Enabling PII Discovery in Textual Data via Outlier Detection

Md. Rakibul Islam[1(✉)], Anne V. D. M. Kayem[2], and Christoph Meinel[2]

[1] Department of Computational Science, University of Potsdam, Potsdam, Germany
md.rakibul.islam@uni-potsdam.de

[2] Hasso-Plattner-Institute for Digital Engineering, University of Potsdam, Potsdam, Germany
Anne.Kayem@hpi.de, Christoph.Meinel@hpi.de

Abstract. Discovering Personal Identifying Information (PII) in textual data is an important pre-processing step to enabling privacy preserving data analytics. One approach to PII discovery in textual data is to characterise the PII as abnormal or unusual observations that can potentially result in privacy violations. However, discovering PII in textual data is challenging because the data is unstructured, and comprises sparse representations of similar text elements. This limits the availability of labeled data for training and the accuracy of PII discovery. In this paper, we present an approach to discovering PII in textual data by characterising the PII as outliers. The PII discovery is done without labelled data, and the PII are identified using named entities. Based on the recognised named entities, we then employ five (5) unsupervised outlier detection models (LOF, DBSCAN, iForest, OCSVM, and SUOD). Our performance comparison results indicate that iForest offers the best prediction accuracy with an ROC AUC value of 0.89. We employ a masking mechanism, to replace discovered PII with semantically similar values. Our results indicate a median semantic similarity score of 0.461 between original and transformed texts which results in low information loss.

Keywords: Outlier Detection · Named Entity Recognition · Data Masking · Personal Identifying Information (PII)

1 Introduction

Growing instances of information and data sharing abound on the Internet, with an increasing representation in the form of free text on social media, forums, blogs, and wikis. According to Gandomi and Haider [9], textual data makes up to 95% of all unstructured data online. Sharing textual data can inadvertently lead to sensitive information disclosure, without either the subjects concerned or the data owners being aware of it.

Discovering personal identifying information (PII) in unstructured textual data is challenging because the data does not lend itself well to labelling. This is

C. Strauss et al. (Eds.): DEXA 2023, LNCS 14147, pp. 209–216, 2023.
https://doi.org/10.1007/978-3-031-39821-6_17

mainly because unstructured textual data is comprised of sparse representations of similar text elements, that do not necessarily obey grammatical structures. This as such limits the availability of labeled data for training and the accuracy of PII discovery. PII discovery is also typically followed by masking and/or deletion which results in high information loss.

In this paper, we present an approach to discovering and masking PII in textual data by characterising PII as outliers. Our results show that iForest predicts outliers with ROC AUC value of 0.89, confirming that iForest performs well for large datasets. Detected outliers are masked to anonymise but preserve semantic similarity. Our similarity scores comparing the original and anonymised text show a median score of 0.461.

The rest of the paper is structured as follows, Sect. 2 presents related work and Sect. 3 presents our outlier detection and masking approach. Section 4 presents our results and Sect. 5, concludes the paper.

2 Related Work

Outlier detection has been researched primarily with respect to structured data [1,10,12]. Recent work also shows that approaches such as deep feature extraction using neural networks [5] and generative neural networks [17] can also be used to predict outliers. However, the correctness of labeled data impacts significantly on the performance and accuracy of these models. Unsupervised approaches such as proximity-based, density, and cluster-based methods [4,10,11] handle low dimensional numerical data well but are prone to overfitting on textual data due to assumptions about data format and distance differences [15,20]. Angle-based vector similarity is useful in estimating divergence in textual documents that are represented as feature vectors based on word occurrences, and vector cosine similarity but is not scalable to large datasets [21]. While cluster-based approaches handle large datasets well by emphasising cluster tightness but are dependent on threshold values and so are not suited to textual data [7]. Furthermore, identifying outliers in textual data using distance and density-based approaches are processing intensive in terms of similarity calculations [1]. Dimension reduction can address this problem, but incurs high information loss when applied to identifying sensitive data [3]. Alternatively, outlier identification approaches based on subspaces can address this issue by integrating pattern analysis of local data with analysis of subspaces [2], but are processing intensive [1,13]. Other work on PII discovery, focuses either on structured data [6,18] or semi-structured data [18] but assumes the availability of labelled data to support training PII discovery models, which is impractical for unstructured textual data.

