2021 IEEE 29th International Requirements Engineering Conference Workshops (REW) | 978-1-6654-1898-0/21/$31.00 ©2021 IEEE | DOI: 10.1109/REW53955.2021.00072

A Named Entity Recognition Based Approach for Privacy Requirements Engineering

Guntur Budi Herwanto
University of Vienna
Faculty of Computer Science
Vienna, Austria
a11947751@unet.univie.ac.at
Universitas Gadjah Mada
Yogyakarta, Indonesia
gunturbudi@ugm.ac.id

Gerald Quirchmayr
University of Vienna
Faculty of Computer Science
Research Group MIS
Währinger Straße 29
A-1090 Vienna, Austria
gerald.quirchmayr@univie.ac.at

A Min Tjoa
Vienna University of Technology
Institute of
Information and Software Engineering
Favoritenstrasse 9-11
A-1040 Vienna, Austria
a.tjoa@tuwien.ac.at

***Abstract*—The presence of experts, such as a data protection officer (DPO) and a privacy engineer is essential in Privacy Requirements Engineering. This task is carried out in various forms including threat modeling and privacy impact assessment. The knowledge required for performing privacy threat modeling can be a serious challenge for a novice privacy engineer. We aim to bridge this gap by developing an automated approach via machine learning that is able to detect privacy-related entities in the user stories. The relevant entities include (1) the Data Subject, (2) the Processing, and (3) the Personal Data entities. We use a state-of-the-art Named Entity Recognition (NER) model along with contextual embedding techniques. We argue that an automated approach can assist agile teams in performing privacy requirements engineering techniques such as threat modeling, which requires a holistic understanding of how personally identifiable information is used in a system. In comparison to other domain-specific NER models, our approach achieves a reasonably good performance in terms of precision and recall.**

***Index Terms*—privacy requirements engineering, named entity recognition, user stories, agile development**

I. Introduction

Privacy regulations increasingly oblige software developers to tailor privacy aspects into their products. Failure to comply with the rules on privacy could lead to a financial burden caused by penalties and fines which result in the loss of sensitive data and reputation. In the European Union, the General Data Protection Regulation (GDPR) has come into force in 2018 and includes regulations requiring organizations to adhere to certain standards, including privacy by design and by default. Since the enforcement, tailoring aspects of data security and privacy into the software development life cycle (SDLC) has become a key concern and a significant challenge for organizations. This challenge is especially felt by developers in agile process, where requirements engineering is a continuous process rather than a well-defined and isolated phase. To address this challenge, privacy engineering emerges as a research framework that is centered on incorporating privacy into the organizational and technical measures of an SDLC [1].

In the SDLC, requirements engineering is essential to the development of the system. Requirement elicitation with the concern of privacy has been developed in various forms such as privacy impact assessment (PIA) [2] and privacy threat modeling [3]. Requirements concerning the processing of personal data leads to a growing need for privacy protection measures. According to Pandit et al. [4], there are over a hundred categories of personal data that has been documented. These categories can be expanded according to properties that can be linked to individual. Due to the nature of natural language, there are numerous textual representations for these categories. For instance, the category of official identification can take the form of a national security number, a student identification number, or an employee identification number. Additionally, personal data is highly contextual. For example, a company's location is not considered personal data, while a person's location is considered personal data. These differences can overwhelm the software analyst / development team when having to digest them in the requirements phase [5].

Agile software development is a significant part of the current software development strategy [6]. One of the most widely used agile method is Scrum. Scrum promotes the use of user stories for requirement elicitation. According to Notario [7], there are two major approaches to elicit and handle privacy requirements using user story. The first alternative is the backlog constraints approach, which tries to link items in the user story to a specific privacy requirement. The second approach is to expand the well-known privacy requirements into user stories. The presence of an expert, such as a data protection officer (DPO) or privacy engineer can help to ease this process. However, in small teams and organizations, hiring dedicated experts can be cumbersome [8] and costly [8]. While tool support on requirement engineering [9], PIA and Threat Modeling [10], [11] can alleviate some of the effort, the manual assessment of modeling the system is still required [12].

