Neural Computing and Applications (2024) 36:15439–15454
https://doi.org/10.1007/s00521-024-09707-w

ORIGINAL ARTICLE

Privacy BERT-LSTM: a novel NLP algorithm for sensitive information detection in textual documents

Janani Muralitharan[1] · Chandrasekar Arumugam[2]

Received: 22 August 2023 / Accepted: 25 March 2024 / Published online: 16 May 2024

Abstract
In this modern digital era, the increasing volume of textual data and the widespread adoption of natural language processing (NLP) techniques have presented a critical challenge in safeguarding sensitive privacy information. As a result, there is a pressing demand to design robust and accurate NLP-based techniques to perform efficient sensitive information detection in textual data. This research paper focuses on the detection and classification of sensitive privacy information in textual documents using NLP by proposing a novel algorithm named Privacy BERT-LSTM. The proposed Privacy BERT-LSTM algorithm employs BERT for obtaining contextual embeddings and LSTM for sequential information processing, facilitating efficient sensitive information detection in textual documents. The BERT with its bidirectional characteristics captures the nuances and meaning of the textual documents, while the LSTM derives the long-range dependencies in the textual data. Moreover, the proposed Privacy BERT-LSTM algorithm with its attention mechanism highlights the important regions of the textual documents, contributing to efficient sensitive information detection. The comprehensive performance evaluation is conducted by employing the SMS Spam Collection dataset in terms of standard performance metrics and comparing it with different state-of-the-art techniques, namely, CASSED, PRIVAFRAME, CNN-LSTM, Conv-FFD, GCSA, TSIIP, and, C-PIIM. The experimental outcomes clearly illustrate that the Privacy BERT-LSTM algorithm demonstrates superior performance in identifying various types of sensitive information by achieving an accuracy of 92.50%, F1-score of 85.02%, and Precision of 89.36%. The proposed algorithm outperforms existing baseline models, providing valuable advancements in sensitive information detection using NLP. Therefore, this research contributes to the advancement of privacy protection in NLP applications and opens avenues for future investigations in the domain of sensitive information detection. Additionally, the proposed algorithm provides valuable insights for researchers and practitioners working on privacy-sensitive NLP tasks.

Keywords Sensitive information detection · NLP · BERT · LSTM · Textual documents · Class imbalance

1 Introduction

1.1 Background knowledge about sensitive information detection in textual documents

In recent years, the rapid proliferation of the internet has increased the digitization of sensitive information [1]. From social media posts to medical records, textual documents encompass a vast repository of information. However, this digital transformation has raised serious concerns about the protection of sensitive privacy information embedded within these texts [2].

Sensitive privacy information, such as personal identifiable information (PII), financial data, and medical

✉ Janani Muralitharan
jananim@stjosephs.ac.in

[1] Department of Information Technology, St.Joseph's College of Engineering, Chennai, Tamil Nadu, India

[2] Department of Computer Science and Engineering, St.Joseph's College of Engineering, Chennai, Tamil Nadu, India

records, is highly susceptible to privacy breaches due to unauthorized access [3]. The leakage of sensitive information leads to financial fraud, identity theft, and business losses [4]. Therefore, several attempts are made to recognize sensitive information in textual documents. Traditional methods for detecting sensitive information often relied on rule-based approaches and keyword-matching techniques. However, these approaches are limited in their ability to handle the complexities and variations present in natural language [5].

Various deep learning (DL) has shown remarkable success in various NLP tasks. Particularly, adaptive multi-scale convolution networks-based manifold embedding networks (AMCMENet) [6], adaptive focal local-based normalized conditional autoencoder (NCVAE-AFL) [7], and hypergraph CNN based on model-assisted multi-source fusion (MAMF-HGCN) [8] with its outstanding feature learning ability have brought a paradigm shift in various fields. Furthermore, the wavelet transform-based CNN [9] also witnessed greater advantages in various detection tasks. Furthermore, several NLP models based on transformer architectures have been used to gather the contextual entity among words and have shown impressive results in text classification, sentiment analysis, and named entity recognition.

The primary motivation of this paper is to surpass the limitations encountered by the traditional rule-based approaches and keyword-matching techniques by proposing a novel NLP-based algorithm for sensitive privacy information detection in textual documents. The Privacy BERT-LSTM model combines the strengths of BERT and LSTM to create a powerful hybrid model that comprehensively analyzes textual data and accurately identifies sensitive information in textual documents.

Moreover, the research addresses the challenge of class imbalance, which is common in privacy information detection, as sensitive information instances are often rarer than non-sensitive ones by implementing strategies to mitigate class imbalance issues for developing a more robust and reliable model.

The interpretability of the Privacy BERT-LSTM algorithm is another crucial aspect of this research. The Privacy BERT-LSTM model includes an attention mechanism, which allows us to visualize the most important tokens influencing the classification decision. This transparency provides insights into the model's reasoning and fosters accountability in sensitive information detection.

1.2 Objectives

The main objectives of this paper are presented below:

1. The research aims to develop a novel NLP-based approach named Privacy BERT-LSTM algorithm for sensitive privacy information detection in textual documents.
2. A Privacy BERT-LSTM algorithm aims to understand the meaning and nuances of natural language more effectively by utilizing BERT for word embeddings and LSTM with an attention mechanism for sequential information processing.
3. In the experimentation phase, the research aims to measuring the performance of the Privacy BERT-LSTM technique by utilizing different standard evaluation metrics and compare against existing baseline models to demonstrate its superiority in sensitive privacy information detection.
4. To perform attention heat map visualization and error analysis to improve the interpretability of our Privacy BERT-LSTM model.

1.3 Contributions

The significant contributions made by the proposed Privacy BERT-LSTM model are provided below:

1. *Privacy BERT-LSTM Algorithm:* The Privacy BERT-LSTM algorithm effectively combines BERT's contextual embeddings and LSTM's sequential information processing to perform accurate classification of various sensitive information types.
2. *Evaluation on SMS Spam Collection Dataset:* The Privacy BERT-LSTM algorithm is thoroughly evaluated on the widely used SMS Spam Collection dataset, originally designed for spam detection. This novel adaptation of the dataset for sensitive information detection adds value to the research, as it allows us to evaluate our model's performance on a various range of textual messages.
3. *Handling Class Imbalance with Data Augmentation:* The research explores and implements data augmentation techniques to address the class imbalance issue commonly present in sensitive information detection tasks that enabled the model to be more sensitive to rare sensitive information instances.
4. *Interpretability through Attention Heat maps:* To enhance model interpretability, the research visualizes attention heat maps and saliency maps, providing valuable insights into the Privacy BERT-LSTM model's decision-making process.
5. *Error Analysis for Model Limitations:* The research conducts an in-depth error analysis to identify patterns in misclassifications and explore potential model limitations.

The remaining of the paper is structured as follows. Section 2 provides the overview of the existing works with their limitations and also the research gaps. Section 3 provides a detailed explanation of the proposed methodology. Section 4 discusses the experimental evaluation of the proposed method. Section 5 provides the conclusion of the research work along with the future scope.

2 Related works

2.1 Review of existing works related to sensitive information detection and classification in textual documents

Early efforts in sensitive information detection often relied on rule-based and pattern-matching techniques. For instance, rules were designed to detect credit card numbers, social security numbers, or email addresses. While these methods were relatively simple and interpretable, they lacked the flexibility to handle variations in language and context approaches Aubaid and Mishra [10]. Moreover, they required manual rule creation, making them less suitable for identifying complex and evolving patterns of sensitive information.

Another class of techniques involves keyword matching, where lists of sensitive keywords or phrases are used to identify potential instances of sensitive information in the text Huo and Jiang [11]. These keywords included terms related to PII, financial information, or medical records. However, it suffered from false positives and false negatives, as sensitive information is referred to using synonyms or different terms.

Recently, machine learning techniques gained more prevalence in sensitive information identification because of their skill to detect patterns in the data and adapt to different contexts Zhang and Jiang [12]. Various supervised and unsupervised machine learning algorithms have been explored for this task. Supervised learning methods involve learning a paradigm on labeled data, wherein an individual case is explained as sensible or insensitive. Several machine learning algorithms, such as Support Vector Machines (SVM), Random Forests, and Naive Bayes, have been used for this purpose.

Researchers often extract features from the text, such as n-grams García et al. [13], bag-of-words Barve et al. [14], or TF-IDF Zhuohao et al. [15], to represent the input data. These features are then fed into the classifier to make predictions. Supervised learning approaches achieved good accuracy, especially when provided with a large and diverse labeled dataset. However, they struggled with rare or emerging sensitive information classes due to data imbalance Luo [16].