We present an approach to solving the problem of PII discovery in unstructured textual data in the next section.

3 PII Discovery and Masking

Our PII discovery and masking mechanism operates in three (3) steps, namely: (1.) Named entity recognition to support feature generation, (2.) Using the named entities to support PII discovery and (3.) Replacing the identified PII with semantically similar but different values.

Table 1. Named entity categories based on spaCy NER system

Feature	Description
PERSON	People, including fictional
EMAIL	Any valid email
PHONE	Any valid phone number
NORP	Nationalities or religious or political groups
FAC	Buildings, airports, highways, bridges, etc
ORG	Companies, agencies, institutions, etc
GPE	Countries, cities, states
LOC	Non-GPE locations, mountain ranges, bodies of water
PRODUCT	Objects, vehicles, foods, etc
EVENT	Named hurricanes, battles, wars, sports events, etc
WORK_OF_ART	Titles of books, songs, etc
LAW	Named documents made into laws
LANGUAGE	Any named language
DATE	Absolute or relative dates or periods
TIME	Times smaller than a day
PERCENT	Percentage, including "%"
MONEY	Monetary values, including unit
QUANTITY	Measurements, as of weight or distance
ORDINAL	"first", "second", etc
CARDINAL	Numerals that do not fall under another type

We define an outlier as the occurrence of PII in a text. To detect outliers (PII), We are only interested in phrases that contain PII such as *name, date of birth, address, etc.*. Typically, these sensitive phrases form named entities, thus requiring the use of Named Entity Recognition (NER) [19]. Most NER systems are largely dependant on plain features and domain-specific information to learn reliably from already available supervised training corpora. We address this issue by identifying named entities (NE) using a pre-trained transition-based parser model [14]. The model constructs portions of the input sequentially using a stack data structure. To generate representations of the stack required for prediction, our NER model employs the Stack-LSTM, which augments the LSTM model with a stack pointer [8]. NER is done by detecting a single word or a collection of words that comprise an entity and classifying them into different categories. So, given a collection of comments $C_1, C_2, ..., C_n$, we want to locate all named entities and calculate their frequency count by category. Each of the named entity (NE) categories is considered a feature for detecting outliers. We selected 20 categories that can represent most of the known named entities. Table 1 describes the NE categories that we used as features to represent documents. Among them, 18 of these categories were selected based on the NER implementation of spaCy library. The remaining two (i.e., EMAIL and PHONE) were manually annotated. Thus the feature extraction process of finding PII in unstructured data reduces to locating named entities for each document and creating a feature matrix with the frequency count by each named entity category. This process gives us a concise representation of a textual document compared to the traditional bag of words model, which requires a large representational space.

Five unsupervised outlier detection models (LOF, DBSCAN, iForest, OCSVM, SUOD) were then employed for outlier detection. The PIIs (outliers) were then transformed by substituting named entities with pseudo-values. Pseudo-values are created as comparable replacement types for the named entities based on the types of named entities in the text. For instance, when an `EMAIL, PHONE,` or `DATE` is discovered as a named entity, the masking algorithm generates entities of a similar kind. We maintain a hash-table lookup approach to produce consistent masking values that translate to the same masking value each time a particular type of named item is discovered. In terms of content replacement, we used pre-defined pseudo-values to replace PII realistically without mapping to a real person. Semantic similarity, based on comparing word embeddings, is used to evaluate the distance between the original and anonymised textual data elements rather than their lexicographical similarity [16]. We trained our Word2Vec embeddings on the Common Bag of Words (CBOW) pre-trained model for performance efficiency and accuracy for representations of more frequently occurring words. The resulting word embeddings are used to calculate document similarity by measuring the cosine angle.

