performance, with each figure representing a different fold or configuration.



(**a**) (**b**)

Figure 6. Evolution of the Training Loss with increasing epochs: (**a**) training-validation-test and k-fold cross-validation; (**b**) stratified k-fold cross-validation.

Figure 7 provides a comprehensive graphical comparison of the results, summarizing the performance differences between all trained models. This comparative analysis helps to highlight the strengths and weaknesses of each approach and allows for a more informed decision when selecting the optimal training method.

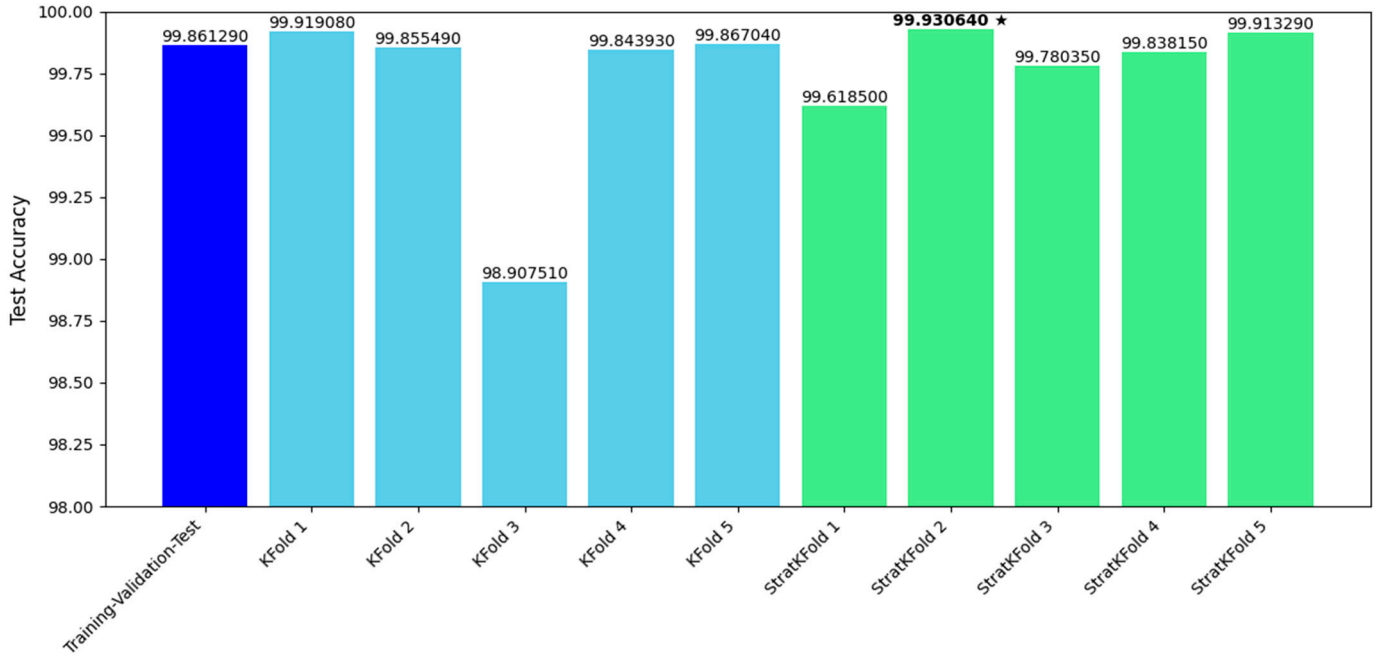


Figure 7. Performance comparison of the trained models based on Test Accuracy metrics. The dark star indicates the model with the highest Test Accuracy. Accuracy values are reported with five decimal digits to allow for clearer comparison.

By analyzing the data collected during the experimentation, we found that the maximum variation in performance across the models was 1.02313%. Despite this relatively small variation, the average test accuracy across all models trained using the k-fold cross-validation technique was 99.68%. For the models trained using the stratified k-fold technique, the average test accuracy was slightly higher, 99.82%.

It is worth noting that, in general, the models trained with k-fold cross-validation performed better overall. Among these, the model with the highest test accuracy—99.93%—was the stratified k-fold 2 model. This model exhibited the most robust generalization capabilities across different datasets and configurations.

4.2. Experimental Results Obtained from Model Testing

During the testing phase, we assessed the performance of the models using not just test accuracy but also other well-known metrics like precision, recall, and F1-score. The experimental results are summarized in Table 5. An analysis of the data in the table reveals that the model with the highest performance across all metrics is the one trained using the training-validation-test technique (this appears in the first row of Table 5). This model achieved the following results:

- Accuracy: 99.558%;
- Precision: 99.564%;
- Recall: 99.558%;
- F1 score: 99.559%.

Table 5. Experimental results obtained during the testing phase.

Model	Accuracy	Precision	Recall	F1-Score
Training-Validation-Test	99.558%	99.564%	99.558%	99.559%
K-fold 1	99.240%	99.262%	99.240%	99.244%
K-fold 2	98.904%	98.945%	98.904%	98.912%
K-fold 3	80.845%	89.393%	80.845%	82.219%
K-fold 4	97.968%	98.106%	97.968%	97.993%
K-fold 5	99.346%	99.361%	99.346%	99.349%
Stratified k-fold 1	98.480%	98.496%	98.480%	98.466%
Stratified k-fold 2	98.657%	98.723%	98.657%	98.669%
Stratified k-fold 3	95.953%	96.542%	95.953%	96.057%
Stratified k-fold 4	99.454%	99.512%	99.387%	99.485%
Stratified k-fold 5	99.417%	99.431%	99.417%	99.419%

These results are very strong, indicating that the model is performing exceptionally well in all aspects of classification.

On the other hand, the model with the worst performance, based on the same metrics, is the 3-fold model. This model showed the following results:

- Accuracy: 80.845%;
- Precision: 89.393%;
- Recall: 80.845%;
- F1 score: 82.219%.

As we can see, the 3-fold model is significantly less effective than the training-validation-test model, with noticeably lower scores in all the metrics.