Unsupervised learning methods attempt to identify sensitive information patterns without the need for labeled data Kulkarni and Cauvery [17]. Techniques like clustering, topic modeling, and anomaly detection have been explored for this purpose. Unsupervised approaches were valuable when labeled data were‘ insufficient. However, they lacked precision, as they did not have access to ground truth labels for evaluation.

In recent years, deep learning models, particularly neural networks, have shown remarkable success in various. Convolutional neural networks (CNNs) have been adapted for text classification tasks, including sensitive information detection Liu et al. [18]. In this approach, 1D convolution was applied to the word embeddings to take local patterns and dependencies in the text. However, they struggled with capturing long-range dependencies in the text.

Recurrent neural networks (RNNs) particularly LSTM networks have been widely used for sequence-to-sequence modeling tasks in NLP Roslan and Foozy [19]. LSTM's ability to process sequential information and capture long-range dependencies makes them suitable for sensitive information detection.

Bi-Directional LSTM (Bi-LSTM) was also widely used for sensitive information detection which reduced the computational complexity of the network and showed improved performance Victor and Lopez [20]. RNNs learn from the context surrounding words to better understand the presence of sensitive information. However, they suffered from vanishing gradient problems and struggled to capture very long dependencies.

Transformer-Based Models, such as BERT, have revolutionized NLP tasks by leveraging self-attention mechanisms to capture bidirectional context in text García-Pablos et al. [21]. Researchers often fine-tune pre-trained BERT models on specific tasks, including sensitive information detection, to improve performance Guo et al. [22].

Transfer learning has been broadly accepted in NLP to leverage pre-trained language models for downstream tasks Qasim et al. [23]. Transfer learning allows models to leverage knowledge learned from vast amounts of text data, which significantly improves performance on specific tasks with limited labeled data.

2.2 Limitations and challenges related to existing solutions

The existing studies had suffered from several challenges and limitations, namely, lack of interpretability and explain ability, data imbalance, gradient vanishing, and poor performance. The limitations and challenges of sensitive information detection are provided below:

Black-box nature: The existing studies clearly showed that the black-box nature of the traditional DL models for sensitive information detection greatly affected the interpretability and explain ability of the model.

Class imbalance: Another limitation was the class imbalance issue which degraded the execution of the classification methodology. The existing studies clearly illustrated that the class imbalance issues reduced the performance of the system particularly for minority classes in different NLP operations.

Gradient vanishing issues: The gradient vanishing issue was also one of the major limitations of deep learning. This issue most commonly occurs during training which leads to instability, slow convergence as well as limits the learning ability of the model.

Poor performance: The existing approaches for sensitive information detection showed poor performance due to an increase in the misclassification rate. If the sensitive information was expressed in the form of synonyms or different terms, it was prone to false positive and false negative rates. Therefore, there is an immediate requirement for an efficient sensitive information detection model arises to address these challenges and limitations in the existing works.

3 Research gap

An effective algorithm that improves interpretability and lowers class imbalance issues is required for developing a trustable as well as efficient and reliable detection model for sensitive information from textual documents. These research gaps need to be addressed to develop an efficient model for sensitive privacy information detection from textual documents. Therefore, our research work considers the issues associated with interpretability and explain ability in order to gain the trust of the users and also to maximize the transparency of the each prediction. Furthermore, the Privacy BERT-LSTM method also incorporates efficient methods like data augmentation and class weighting to solve the class imbalance problems.

4 Proposed methodology

This section provides a detailed overview of our proposed methodology which involves three steps, namely, data preprocessing, BERT for word embeddings, and a bidirectional LSTM encoder with an attention mechanism for processing the sequential information. The overall framework of our proposed technique is illustrated in Fig. 1.

The proposed Privacy BERT-LSTM methodology initially performs pre-processing of the SMS Spam Collection dataset prepared for sensitive information detection. It involves text cleaning, tokenization, data transformations, and encoding. The text cleaning reduces the existence of white spaces, special characters, symbols, and irrelevant characters, while the tokenization converts the cleaned textual messages into separate tokens. The data transformation including lemmatization or stemming is performed to diminish the words to their base forms and the encoding normalizes the tokenized text data into numerical vectors for further processing. The preprocessed textual data are fed to BERT for obtaining contextual embeddings. BERT consists of multiple transformer layers that process the preprocessed textual data through a series of attention mechanisms for obtaining the contextual information in the word embeddings by considering the relationships between words in the entire sequence. The bidirectional nature of the BERT facilitates understanding the context in which a word appears for capturing the meaning and nuances of textual messages more effectively. The contextual embeddings are fed to the bidirectional LSTM layer with an attention mechanism for deriving long-range dependencies in the textual documents. The bidirectional LSTM layer processes the contextual embeddings in forward and backward directions. This bidirectional nature allows the LSTM to understand the sequential dependencies and context in the text messages. The attention mechanism is incorporated to highlight important regions of the textual document and facilitates the generation of attention-weighted contextual embedding. This embedding captures the contextual information while highlighting the significant tokens for classification. Finally, these embeddings are federal to the fully connected layer along with a sigmoid activation function for performing sensitive information detection. The detailed description of every step of the proposed Privacy BERT-LSTM algorithm is delineated below:

4.1 Data preprocessing

The first step of the Privacy BERT-LSTM methodology is data preprocessing. The SMS Spam Collection dataset, initially designed for spam detection, is prepared for sensitive information detection. This process involves cleaning the text data to remove any irrelevant or noisy information that might hinder the model's performance. The data are then tokenized, where each word is converted into a numerical representation to be processed by the model. Additionally, any necessary data transformations or encoding tasks are performed during this phase.

1. Text cleaning

Let D be the set of textual documents in the SMS Spam Collection dataset, where $D = \{d_1, d_2, ...d_N\}$, and the

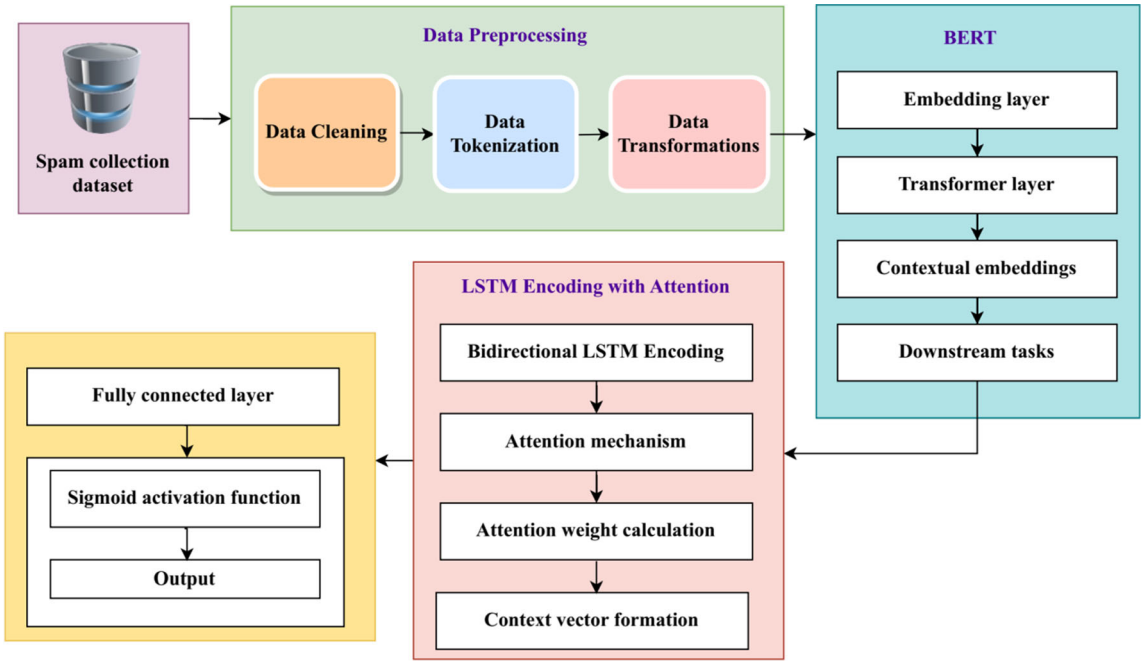

Fig. 1 Overall framework of our privacy BERT-LSTM methodology

number of documents is denoted as N. Text cleaning involves removing irrelevant characters, symbols, special characters, and white spaces from each document $d_i \in D$. This process is represented as follows:

$$d_{i\,cleaned} = \text{clean}_{\text{text}}(d_i) \tag{1}$$

The term $\text{clean}_{\text{text}}(\cdot)$ is a function that removes unwanted characters and symbols from the text.