4 Experimental Evaluation and Results

Code for our implementation can be found at[1]. We used AirBnB review data for Berlin, Germany, compiled on 17 December 2021 containing 410, 291 reviews including spam[2]. We considered comments written in English only, for a total of 253, 908 reviews.

Using the named entities in Table 1, we pre-trained an NER system to identify named entities and calculated their frequency count by category, giving a $253,908 \times 20$ initial feature matrix. We applied dimension reduction using Principal Component Analysis (PCA) and Singular Value Decomposition (SVD)to reduce the sparsity of the feature vectors.

[1] Github Code.

[2] AirBnB Dataset.

Since we do not have any ground truth about what an outlier (PII) looks like in our context, we used domain knowledge and data analysis, to make two assumptions for labeling comments as an outlier (PII). (1) If `EMAIL` or `PHONE` is present as a named entity, we consider the comment to be an outlier (PII), and (2) likewise, for the presence of `PERSON` or `ORG` with other named entities.

$$v_i > 0;\ i \in \{EMAIL, PHONE\}$$

$$v_i > 0\ and\ v_j > 0;\ i \in \{PERSON, ORG\};\ j \notin \{PERSON, ORG\}$$

Here v_i is an element of the feature vector and i represents a category of named entities. We take the same sample of 50, 782 reviews that are used in the model implementation part. After labeling, we get 42% outlier reviews and 58% not outlier reviews.

Table 2. Execution time comparison for base models

Model	n_jobs = 1	n_jobs = −1
LOF	**3.51** s ± **274** ms	**2.32** s ± **92.9** ms
DBSCAN	8.54 s ± 799 ms	3.74 s ± 346 ms
iForest	5.23 s ± 485 ms	4.2 s ± 109 ms
OCSVM	7 min 23 s ± 8.99 s	7 min 2 s ± 24.8 s
SUOD	5 min 32 s ± 1 min 19 s	4 min 46 s ± 16.6 s

Table 2 shows the execution time of the five models. As the density calculation depends on the dimension of the dataset, dimension reduction helps run LOF and DBSCAN faster, but both do not scale well for PII discovery in unstructured textual data. iForest is slower, but scales well with growing data sizes and, due to the isolation property, is faster than density-based approaches. Also, iForest has linear time complexity and requires low memory, while SVM is based on a nonlinear kernel function which can have a complexity of up to $O(n_{features} \times n_{samples}^3)$. Table 3 illustrates the outlier score threshold, precision, recall, F1-score, ROC AUC, and PR AUC score for five models. For model evaluation, recall is the most important metric as we interested in reducing the false negative value. The table shows that based on the recall and F1-score value, SUOD, iForest, and OCSVM perform well with recall values of 0.70, 0.69, and 0.68, respectively. On the other hand, LOF and DBSCAN perform worst, with 0.33 and 0.18 recall values, respectively. Figure 1 shows the TPR (True Positive Rate)/recall versus the FPR(False Positive Rate) at various outlier score thresholds. In this case, iForest performs best with a ROC AUC of 0.86, followed by SUOD and OCSVM. LOF and DBScan performed worst, which is aligned with our previous result based on recall and F1-score (Table 3). Figure 2 and Table 3 show results of the PR curve, indicating that iForest performs best with a PR AUC of 0.78, followed by SUOD, OCSVM, LOF, and DBSCAN.