To further illustrate these differences, we can refer to the four images in Figure 8. These images show the average values of accuracy, precision, recall, and F1-score for the models. From top to bottom, in a clockwise direction, we can observe the following trend:

- The model trained with the training-validation-test technique provides the most reliable results overall, with the stratified k-fold being a better alternative than the regular k-fold for ensuring balanced performance in classifying PII. The training-validation-test leads to more accurate and stable classification outcomes.

- The models trained with the stratified k-fold technique tend to have higher performance across all the metrics compared to the models trained with the standard k-fold method. This suggests that ensuring a balanced distribution of classes within the folds helps improve the model's performance in detecting PII.
- The stratified k-fold technique makes sure that each fold has a similar proportion of each class, which is especially important when dealing with unbalanced data. In contrast, with regular k-fold cross-validation, the data might not be as well balanced across the different folds, which could lead to poorer generalization and performance.

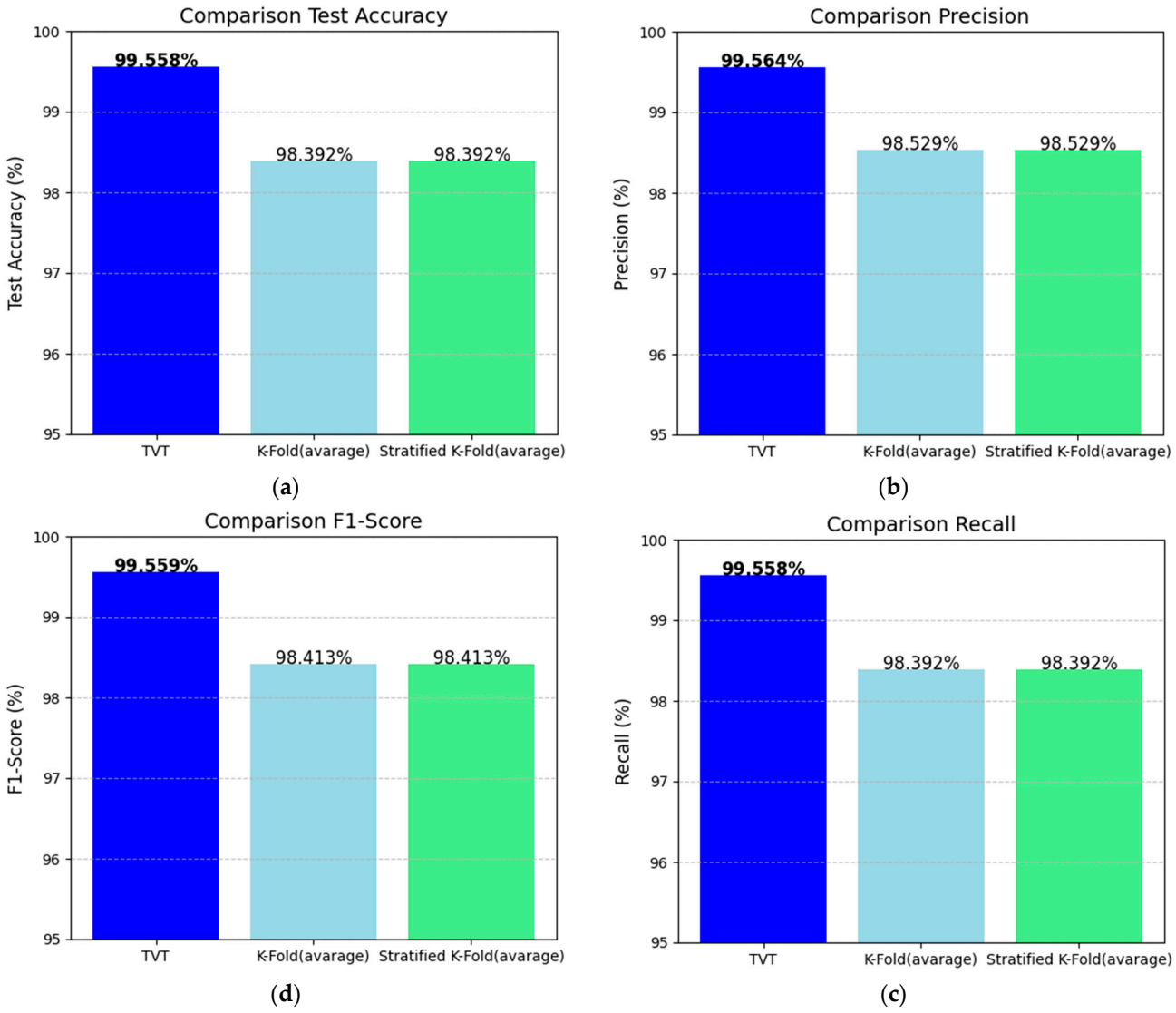


Figure 8. Comparison of the performance metrics obtained during the testing phase: (**a**) accuracy, (**b**) precision, (**c**) recall, and (**d**) F1-score. Values in bold indicate the best performance.

5. Discussion

In this section, we discuss the results, highlighting key findings. We also conduct a comparative analysis to determine which model provides the best performance to minimize the risk of exposing PII through effective detection capabilities. Furthermore, we delineate the limitations of our experimental study and highlight future research directions.

5.1. Evaluation

This study reveals that the training-validation-test approach, when combined with stratified k-fold cross-validation, is the most effective method for training models to detect

PII. The training-validation-test approach achieved the highest performance, with a test accuracy of 99.558% and an F1-score of 99.559%, showcasing strong generalization and robustness. This suggests that the model trained with this approach is not only accurate but also able to effectively handle unseen data, which is crucial for real-world applications where data continuously evolve.

In contrast, k-fold cross-validation showed more variability in its results. For instance, 3-fold cross-validation performed poorly in some cases, while other configurations performed better. This variability indicates that the performance of k-fold cross-validation depends heavily on how the dataset is split. More folds could potentially improve generalization by providing more diverse subsets for training, but this does not always guarantee better performance.

The stratified k-fold method, however, consistently outperformed regular k-fold. In particular, the stratified 4-fold cross-validation achieved the highest accuracy at 99.454%. The key advantage of stratified k-fold is its ability to maintain the class distribution in each fold, which is critical when dealing with imbalanced datasets. In the context of PII detection, certain types of sensitive data (such as medical or financial records) may be less frequent than others, leading to class imbalances. Stratified k-fold mitigates this issue, ensuring that each fold includes a representative sample of each class, which helps in producing more reliable performance metrics, especially in terms of precision and recall. Regular k-fold, in contrast, struggles with class imbalances, often resulting in poor performance for rare classes.

