Exploiting Large Language Models to Train Automatic Detectors of Sensitive Data

Simone **De Renzis**[1], Dennis **Dosso**[2] and Alberto **Testolin**[1,3]

[1]*Department of Mathematics, University of Padova, Italy*
[2]*Siav S.p.A., Italy*
[3]*Department of General Psychology, University of Padova, Italy*

Abstract

This paper describes a machine learning system designed to identify sensitive data within Italian text documents, aligning with the definitions and regulations outlined in the General Data Protection Regulation (GDPR). To overcome the lack of suitable training datasets, which would require the disclosure of sensitive data from real users, the proposed system exploits a Large Language Model (LLM) to generate synthetic documents that can be used to train supervised classifiers to detect the target sensitive data. We show that "artificial" sensitive data can be generated using both proprietary or open source LLMs, demonstrating that the proposed approach can be implemented either using external services or by relying on locally runnable models. We focus on the detection of six key domains of sensitive data, by training supervised classifiers based on the BERT Transformer architecture adapted to carry out text classification and Named-Entity Recognition (NER) tasks. We evaluate the performance of the system using fine-grained metrics, and show that the NER model can achieve a remarkable detection performance (over 90% F1 score), thus confirming the quality of the synthetic datasets generated with both proprietary and open source LLMs. The dataset we generated using the open source model is made publicly available for download.

Keywords

Generative Artificial Intelligence, Sensitive data detection, NER, BERT, LLM

1. Introduction

In today's digital era safeguarding personal data has become a priority, especially with the advent of the GDPR [1]. For digital archives, it's essential to identify documents containing sensitive data, ensuring compliance and effective information management. The GDPR details two main categories of personal data: the first one includes information that can directly lead to the identification of an individual, while the second one includes a broader range of expressions that disclose sensitive aspects of a person's life. This second category is the focus of the present work and will be referred to as *sensitive data*. In particular, we deal with six key categories of sensitive data: (i) **Health**: Physical and mental well-being of individuals, with details regarding existing diagnoses, medical conditions, and disabilities; (ii) **Political**: Individual's political beliefs, their political orientation, specific party affiliation, as well as membership in

work unions or similar organizations; (iii) **Sexuality**: Individual's sexual orientation, habits, and gender identity; (iv) **Judicial**: Legal matters, such as offenses, crimes, charges, pending criminal proceedings, accusations, and trial proceedings involving an individual; (v) **Philosophy**: Individual's philosophical and religious beliefs and affiliations; (vi) **Ethnic**: Individual's ethnic origin and heritage.

The present article describes an original approach to implement a system based on machine learning classifiers to automatically detect sensitive data in text documents. The proposed method relies on Large Language Models (LLMs) to generate synthetic documents with "artificial" sensitive data, which can then be used to train Transformer-based text classifiers [2]. Our empirical investigations show that a neural model based on the Bidirectional Encoder Representations from Transformers (BERT) [3] architecture adapted for Named Entity Recognition (NER) achieves the best detection performance, both when trained using data generated by proprietary LLMs like GPT-4 [4], but also when the synthetic data is generated using open source LLMs such as OpenLLaMa [5]. The dataset generated using the open source LLM is made publicly available for download to promote further research on this domain.

The paper is structured as follows: Section 2 presents the current state of research on sensitive data detection, Section 3 details the process of automated generation and labeling of synthetic corpora, and our method based on BERT. Section 4 reports the experimental results and Section 5 discusses some limitations of our method and possible directions for future research.

2. Related Work

While the problem of detecting Personally Identifiable Information (PII) has been extensively studied in both academic and industrial settings [6], the task of identifying sensitive data has been much less explored [7].

2.1. Training corpora with sensitive data

The nature of this topic makes it difficult to find real-world documents containing sensitive data, since organizations are generally unwilling to grant access to private documents due to concerns regarding proper data handling protocols [8, 9, 10]. This is especially true in the Italian scenario, which is the specific focus of our inquiry, where research on sensitive data detection is primarily based on manually curated datasets that are not released for public use [11].