Table 3. Evaluation matrices for the models

Model	Threshold	Precision	Recall	F1-score	ROC AUC	PR AUC
LOF	1.00	0.84	0.33	0.47	0.61	0.55
DBSCAN	–	0.90	0.18	0.30	0.58	0.51
iForest	0.00	0.73	0.69	0.71	**0.86**	**0.78**
OCSVM	500.77	0.74	0.68	0.71	0.79	0.73
SUOD	−0.16	0.75	**0.70**	**0.72**	0.84	0.77

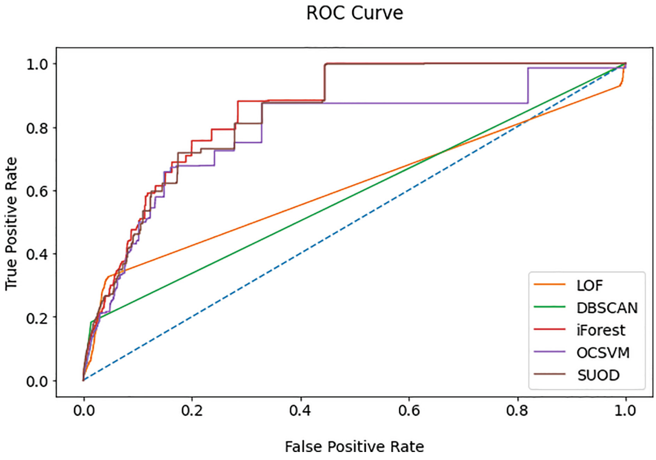

Fig. 1. ROC curve

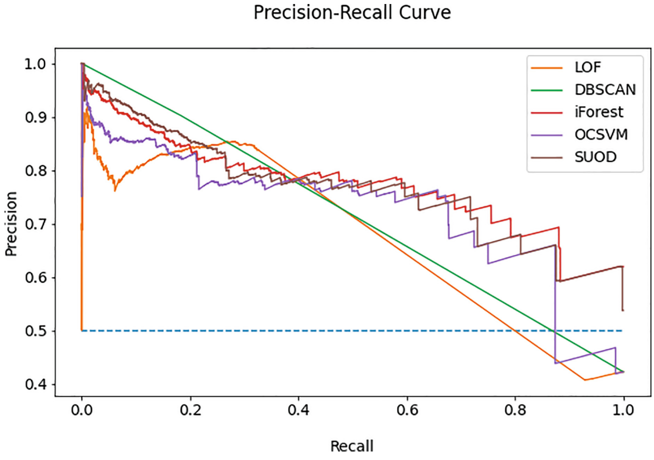

Fig. 2. Precision-Recall curve

During the data masking step, we only substitute the named entities from the outlier comments and use these named entities to generate document embeddings. This avoids the remaining terms from affecting the embeddings that are unchanged in both original and transformed comments. The results show that 50% of the anonymised comments have a similarity score between 0.357 to 0.554 with a median score of 0.461 while only 7% of the transformed comments have a similarity score less than or equal to 0. As the majority of the similarity score has a value greater than 0, we can conclude that our proposed data masking approach preserves most of the semantic properties of the original comments. LOF performs best after tuning with a recall value of 0.74. The Receiver Operating Characteristic (ROC) curves of the tuned iforest model performs best with a ROC AUC of 0.89, followed by SUOD, LOF, and OCSVM. Furthermore, iforest performs best with Precision-Recall (PR) AUC of 0.81, followed by SUOD, OCSVM, LOF, and DBSCAN.

5 Conclusion

We presented an approach to discovering personal identifying information (PII) in unstructured textual data, by characterising PIIs as outliers. We show that by using named entities it is possible to detect outliers (PIIs) using traditional unsupervised outlier detection models. Our experiments show that iForest predicts outliers with a ROC AUC score of 0.86 and a recall value of 0.69. Detected outliers are masked to anonymise but preserve semantic similarity. Our similarity scores comparing the original and anonymised text show a median score of 0.461.