The study highlights the importance of proper data distribution and generalization. Models trained using the training-validation-test approach and stratified k-fold methods performed the best in terms of generalization. These methods provided a more reliable representation of how the model would perform on unseen data, reducing the risk of overfitting. In contrast, models trained using k-fold cross-validation showed significant drops in performance due to overfitting, particularly when the dataset was imbalanced. Overfitting occurs when a model becomes too specialized to the training data, leading to poor performance on new, unseen data.

The implications of these findings are significant for both public and private organizations. The study demonstrates that pre-existing models for PII detection are already reliable when trained properly. Organizations can fine-tune these pre-trained models with relevant and well-structured datasets to achieve high accuracy and robust generalization. This is especially critical in privacy-sensitive applications, such as those in the healthcare and finance sectors, where consistent and accurate PII detection is essential to comply with privacy regulations and protect individuals' sensitive information.

Furthermore, the results underscore the importance of addressing class imbalances in data. Stratified sampling, which is a key feature of stratified k-fold, ensures that rare or underrepresented PII categories are properly represented in the training process. This is crucial for detecting and safeguarding less common but highly sensitive forms of PII, such as medical conditions or financial records.

During the model testing phase, we analyzed which PII classes posed the greatest detection challenges, specifically examining the relationship between detection difficulty and the sensitivity level of the PII. Table 6 presents a subset of the experimental results, showing the false predictions made by the three best-performing models (TVT, k-fold 1, and stratified k-fold 4) across 10 PII classes. Our analysis reveals that the PII classes with the greatest detection difficulty (i.e., higher false prediction rates) are those at the 'sensitive' level, particularly when only one PII appears per sentence in the dataset. Conversely, when multiple PII instances occur in the same sentence, the models demonstrate better detection performance, likely due to the increased semantic context provided to the BERT schema.

Table 6. False prediction summary for different PII sensitivity levels.

		TVT	K-Fold 1	Strat. k-Fold 4
PII Name	**Total Classes**	**False Predictions**	**False Predictions**	**False Predictions**
Full name	10,430	26 (0.249%)	9 (0.086%)	17 (0.163%)
Surname	6428	24 (0.373%)	8 (0.124%)	17 (0.264%)
No PII	14,404	10 (0.069%)	7 (0.049%)	10 (0.069%)
Address	1131	7 (0.619%)	5 (0.442%)	5 (0.442%)
Job area	236	2 (0.847%)	4 (1.695%)	4 (1.695%)
First name	1384	2 (0.145%)	2 (0.145%)	5 (0.361%)
Job area + Full name	1896	2 (0.105%)	2 (0.105%)	2 (0.105%)
State	464	1 (0.216%)	3 (0.647%)	2 (0.431%)
Surname + Address	985	3 (0.305%)	0 (0.000%)	1 (0.102%)
Address + Full name	1728	3 (0.174%)	0 (0.000%)	1 (0.058%)

While the primary goal of this research was not to assess the scalability of the trained BERT models (a topic we plan to explore in future work, as discussed in Section 5.3), in addition to providing information to assess the computational complexity during the training phase, we measured the inference time—i.e., the time it takes for a model to make predictions after being trained—within the experimental environment. All the model instances we generated showed similar times, with average values reported as follows: 2.322 ms for sensitive PII, 2.133 ms for very sensitive PII, and 2.141 ms for less sensitive PII.

We believe that adopting these improved strategies could establish new standards for model evaluation in data privacy tasks. By prioritizing approaches like stratified k-fold cross-validation and the training-validation-test split, organizations can develop more trustworthy AI systems that not only comply with privacy regulations but also safeguard sensitive information more effectively. These strategies can lead to the creation of AI models that are both more accurate and more reliable, which is essential in maintaining public trust in AI systems used for privacy protection.

Comparison

In our evaluation, we conducted a quantitative comparison with existing works in the literature, focusing on studies with open English documentation. This approach allowed us to contextualize our results and assess our model's performance relative to prior work.

Specifically, Chen et al. [45] evaluated the performance of several models for PII recognition tasks, including Mistral 7B, CodeLlama 7B, Long Coderbase, GraphCodeBERT, CodeT5-220m, CodeT5Plus-770m, and CodeBERT. The dataset used in their study focuses on privacy elements from the California Consumer Protection Act (CCPA) and the General Data Protection Regulation (GDPR). It consists of records from public databases, with annotation on the following categories of PII:

- Device unique ID;
- Account or person identifier;
- Demographic data;
- Commercial or financial information;
- Biometric data;
- Employment information;
- Education level information;
- Job positions held.

Chen et al. [45] primarily focused on assessing the ability of large language models (LLMs) to challenge inaccurate or misleading security advice, aiming to determine if these models can be reliable sources of guidance in real-world scenarios.