2. Tokenization

Next, we tokenize the cleaned text into individual words or tokens. Each document $d_i \in D$ is converted into a sequence of tokens $X_i = [x_1, x_2, ...x_{M_i}]$, where M_i is the number of tokens in document d_i. This step is represented as:

$$X_i = \text{tokenize}\,(d_{i\,\text{cleaned}}) \tag{2}$$

The term $\text{tokenize}\,(\cdot)$ is a function that breaks down the text into tokens.

3. Data transformations

In some cases, additional data transformations are required to prepare the text data for NLP models. For example, we might perform stemming or lemmatization to reduce words to their base forms and handle variations in word forms. This normalization step is represented as follows:

$$X_{\text{i normalized}} = \text{normalize}\,(X_i) \tag{3}$$

The term $\text{normalize}\,(\cdot)$ is a function that performs data transformations, such as stemming or lemmatization.

4. Encoding

Finally, we encode the normalized tokenized text data into numerical vectors for processing by the Privacy BERT-LSTM algorithm. One common method is to employ word embeddings, which are dense phase vectors that derives semantic relationships among words. The encoding process is represented as:

$$E_i = \text{word}_{\text{embeddings}}(X_{i\,\text{normalized}}) \tag{4}$$

where $E_i = [e_1, e_2, ...e_{M_i}]$ is the set of word embeddings for the document d_i. Each embedding e_i is a numerical vector that represents the contextual meaning of the token x_i in the document d_i.

4.2 Word embeddings with BERT (bidirectional encoder representations from transformers)

BERT is a bidirectional model because it takes into account both the left and right contexts of each word during its training phase [24]. This bidirectional nature allows BERT to understand the context in which a word appears, enabling it to capture the meaning and nuances of natural language more effectively.

To obtain contextual embeddings for the text data, we utilize BERT, which is a transformer-based language model. BERT generates word embeddings $E = [e_1, e_2, ...e_n]$ by processing the input sequence X through multiple transformer layers. Each embedding e_i captures the contextual relationship between a word x_i and its surrounding words in the text.

The process of obtaining word embeddings with BERT is summarized as follows:

1. Input representation

The input sequence $X = [x_1, x_2, ...x_n]$ is tokenized and converted into numerical representations. The number of tokens is represented as n. Each token x_i is mapped to its corresponding index in BERT's vocabulary.

2. Embedding layer

The tokenized input sequence X is fed into the embedding layer of BERT, which maps each token x_i to its corresponding word embedding e_i. For pre-training the embedding layer by using a huge corpus of text data which learns to portray each token based on its context of training data in a high-dimensional vector space.

3. Transformer layers

BERT consists of multiple transformer layers, typically 12 or 24, which process the input sequence X through a series of attention mechanisms and feed-forward neural networks. Each transformer layer refines the contextual information in the word embeddings by considering the relationships between words in the entire sequence.

4. Contextual embeddings

After passing through the transformer layers, BERT produces contextual embeddings $E = [e_1, e_2, ...e_n]$, where each embedding e_i represents the contextual information of the corresponding token x_i in the input sequence. These embeddings capture the contextual relationships between words, allowing the model to understand the meaning of each word in the context of the entire text.

5. Downstream tasks

The pre-trained BERT model is fine-tuned on specific downstream tasks, such as sensitive information detection, by adding task-specific layers and training the model on labeled data. Fine-tuning allows BERT to adapt its contextual embeddings for the specific task and achieve high performance on that task.

4.3 LSTM encoding with attention

After obtaining contextual embeddings from BERT, the algorithm employs a bidirectional LSTM layer to encode the sequential information in the embeddings. The bidirectional LSTM processes the input sequence $E = [e_1, e_2, ...e_n]$ both forward and backward to capture long-range dependencies in the text [25].

1. Bidirectional LSTM encoding

The bidirectional LSTM layer processes the input array E in forward and backward direction. It enables the LSTM to deduce the information with past and future contexts of each word, allowing it to understand the sequential dependencies and context in the text more comprehensively. The LSTM layer generates two sets of hidden states, namely $H_{\text{forward}} = \left[h_1^f, h_2^f, ...h_n^f\right]$ for the forward pass and $H_{\text{backward}} = \left[h_1^b, h_2^b, ...h_n^b\right]$ for the backward pass.

2. Attention mechanism

In addition to bidirectional LSTM encoding, the Privacy BERT-LSTM algorithm incorporates an attention mechanism to highlight relevant parts of the document [26]. The attention mechanism computes attention weights $A = [a_1, a_2, ...a_n]$ for each word embedding e_i, where a_i represents the importance of the word in the classification decision.

3. Attention weights calculation

The attention weights are computed using a Soft max function that takes into account the similarity between each word embedding e_i and a learnable weight matrix W_a. The similarity score reflects the importance of each word embedding in contributing to the classification decision for sensitive information. The attention weights A obtained through a Soft max function are expressed as follows:

$$a_i = \frac{\exp(e_i \cdot W_a)}{\sum_{j=1}^{n} \exp(e_i \cdot W_a)} \tag{5}$$

The term W_a is a learnable weight matrix that captures the importance of each word embedding, e_i is the contextual embedding of the ith word in the input sequence, and $\exp(\cdot)$ is the exponential function. The Soft max function normalizes the similarity scores, ensuring that the attention weights sum to 1, which allows it to concentrate on the more relevant words in the document.

4. Context vector formation

The attention weights A are multiplied element-wise with the original embeddings E, generating an attention-weighted contextual embedding $C = [c_1, c_2, ...c_n]$. The context vector C captures the contextual information while highlighting the important tokens for classification.

The context vector C is then used as input to the classification layer, where the binary classification is performed using fully connected layer along with a sigmoid activation performs binary classification to identify sensitive entities in the text.

4.4 Privacy BERT-LSTM model architecture

After LSTM encoding with attention, the Privacy BERT-LSTM algorithm combines the LSTM output with the attention-weighted embeddings to form the context vector $C = [c_1, c_2, ...c_n]$. The context vector captures the contextual information and highlights important tokens for classification.

1. Concatenation

The LSTM output H_{forward} *and* H_{backward} is concatenated with the attention-weighted embeddings C to form the context vector $C = [c_1, c_2, ...c_n]$.

$$C = \text{concatenate}([H_{\text{forward}}, H_{\text{backward}}, C]) \tag{6}$$

The concatenation operation combines the information from both the forward and backward passes of the bidirectional LSTM with the attention-weighted embeddings, ensuring that the context vector captures comprehensive information about the text. For instance, consider the sentence " Mobile club: Choose any of the top quality items for your mobile.7cfca1a." In this example, the Privacy BERT-LSTM model correctly identifies the message as sensitive due to the words "Mobile club" and "7cfca1a." The model focuses on these words to classify the message as sensitive, indicating that it has learned to recognize phrases with its ability to derive comprehensive information about the messages. Furthermore, for the SMS Messages "Money I have won wining number 946 wot do I next" the attention mechanism assigns higher weights to the number "946," for identifying sensitive information. This highlights the model's ability to catch important tokens in the text, contributing to accurate sensitive information detection. For the message "ECPT 1/3. You have ordered a Ringtone. Your order is being processed…" the model focuses on contextually important tokens such as "ECPT 1/3″ and "Ringtone," enabling it to distinguish between sensitive and non-sensitive messages. For the SMS Message "Please CALL 08712402578 immediately as there is an urgent message waiting for you" the model captures the contextual information of those messages and enhances its ability to concentrate on contextually significant words and distinguish between sensitive and non-sensitive messages. At last, for the message "You have an important customer service announcement Call Free phone 0800542 0825 now!" the model captures significant tokens including 0800542 0825 and Free phone in the textual documents, contributing to efficient sensitive information detection.

2. Fully connected layer

The context vector C is intersecting a fully connected layer along with a sigmoid activation function to perform binary classification. The fully connected layer applies linear transformations to the input context vector C using a learnable weight matrix W_c.

$$Z = C \cdot W_c + b_c \tag{7}$$

The output of the fully connected layer is represented as the term Z and b_c is a learnable bias term.

3. Sigmoid activation function

The output Z is later gone with a sigmoid activation function to obtain the probability p_i that each word x_i in the text is classified as sensitive:

$$p_i = \text{sigmoid}(Z) \tag{8}$$

The sigmoid activation function is mapping the output values to a extent 0 to 1, where values closer to 1 represents a high probability that the word is sensitive, and values closer to 0 indicate a low probability that the word is sensitive. Figure 2 represents the architecture of the Privacy BERT-LSTM technique algorithm for sensitive information detection.