Some publicly available datasets involve classifying emails from the Enron corpus [12], detecting privacy leaks in Tweets [13] and health-related information [14]. One approach, employed by Petrolini et al. [15], involves extracting conversations from specific subsections of the Reddit forum that deal with sensitive topics. Although collecting datasets from scraped tweets or Reddit messages is a cost-effective way to obtain sensitive data, their lack of diversity may hinder their effectiveness in training models for various types of documents. Gambarelli et al. [16] manually curated two datasets containing various categories of sensitive data. Such corpora are undoubtedly of higher quality, but are also more expensive to build due to the need for manual labeling, often requiring the involvement of domain experts.

2.2. Machine learning models

A variety of machine learning models and deep learning architectures have been employed to perform Natural Language Processing (NLP) tasks, such as text classification and NER. Various architectures are involved in the domain of PII and sensitive data detection, from Convolutional Neural Networks (CNN) [17] to Transformer-based models like BERT [14]. In Karl and Scherp [18] a comparative investigation is carried out to evaluate the performance of various methods in the domain of short text classification, highlighting Transformer-based models as the best performing in terms of accuracy and speed. The BERT model is used also by Petrolini et al. [15] and Gambarelli et al. [16]. The first work proposes a method that relies on identifying a "sensitive topic" and a PII that can be linked to it. However, personal data is often mentioned separately from the related sensitive topic or may not be actually related to it. Our approach aims to make detection more robust by feeding the entire document to the classifier: this enables the model to consider the complete context and develop an understanding of the relationship between the person and its sensitive data disclosure. The second work instead introduces a multi-step inference pipeline in which a first prediction is done to distinguish between sensitive and non sensitive sentences, and then a finer inference is done to classify the category of the sensitive sentence. Our approach uses a single BERT model for prediction that discriminates between the six sensitive categories and a non sensitive one, thus speeding up the process of inference and decreasing the memory load.

3. Methods

Our proposal involves leveraging the generation capabilities of recent LLM architectures to generate documents and perform automatic labeling, reducing the data acquisition costs.

3.1. Document generation and data labeling

The procedure we propose for creating synthetic training data involves two distinct phases: *document generation*, which consists in the creation of documents of specific types, and *span labeling*, which requires to explicitly detect and categorize the sensitive data spans within the generated documents. We use the term *span* to denote a segment of text, varying in size, that is of particular interest—specifically, one that reveals sensitive information. In our experiments, we used two families of LLMs: BingAI, a chat interface integrated into Microsoft browser which is powered by GPT-4 [4]; and OpenLLaMa [5], an open and commercially permissive reimplementation of LLaMa[1].

For document generation, we defined a list of document types (e.g., clinical records, medical prescriptions, criminal records, etc.) that might contain sensitive data belonging to one of the six categories mentioned in Section 1. An automated script was devised to prompt the LLMs to generate such documents containing sensitive data. The template of the prompt looks like this: "*Puoi generare un documento di finzione ma realistico riguardante NAME del tipo "DOCUMENT_TITLE", che includa informazioni riguardo SENSITIVE_ INFO di NAME?*". This procedure was similar between BingAI and OpenLLaMA, but for the latter model a custom

---

[1]https://github.com/facebookresearch/llama/blob/main/LICENSE (Last visited: June 2023)

system prompt was instantiated, asking it to act as a document generator: this trick belongs to a set of techniques that consists on carefully crafting prompts that have been shown to improve the quality of text generation [19].

The span labeling phase has been approached in two distinct ways. For the BingAI model, a prompt was built based on the type of sensitive data the document is supposed to contain: the prompt asks to generate the document provided as input, but with the sensitive information spans "censored" or concealed with a specific tag. To guide the model in detecting specific types of sensitive information, the prompt is automatically constructed based on the known sensitive category data associated with the given document. This approach has been found to be more effective than simply asking to return the sensitive spans themselves. Similar to the prompt used for document generation, the labeling prompt also follows a structured format with specific variable words that are filled based on the document type and the associated sensitive information: "*Puoi censurare tutte e sole le porzioni di frasi che contengono informazioni o possono ricondursi a SENSITIVE_ INFO di NAME? Fornisci il documento con sole frasi che non hanno niente a che fare con SENSITIVE_ INFO di NAME. Leggendo il documento non devo essere in grado di ricostruire alcun'informazione relativa a SENSITIVE_ INFO di NAME. Usa l'etichetta [LABEL] per sostituire le porzioni di frase che contengono informazioni relative a SENSITIVE_ INFO di NAME.*"