In this paper, we aim to bridge the gap for adaptation of privacy aspects into the requirement engineering by providing an automatic approach of privacy-related entities in

978-1-6654-1898-0/21/$31.00 ©2021 IEEE
DOI 10.1109/REW53955.2021.00072

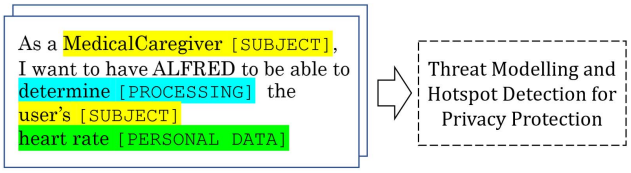

Fig. 1. Overview of automatic detection of Privacy Related Entities in a user story via Named Entity Recognition. The automated detection of entities could be useful for hotspot detection in threat modeling.

user stories. The extensive study by Aberkane et al. [13] have outlined the possibilities for Natural Language Processing (NLP) techniques to automate the privacy requirement engineering process. We developed a machine learning model that enable the detection of the privacy related entity in the user stories. The entity includes the personal data categories, the data subject, and the processing. We argue that an automated approach would support in the system's modeling process, allowing for a more agile approach to requirements engineering. We use state-of-the art Named Entity Recognition (NER) models, along with recent contextual embedding techniques. The overview of our approach is shown in Figure I. To the best of our knowledge, this is the first attempt of applying NER in the domain of privacy requirements engineering.

II. Background and Related Work

In this section, we focus on risk-based approaches and assisting tools in the privacy requirements engineering. We also provide background and related work on the named entity recognition.

A. Privacy Requirements Engineering

According to Notario [7], eliciting privacy requirements can be performed via a goal-oriented and risk-based approach. Both processes should be performed under the system development to achieve the best coverage of privacy requirements. Notario et al. [7] stated that a goal-oriented approach is easier to follow than a risk-based approach for those with less expertise in privacy engineering. In a risk-based approach, we must identify both the assets to be protected and the threats that could compromise those assets. Data Protection Impact Assessments (PIA) emerged as a systematic process for identifying and mitigating privacy risks. Supporting tools have been developed to facilitate DPIA adaptation, for example, by CAIRIS [10] and CNIL PIA [11]. To conduct a DPIA in CAIRIS, we must define roles, personas, processing, assets modeling, data flow, and perform risk analysis. Based on this requirement, extensive knowledge is required for performing DPIA, even with assisted tools. To alleviate this problem, some researchers attempt a lightweight approach on some components, such as threat modeling using a card game to simulate threat elicitation [12]. Zibuschka's [14] envisioned an automated DPIA solution focused on deriving meta-information from the current system in order to provide privacy engineers with the information they need. We propose a NER model that is capable of detecting the privacy-related entities (hotspots) automatically.

B. Named Entity Recognition

Named Entity Recognition (NER) is a task for identifying a collection of words or phrases belonging to particular Named Entity (NE) [15]. The most common entities in a NER task are person, organization, and location [16]. NER is usually performed through a number of tasks, including information retrieval [17], and knowledge based construction [18].

In SDLCs NER is used in the variety of tasks, including requirement, testing, and debugging. Mahalakshmi et al. uses NER for automatically generating test cases [19]. Entities are detected from a set of use cases based on text features such as n-gram, term frequency, dictionary reference, and some minor features. Then, these features constitute the input for the Maximum Entropy Model (MEM) to detect the named entity. Further, test cases are generated based on the named entity. NER is also used in privacy contexts, such as automatic anonymization for sensitive documents [20], automatic detection of data privacy protection (DPP) entities in legal contract documents [21], and detection of opt-out statements in privacy policies [22]. However, the use of NER to target user stories for privacy requirement engineering has yet to be investigated.