4.5 Model training and fine-tuning

To train the Privacy BERT-LSTM model, we utilize the SMS Spam Collection dataset, where each instance is labeled as sensitive or non-sensitive. The binary cross-entropy loss is used to optimize the predicted probabilities p_i and the corresponding ground truth labels. During training, the model's parameters are fine-tuned using the Adam optimizer. The Adam optimizer adjusts the learning rate adaptively based on the gradient information, facilitating faster convergence and improved model performance. The Privacy BERT-LSTM model undergoes several training epochs, iterating over the dataset multiple times to gradually improve its performance. After training, the model is ready for evaluation and testing on new, unseen data to assess its accuracy in detecting and classifying sensitive information in textual documents.

1. Binary cross-entropy loss

Binary cross-entropy loss is deployed for binary categorization tasks. For each word x_i in the text, the binary cross-entropy loss measures the difference between the predicted probability p_i and the actual label y_i, which is 1 for sensitive words and 0 for non-sensitive words. The loss function is defined as follows:

$$\text{Loss}_i = -[y_i * \log(p_i) + (1 - y_i) * \log(1 - p_i)] \tag{9}$$

The term y_i is the ground truth label for a word x_i (1 for sensitive, 0 for non-sensitive) and p_i is the predicted probability that the word x_i is classified as sensitive.

2. Adam optimizer

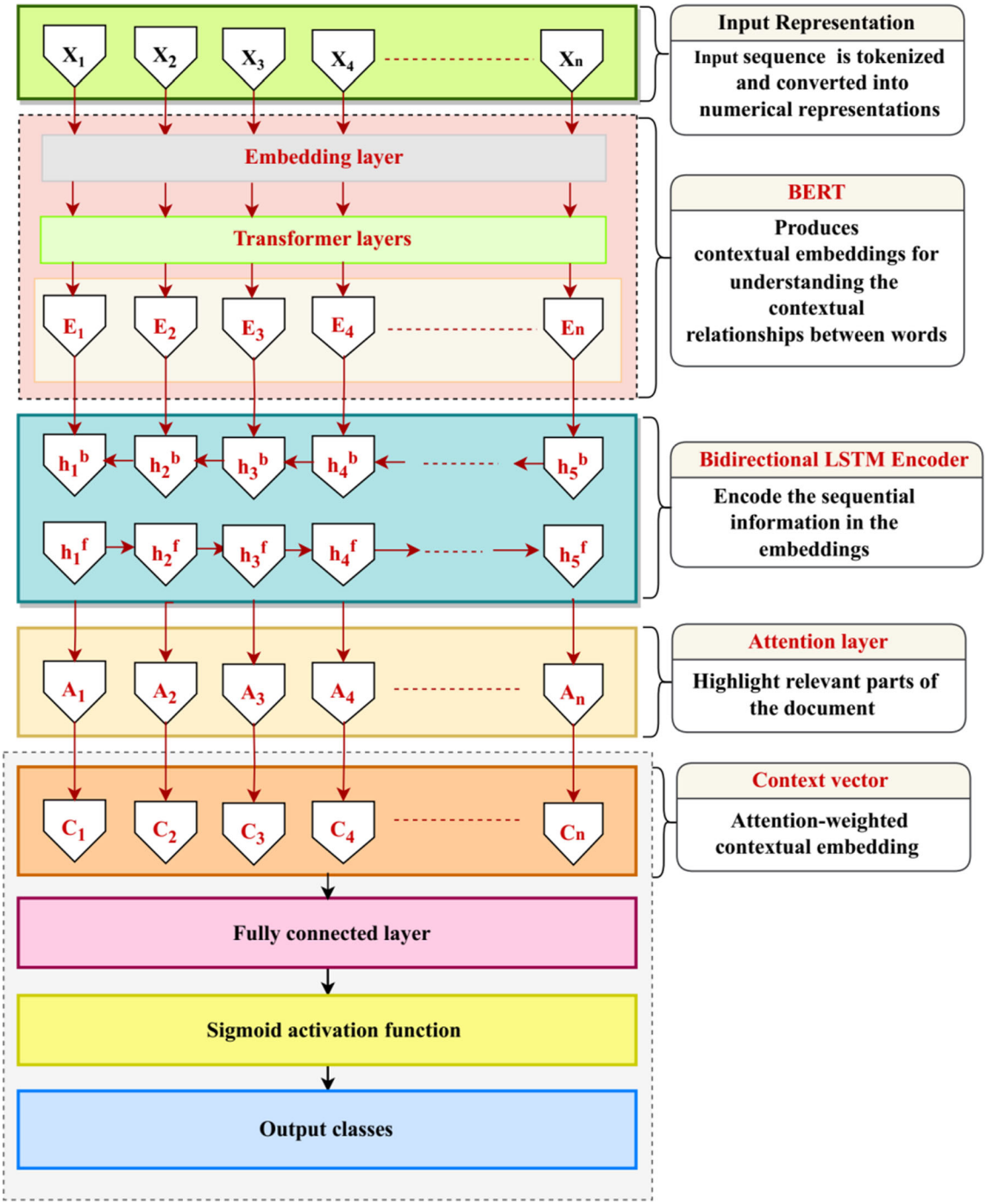

Fig. 2 Architecture of privacy BERT-LSTM algorithm for sensitive information detection

During training, the Privacy BERT-LSTM model's parameters are fine-tuned using the Adam optimizer. The Adam optimizer adjusts the learning rate adaptively based on the gradient information, which helps the model converge faster and improve its performance. The optimization process involves updating the model's parameters iteratively to enhance the model's ability to classify sensitive information and reduce the binary cross-entropy loss and.

3. Training epochs

The Privacy BERT-LSTM model undergoes several training epochs, where an epoch refers to one complete iteration over the entire training dataset. During each epoch, the model processes and learns from all instances in the dataset, gradually improving its performance over successive iterations.

4. Model evaluation

After training, the Privacy BERT-LSTM model is evaluated and tested on new, unseen data to assess its accuracy in detecting and classifying sensitive information in textual documents. Its performance is measured using recall, precision, F1-score, and overall accuracy metrics, to measure its effectiveness in identifying sensitive entities in the text.

The binary cross-entropy loss and the Adam optimizer deploy to fine-tuning the model's parameters by Privacy BERT-LSTM become optimized for sensitive information detection. The iterative training process over several epochs' aids the model learns from the data and enhances its execution eventually. From training, the method is capable of accurately classifying sensitive and non-sensitive words, making it a powerful tool for protecting privacy in textual documents. The evaluation phase ensures that the model's performance is robust and generalizes well to unseen data, which is crucial for real-world applications of sensitive information detection in various domains.

5 Results and discussion

5.1 Dataset description

The SMS Spam Collection dataset employed in our research contains labeled SMS messages, where all messages are classified as "spam" or "Hams" as given in Table 1. The dataset was initially designed for spam detection tasks but has been adapted for sensitive information detection in textual documents. The dataset contains the following attributes Text (SMS Message) and Label. The "Text" attribute represents the content of the SMS message. It contains unstructured textual data that includes sensitive privacy information, such as personal identifiable information (PII), financial records, or medical details. The "Label" attribute indicates the classification of each SMS message. It is binary, with "spam" denoting sensitive messages containing private information and "ham" representing non-sensitive messages.

The SMS Spam Collection dataset [27] typically comprises a significant number of SMS messages, making it suitable for training and evaluating machine learning models. The dataset's size depends on the specific version used, but it typically contains thousands of instances. The dataset is often characterized by class imbalance, as the number of "spam" (sensitive) messages is relatively small compared to the number of "ham" (non-sensitive) messages. This class imbalance poses a challenge in training models, as it leads to biased predictions and reduced accuracy in detecting sensitive information. For that purpose, we employ data augmentation and class weighting strategies for improving the model's performance in detecting rare sensitive information instances.

Data Source and Ethics: The SMS Spam Collection dataset is generally collected from diverse sources, such as social media platforms, communication logs, and online forums. Additionally, researchers carefully anonymize and sanitize the data to protect the privacy of individuals whose messages are included in the dataset. When using this dataset, ethical considerations are taken into account to ensure the responsible handling of sensitive information and also measures are taken to ensure data privacy and prevent the misuse of any private information present in the dataset.