The OpenLLaMa model, being a much smaller (13 billions parameters) and less capable model, required a few shot learning approach to get the best results. A predefined set of sentences, each with corresponding labels, is incorporated into the prompt tailored on the type of sensitive data to be labeled. Subsequently, the document to be labeled is tokenized into sentences, maintaining a consistent format with the provided examples. This approach proves effective in guiding the model to both comprehend the nuances of sensitive data and to adhere to a programmatically exploitable format for document labeling.

Supplementary documents, consisting of paragraphs extracted from Wikipedia and covering specific categories related to sensitive data, were also included in the dataset. The addition of text addressing sensitive topics, without disclosing sensitive information about individuals (e.g., general articles about politics, illnesses, etc.) was aimed to enhance the robustness of the models. In particular, this strategy helps preventing models from incorrectly associating the vocabulary of sensitive topics with the actual disclosure of sensitive information.

As a comparison, our dataset generated by OpenLLaMa comprises 26'821 data points if split at a sentence level, largely exceeding the dataset proposed by Gambarelli et al. [16], which contains 5'562 sentences in its fine-grained version. In particular, our open dataset features 370 documents related to the categories health and sexuality, 191 judicial, 96 political, 132 philosophical, 134 ethnic, 638 non sensitive and 490 of mixed categories, for a total of 2051 documents. The dataset is freely available for download along with a detailed description of its structure[2].

3.2. Sensitive data detection

We tested three different classification models, each based on a different variation of the basic BERT architecture. Due to the imbalanced distribution in the training data, where over

---

[2]https://github.com/SimoDR/sensitive-data-detection

70% of tokens correspond to non-sensitive spans, we employed a weighted softmax loss for all classification models. This approach assigns higher weights to the sensitive data class, mitigating the bias inherent in favor of the majority class, as discussed in [20]. To evaluate the models, a test set composed of 50 documents generated with BingAI was created. Notably, the test dataset was built to include also document types that were non present in the training set to further test the robustness of sensitive data detection models.

The results were evaluated in terms of precision, recall and F1 scores on the categories of sensitive data in the task of span detection. The evaluation metrics are based on Segura-Bedmar et al. [21] methodologies for NER evaluation and individual tokens serve as the unit for counting True Positives, False Negatives, and False Positives. We also tested the BingAI model as a zero shot detection model, i.e., we prompted it asking to perform NER on a document, without any other form of example.

3.2.1. Sentence Classification (SC)

We used BERT as a text classifier, where each sentence is classified into one of six sensitive categories plus a non-sensitive one. This corresponds to a *multi-class text classification task*, where each sentence serves as a distinct data point in the dataset. As discussed in Section 2, determining whether a sentence is sensitive or not is also dependent on the context in which the sentence is embedded.

3.2.2. Sentence Classification with Context (SCC)

To address the limitations of the SC model, in this version we included contextual information from the surrounding text along with each sentence to improve the classification task. Therefore, as input to this model we used two chunks of text: the one to be classified and the surrounding text, forming the context. They are separated by the special token [SEP], here used to help BERT consider the difference between the two chunks. Notably, the chunks are of fixed length, thereby obviating the need for sentence tokenization. The context also adheres to a predetermined length, ensuring consistency across the training examples. To generate the training examples for the SCC model, a sliding window approach is used. By using a stride, the training examples are partially overlapping, effectively introducing a form of data augmentation. Although this approach resolves the issue encountered in the SC model by incorporating contextual information within each chunk, the sliding window approach requires the model to perform inference on a significantly larger number of inputs, limiting its computational efficiency. As a result, this limitation has led us to treat the task as a token classification problem instead of a sequence classification problem.