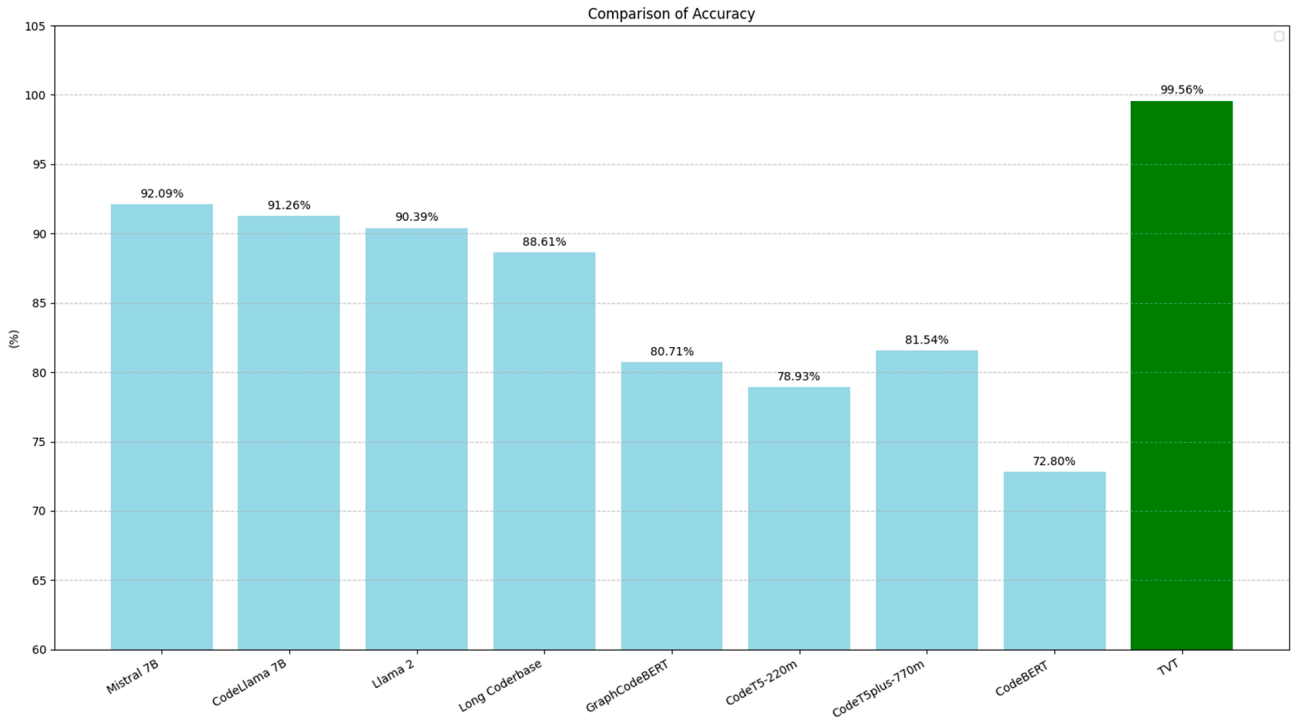
Figure 9 compares the performance of the models from [45] with our BERT model across four evaluation metrics: accuracy (9a), precision (9b), recall (9c), and F1-score (9d). These metrics were computed based on the following standard definitions:

- Accuracy = (True Positives + True Negatives)/(Total Samples);
- Precision = True Positives/(True Positives + False Positives);
- Recall = True Positives/(True Positives + False Negatives);
- F1-Score = 2 * (Precision * Recall)/(Precision + Recall).

In all cases, our BERT model shows clear improvements over the models in [45], achieving a 7.47-point increase in accuracy, a 6.99-point increase in precision, a 7.57-point increase in recall, and a 7.52-point increase in F1-score.

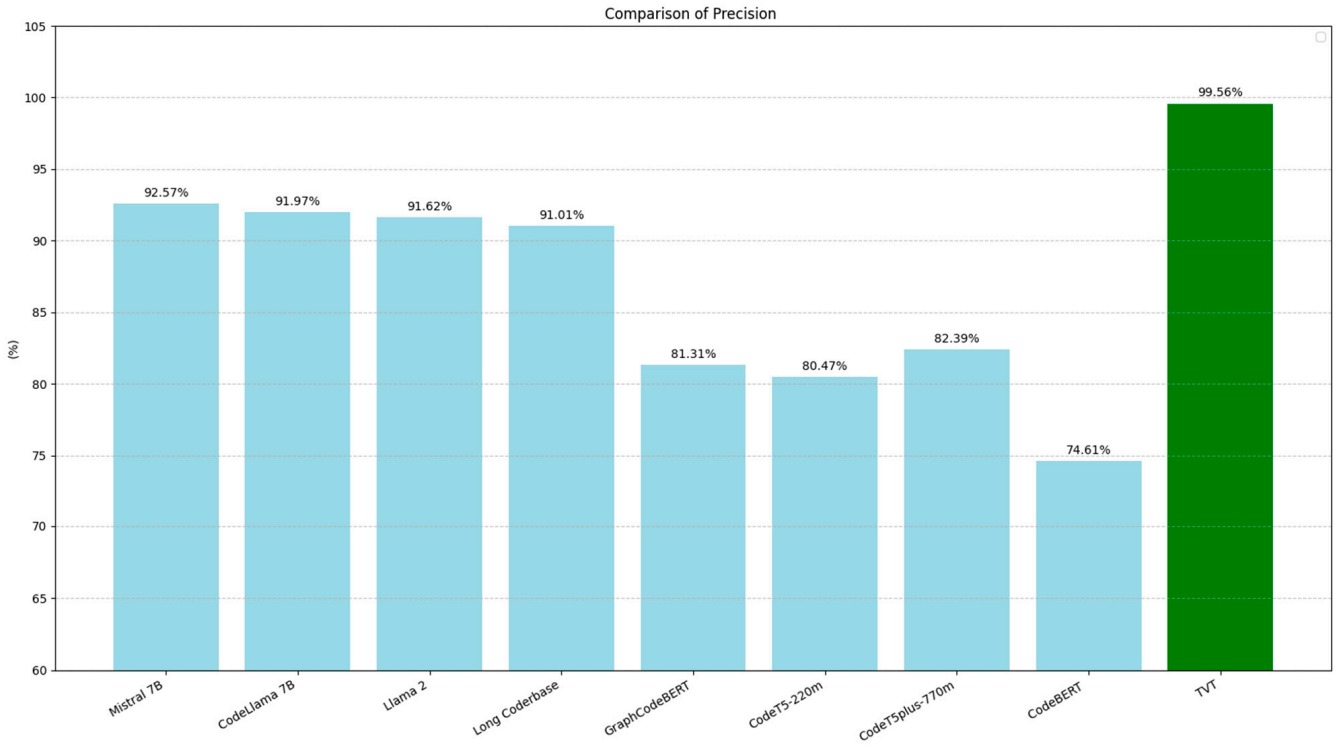
The second comparison we conducted focused on the experimental work by Shreya et al. [46], where the GPT-4 model was used to identify students' personal information in an educational setting. The dataset analyzed consisted of student posts from various subjects, including economics, mathematics, design, gamification, business trends, poetry, mythology, probability, and immunology, written between 2012 and 2015. After applying filtering and balancing operations, the dataset was reduced to 3505 posts from 2882 unique students.