Table 1 Details of SMS spam collection dataset

Labels	Number of sentences
Spams	4827
Hams	747
Total	5574

5.2 Performance metrics

In this section, we assess the effectiveness of the Privacy BERT-LSTM model in sensitive information detection on the SMS Spam Collection dataset. The following standard performance metrics are utilized for evaluation:

Precision $(\mathrm{SID}_{\mathrm{preci}})$*:*

Precision is the proportion of true positive rate to the sum of positive predictions produced by the model. It evaluates the correctness of the model in determining true positives as decreasing false positives. The formula to express precision is given below:

$$\mathrm{SID}_{\mathrm{preci}} = \frac{\mathrm{SID}_{\mathrm{TP}}}{\mathrm{SID}_{\mathrm{TP}} + \mathrm{SID}_{\mathrm{FP}}} \tag{10}$$

Recall $(\mathrm{SID}_{\mathrm{recall}})$*:*

Recall is the fraction of correctly detected samples to the total number of actual positive instances in the dataset. It evaluates the model's capacity to capturing all positive instances, thereby minimizing false negatives. The formula to compute recall is expressed below:

$$\mathrm{SID}_{\mathrm{recall}} = \frac{\mathrm{SID}_{\mathrm{TP}}}{\mathrm{SID}_{\mathrm{TP}} + \mathrm{SID}_{\mathrm{FN}}} \tag{11}$$

F1-Score $(\mathrm{SID}_{F1-\mathrm{score}})$*:*

The F1-score is the arithmetic mean of precision and recall which provides a rational assessment of the model's performance, of false positives and false negatives. A value refers a well-balanced between recall and precision. The F1-score is formulated as:

$$\mathrm{SID}_{F1-\mathrm{score}} = 2 * \frac{\mathrm{SID}_{\mathrm{recall}} * \mathrm{SID}_{\mathrm{preci}}}{\mathrm{SID}_{\mathrm{recall}} + \mathrm{SID}_{\mathrm{preci}}} \quad (12)$$

Accuracy $\left(\mathrm{SID}_{\mathrm{accuracy}}\right)$*:*

Accuracy is the fraction of correctly identified samples to the sum of all samples in the dataset. It evaluates the exactness of the detection. The mathematical expression for accuracy is:

$$\mathrm{SID}_{\mathrm{accuracy}} = \frac{\mathrm{SID}_{\mathrm{TP}} + \mathrm{SID}_{\mathrm{TN}}}{\mathrm{SID}_{\mathrm{TP}} + \mathrm{SID}_{\mathrm{TN}} + \mathrm{SID}_{\mathrm{FP}} + \mathrm{SID}_{\mathrm{FN}}} \quad (13)$$

The term $\mathrm{SID}_{\mathrm{TP}}$ indicates the True Positives (number of correctly classified sensitive words), $\mathrm{SID}_{\mathrm{TN}}$ represents the True Negatives (number of correctly classified non-sensitive words), $\mathrm{SID}_{\mathrm{FP}}$ describes the False Positives (number of non-sensitive words classified as sensitive), and $\mathrm{SID}_{\mathrm{FN}}$ indicates the False Negatives (number of sensitive words classified as non-sensitive).

5.3 Experimental setup

The proposed method for sensitive information detection is implemented in the Python platform with the system configuration of Intel(R) Core(TM) i5-6300U processor, 16 GB RAM 2.40 GHz CPU, and 64-bit operating system on Windows 10.

The data splitting process involves dividing the SMS Spam Collection dataset into three subsets in the ratio of 60:20:20. This division is typically performed randomly to ensure that each subset is representative of the overall dataset. The training set constitutes the largest portion of the dataset of around 3344 messages and is used to train the Privacy BERT-LSTM model. During training, the model learns from the labeled examples in the training set, adjusting its internal parameters (weights and biases) to minimize the loss function (binary cross-entropy) and make accurate predictions. The validation set consisting of 1114 messages is used to tune hyperparameters and prevent over fitting. Hyperparameters are the configurations of the model that are set before training begins, such as learning rate, number of layers, attention mechanism, and dropout rate. By evaluating the model's performance on the validation set, we select the best hyperparameter values that maximize the model's effectiveness in sensitive information detection. The validation set also helps identify if the model is over fitting to the training set. If over fitting occurs, adjustments are made to prevent it. The test set consisting of 1114 messages is only deployed at the end of the training phase. The primary purpose of the test set is to evaluate the Privacy BERT-LSTM model's generalization performance on new, previously unseen data. The model's accuracy in detecting sensitive information in real-world scenarios is measured using performance metrics like precision, recall, F1-score, and accuracy. The test set provides an unbiased assessment of the model's effectiveness and helps gauge how well it handles new textual documents beyond those used in training. For optimizing, the hyperparameter tuning involves searching for the best hyperparameter configurations. Different combinations of hyperparameter values are tested on the validation set, and the model's performance is evaluated for each configuration. The hyperparameters that yield the best results, maximizing the model's accuracy on the validation set, are selected for the final model. Hyperparameter tuning ensures that the Privacy BERT-LSTM model is well-configured to achieve optimal performance in sensitive information detection. Moreover, over fitting occurs when the model becomes too specific to the training data and fails to generalize to new data. Techniques like early stopping and dropout are employed to prevent over fitting. Early stopping observes the model's execution on the validation set over training. If the execution starts degrading, training is ended to inhibit the model from over fitting. Dropout randomly deactivates some neurons during training, forcing the model to rely on a diverse set of features and preventing it from becoming overly dependent on specific patterns in the training data.

The hyperparameters including learning rate, attention mechanism, batch size, number of LSTM layers, dropout rate, maximum sequence length, maximum vocabulary size, and LSTM hidden size are the hyperparameters of the proposed Privacy BERT-LSTM algorithm which is provided in Table 2.

5.4 Performance analysis

The performance analysis is carried out to determine the reliability and efficiency of the proposed Privacy BERT-LSTM algorithm. For testing purpose, a total number of 1114 messages are considered, out of which 965 messages are Spams and 150 messages are Hams. The overall performance of the proposed method is provided in Table 3.

The AUC/ROC curve describes the effectiveness of the Privacy BERT-LSTM model in distinguishing sensitive and non-sensitive information. It is a two-dimensional curve that visualizes the model performance at various threshold levels. Figure 3 represents the AUC/ROC analysis of the Privacy BERT-LSTM model. The graphical representation clearly illustrates that the proposed method is efficient and reliable in sensitive privacy information detection.

5.4.1 Effect of class imbalance

The class imbalance in the SMS Spam Collection dataset, where sensitive information instances are rare compared to

Table 2 Hyperparameters settings of the privacy BERT-LSTM algorithm

Hyperparameters	Description	Value
Learning rate	Step size for adjusting model weights	0.0001
Batch size	Number of instances in each training batch	32
Number of LSTM layers	Number of LSTM layers in the model	2
LSTM hidden size	Number of units in LSTM hidden layers	128
Attention mechanism	Type of attention mechanism used	Bahdanau Attention
Dropout rate	Fraction of neurons to deactivate during training	0.2
Maximum sequence length	Maximum length of input sequences	100
Maximum vocabulary size	Maximum size of the model's vocabulary	10,000
Number of epochs	Number of training epochs	10

Table 3 Overall performance of the proposed method

Name of the technique	Performance measures	Values (%)
Privacy BERT-LSTM algorithm	Accuracy	92.50
	Precision	89.36
	Recall	81.47
	F1-Score	85.02

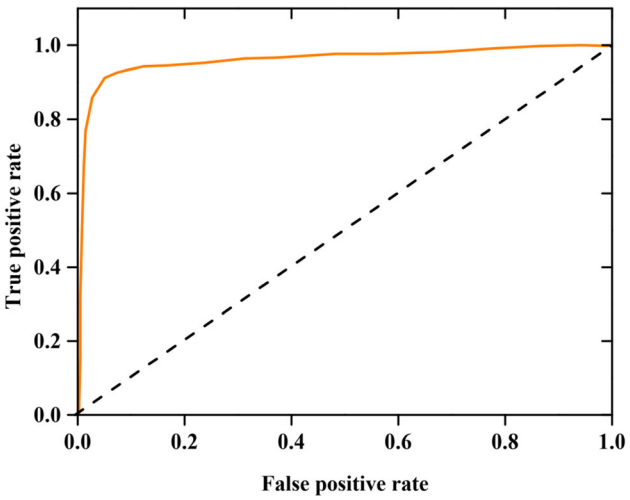

Fig. 3 AUC/ROC analysis of privacy BERT-LSTM

non-sensitive text, significantly impacts the model's performance. We conduct an analysis to understand how class imbalance affects the metrics of the Privacy BERT-LSTM model and explore strategies to address this issue. Table 4 represents the performance of the Privacy BERT-LSTM method under different scenarios related to class imbalance.