3.2.3. Named Entity Recognition (NER)

This approach involves the identification and categorization of significant information, known as named entities, within a given text. By classifying each token and identifying consecutive tokens with the same label, we can concatenate them to form spans that represent specific categories. In this case, we used the BERT model with a linear layer that performs classification for each token, using a softmax function to determine the most probable label for each token.

For labeling, we adopted a variation of the BIO format: tokens are tagged as either B (beginning), I (inside), or O (outside) of an entity [22]. In our implementation, we do not use the B tag, as the frequency of chunk beginnings is relatively low compared to tokens inside and outside of chunks. We also split the documents into fixed-length chunks with a specified stride. This approach augments the data and allows the model to focus on shorter paragraphs within the text, as opposed to processing the entire document. The final dataset results to assign for each token its respective label in the format "I-" followed by one of the six sensitive categories, or "O" for the non sensitive one.

4. Results

The two rows in Table 1 present the performance metrics of the three classification models when trained on either the proprietary dataset or the open synthetic dataset All classifiers significantly outperformed the "Zero-shot BingAI" model, and the NER model achieved superior performance compared to the other classifiers on both datasets. This might be attributed to the fact that the training dataset for the NER model incorporates documents that were not specifically generated and labeled by BingAI. In this context, a unique prompt was utilized for BingAI, which differs from the labeling stage where each category of sensitive data had a distinct and personalized prompt. It also is worth mentioning that if reference examples were provided to BingAI as part of the prompt, the results might have been considerably improved. However, in this experimental setting, our objective was to evaluate the zero-shot capabilities of the model as an out-of-the-box tool.

Table 1
Performance comparison between the detection models trained on the BingAI and OpenLLaMa generated datasets. Precision, recall and F1 are weighted averages. The first two rows are referred to the span taks, the second two to the document level task.

| Detection model | SC | | | SCC | | | NER | | | BingAI (Zero-shot) | | |
Training dataset	Prec	Recall	F1	Prec	Recall	F1	Prec	Recall	F1	Prec	Recall	F1
BingAI	0.611	0.651	0.631	0.663	0.673	0.668	0.815	0.690	**0.735**	0.642	0.332	0.437
OpenLLaMa	0.649	0.700	0.663	0.620	0.710	0.654	0.734	0.728	**0.731**	0.642	0.332	0.437
BingAI	0.717	0.837	0.770	0.769	0.911	0.820	0.929	0.914	**0.921**	0.667	1.000	0.789
OpenLLaMa	0.621	0.937	0.744	0.669	1.000	0.791	0.915	0.906	**0.910**	0.667	1.000	0.789

The lower quality of the OpenLLaMa dataset results in a slightly lower, though almost negligible, detection accuracy. The graph in Figure 1 further investigates this issue by comparing how the performance of the NER model changes with different sizes of the artificial training datasets. The lower quality of the OpenLLaMA dataset requires to generate a significantly large amount of artificial samples to achieve a similar classification accuracy (2K documents vs 860).

The second two rows of Table 1 show the same comparison, but the metrics are applied at document level. Since our primary goal is to detect whether a document contains sensitive data or not, this task evaluates the models' capability to classify documents into one of the six sensitive classes or the non-sensitive class. In this assessment, each document is assigned one or more labels based on the presence of at least one span corresponding to each sensitive class in its
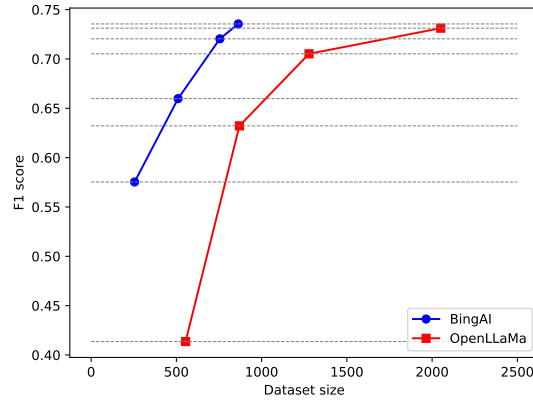
Figure 1: Span level scores (weighted F1) obtained by training the NER model on various size variations of the two datasets, generated with BingAI and with OpenLLaMa.

text. Results show that the NER model still significantly outperforms all the other approaches, reaching over 90% of weighted F1 score when trained on any of the artificial datasets.

