

Privacy Impact Assessment of Third-Party Dependencies

MSc Industrial Internship
Cyber Security

Kevin Shaji
Student ID: x22108718

School of Computing
National College of Ireland

Supervisor: Vikas Sahni

National College of Ireland
MSc Project Submission Sheet
School of Computing



Student Name:Kevin Shaji.....
Student ID: x22108718.....
Programme:MSc Cyber Security..... **Year:** ...2023.....
Module:Industrial Internship.....
Supervisor:Vikas Sahni.....
Submission Due Date:January 5th 2024.....
Project Title:Privacy Impact Assessment of Third-Party Dependencies.
Word Count:6657..... **Page Count:**.....22.....

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: Kevin Shaji.....

Date: January 5th 2024.....

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	<input type="checkbox"/>
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	<input type="checkbox"/>
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	<input type="checkbox"/>

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Privacy Impact Assessment of Third-Party Dependencies

Kevin Shaji
X22108718

Abstract

This research focuses on developing a novel framework for Privacy Impact Assessment (PIA) within the context of third-party dependencies in software development. The increasing reliance on third-party libraries and services has escalated data privacy and security concerns among users. However, existing research does not systematically or standardize the assessment of these dependencies privacy implications.

Recognizing the critical need for data protection and the absence of a specific evaluation framework for third-party dependencies, this study aims to fill the gaps in the current literature by developing a robust PIA methodology. The methodology involved examining privacy policies and data access behaviours of third-party dependencies using Natural Language Processing (NLP) techniques to identify potential privacy risks.

The research made valuable contributions, such as creating a detailed framework for Privacy Impact Assessments (PIA) and establishing a risk rating score. This score offers valuable insights for safely integrating third-party dependencies. Ultimately, this research aims to enhance data protection and user privacy within the software development lifecycle. The findings of the completed research contributed significantly to enhancing privacy and security practices concerning third-party dependencies in software development.

Keywords: PIA: Privacy impact assessments, TPL: Third party libraries, Privacy Compliance.

1 Introduction

In the rapidly evolving landscape of digital technologies, the increasing need for third-party dependencies in software development poses significant challenges to the preservation of user privacy. Using external tools like libraries, frameworks, and services has become a common practice in today's software development. Although these tools bring various benefits such as improved efficiency, enhanced functionality, and faster project developments, The history of this practice has led to a concerning lack of attention to privacy. As digital systems expanded, concerns about privacy is also getting higher, emphasizing the need to carefully examine how the inclusion of external components in software projects impacts user privacy and information security.

The historical evolution of third-party dependencies has given rise to several critical issues. Privacy policies are the main documents that explain how dependencies handle data. They are usually long, complicated, and full of legal terms. This makes it difficult for developers and compliance teams to understand and assess the privacy impacts described in these documents from different software's. Consequently, the lack of a standardized methodology for assessing the privacy impact of third-party dependencies also affect the developers ability to make informed decisions about the software components they integrate into the systems they are building.

This research addresses the need to conduct a thorough Privacy Impact Assessment (PIA) of the third- party dependencies embedded within Cubic Telecom's¹ automotive software solutions. The primary contribution of this research lies in the development and validation of a practical Privacy Impact Assessment framework dedicated for third-party dependencies. By offering a structured approach to assess and compare the privacy issues of various dependencies.

1.1 Research Question

The central research question guiding this study is:

How can a systematic analysis of privacy policy documents for third-party dependencies be leveraged to provide valuable insights and facilitate informed decision-making in software development ?

The automotive industry, in particular, stands at the forefront of this transformative wave, as it shifts towards connected and autonomous vehicles. Cubic Telecom, a pioneering company specializing in automotive software solutions, is not exempt from these challenges. Project teams stand to benefit significantly from a standardized PIA framework, which streamlines the process of evaluating third-party dependencies, enhances transparency, and ultimately supports the creation of more privacy-respecting software. Additionally, policymakers, and legal teams within the organization is also benefited from the knowledge generated, as it informs the development of more robust privacy regulations and standards of third-party dependencies in automotive software.