Although there are several entities to consider when conducting privacy requirements engineering, we focus on three distinct categories of privacy-related entities: personal data attributes, data subject, and processing. Personal data is defined by GDPR in Article 4(1). This GDPR article provides some examples of personal data categories, including a name, an identification number, and location data. Additionally, Pandit et al defined a more detailed description of what constitutes personal data categories [4]. They introduced several types of personal data that are not explicitly mentioned in GDPR Article 4(1), including browsing habits, family data, financial data, health records, and employment history. Aside from these examples, Finsk has built an assessment scheme in the form of a flowchart to determine whether an attribute belongs to personal data or not [23]. According to Finsk, data are classified as personal data when there is information that can be resolved from the relation between the data that can be resolved through inference, interlinking or other advanced methods. We refer to this method when performing a manual annotation on a personal data entity. To the best of our knowledge, this is the first attempt performing NER in Privacy Engineering.

III. NER Model for Privacy Related Entities

In this section, we introduce our approach to building a named-entity-recognition model for privacy related entities. Since we opted for a completely supervised model, we need a data-set that has been labeled by humans. We start with the annotation process, followed by the data augmentation approach. Once the data is established, it is passed to a semantic feature representation and a model of recurrent neural networks.

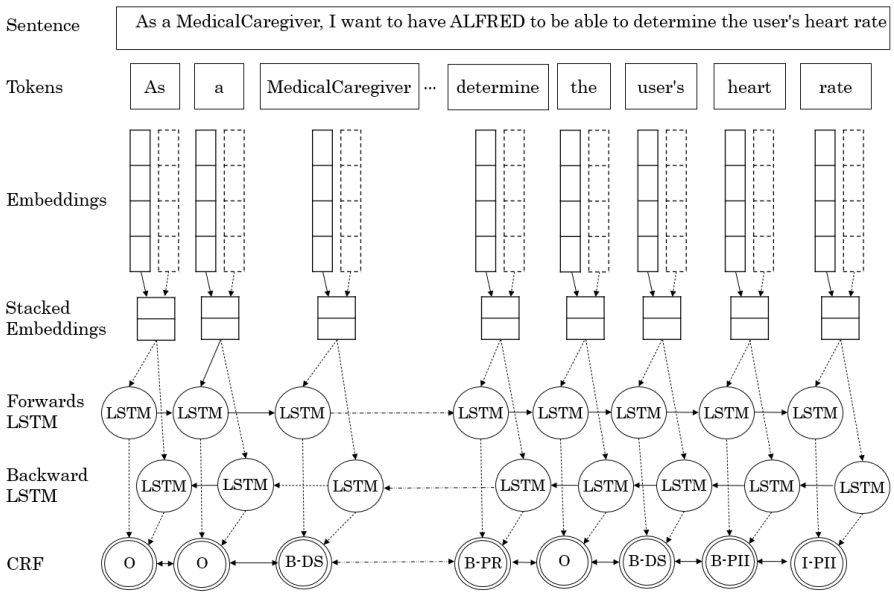

Fig. 2. The pipeline and the architecture of our model. The primary input of our model is the sentence of the user story. We tokenize the sentence and label it with the "BIO" scheme. These tokens are then passed to the embedding layer to provide contextual representations. We optionally add more (stacking) embedding to the existing embedding layer to provide richer representations. These embeddings become the input to the Forward and Backward LSTM layer. The output of the LSTM layer is then passed to the CRF layer. In the CRF layer, we denote B-DS as the beginning of the Data Subject entity, B-PR as the beginning of the Processing entity, B-PII and I-PII as the beginning and inside of a Personally Identifiable Information entity, whereas O is none entity.

A. Annotation

Our literature study has not revealed a specific data set of privacy-related entities in user requirements or user stories. As a result, we perform a manual annotation to label the privacy-related entities in the user story. We use the user stories from Dalpiaz [24] as the initial data set. The data set contains 1.680 individual user stories from 22 projects. In addition we use the description of the DPVC vocabulary [4] to add more coverage to the personal data entity.