As a final analysis, we compared the execution time and the throughput of the three classifiers by collecting data from 10 distinct runs. The SC model exhibited the lowest latency in terms of average time per document (2.09 ± 0.15 s) and the highest throughput (0.46 ± 0.03 q/s). The NER model lagged slightly behind, both in terms of average time per document (2.30 ± 0.19 s) and throughput (0.42 ± 0.04 q/s). The slowest model was SCC both for average time (13.35 ± 1.01 s) and throughput (0.08 ± 0.01 q/s). Such evaluation does not include the BingAI solution due to various factors that influence the speed of inference, such as network connection quality and current traffic conditions.

5. Conclusions

This paper introduced a novel approach to identify sensitive data in text documents, aligning with the GDPR legal foundation. The proposed method relies on LLMs to generate artificial datasets containing sensitive data: our results show that open source, smaller LLMs running on local environments (OpenLLama) can produce text of sufficient quality to train classification models that perform nearly as well as those trained on higher-quality data generated by proprietary LLMs (BingAI). Among the considered models, the NER-based model achieved a remarkable performance, with over 70% of weighted F1 score for the sentence classification task and over 90% of F1 score for the document classification task.

Future research should explore the performance of more recent open source LLMs architectures, potentially yielding superior performance in text generation and labeling accuracy. In addition, it might be interesting to test the zero-shot learning capabilities of open source LLMs: such investigation would allow to include an assessments on resource utilization, including considerations of speed, weight, and overall performance.

References

[1] European Commission, Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation) (Text with EEA relevance), 2016. URL: https://eur-lex.europa.eu/eli/reg/2016/679/oj.

[2] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, I. Polosukhin, Attention is all you need, in: I. Guyon, U. von Luxburg, S. Bengio, H. M. Wallach, R. Fergus, S. V. N. Vishwanathan, R. Garnett (Eds.), Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 2017, pp. 5998–6008. URL: https://proceedings.neurips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html.

[3] J. Devlin, M. Chang, K. Lee, K. Toutanova, BERT: pre-training of deep bidirectional transformers for language understanding (2019) 4171–4186. URL: https://doi.org/10.18653/v1/n19-1423. doi:10.18653/V1/N19-1423.

[4] OpenAI, GPT-4 technical report, CoRR abs/2303.08774 (2023). URL: https://doi.org/10.48550/arXiv.2303.08774. doi:10.48550/ARXIV.2303.08774. arXiv:2303.08774.

[5] X. Geng, H. Liu, OpenLLaMA: An open reproduction of LLaMA, 2023. URL: https://github.com/openlm-research/open_llama, online, last accessed 2023-06-19.

[6] T. Paccosi, A. P. Aprosio, REDIT: A tool and dataset for extraction of personal data in documents of the public administration domain, in: E. Fersini, M. Passarotti, V. Patti (Eds.), Proceedings of the Eighth Italian Conference on Computational Linguistics, CLiC-it 2021, Milan, Italy, January 26-28, 2022, volume 3033 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2021. URL: https://ceur-ws.org/Vol-3033/paper58.pdf.

[7] Z. Yang, Z. Liang, Automated identification of sensitive data from implicit user specification, Cybersecurity 1 (2018) 13. URL: https://doi.org/10.1186/s42400-018-0011-x. doi:10.1186/S42400-018-0011-X.

[8] G. Wilms, Guide on good data protection practice in research, European University Institute (2019). URL: https://www.eui.eu/documents/servicesadmin/deanofstudies/researchethics/guide-data-protection-research.pdf, online, last accessed 2023-11-24.

[9] G. Williams, I. Pigeot, Consent and confidentiality in the light of recent demands for data sharing, Biometrical journal 59 (2017) 240–250.

[10] C. Borgerud, E. Borglund, Open research data, an archival challenge?, Archival Science 20 (2020) 279–302.