1.2 Report Structure

The remaining structure of this research paper is outlined as follows: Section 2 focuses into related work regarding privacy impact assessments. Section 3 details the methodology for executing the proposed approach. Section 4 presents the design specifications for the given approach. Section 5 offers detailed steps of the implementation process. Lastly, Sections 6 and 7 discuss the evaluation results and conclusions derived from the findings.

¹ <https://www.cubictelecom.com>

2 Related Work

Previous works have investigated the privacy and information assurance risks arising from the use of third-party dependencies in software development.

The research conducted by Zhao et al. (2023) is strictly focused on privacy issues in Android applications, They pointed out the importance of safeguarding personal information privacy in mobile software and the increasing issues regarding privacy adherence concerning third-party libraries (TPLs).(Cheng *et al.*, 2023) introduced methods for identifying conflicts between app behaviour and privacy policies, there still remains a lack of understanding regarding TPLs' privacy compliance.

In 2022, (Del Alamo *et al.*, 2022) presented a summary of methods for automatically analysing privacy policy texts. This work is essential for grasping the basics of automated privacy policy analysis. The researchers categorized the different approaches employed, pointed out the different types of information extracted, and pointed into the goals making these analyses. This paper serves as the foundation for gaining insight into the field of automated privacy policy analysis.

Wilson (Wilson *et al.*, 2016) concentrated on automating the identification of key information from privacy policy texts. They achieved this through a combination of crowdsourcing, natural language processing (NLP), and machine learning techniques. One of the notable contribution of their work was the development of a corpus with manual annotations, specifically focusing on detailed data practices. This effort is significant because it helps overcome the limited availability of datasets that can be used to identify data practices within privacy policies.

Farhana Nazir et al. (2017) explored NLP applications in Software Requirement Engineering (SRE),(Nazir *et al.*, 2017) highlighting the need for manual operations on initial plain text software requirements before applying NLP techniques. (Tulili, Capiluppi and Rastogi, 2023)Guzman et al. conduct a systematic mapping study focused on software quality control techniques for assessing privacy in information systems. The study provides an overview of state-of-the-art techniques, offering insights into the landscape of privacy-focused software quality control. (Tsfay *et al.*, 2018) proposed a machine learning-based approach to summarize lengthy privacy policies into concise notes, aligning with the risk-based approach and European Union General Data Protection Regulation (GDPR) aspects. The paper contributes to addressing the challenge of condensing and assessing privacy policies effectively.

(Chaw and Chua, 2021) introduced a framework system using the Word Mover's Distance text similarity algorithm for assessing privacy policy compliance. (Koroteev, 2021) conducted a comprehensive review of BERT's applications in natural language processing. (Qi *et al.*, 2023) contributed to automated privacy policy analysis by introducing PoliGraph, a knowledge graph for capturing statements in privacy policies. These works, while invaluable, do not provide a standardized framework for evaluating the privacy impact of third-party dependencies, which this research aims to introduce.

Building on these foundations, new studies have emerged to address the nuanced challenges of third-party dependencies. (van Daalen, 2023) evaluated the data privacy implications of third-party libraries on iOS and Android, providing insights into compliance discrepancies across platforms. (Clarke, 2009) focused on the risks third-party service providers pose to enterprise data privacy, underscoring the need for meticulous PIAs in managing vendor relations.

Furthermore, (Wright and De Hert, 2012) discussed the pivotal role of consent management in data privacy, particularly within third-party ecosystems, aligning with GDPR compliance. (Wagner and Ford, 2020) offered methodologies to incorporate privacy risk assessments within agile software development, advocating for 'Data Protection by Design'. The automation of compliance for software development components was explored by (Castellanos-Ardila, Gallina and Governatori, 2021), while (Chhetri *et al.*, 2022) evaluated the efficacy of automated tools in maintaining GDPR compliance for software systems. (Ravichander *et al.*, 2021) explored how NLP enhances the accessibility of privacy policies, improving the clarity of privacy practices related to third-party dependencies for users.