In our comparison, we evaluated the precision and recall of the models presented in Shreya et al. [46] and compared them with those of our BERT model, which was trained using a separate training-validation-test procedure. Precision and recall were calculated using the standard formulas outlined earlier.

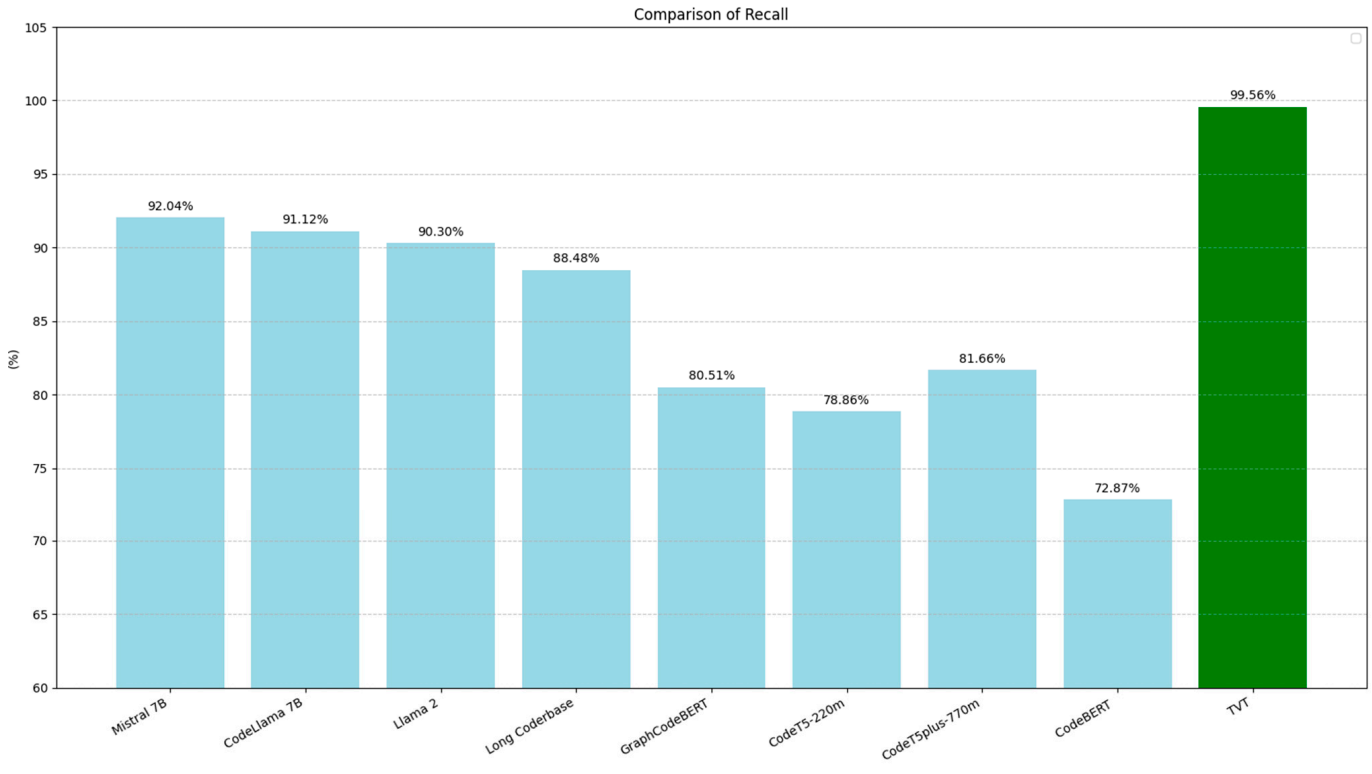


(**a**)