[11] F. Lorè, P. Basile, A. Appice, M. de Gemmis, D. Malerba, G. Semeraro, An AI framework to support decisions on GDPR compliance, J. Intell. Inf. Syst. 61 (2023) 541–568. URL: https://doi.org/10.1007/s10844-023-00782-4. doi:10.1007/S10844-023-00782-4.

[12] B. Klimt, Y. Yang, The enron corpus: A new dataset for email classification research, in: J. Boulicaut, F. Esposito, F. Giannotti, D. Pedreschi (Eds.), Machine Learning: ECML 2004, 15th European Conference on Machine Learning, volume 3201 of *Lecture Notes in Computer Science*, Springer, 2004, pp. 217–226. URL: https://doi.org/10.1007/978-3-540-30115-8_22. doi:10.1007/978-3-540-30115-8_22.

[13] H. Mao, X. Shuai, A. Kapadia, Loose tweets: an analysis of privacy leaks on twitter, in:

Y. Chen, J. Vaidya (Eds.), Proceedings of the 10th annual ACM workshop on Privacy in the electronic society, WPES, ACM, 2011, pp. 1–12. URL: https://doi.org/10.1145/2046556.2046558. doi:10.1145/2046556.2046558.

[14] A. G. Pablos, N. Pérez, M. Cuadros, Sensitive data detection and classification in spanish clinical text: Experiments with BERT, CoRR abs/2003.03106 (2020). URL: https://arxiv.org/abs/2003.03106.

[15] M. Petrolini, S. Cagnoni, M. Mordonini, Automatic detection of sensitive data using transformer- based classifiers, Future Internet 14 (2022) 228. URL: https://doi.org/10.3390/fi14080228. doi:10.3390/fi14080228.

[16] G. Gambarelli, A. Gangemi, R. Tripodi, Is your model sensitive? SPEDAC: A new resource for the automatic classification of sensitive personal data, IEEE Access 11 (2023) 10864–10880. URL: https://doi.org/10.1109/ACCESS.2023.3240089. doi:10.1109/ACCESS.2023.3240089.

[17] C. Pearson, N. Seliya, R. Dave, Named entity recognition in unstructured medical text documents, CoRR abs/2110.15732 (2021). URL: https://arxiv.org/abs/2110.15732. arXiv:2110.15732.

[18] F. Karl, A. Scherp, Transformers are short text classifiers: A study of inductive short text classifiers on benchmarks and real-world datasets, CoRR abs/2211.16878 (2022). URL: https://doi.org/10.48550/arXiv.2211.16878. doi:10.48550/ARXIV.2211.16878. arXiv:2211.16878.

[19] J. White, Q. Fu, S. Hays, M. Sandborn, C. Olea, H. Gilbert, A. Elnashar, J. Spencer-Smith, D. C. Schmidt, A prompt pattern catalog to enhance prompt engineering with ChatGPT, CoRR abs/2302.11382 (2023). URL: https://doi.org/10.48550/arXiv.2302.11382. doi:10.48550/arXiv.2302.11382. arXiv:2302.11382.

[20] H. Zhu, Y. Yuan, G. Hu, X. Wu, N. Robertson, Imbalance robust softmax for deep embedding learning, in: Proceedings of the Asian Conference on Computer Vision, 2020.

[21] I. Segura-Bedmar, P. Martínez, M. Herrero-Zazo, Semeval-2013 task 9 : Extraction of drug-drug interactions from biomedical texts (ddiextraction 2013), in: M. T. Diab, T. Baldwin, M. Baroni (Eds.), Proceedings of the 7th International Workshop on Semantic Evaluation, SemEval@NAACL-HLT 2013, The Association for Computer Linguistics, 2013, pp. 341–350. URL: https://aclanthology.org/S13-2056/.

[22] L. A. Ramshaw, M. Marcus, Text chunking using transformation-based learning, in: D. Yarowsky, K. Church (Eds.), Third Workshop on Very Large Corpora, VLC@ACL 1995, 1995. URL: https://aclanthology.org/W95-0107/.