Additional works that contribute to this area include (Amos *et al.*, no date), which compares computational techniques for parsing privacy policies; (Andrade *et al.*, 2022) reviewing Privacy by Design principles in software engineering; and (Mannan *et al.*, 2019), presenting a framework for assessing the privacy controls of SDKs.

While the existing body of research provides valuable insights into automated privacy policy analysis, corpus creation, NLP applications in software development, and software quality control for privacy, there remains a gap in the development of a standardized framework specifically developed for evaluating the privacy impact of third-party dependencies. The existing solutions don't meet the requirement for a structured Privacy Impact Assessment (PIA) framework to help developers make informed choices when picking third-party tools, considering their impact on privacy. This research seeks to bridge this gap by introducing a new PIA framework. This will improve the tools currently at developers' disposal for creating software that prioritizes privacy-conscious software development.

In conclusion, The evaluation system for assessing the Privacy Impact of third-party dependencies provides a comprehensive method to deal with privacy risks that may arise when incorporating external libraries into applications. By introducing a detailed framework, the study can make a significant contribution to enhancing the privacy protection of software applications and the responsible management of user data in relation to third-party dependencies.

3 Research Methodology

The methodological approach involved an analysis of the potential risks introduced by using third-party libraries, frameworks, and services into software applications. This research developed and evaluated a PIA framework. The methodology followed a systematic process which includes data collection, data cleaning, manual and automated annotations, and the development of a scoring framework for privacy associated with third-party dependencies.

Conducted a thorough research from the inputs from the project teams and the compliance teams regarding the insights required from a privacy policy document. The methodology steps in Figure 1 was followed throughout the research for the development of a novel framework for analysis. The ultimate goal is to provide project teams and organizations the information of potential privacy risks and ensure compliance with relevant regulations and best practices.

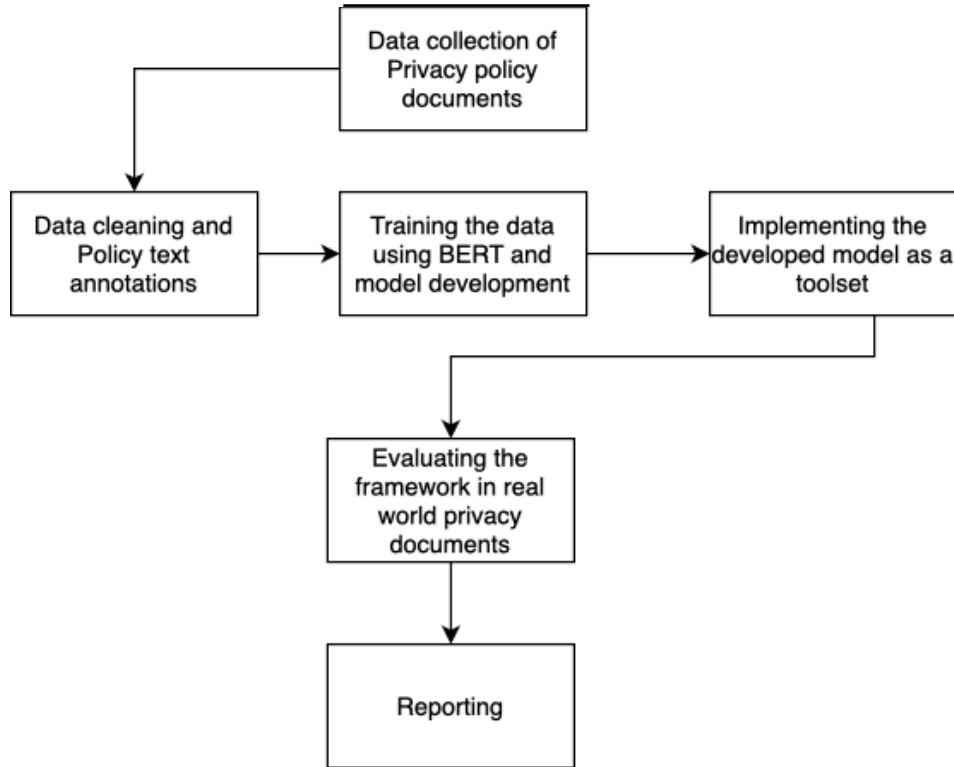


Figure 1: Research Methodology steps.