Figure 9. *Cont.*

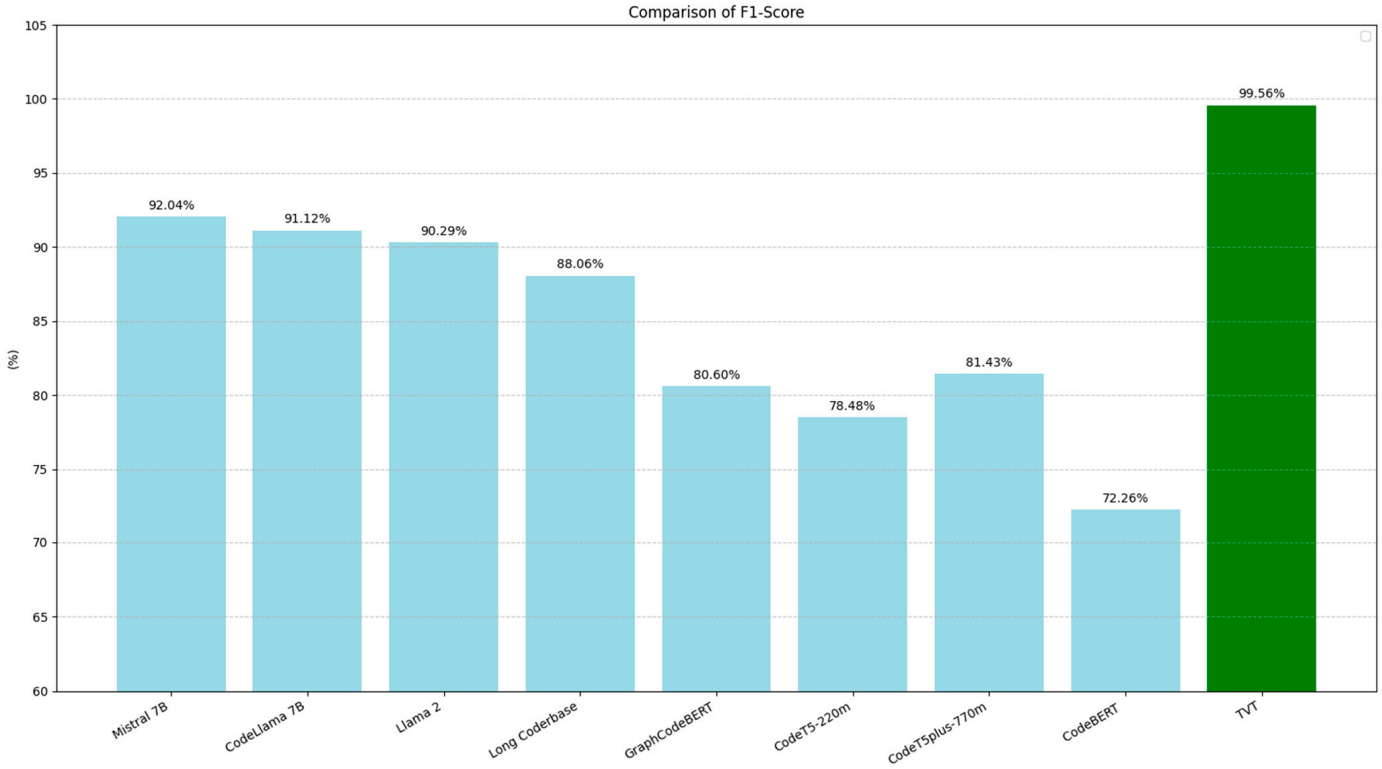


(**b**)



(**c**)

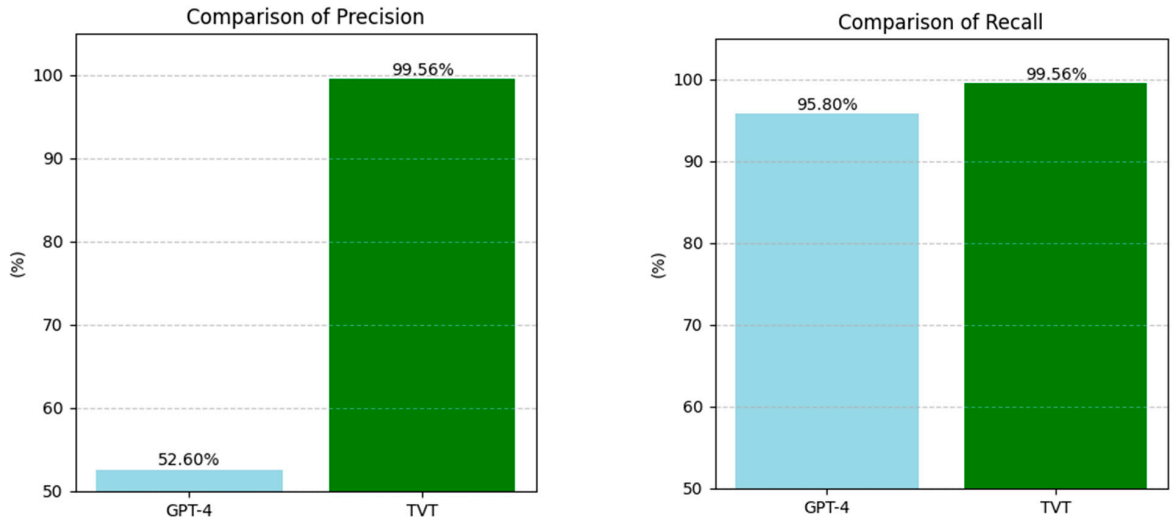
Figure 9. *Cont.*



(**d**)

Figure 9. Comparison of the performance metrics in PII recognition among the models evaluated by Chen et al. [45] with that of the BERT-trained training-validation-test model: (**a**) accuracy, (**b**) precision, (**c**) recall, and (**d**) F1-score.

Figure 10 compares the precision and recall indices between the models from Shreya et al. [46] and our BERT-trained model. While the improvement in precision is substantial, with an increase of 46.96 points in absolute value, the recall index shows a more modest improvement of 3.76 points. This suggests that our BERT model is particularly effective at minimizing false positives, though the improvement in capturing all relevant instances (i.e., reducing false negatives) is more incremental.



(**a**) (**b**)

Figure 10. Comparison of the performance metrics in PII recognition among the models evaluated by Shreya et al. [46] with that of the BERT-trained training-validation-test model: (**a**) precision, (**b**) recall.

5.2. Limitations

This study had several limitations. Firstly, the authors' experience in preparing the validation and testing datasets, coupled with the time-consuming nature of this process, constrained the study. Additionally, the study relied solely on BERT as a pre-trained model, while alternative variants such as RoBERTa, DeBERTa, and CodeBERT, which could offer efficiency improvements, were not explored. Lastly, the comparison of performance metrics between the model trained in this study and those in the experiments by Chen et al. [45] and Shreya et al. [46] is affected by the use of different testing datasets.

5.3. Future Work

Our future research will be divided into three main streams, each tackled by a different workgroup. First, we plan to assess the performance of our model on real-world datasets. We already have access to a dataset from an airport information system for this purpose. Second, we will study how our model performs with different variants of the testing dataset, focusing on privacy-sensitive domains such as healthcare, finance, and education. Lastly, we aim to develop a version of the BERT model specifically designed to ensure data privacy, with a focus on its scalability properties.

6. Conclusions

This research showed how the detection of PII can be improved using well-established NLP techniques and artificial intelligence algorithms. The experimental prototype we developed outperforms previous approaches in the field due to its use of a freely available, pre-trained BERT model. Our goal was to provide a valuable contribution to researchers and both public and private organizations by demonstrating how data privacy protection can be enhanced without requiring significant investments. The solution we developed can be used for real-time data stream PII detection as well as for analyzing data involved in events that lead to malicious exfiltration. It should be emphasized that informing individuals whose personal data are compromised in a breach is a legal requirement in many countries.

Author Contributions: Conceptualization, L.M. and A.E.; Methodology, A.E.; Software, A.E.; Validation, L.M. and A.E.; Formal analysis, L.M. and A.E.; Investigation, L.M. and A.E.; Resources, L.M. and A.E.; Data curation, L.M. and A.E.; Writing—original draft preparation, L.M.; Writing—review and editing, L.M.; Visualization, L.M.; Supervision, L.M.; Project administration, L.M.; Funding acquisition, L.M. All authors have read and agreed to the published version of the manuscript.

Funding: This work partially fulfills the research objectives of the Secure Safe Apulia project that was funded by the Apulia Region (Italy) under the 6ESURE5 grant agreement PO FESR 2014–2020.