We define three entity categories, which are Personal Data, Data Subject, and Processing. Due to the currently limited availability of user stories containing significant information related to data controllers and data processor, these subjects will have to be added in a further step. To define the category of personal data from user stories, we refer to the Data Privacy Vocabulary [4]. However, these categories are limited and do not cover all of the personal data we found in the user stories. Thus, we took the assessment scheme of Finck and Pallas [23] to determine the personal data categories. For processing, we primarily focus on the verb that follows the personal data categories. Regarding the data subject, we refer to the identifiable natural person whose personal data are being used. We base our sequence labeling task on an open-source text annotation tool called doccano [25].

B. Data Augmentation

We realize that the amount of user stories available is relatively small for training our model. Even though the data privacy vocabulary covers most of the personal data categories, it can be written differently in a user story due to the nature of human language. This problem might affect the recall of our model. To overcome this problem, we perform data augmentation.

Data augmentation is a technique to create synthetic data based on the human-labeled data-set. According to Dai and Adel [26], data augmentation is best performed in low resource scenarios, which matches our case. We use synonym replacement for the data augmentation approach. Synonym replacement aims to replace the entity with the synonym generated from the dictionary. Thus, the semantic information of the mention would not be changed. We use WordNet [27] and PPDB 2.0 [28] as our dictionary for the synonym set. Lastly, we use the nlpaug library for the implementation of the synonym replacement [29].

C. Feature Representation

To provide a semantic representation of user stories in model training, we explore various word embedding and contextual embedding methods. We use a classic word embedding model called GLoVe for the pre-trained word embedding [30]. As contextual embedding method we use Flair Embedding [31]. We also apply the transformer models BERT [32] and RoBERTa [33]. The key difference between classic and contextual embedding is that the word representation is determined by the document's context in contextual embedding. As a consequence, the vector for the same term may be different. Whereas in the classical word embedding, every word can exactly belong to one vector.

TABLE I
ANNOTATION RESULT WITH DATA SPLIT

Data	Entities		
	Data Subject	Processing	Personal Data
Training	587	527	722
Validation	214	185	239
Testing	177	142	224

D. Model Training

We address the problem of model training as a sequence labeling task. The sequence is the word or phrase that has been assigned to a particular entity. These sequences are essential to learning, due to the appearance of one entity which usually depends on the other entity. For example, the appearance of personal data attributes usually occurs before processing or vice versa. Thus, we use the supervised model approach that very well fits at dealing with bidirectional sequential data. We use the architecture proposed by Huang [34], which combines a Bidirectional Long Short Term Memory (BiLSTM) layer with conditional random field (CRF) for sequence labeling task. This architecture proved to be good at dealing with bidirectional sequence data and long-tail sequence tasks.

IV. EXPERIMENTAL RESULTS

In this section, we describe the experimental setup and presents the results and findings.

A. Experimental Setup

The main goal of our experiment is to find the best embedding features for our NER model. We use seven embedding variations, including classical word embedding, contextual embedding, and the stacked embedding between the classical word embedding and contextual embedding. We split the data into training, validation, and testing with 70 percent of training, 20 percent of validation, and 10 percent of testing. After splitting, we complement the data training with the augmented data. We run each of the augmentation methods individually, and the results of one data augmentation are not combined with another. Thus, the portion of data training for each of the augmentation methods will be doubled. Our training pipeline is built on the Flair Framework [35]. We have decided to publish our code and annotation results [1] for validation and further research.

B. Evaluation Metrics

To evaluate our model's performance, we use the standard Named Entity Recognition metrics, such as Precision, Recall, and F1-Measure. We refer to the CoNLL evaluation as it considers a full entity-level match. Thus, precision measures the ability of the system to predict the named entity correctly. Recall measures the ability of the system to predict the existence of the named entity. The F-measure is the harmonic mean of precision and recall.