3.1 Framework Development

Framework Development: Based on the findings from the literature review, the research was designed and developed the Privacy Impact Assessment framework for third-party dependencies. The framework considered various factors, including data flow, data access, data storage, and data sharing patterns of a dependency using the privacy policy documents within the software system. The developed program can be used as a command line tool to analyse the privacy policy documents on the fly and as well as implemented as an API (Application programming interface) service which can be used to use it as a service based application within the organisation. The following categories of insights are derived from the model after a successful analysis of a privacy document.

Data Minimization: Evaluate whether the dependency collects only the necessary data required for its intended purpose. Minimizing the amount of data collected helps reduce privacy risks.

Consent Management: Evaluate the presence of granular and explicit consent options within the document. It falls under the category of how user consent is managed, indicating a user-friendly approach to data processing.

Data Security: Evaluate the use of encryption during data transmission and for storage. It falls under the category of data security, indicating measures taken to protect data during specific stages of processing.

Data retention and Deletion: Evaluate the document to find out limitations in the options for deleting user data which is essential for managing user data according to privacy principles. This is categorised into limited or no data deletion options and options to delete the user data.

Data sharing practices: Evaluate the document to analyse the category of data sharing practices. This is find out whether there is data sharing with third parties aligns with privacy-friendly practices.

GDPR Complaint: Evaluate the document to find out whether the dependency is GDPR complaint or not. Being GDPR compliant is related to legal compliance. It indicates that the privacy practices align with the requirements set forth by the General Data Protection Regulation (GDPR).

3.2 Data Collection and Selection

Training a model is achieved with the help of diverse set of privacy policy documents available from the web² by utilizing a publicly available dataset of privacy policies and contents. The following data was be collected as part of the research:

Privacy Policy Dataset : Utilized prebuilt collection of privacy policy documents available from the web to identify the structure and select the relevant strings based on the insights. The python scripting and libraries were used to extract the specific strings for labelling, tokenization and training of the model.

Identify Third-Party Dependencies: Created a list of all third-party dependencies for analysis and validation. This includes services, libraries, plugins, and any other external components that handle or process data on behalf of an application and made sure that the selected documents and different from each other in terms of privacy impact in order to test the efficiency of the framework.

3.3 NLP Analysis

Natural Language Processing (NLP) was employed in the analysis of privacy policies by utilizing tokenization and in the annotation of privacy policies for training, NLP techniques, specifically tokenization and Named Entity Recognition (NER). These methods help break down the document, identify important units, and recognize entities like data encryption, data processing activities, and security measures for text based classification. BERT BASE model is chose for training and developing the analysis model after reviewing relevant studies in this area and considering recent progress in Natural Language Processing and large language models.

² Dataset: <https://github.com/citp/privacy-policy-historical>

The Bidirectional Encoder Representations from Transformers (BERT)³ model emerged as a suitable tool for this research due to its state-of-the-art performance in natural language understanding tasks. BERT's deep learning algorithm, pre-trained on a vast corpus of text, captures complex language nuances. Its bidirectional training, which considers each word in the context of all the words in a sentence, rather than in isolation. This helps it understand the language better and allows for a more advanced analysis. This is especially useful when studying privacy policies, which usually have complex and conditional language. BERT is considered one of the top models for tasks like sorting and organizing different types of text.

Different categories including aspects like data retention, data sharing, and consent management was used to label the data. Finally the classification task was conducted by fine-tuning a pre-trained BERT model with a labelled dataset of privacy policy excerpts that had been annotated based on these categories.