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding author.

Acknowledgments: The authors thank the engineers of Exprivia SpA, an Italian company, for their valuable collaboration during the research and validation activities.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Cost of a Data Breach 2024 | IBM. Available online: https://www.ibm.com/reports/data-breach (accessed on 9 January 2025).
2. IT Governance USA. Data Breaches and Cyber Attacks–USA Report 2024. IT Governance USA Blog. Available online: https://www.itgovernanceusa.com/blog/data-breaches-and-cyber-attacks-in-2024-in-the-usa (accessed on 1 January 2025).
3. General Data Protection Regulation (GDPR)—Legal Text. General Data Protection Regulation (GDPR). Available online: https://gdpr-info.eu/ (accessed on 1 January 2025).

4. Jahan, M.S.; Oussalah, M. A Systematic Review of Hate Speech Automatic Detection Using Natural Language Processing. *Neurocomputing* **2023**, *546*, 126232. [CrossRef]
5. Liu, Y.; Song, H.H.; Bermudez, I.; Mislove, A.; Baldi, M.; Tongaonkar, A. Identifying Personal Information in Internet Traffic. In Proceedings of the 2015 ACM on Conference on Online Social Networks, Palo Alto, CA, USA, 2–3 November 2015; ACM: Palo Alto, CA, USA, 2015; pp. 59–70. [CrossRef]
6. Go, S.J.Y.; Guinto, R.; Festin, C.A.M.; Austria, I.; Ocampo, R.; Tan, W.M. An SDN/NFV-Enabled Architecture for Detecting Personally Identifiable Information Leaks on Network Traffic. In Proceedings of the 2019 Eleventh International Conference on Ubiquitous and Future Networks (ICUFN), Zagreb, Croatia, 2–5 July 2019; IEEE: Piscataway, NJ, USA, 2019; pp. 306–311. [CrossRef]
7. Ren, J.; Rao, A.; Lindorfer, M.; Legout, A.; Choffnes, D. ReCon: Revealing and Controlling PII Leaks in Mobile Network Traffic. In Proceedings of the 14th Annual International Conference on Mobile Systems, Applications, and Services, Singapore, 26–30 June 2016; ACM: Singapore, 2016; pp. 361–374. [CrossRef]
8. Noever, D. The Enron Corpus: Where the Email Bodies Are Buried? *arXiv* **2020**. [CrossRef]
9. Chan, K.-H.; Im, S.-K.; Ke, W. Multiple Classifier for Concatenate-Designed Neural Network. *Neural Comput. Applic.* **2022**, *34*, 1359–1372. [CrossRef]
10. Bader, M.D.M.; Mooney, S.J.; Rundle, A.G. Protecting Personally Identifiable Information When Using Online Geographic Tools for Public Health Research. *Am. J. Public Health* **2016**, *106*, 206–208. [CrossRef]
11. Alnemari, A.; Raj, R.K.; Romanowski, C.J.; Mishra, S. Protecting Personally Identifiable Information (PII) in Critical Infrastructure Data Using Differential Privacy. In Proceedings of the 2019 IEEE International Symposium on Technologies for Homeland Security (HST), Woburn, MA, USA, 5–6 November 2019; IEEE: Piscataway, NJ, USA, 2019; pp. 1–6. [CrossRef]
12. AL Ghazo, A.T.; Abu Mallouh, M.; Alajlouni, S.; Almalkawi, I.T. Securing Cyber Physical Systems: Lightweight Industrial Internet of Things Authentication (LI2A) for Critical Infrastructure and Manufacturing. *Appl. Syst. Innov.* **2025**, *8*, 11. [CrossRef]
13. Onik, M.M.H.; Kim, C.-S.; Lee, N.-Y.; Yang, J. Personal Information Classification on Aggregated Android Application's Permissions. *Appl. Sci.* **2019**, *9*, 3997. [CrossRef]
14. Majeed, A.; Ullah, F.; Lee, S. Vulnerability- and Diversity-Aware Anonymization of Personally Identifiable Information for Improving User Privacy and Utility of Publishing Data. *Sensors* **2017**, *17*, 1059. [CrossRef]
15. Venkatanathan, J.; Kostakos, V.; Karapanos, E.; Goncalves, J. Online Disclosure of Personally Identifiable Information with Strangers: Effects of Public and Private Sharing. *Interact. Comput.* **2014**, *26*, 614–626. [CrossRef]
16. Tesfay, W.B.; Serna, J.; Pape, S. Challenges in Detecting Privacy Revealing Information in Unstructured Text. In Proceedings of the 4th Workshop on Society, Privacy and the Semantic Web—Policy and Technology (PrivOn2016), Kobe, Japan, 18 October 2016; Brewster, C., Cheatham, M., d'Aquin, M., Decker, S., Kirrane, S., Eds.; CEUR Workshop Proceedings. CEUR: Kobe, Japan, 2016; Volume 1750.
17. Liu, Y.; Song, T.; Liao, L. TPII: Tracking Personally Identifiable Information via User Behaviors in HTTP Traffic. *Front. Comput. Sci.* **2020**, *14*, 143801. [CrossRef]
18. Vishwamitra, N.; Li, Y.; Wang, K.; Hu, H.; Caine, K.; Ahn, G.-J. Towards PII-Based Multiparty Access Control for Photo Sharing in Online Social Networks. In Proceedings of the 22nd ACM on Symposium on Access Control Models and Technologies, Indianapolis, IN, USA, 21–23 June 2017; ACM: Indianapolis, IN, USA, 2017; pp. 155–166. [CrossRef]
19. Conti, M.; Li, Q.Q.; Maragno, A.; Spolaor, R. The Dark Side(-Channel) of Mobile Devices: A Survey on Network Traffic Analysis. *IEEE Commun. Surv. Tutor.* **2018**, *20*, 2658–2713. [CrossRef]
20. Wongwiwatchai, N.; Pongkham, P.; Sripanidkulchai, K. Detecting Personally Identifiable Information Transmission in Android Applications Using Light-Weight Static Analysis. *Comput. Secur.* **2020**, *99*, 102011. [CrossRef]
21. Lee, J.; De Guzman, M.C.; Wang, J.; Gupta, M.; Rao, H.R. Investigating Perceptions about Risk of Data Breaches in Financial Institutions: A Routine Activity-Approach. *Comput. Secur.* **2022**, *121*, 102832. [CrossRef]
22. Tavana, M.; Khalili Nasr, A.; Ahmadabadi, A.B.; Amiri, A.S.; Mina, H. An Interval Multi-Criteria Decision-Making Model for Evaluating Blockchain-IoT Technology in Supply Chain Networks. *Internet Things* **2023**, *22*, 100786. [CrossRef]
23. Tavana, M. Decision Analytics in the World of Big Data and Colorful Choices. *Decis. Anal. J.* **2021**, *1*, 100002. [CrossRef]
24. Ayyagari, R. An Exploratory Analysis of Data Breaches from 2005-2011: Trends and Insights. *J. Inf. Priv. Secur.* **2012**, *8*, 33–56. [CrossRef]
25. Stiennon, R. Breach Level Index. Categorising Data Breach Severity with a Breach Level Index. 2013. Available online: http://breachlevelindex.com/pdf/Breach-Level-Index-WP.pdf (accessed on 30 January 2025).
26. Ayaburi, E.W. Understanding Online Information Disclosure: Examination of Data Breach Victimization Experience Effect. *ITP* **2023**, *36*, 95–114. [CrossRef]
27. Zadeh, A.; Lavine, B.; Zolbanin, H.; Hopkins, D. A Cybersecurity Risk Quantification and Classification Framework for Informed Risk Mitigation Decisions. *Decis. Anal. J.* **2023**, *9*, 100328. [CrossRef]