[1]https://github.com/gunturbudi/ner-privacy-engineering

C. Results and Findings

This section shows the overview of our annotation result, lesson learned, and the intuition on the annotation process. The performance of our model is then presented.

1) Annotation Result: We went through all 1.680 user stories and annotated 635 stories that contained entities with privacy-related entities. We discovered that some of the projects in our dataset are not privacy-sensitive.

Duraspace projects, which serve as a repository for sharing free and open-source material, contain just 10% of privacy-sensitive user stories. On the contrary, we can capture 84% of the CamperPlus project's privacy-related user stories, which mention the processing of sensitive information about children and parents. The first findings on the annotation process is that a story followed by *my* has a high probability of becoming a personal data attribute. One typical example from the Camper Plus user stories is: *As a parent, I want to be able to track my child's activity and schedule at camp so that I can have peace of mind*. The second finding is that we can infer personal data attributes under the processing entity or the data subject itself. The example is in Alfred user stories: *As a MedicalCaregiver, I want to locate the ALFRED user*. Based on the user story, we can infer a privacy risk based on the *locate* action and *ALFRED user* as the data subject. However, the personal data attributes are not explicitly mentioned in the story. Thus, the entity data subject and processing must be considered when deciding the usage of personal data.

We are fully aware that not all personal data attributes are mentioned in the user story. Despite that, the early detection of personal data can launch further processes, such as Data Protection Impact Assessment and Threat Modeling. In addition, some user stories are already written as a privacy requirement. It might be suitable to perform a text classification rather than NER. Further, we can promote the detected entity in the user story into the threat modeling and risk assessment process.

2) Model Performance: The experiment aims to find the best contextual embedding representation and data augmentation method for our NER model. Table II depicts our NER model for detecting data subject entity, processing entity, and private data entity. We compare the performance between different embedding models as detailed in Section III-C. Additionally, we compare the output of the original (base) and augmented data sets. We only report the best augmentation method for each embedding in the Table II.

Based on the experiment, both the transformers models of BERT and RoBERTa outperforms other embedding representation. Among the three entities, Data Subject achieves the best performance. We expect this to occur due to the maturity of the NER model to detect the PERSON entity. The best model to detect the Data Subject entity is achieved by BERT, with the F_1 91,2. In processing entities, RoBERTa outperforms all the other embeddings in the processing entity, with the F_1 score of 74,4. For the personal data entity, BERT is the only embedding method able to achieve 70% for F_1 threshold with the harmonic precision and recall.

TABLE II
EXPERIMENTAL RESULTS ON NAMED ENTITY RECOGNITION

		Data Subject			Processing			Personal Data		
		Pr	Rec	F_1	Pr	Rec	F_1	Pr	Rec	F_1
GloVe	base	89,9	80,8	85,1	74,8	62,7	68,2	61,7	44,6	51,8
	syn-ppdb	91,3	82,5	86,7	78,2	68,3	72,9	65,0	47,3	54,8
GloVe+Pooled Flair	base	89,0	91,0	89,9	72,9	66,2	69,4	67,5	59,4	63,2
	syn-wordnet	91,0	91,0	91,0	72,4	73,9	73,2	67,6	64,3	65,9
BERT	base	85,1	93,2	89,0	67,1	69,0	68,1	71,8	69,2	70,5
	syn-wordnet	**89,2**	**93,2**	**91,2**	72,5	72,5	72,5	**74,7**	**71,0**	**72,8**
RoBERTa	base	85,6	94,4	89,8	71,4	73,9	72,7	69,9	68,3	69,1
	syn-ppdb	88,3	89,3	88,8	**76,3**	**72,5**	**74,4**	70,1	67,9	68,9

BERT has the best overall and average results in all three categories. In the augmented results, none of the three entities have any metrics that fall below a 70% score. It can be shown that the augmentation method successfully improves the performance on all of the original data, on almost all embeddings. Aligned with Dai and Adel's findings, [26] synonym replacement performs best on the transformers model. In the synonym replacement, PPDB 2.0 and WordNet do not show a single superiority over another, thus, both can be used as basis for the augmentation approach to synonym replacement.