Furthermore, BERT's ability in grasping the context compared to other NLP libraries like CNN or XLNet allowed it to excel at recognizing and extracting particular data practices and conditions outlined in the policies. By fine-tuning BERT on a set of annotated privacy policies, the model learned to identify patterns and legal jargon indicative of compliance or non-compliance with data protection standards.

3.4 Annotation and Tokenization

The privacy policy dataset has been annotated and tokenized to identify specific information related to privacy, including information that may raise concerns about user privacy has completed by following the specified steps. The annotation of the data was done manually. After finding out the relevant text sentences in the privacy policy document it is extracted and cleaned. Extraction of the relevant data from a long privacy policy document was achieved with the help of beautiful soup⁴ library available in python.

Table 1: Data annotation breakdown

Categories	Number of annotations
Data Security	212
Consent Management	188
Data Encryption	201
Data retention and Deletion	202
Data sharing practices	213
GDPR compliance	280
Total annotated data	1296

Data Pre-processing

Data pre-processing is a crucial step in preparing your privacy policy documents for training with the BERT model. Initially, raw text data were extracted from the privacy policies, which were then subjected to a series of pre-processing tasks to ensure uniformity and readability for the BERT model. This process included tokenization, where text was split into meaningful tokens, and normalization, which involved converting text to a standard format by lowercasing

³ BERT: <https://blog.research.google/2018/11/open-sourcing-bert-state-of-art-pre.html>

⁴ BeautifulSoup: <https://pypi.org/project/beautifulsoup4/>

and removing special characters and white spaces. Additionally, sentence segmentation was performed to divide the text into individual sentences.

Manual Annotation

The privacy policies and the respective texts are manually annotated into dataset. Identified and tagged phrases and sentences relevant to the six target areas. A set of experts reviewed the annotations for consistency and accuracy. Each sentence or relevant text snippet in the dataset from the privacy policies is annotated and labelled according to the six focus areas:

- Data Minimization: Sentences relating to how much data is collected.
- Consent Management: Information about how user consent is obtained and managed.
- Data Encryption: Details on how data is encrypted during transmission and/or storage.
- Data Retention and Deletion: Policies regarding how long data is kept and how it can be deleted.
- Data Sharing Practices: Information on if and how data is shared with third parties.
- GDPR Compliance: Statements regarding adherence to GDPR regulations.

D3		
	A	B
1	text	category
2	We secure your personal data by implementing SSL encryption technology during data transmission.	Data Security
3	Your data is stored on our secure servers, but we do not explicitly mention the use of encryption technologies.	Data Security
4	You can customize your privacy settings to control the different types of data you share with us, including location, browsing history, and contact information	Consent Options
5	By using our services, you agree to our collection and use of your personal information as outlined in this policy without options for specific consent preferences.	Consent Options
6	We use advanced AES-256 encryption to protect your data both at rest and in transit.	Data Encryption
7	Data encryption is applied during transmission over the internet, but not for data stored on our servers.	Data Encryption
8	You can easily request the deletion of your account and associated data at any time through your account settings.	Data Retention
9	Data deletion requests must be submitted in writing and sent to our legal department for processing, which may take up to 30 days.	Data Retention
10	We do not share your personal data with any third parties for their marketing purposes.	Data Sharing

Figure 2: Annotated dataset for training.

Model Architecture

The core of the model architecture is the BERT pre-trained model, which is further fine-tuned to classify specific aspects of privacy policy texts. To adapt BERT for the specific task of privacy policy analysis, A classification layer on top of the pre-trained model was added. This layer is a fully connected neural network that outputs a probability distribution over the predefined labels in our dataset, such as "Encryption Used", "Encryption Not Used", "Granular", and others. Input texts are pre-processed and tokenized using the BERT tokenizer. This tokenizer converts text into tokens that the BERT model can understand, adds special tokens required for classification tasks and pads or truncates sentences to a uniform length.