In Table 4, we compare the model's performance metrics under different scenarios of handling class imbalance. The class imbalance column represents the percentage of sensitive information instances in the dataset, highlighting the significant class imbalance issue. Without Handling column shows the model's performance metrics when trained on the original imbalanced dataset without any handling of the class imbalance issue. The Data Augmentation column presents the results after applying data augmentation techniques to increase the number of sensitive information instances. Data augmentation helps balance the dataset by creating synthetic samples, thereby mitigating the class imbalance. The class weighting column showcases the performance metrics when incorporating class weighting during model training. Class weighting assigns higher weights to the minority class (sensitive information) to give it more importance during optimization.

The execution of our technique in handling class imbalance under diverse scenarios is clearly illustrated in

Table 4 Performance metrics with class imbalance

Metric	Class imbalance (%)	Without handling (%)	Data augmentation (%)	Class weighting (%)
Precision	85.21	88.57	87.90	89.36
Recall	70.58	74.23	79.12	81.47
F1-score	77.22	80.06	83.02	85.02
Accuracy	90.45	91.35	91.78	92.50

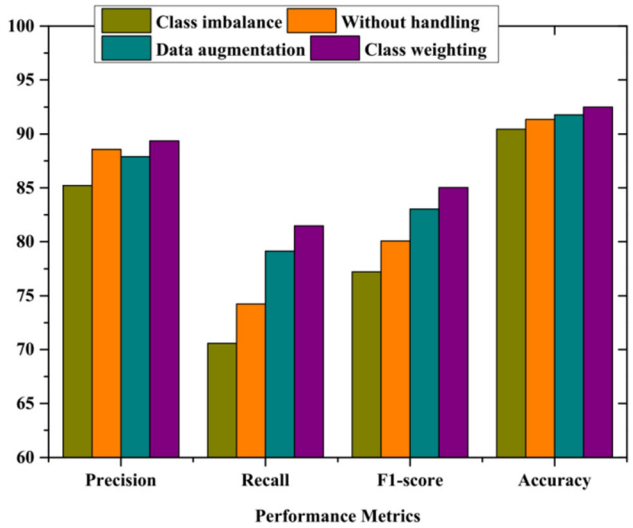

Fig. 4 Performance metrics under different scenarios related to class imbalance

Fig. 4. Figure 4 clearly shows that the utilization of data augmentation and class weighting techniques exhibits significant improvement in models performance. Therefore, we came to know that the proposed Privacy BERT-LSTM model addressed the class imbalance issue by utilizing efficient strategies to solve the data imbalance in the datasets.

5.4.2 Model interpretability analysis

The interpretability of the Privacy BERT-LSTM model is crucial for building trust and understanding its decision-making process in sensitive information detection. In this section, we analyze how the attention mechanism highlights important tokens in the text, shedding light on the model's classification decisions. We showcase specific examples to demonstrate how the model identifies sensitive information and discuss instances where it correctly or incorrectly classifies words.

The attention mechanism in the Privacy BERT-LSTM model plays a critical role in capturing the relevance and importance of individual words in the text. By computing attention weights for each word embedding, the model focuses on the most informative tokens for sensitive information detection. We visualize the attention weights to show which words the model prioritizes in its classification decisions. These visualizations make it easier to interpret the model's behavior and identify regions of high importance in the text for sensitive information classification. We present specific examples of SMS messages from the SMS Spam Collection dataset to illustrate how the Privacy BERT-LSTM model identifies sensitive information.

Figure 5 represents the attention heat map of the five SMS messages selected randomly from the SMS spam collection dataset. The words having high attention weights are represented in dark color in Fig. 5. From the figure, it is evident that the Privacy BERT-LSTM model assigns higher attention weights to sensitive words, indicating the efficiency and superiority of the proposed method in sensitive information identification.

To identify patterns in misclassifications and understand the model's limitations, we conduct an error analysis on the Privacy BERT-LSTM model's performance on the SMS Spam Collection dataset. We investigate whether certain types of sensitive information are consistently misclassified and analyze potential areas of model uncertainty. Table 5 represents the misclassifications by sensitive information type.

In Table 5, we present the count of correctly classified (True Positives) and misclassified instances (False Positives and False Negatives) for various category of sensible information in the SMS Spam Collection dataset. The Accuracy column shows the overall precision of our proposed approach in detecting each sensitive information type. From the table, we conclude that the model performs well in identifying PII, with a high number of True Positives. However, there are some instances of False Positives, indicating that the model occasionally misclassifies non-sensitive information as PII. Furthermore, the model achieves good accuracy in detecting financial information. Still, it shows a higher rate of False Positives compared to PII, suggesting that some non-sensitive messages are misclassified as containing financial data. The table also describes that the model performs reasonably well in identifying medical records, but it shows a relatively higher number of False Negatives. This indicates that some messages containing medical information are misclassified as non-sensitive. Also, the model struggles more with account numbers. This indicates that the model misclassifies some messages with account numbers as either sensitive or non-sensitive.

5.5 Comparative analysis

In this section, the performance of the Privacy BERT-LSTM model is compared with state-of-the-art baseline methods commonly used in sensitive information detection tasks. For each baseline method, we evaluate the F1-score, recall, precision, and accuracy on the SMS Spam Collection dataset. The Privacy BERT-LSTM model's performance is also evaluated using the same metrics. The objective is to demonstrate the superiority of the Privacy BERT-LSTM algorithm over existing methods in accurately classifying sensitive information in textual documents.

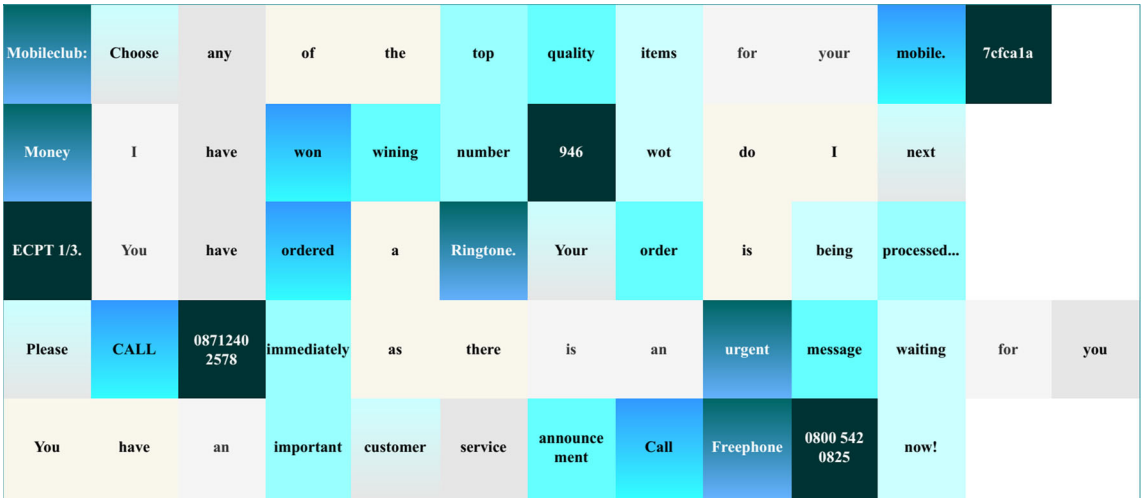

Fig. 5 Attention heat map visualization of five SMS messages from the SMS spam collection dataset

Table 5 Misclassifications by sensitive information type

Sensitive information type	True positives	False positives	False negatives	Accuracy (%)
Personal identifiable info	500	30	20	92.50
Financial information	320	45	15	88.96
Medical records	100	5	10	89.55
Account numbers	200	20	30	86.96

PRIVAFRAME, TSIIP, CNN-LSTM, Conv-FFD, C-PIIM, GCSA, and, CASSED are the existing works employed for the identification of sensitive information in the textual documents. PRIVAFRAME [5] was a knowledge graph based on a frame approach developed for the detection of sensible data specified in the Data Privacy vocabulary (DPV). TSIIP [11] was an information intelligent text-sensitive perception method for predicting sensible information. The experimental evaluation shows that the TSIIP technique surpassed other existing approaches and showed better performance. C-PIIM [17] was a clustering-based PII detection model which employed NLP and Byte-mLSTM, an unsupervised learning algorithm for the detection of PII from the textual documents. GCSA [18] was a graph convolutional network and self-attention-based algorithm for sensitive word detection. The GCSA technique employed a graph convolutional network for the extraction of structural and textual information and a self-attention mechanism for determining the degree of similarity among the candidate words and corpus. CASSED [28] was a structured sensitive data detection using the context-based model which utilizes a transformer-based BERT technique for sensitive data detection. This method was utilized as a classifier for predicting one or more sensitive data in the databases. The CNN-LSTM [29] was designed to identify the sensitive information in the text data through its superior feature representation ability. The Conv-FFD [30] was a convolution-based neural computing approach for identifying fake information in social media. This method is utilized to derive feature representation from Chinese short texts.