3) Use Case: To demonstrate our NER model, we have inferred privacy-related entities from a collection of user stories that was not part of our corpus. We choose an open source project called Solid Project [2], which hosts the user stories in their repository [3]. We apply a BERT model that was trained on WordNet synonym replacement.

Based on the inference process, we were able to capture 59 distinct Data Subject, 138 Processing and 100 distinct Personal Data categories from the 114 individual Solid user stories. Here is one example of an inferred user story:

> As a **existing Solid user** [Data Subject], I would like to **use** [Processing] my **identity** [Personal Data] to **register** [Processing] a Pod with another Provider so that I can have different Pods for different purposes, or to **migrate data** [Processing].

The example demonstrates the ability of our model to capture the privacy-related entity. Due to the nature of supervised model, our model is able to capture the entity outside of the trained data. Our model also is able to differentiate when an entity is not a personal data, even if the pattern is likely personal data. Consider the following user story:

> As an **IoT developer** [Data Subject], I want to have a lightweight authentication mechanism so that my very small microcontrollers can **authenticate** [Processing] themselves

The example demonstrates that the phrase "my very small microcontrollers" is not a PII entity. However, we consider that "IoT developer" is a False Positive for the Data Subject prediction, as it cannot be considered a data subject for this user story.

4) Failure Analysis: The performance of the processing and personal data entity prediction fall short compared to the data subject entity. Based on our analysis, the ambiguity of the entity and the typical pattern can result in false detection. Due to the fact that not all of the verb in user story is a processing entity, and not all of the data in users story is personal data, false predictions are possible. As shown in the previous example, the typical pattern of data subjects results in a false positive prediction.

V. THREATS TO VALIDITY

As we use supervised NER model, the annotation data relies on human judgment, which leads to a major limitation for the objectivity of the approach. Therefore, we provide annotation guidelines for the annotator to mitigate the validity threat. We derive the guidelines from the data privacy vocabulary [4] which describes the data subject, processing, and the categories of personal data from the legal perspective. We also provided other supporting research [23] that can strengthen our view on the personal data. Additionally, we recheck and discuss the annotation result with the expert. However, the limitation of cost and human resources in our research restricts us from obtaining inter-coder reliability of data annotation. Nevertheless, we believe our research can lay the groundwork for the broader adoption of applying the state-of-the-art in NLP [13] in the context of privacy requirements engineering.

VI. CONCLUSION AND FUTURE WORK

The research described in this paper was aimed at laying the foundations for automating the privacy engineering process by named entity recognition on personal data, data subject, and processing. By leveraging state-of-the-art feature representation and deep learning algorithms for our task, we can achieve a reasonably good performance on all three entities.

We plan to extend our model's coverage to include GDPR-mandated entities such as data controllers and data processors in the near future. Additionally, we intend to integrate our NER model into further privacy requirement engineering processes, including threat modeling and a model-based approach. A group study examining the effectiveness of this approach in

[2]https://solidproject.org/
[3]https://github.com/solid/user-stories/issues

real-world privacy requirement engineering situations would also be necessary. These processes are typically performed by a human data protection expert who identifies the potential privacy issues in the requirements. By leveraging NLP, the possibility of automating the process of privacy requirement engineering can alleviate the burden of providing an expert on a small team. This model also supports software development teams in identifying privacy issues early in the development process.

ACKNOWLEDGMENT

The authors acknowledge the scholarship granted by the Indonesia Endowment Fund for Education (IEFE/LPDP), Ministry of Finance, Republic of Indonesia.

REFERENCES

[1] S. Gürses and J. M. Del Alamo, "Privacy Engineering: Shaping an Emerging Field of Research and Practice," *IEEE Security and Privacy*, vol. 14, no. 2, pp. 40–46, 2016.