28. Pool, J.; Akhlaghpour, S.; Fatehi, F.; Burton-Jones, A. A Systematic Analysis of Failures in Protecting Personal Health Data: A Scoping Review. *Int. J. Inf. Manag.* **2024**, *74*, 102719. [CrossRef]
29. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention Is All You Need. In Proceedings of the 31st International Conference on Neural Information Processing Systems, NIPS'17, Long Beach, CA, USA, 4–9 December 2017; Curran Associates Inc.: Red Hook, NY, USA, 2017; pp. 6000–6010.
30. Im, S.-K.; Chan, K.-H. Neural Machine Translation with CARU-Embedding Layer and CARU-Gated Attention Layer. *Mathematics* **2024**, *12*, 997. [CrossRef]
31. Transformers. Available online: https://huggingface.co/docs/transformers/index (accessed on 20 January 2025).
32. Joshi, M.; Chen, D.; Liu, Y.; Weld, D.S.; Zettlemoyer, L.; Levy, O. SpanBERT: Improving Pre-Training by Representing and Predicting Spans. *Trans. Assoc. Comput. Linguist.* **2020**, *8*, 64–77. [CrossRef]
33. Clark, K.; Luong, M.-T.; Le, Q.V.; Manning, C.D. ELECTRA: Pre-Training Text Encoders as Discriminators Rather Than Generators. *arXiv* **2020**. [CrossRef]
34. Devlin, J.; Chang, M.-W.; Lee, K.; Toutanova, K. BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding. *arXiv* **2019**. [CrossRef]
35. ai4privacy/pii-Masking-200k | Datasets at Hugging Face. Available online: https://huggingface.co/datasets/ai4privacy/pii-masking-200k (accessed on 22 January 2025).
36. Holmes, L.; Crossley, S.; Wang, J.; Zhang, W. The Cleaned Repository of Annotated Personally Identifiable Information. 2024. Available online: https://zenodo.org/records/12729952 (accessed on 30 January 2025).
37. Wen, Y.; Marchyok, L.; Hong, S.; Geiping, J.; Goldstein, T.; Carlini, N. Privacy Backdoors: Enhancing Membership Inference through Poisoning Pre-Trained Models. *arXiv* **2024**. [CrossRef]
38. Rashid, M.R.U.; Liu, J.; Koike-Akino, T.; Mehnaz, S.; Wang, Y. Forget to Flourish: Leveraging Machine-Unlearning on Pretrained Language Models for Privacy Leakage. *arXiv* **2024**. [CrossRef]
39. Hastings, J.D.; Weitl-Harms, S.; Doty, J.; Myers, Z.J.; Thompson, W. Utilizing Large Language Models to Synthesize Product Desirability Datasets. In Proceedings of the 2024 IEEE International Conference on Big Data (BigData), Washington, DC, USA, 15–18 December 2024; IEEE: Piscataway, NJ, USA, 2024; pp. 5352–5360. [CrossRef]
40. Generic Sentiment | Multidomain Sentiment Dataset. Available online: https://www.kaggle.com/datasets/akgeni/generic-sentiment-multidomain-sentiment-dataset (accessed on 22 January 2025).
41. ai4privacy/pii-Masking-43k | Datasets at Hugging Face. Available online: https://huggingface.co/datasets/ai4privacy/pii-masking-43k (accessed on 24 January 2025).
42. Dataset for Chatbot. Available online: https://www.kaggle.com/datasets/grafstor/simple-dialogs-for-chatbot (accessed on 24 January 2025).
43. IMDB Dataset of 50K Movie Reviews. Available online: https://www.kaggle.com/datasets/lakshmi25npathi/imdb-dataset-of-50k-movie-reviews (accessed on 24 January 2025).
44. Open AI | GPT-4. Available online: https://openai.com/index/gpt-4/ (accessed on 22 January 2025).
45. Chen, Y.; Arunasalam, A.; Celik, Z.B. Can Large Language Models Provide Security & Privacy Advice? Measuring the Ability of LLMs to Refute Misconceptions. In Proceedings of the Annual Computer Security Applications Conference, Austin, TX, USA, 4–8 December 2023; ACM: Austin, TX, USA, 2023; pp. 366–378. [CrossRef]
46. Singhal, S.; Zambrano, A.F.; Pankiewicz, M.; Liu, X.; Porter, C.; Baker, R.S. De-Identifying Student Personally Identifying Information with GPT-4. In Proceedings of the EDM2024, Atlanta, GA, USA, 14–17 July 2024. [CrossRef]