Fine Tuning

The model is fine-tuned on our annotated dataset of privacy policy texts. During this process, the weights of the pre-trained BERT layers and the added classification layer are adjusted to minimize the classification loss on our specific dataset. This step tailors the BERT model to effectively understand and categorize the of privacy policy language.

3.5 Scoring Algorithm

A scoring framework for the PIA framework was systematically developed, including the quantification and assessment of the privacy impact of third-party dependencies based developed model. Certain set of criteria's such as using specific terms, following rules, and other important factors were considered before applying the score metrics. Assigned different scores to different parts of the metrics, making sure to assess it thoroughly. The framework was designed to generate an overall privacy impact score, providing a clear and interpretable representation of the assessed privacy risks.

The following score metrics is mapped in the framework:

```
score_metrics = {  
    "Clear and detailed privacy policy": 2,  
    "Adequate privacy policy but lacks details": 1,  
    "Granular and explicit consent options": 2,  
    "Consent options available but not granular": 1,  
    "Limited or no consent options": 0,  
    "Strong encryption for data transmission and storage": 1.5,  
    "Encryption used for data transmission but not storage": 0.5,  
    "Limited or no encryption": 0,  
    "Easy and accessible options for data deletion": 1.5,  
    "Options available but not user-friendly": 0.5,  
    "Limited or no data deletion options": 0,  
    "No data sharing with third parties": 2,  
    "Limited and transparent data sharing": 1,  
    "Extensive data sharing without clear transparency": 0,  
    "GDPR complaint": 1,  
    "Not mentioned regarding GDPR compliance": 0,  
}
```

Figure 3: Score metrics used in the framework.

3.6 Privacy Risk Assessment

The selected third party dependencies has undergone a thorough privacy analysis using the developed framework and the collected data. This analysis involved identifying potential privacy concerns, and assessing the compliance with relevant regulations and privacy policies. The final score is generated using the metrics and weights.

The framework takes the user inputs as a third party dependency name. Subsequently, it conducts a search for the official privacy policy of the specified dependency utilizing the Google Search API. Upon locating the document, the framework extracts its contents and transforms them into a text format. It then isolates and extracts sections relevant to the predetermined areas of interest within the privacy policy. These extracted segments are subsequently input into the trained model for thorough analysis. Based on the insights derived from the model, the framework calculates a privacy impact assessment score for the given third-party dependency. This score is then presented in the output results.

4 Design Specification

The Privacy Policy Analysis Tool is a python-based project that leverages Natural Language Processing techniques to analyse privacy policies fetched from online sources. The tool will use BeautifulSoup for web scraping, BERT for NLP analysis finally based on the results from these collected and analysed data a privacy impact scoring will be developed for each dependency. The following are the technical methodologies, and the associated requirements that enable the functionality of the developed solution.

4.1 Policy Scrapping with BeautifulSoup

Automated scripts to fetch privacy policy documents via web APIs is the initial point of this framework. Utilized BeautifulSoup to scrape privacy policy documents. BeautifulSoup is a Python library that is used commonly for parsing HTML, making it an ideal choice for extracting information from web pages, including privacy policy documents. By leveraging BeautifulSoup's powerful features, navigated through the document's structure, locate specific elements, and extract relevant data pertaining to privacy policies. Moreover, the use of third-party dependency names as input needs a well-defined scrapping process because each of the privacy policy will have a different structure and format. This approach allowed for dynamic and adaptable scraping. Additionally, proper error handling mechanisms has been implemented within the code ensuring a robust and reliable scraping process. This includes addressing potential issues such as network errors, malformed HTML structures which can impact the scraping workflow.

4.2 Model development with BERT

The core of the PIA program is built upon a NLP model powered by BERT (Bidirectional Encoder Representations from Transformers), This framework is enhanced to carefully examine privacy policies by taking into account their dependencies. BERT BASE model is chosen for the specified task and since it is a pre trained model a classification layer with specific feature has been applied to the output layer of the model with the help of annotated

dataset. PyTorch⁵ was used for the deep learning functionality, the transformers⁶ library for BERT, and essential methods from the scikit-learn library were employed to handle data.