[2] M. C. Oetzel and S. Spiekermann, "A systematic methodology for privacy impact assessments: A design science approach," *European Journal of Information Systems*, vol. 23, no. 2, pp. 126–150, 2014.

[3] M. Deng, K. Wuyts, R. Scandariato, B. Preneel, and W. Joosen, "A privacy threat analysis framework: supporting the elicitation and fulfillment of privacy requirements," *Requirements Engineering*, vol. 16, no. 1, pp. 3–32, 2011.

[4] H. J. Pandit, A. Polleres, B. Bos, R. Brennan, B. Bruegger, F. J. Ekaputra, J. D. Fernández, R. G. Hamed, E. Kiesling, M. Lizar, E. Schlehahn, S. Steyskal, and R. Wenning, "Creating a vocabulary for data privacy," in *On the Move to Meaningful Internet Systems: OTM 2019 Conferences* (H. Panetto, C. Debruyne, M. Hepp, D. Lewis, C. A. Ardagna, and R. Meersman, eds.), (Cham), pp. 714–730, Springer International Publishing, 2019.

[5] A. Senarath, M. Grobler, and N. A. G. Arachchilage, "Will they use it or not? Investigating software developers' intention to follow privacy engineering methodologies," *ACM Transactions on Privacy and Security*, vol. 22, no. 4, 2019.

[6] B. Kostova, S. Gürses, and C. Troncoso, "Privacy engineering meets software engineering. on the challenges of engineering privacy bydesign," 2020.

[7] N. Notario, A. Crespo, Y. S. Martin, J. M. Del Alamo, D. L. Metayer, T. Antignac, A. Kung, I. Kroener, and D. Wright, "PRIPARE: Integrating privacy best practices into a privacy engineering methodology," *Proceedings - 2015 IEEE Security and Privacy Workshops, SPW 2015*, pp. 151–158, 2015.

[8] J. Zibuschka and C. Zimmermann, "Lean privacy by design," *SICHERHEIT 2020*, 2020.

[9] N. Kiyavitskaya and N. Zannone, "Requirements model generation to support requirements elicitation: The secure tropos experience," *Automated Software Engg.*, vol. 15, p. 149–173, June 2008.

[10] J. Coles, S. Faily, and D. Ki-Aries, "Tool-supporting data protection impact assessments with cairis," in *2018 IEEE 5th International Workshop on Evolving Security & Privacy Requirements Engineering (ESPRE)*, pp. 21–27, IEEE, 2018.

[11] CNIL, "CNIL PIA Software," 2019.

[12] K. Wuyts, L. Sion, and W. Joosen, "Linddun go: A lightweight approach to privacy threat modeling," in *2020 IEEE European Symposium on Security and Privacy Workshops (EuroS PW)*, pp. 302–309, 2020.

[13] A.-J. Aberkane, G. Poels, and S. V. Broucke, "Exploring automated gdpr-compliance in requirements engineering: A systematic mapping study," *IEEE Access*, vol. 9, pp. 66542–66559, 2021.

[14] J. Zibuschka, "Analysis of automation potentials in privacy impact assessment processes," in *Computer Security* (S. Katsikas, F. Cuppens, N. Cuppens, C. Lambrinoudakis, C. Kalloniatis, J. Mylopoulos, A. Antón, S. Gritzalis, F. Pallas, J. Pohle, A. Sasse, W. Meng, S. Furnell, and J. Garcia-Alfaro, eds.), (Cham), pp. 279–286, Springer International Publishing, 2020.

[15] A. Mikheev, M. Moens, and C. Grover, "Named entity recognition without gazetteers," in *Ninth Conference of the European Chapter of the Association for Computational Linguistics*, 1999.

[16] J. Li, A. Sun, J. Han, and C. Li, "A survey on deep learning for named entity recognition," *IEEE Transactions on Knowledge and Data Engineering*, pp. 1–1, 2020.