References

1. Aggarwal, C.C.: An introduction to outlier analysis. In: Aggarwal, C.C. (ed.) Outlier Analysis, pp. 1–34. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-47578-3_1
2. Aggarwal, C.C., Sathe, S.: Theoretical foundations and algorithms for outlier ensembles. ACM SIGKDD Explor. Newsl. **17**(1), 24–47 (2015)
3. Blouvshtein, L., Cohen-Or, D.: Outlier detection for robust multi-dimensional scaling. IEEE Trans. Pattern Anal. Mach. Intell. **41**(9), 2273–2279 (2019)
4. Breunig, M.M., et al.: LOF: identifying density-based local outliers. In: Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data, SIGMOD 2000, pp. 93–104. Association for Computing Machinery (2000)
5. Chakraborty, D., Narayanan, V., Ghosh, A.: Integration of deep feature extraction and ensemble learning for outlier detection. Pattern Recogn. **89**, 161–171 (2019)
6. Domingo-Ferrer, J., Sánchez, D., Soria-Comas, J.: Database Anonymization: Privacy Models, Data Utility, and Microaggregation-Based Inter-model Connections, vol. 8. Morgan & Claypool Publishers (2016)
7. Duan, L., et al.: Cluster-based outlier detection. Ann. Oper. Res. **168**(1), 151–168 (2009)
8. Dyer, C., et al.: Transition-based dependency parsing with stack long short-term memory. arXiv preprint arXiv:1505.08075 (2015)

9. Gandomi, A., Haider, M.: Beyond the hype: big data concepts, methods, and analytics. Int. J. Inf. Manag. **35**(2), 137–144 (2015)
10. Hautamaki, V., Karkkainen, I., Franti, P.: Outlier detection using k-nearest neighbour graph. In: Proceedings of the 17th International Conference on Pattern Recognition (ICPR 2004), vol. 3, pp. 430–433 (2004)
11. Knorr, E.M., Ng, R.T.: Algorithms for mining distance-based outliers in large datasets. In: Proceedings of the 24rd International Conference on Very Large Data Bases, VLDB 1998, pp. 392–403 (1998)
12. Kokkula, S., Musti, N.M.: Classification and outlier detection based on topic based pattern synthesis. In: Perner, P. (ed.) MLDM 2013. LNCS (LNAI), vol. 7988, pp. 99–114. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-39712-7_8
13. Kriegel, H.-P., Kröger, P., Schubert, E., Zimek, A.: Outlier detection in axis-parallel subspaces of high dimensional data. In: Theeramunkong, T., Kijsirikul, B., Cercone, N., Ho, T.-B. (eds.) PAKDD 2009. LNCS (LNAI), vol. 5476, pp. 831–838. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-01307-2_86
14. Lample, G., et al.: Neural architectures for named entity recognition. In: Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 260–270. Association for Computational Linguistics (2016)
15. Liu, H., et al.: Efficient outlier detection for high-dimensional data. IEEE Trans. Syst. Man Cyberne.: Syst. **48**(12), 2451–2461 (2018)
16. Liu, Y., et al.: Computing semantic text similarity using rich features. In: Proceedings of the 29th Pacific Asia Conference on Language, Information and Computation, pp. 44–52 (2015)
17. Liu, Y., et al.: Generative adversarial active learning for unsupervised outlier detection. IEEE Trans. Knowl. Data Eng. **32**(8), 1517–1528 (2020)
18. Mrabet, A., Bentounsi, M., Darmon, P.: SecP2I a secure multi-party discovery of personally identifiable information (PII) in structured and semi-structured datasets. In: 2019 IEEE International Conference on Big Data (Big Data), pp. 5028–5033 (2019). https://doi.org/10.1109/BigData47090.2019.9006605
19. Nadeau, D., Sekine, S.: A survey of named entity recognition and classification. Lingvisticae Investigationes **30**(1), 3–26 (2007)
20. Sutanto, T., Nayak, R.: Semi-supervised document clustering via Loci. In: Wang, J., et al. (eds.) WISE 2015. LNCS, vol. 9419, pp. 208–215. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-26187-4_16
21. Wong, S.K.M., Ziarko, W., Wong, P.C.N.: Generalized vector spaces model in information retrieval. In: Proceedings of the 8th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 1985, pp. 18–25. Association for Computing Machinery (1985)