Accuracy is the proportion of correctly classified instances to the total number of instances in the dataset. It is used for determining the overall correctness of the

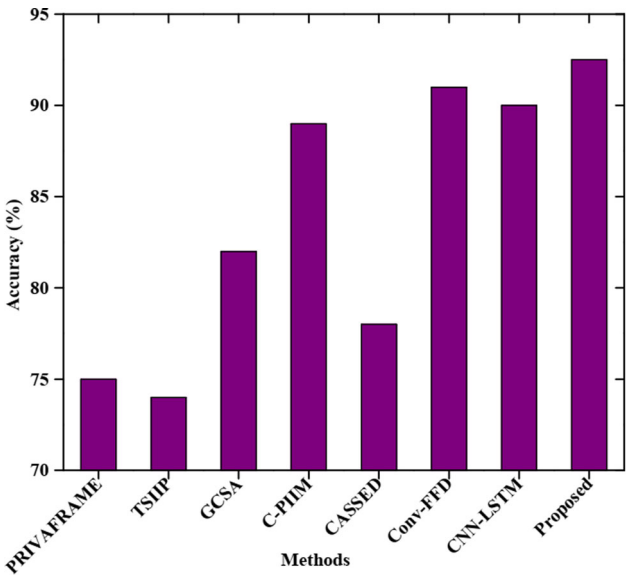

Fig. 6 Accuracy analysis

proposed Privacy BERT-LSTM model's predictions. Figure 6 represents the comparative analysis of the proposed method with other different state-of-the-art methods in terms of accuracy. From the figure, we came to know that the proposed method attains higher accuracy values, while other traditional techniques employed for comparison achieve lower accuracy values, indicating the superiority of the proposed Privacy BERT-LSTM model in sensitive data detection. The graphical representation clearly reveals that the proposed method achieves an accuracy of 92.50% illustrating the correctness of the proposed method in sensitive information detection.

The comparative analysis of the proposed method is performed by utilizing different techniques, namely, C-PIIM, TSIIP, GCSA, CNN-LSTM, Conv-FFD, PRIVAFRAME, and, CASSED, to determine the effectiveness of our proposed approach in determining true positives. The graphical representation of the precision analysis is given in Fig. 7. The graphs are plotted and the graphical findings reveal that the proposed method achieves relatively higher precision among all other existing techniques. The graph clearly illustrates that the proposed Privacy BERT-LSTM model obtains a precision of 89.36%. Therefore, the proposed method is an efficient and reliable approach for detecting true positives.

Here, the comparison is made for the proposed method with existing methods in terms of F1-Score. The different state-of-the-art methods employed for comparisons are C-PIIM, CNN-LSTM, Conv-FFD, PRIVAFRAME, TSIIP, CASSED, and, GCSA. The graphical representation of the proposed Privacy BERT-LSTM is given in Fig. 8. The x-axis describes the different existing methods as well as the proposed method. The y-axis indicates the F1-Score.

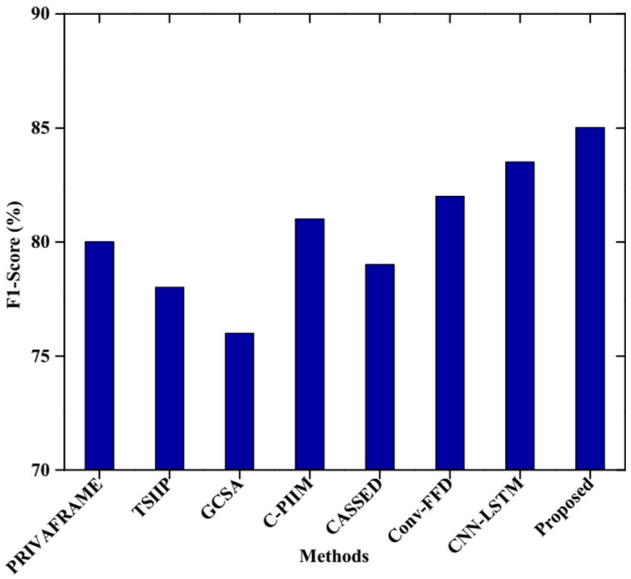

Fig. 8 F1-score analysis

From Fig. 8, we came to know that the proposed model obtains a higher F1-Score, while other previous approaches obtain a lower F1-Score. The proposed method achieves a precision of 85.02%. A high F1-score obtained by the proposed Privacy BERT-LSTM illustrates the well-balanced trade-off that exists between precision and recall.

A comprehensive comparative analysis of the proposed method with different existing techniques in terms of recall is performed. The graphical representation of the analysis of recall is provided in Fig. 9. The graph clearly states that the proposed Privacy BERT-LSTM method obtains a recall of 81.47% which is higher among all other existing techniques. Therefore, the proposed method surpassed other

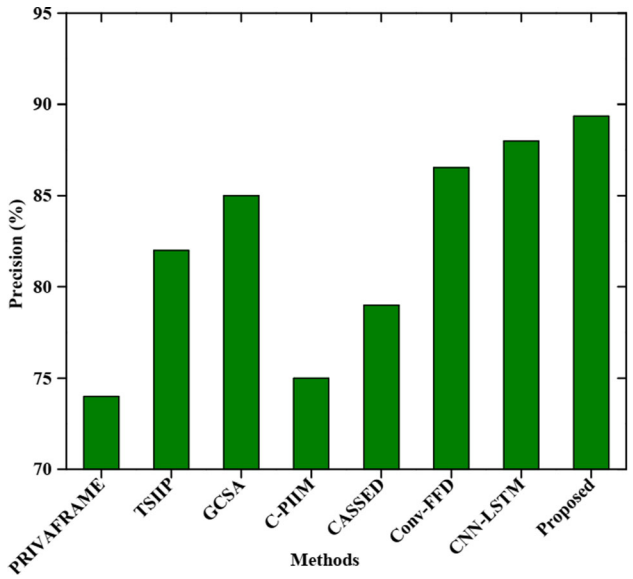

Fig. 7 Precision analysis

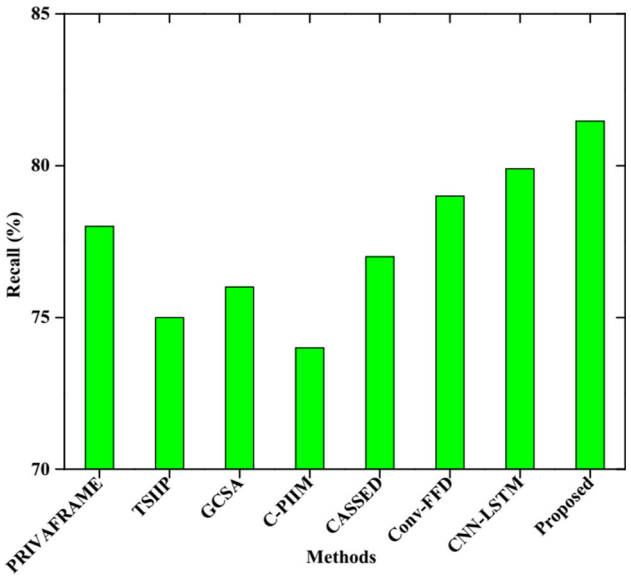

Fig. 9 Recall analysis

Table 6 Comparing computational efficiency of Privacy BERT-LSTM algorithm with other existing methods

Metrics	Methods							
	Proposed	C-PIIM	CNN-LSTM	Conv-FFD	PRIVAFRAME	TSIIP	CASSED	GCSA
Training time	45 min	50 min	1 h 5 min	55 min	1 h 20 min	1 h 35 min	1 h 40 min	1 h 55 min
Testing time (sec per instance)	0.683	2.456	6.892	3.927	9.892	8.920	10.893	11.893

existing techniques deployed for sensitive information detection in terms of recall. A high recall rate achieved by the Privacy BERT-LSTM model clearly illustrates that the proposed model is an efficient and reliable approach in determining most of the sensitive words in the text.

Table 6 represents the computational efficiency of the Privacy BERT-LSTM algorithm with other existing methods including C-PIIM, CNN-LSTM, Conv-FFD, PRIVAFRAME, TSIIP, CASSED, and, GCSA. The table clearly reveals that our Proposed Privacy BERT-LSTM algorithm is computationally efficient compared with other baseline models by achieving a training time of 45 min and a testing time of 0.683 per instance.