[17] D. Ye, Z. Xing, C. Y. Foo, Z. Q. Ang, J. Li, and N. Kapre, "Software-specific named entity recognition in software engineering social content," in *2016 IEEE 23rd International Conference on Software Analysis, Evolution, and Reengineering (SANER)*, vol. 1, pp. 90–101, 2016.

[18] C. Chen, Z. Xing, and X. Wang, "Unsupervised software-specific morphological forms inference from informal discussions," in *2017 IEEE/ACM 39th International Conference on Software Engineering (ICSE)*, pp. 450–461, IEEE, 2017.

[19] G. Mahalakshmi, V. Vijayan, and B. Antony, "Named entity recognition for automated test case generation," *Int. Arab J. Inf. Technol.*, vol. 15, pp. 112–120, 2018.

[20] F. Hassan, D. Sánchez, J. Soria-Comas, and J. Domingo-Ferrer, "Automatic anonymization of textual documents: Detecting sensitive information via word embeddings," in *2019 18th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/13th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE)*, pp. 358–365, 2019.

[21] S. P. Nayak and S. Pasumarthi, "Automatic detection and analysis of dpp entities in legal contract documents," in *2019 First International Conference on Digital Data Processing (DDP)*, pp. 70–75, 2019.

[22] V. Bannihatti Kumar, R. Iyengar, N. Nisal, Y. Feng, H. Habib, P. Story, S. Cherivirala, M. Hagan, L. Cranor, S. Wilson, *et al.*, "Finding a choice in a haystack: automatic extraction of opt-out statements from privacy policy text," in *Proceedings of The Web Conference 2020*, pp. 1943–1954, 2020.

[23] M. Finck and F. Pallas, "They who must not be identified-distinguishing personal from non-personal data under the GDPR," *Forthcoming, International Data Privacy Law*, pp. 14–19, 2020.

[24] F. Dalpiaz, "Requirements data sets (user stories)," *Mendeley Data, V1*, 2018.

[25] H. Nakayama, T. Kubo, J. Kamura, Y. Taniguchi, and X. Liang, "doccano: Text annotation tool for human," 2018. Software available from https://github.com/doccano/doccano.

[26] X. Dai and H. Adel, "An analysis of simple data augmentation for named entity recognition," in *Proceedings of the 28th International Conference on Computational Linguistics, COLING 2020, Barcelona, Spain (Online), December 8-13, 2020* (D. Scott, N. Bel, and C. Zong, eds.), pp. 3861–3867, International Committee on Computational Linguistics, 2020.

[27] C. Fellbaum, "Wordnet," in *Theory and applications of ontology: computer applications*, pp. 231–243, Springer, 2010.

[28] E. Pavlick, P. Rastogi, J. Ganitkevitch, B. Van Durme, and C. Callison-Burch, "PPDB 2.0: Better paraphrase ranking, fine-grained entailment relations, word embeddings, and style classification," in *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, (Beijing, China), pp. 425–430, Association for Computational Linguistics, July 2015.

[29] E. Ma, "Nlp augmentation." https://github.com/makcedward/nlpaug, 2019.

[30] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1532–1543, 2014.

[31] A. Akbik, T. Bergmann, and R. Vollgraf, "Pooled contextualized embeddings for named entity recognition," in *NAACL 2019, 2019 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, p. 724–728, 2019.

[32] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[33] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "Roberta: A robustly optimized bert pretraining approach," 2019.

[34] Z. Huang, W. Xu, and K. Yu, "Bidirectional lstm-crf models for sequence tagging," *arXiv preprint arXiv:1508.01991*, 2015.

[35] A. Akbik, T. Bergmann, D. Blythe, K. Rasul, S. Schweter, and R. Vollgraf, "Flair: An easy-to-use framework for state-of-the-art nlp," in *NAACL 2019, 2019 Annual Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pp. 54–59, 2019.