6 Conclusion

In this research paper, we presented Privacy BERT-LSTM, a novel NLP-based algorithm for sensitive privacy information detection and classification in textual documents. The proposed method is developed by leveraging the power of BERT's contextual embeddings and LSTM's sequential information processing, for identifying the various types of sensitive information, including personal identifiable information (PII), financial data, and medical records. The algorithm's effectiveness is evaluated and compared with other state-of-the-art baseline models using the publicly available SMS Spam Collection dataset. The experimental evaluation reveals that the proposed Privacy BERT-LSTM model outperformed existing baseline models in terms of precision, recall, F1-score, and accuracy. It also states that the proposed model is a robust and interpretable algorithm that has made several significant contributions to the field of sensitive information detection using NLP. By effectively combining BERT and LSTM, our algorithm shows promising potential for real-world applications, enabling better data security, privacy compliance, and risk mitigation in organizations dealing with vast amounts of textual data containing sensitive information. Handling class imbalance is a common challenge in sensitive information detection tasks. To address this issue, we explored and implemented data augmentation and class weighting strategies. The result illustrates this model maximizing our Privacy BERT-LSTM model performance, making it more sensitive to rare sensitive information instances. The interpretability of the Privacy BERT-LSTM model is another crucial aspect of our research. By visualizing attention heat maps and saliency maps, we gained insights into the model's decision-making process. This interpretability not only enhanced stakeholders' trust in the model's decisions but also provided valuable information for understanding the model's strengths and limitations. In conclusion, our research contributes to the advancement of privacy protection in NLP applications. Privacy BERT-LSTM offers a robust and interpretable solution for sensitive information detection, enabling organizations to better safeguard sensitive data and ensure compliance with privacy regulations. Our findings foster further research and innovation in the field, promoting the development of more accurate and reliable sensitive information detection systems.

7 Limitations and future work

Even though the Privacy BERT-LSTM model has shown outstanding performance in sensitive information detection, it also possesses some limitations which open new avenues for future research work. Firstly, the Privacy BERT-LSTM model finds it difficult to handle more complex linguistic variations and semantic understanding, thus compromising model efficiency. Hence, exploring advanced attention mechanisms, transfer learning, and multi-task learning could lead to further improvements in the model's performance. Furthermore, the Privacy BERT-LSTM model was evaluated with a limited number of data which hinders its generalizability to unseen data. Therefore, we plan to conduct a comprehensive evaluation with more number of datasets in the future.

Funding Not applicable.

Data availability The data that support the findings of this study are available from the corresponding author upon reasonable request.

Declarations

Conflicts of interest Not applicable.

Consent to participate Not applicable.

Consent for publication Not applicable.

Human and animal rights This article does not contain any studies with human or animal subjects performed by any of the authors.

Informed consent Informed consent was obtained from all individual participants included in the study.

References

1. Ohata EF, Mattos CLC, Gomes SL, Rebouças EDS, Rego PAL (2022) A text classification methodology to assist a large technical support system. IEEE Access 10:108413–108421
2. Hassan F, Sánchez D, Domingo-Ferrer J (2021) Utility-preserving privacy protection of textual documents via word embeddings. IEEE Trans Knowl Data Eng 35(1):1058–1071
3. Lynn HM, Kim P, Pan SB (2021) Data independent acquisition based bi-directional deep networks for biometric ECG authentication. Appl Sci 11(3):1125
4. Khan AR, Yasin A, Usman SM, Hussain S, Khalid S, Ullah SS (2022) Exploring lightweight deep learning solution for malware detection in IoT constraint environment. Electronics 11(24):4147
5. Gambarelli G, Gangemi A (2022) PRIVAFRAME: a frame-based knowledge graph for sensitive personal data. Big Data Cognit Comput 6(3):90
6. Zhao M, Fu X, Zhang Y, Meng L, Tang B (2022) Highly imbalanced fault diagnosis of mechanical systems based on wavelet packet distortion and convolutional neural networks. Adv Eng Inform 51:101535
7. Zhao X, Zhu X, Liu J, Hu Y, Gao T, Zhao L, Yao J, Liu Z (2024) Model-assisted multi-source fusion hypergraph convolutional neural networks for intelligent few-shot fault diagnosis to electro-hydrostatic actuator. Inf Fus 104:102186
8. Zhao X, Yao J, Deng W, Jia M, Liu Z (2022) Normalized conditional variational auto-encoder with adaptive focal loss for imbalanced fault diagnosis of bearing-rotor system. Mech Syst Signal Process 170:108826
9. Zhu X, Zhao X, Yao J, Deng W, Shao H, Liu, Z (2023) Adaptive multiscale convolution manifold embedding networks for intelligent fault diagnosis of servo motor-cylindrical rolling bearing under variable working conditions. IEEE/ASME Transactions on Mechatronics.
10. Aubaid AM, Mishra A (2020) A rule-based approach to embedding techniques for text document classification. Appl Sci 10(11):4009
11. Huo L, Jiang J (2023) Research on intelligent perception algorithm for sensitive information. Appl Sci 13(6):3383
12. Zhang K, Jiang X (2023) Sensitive data detection with high-throughput machine learning models in electrical health records. arXiv preprint arXiv:2305.03169.
13. García M, Maldonado S, Vairetti C (2021) Efficient n-gram construction for text categorization using feature selection techniques. Intell Data Anal 25(3):509–525
14. Barve Y, Saini JR, Pal K, Kotecha K (2022) A novel evolving sentimental bag-of-words approach for feature extraction to detect misinformation. Int J Adv Comput Sci Appl 13(4):266–275
15. Zhuohao WANG, Dong WANG, Qing LI (2021) Keyword extraction from scientific research projects based on SRP-TF-IDF. Chin J Electron 30(4):652–657
16. Luo X (2021) Efficient English text classification using selected machine learning techniques. Alex Eng J 60(3):3401–3409
17. Kulkarni P, Cauvery NK (2021) Personally identifiable information (pii) detection in the unstructured large text corpus using natural language processing and unsupervised learning technique. Int J Adv Comput Sci Appl 12(9):508–517
18. Liu Y, Yang CY, Yang J (2021) A graph convolutional network-based sensitive information detection algorithm. Complexity 2021:1–8
19. Roslan NIM, Foozy CFM (2022) A comparison of sensitive information detection framework using LSTM and RNN techniques. J Soft Comput Data Min 3(2):92–103
20. Victor N, Lopez D (2020) Sl-LSTM: a Bi-directional LSTM with stochastic gradient descent optimization for sequence labeling tasks in big data. Int J Grid High Perform Comput (IJGHPC) 12(3):1–16
21. García-Pablos A, Perez N, Cuadros M (2020) Sensitive data detection and classification in Spanish clinical text: Experiments with BERT. arXiv preprint arXiv:2003.03106.
22. Guo Y, Liu J, Tang W, Huang C (2021) Exsense: extract sensitive information from unstructured data. Comput Secur 102:102156
23. Qasim R, Bangyal WH, Alqarni MA, Ali Almazroi A (2022) A fine-tuned BERT-based transfer learning approach for text classification. J Healthc Eng 2022:1–17
24. Devlin J, Chang MW, Lee K, Toutanova K (2018) Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.
25. Yuan Y, Lin L, Huo LZ, Kong YL, Zhou ZG, Wu B, Jia Y (2020) Using an attention-based LSTM encoder–decoder network for near real-time disturbance detection. IEEE J Sel Top Appl Earth Obs Remote Sens 13:1819–1832
26. Deng J, Cheng L, Wang Z (2021) Attention-based BiLSTM fused CNN with gating mechanism model for Chinese long text classification. Comput Speech Lang 68:101182
27. Almeida T, Hidalgo J (2012) SMS spam collection. UCI Mach Learn Repos. https://doi.org/10.24432/C5CC84
28. Kužina V, Petric AM, Barišić M, Jović A (2023) CASSED: context-based approach for structured sensitive data detection. Expert Syst Appl 223:119924
29. Butt UA, Amin R, Aldabbas H, Mohan S, Alouffi B, Ahmadian A (2023) Cloud-based email phishing attack using machine and deep learning algorithm. Complex Intell Syst 9(3):3043–3070
30. Zhang Q, Guo Z, Zhu Y, Vijayakumar P, Castiglione A, Gupta BB (2023) A deep learning-based fast fake news detection model for cyber-physical social services. Pattern Recogn Lett 168:31–38