4.3 Storage

The PIA framework makes use of SQL database for the storage of generated data and results. The database is crucial as the storage layer for the program, enabling structured storage and efficient retrieval of data. The schema is designed to store dependency name, policy link, insights results, and PIA scores. Tables are normalized to reduce redundancy and improve data integrity. An ORM (Object-Relational Mapping) layer is used to facilitate the interaction between the Python codebase and the SQL database, making data manipulation more flexible.

4.4 System Design

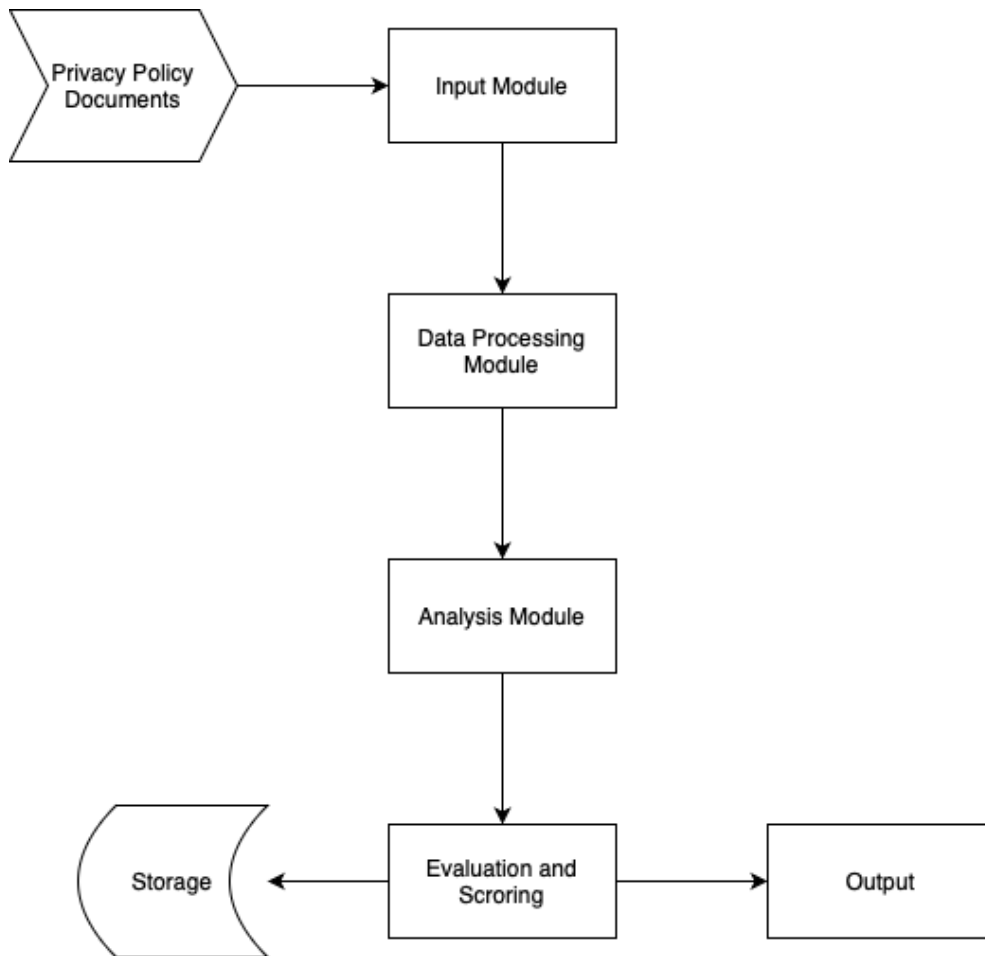


Figure 4: System design workflow of framework.

⁵ Pytorch: <https://pytorch.org>

⁶ Transformers: <https://huggingface.co/docs/transformers/index>

Input Module: Captures user input, which consists of third-party dependency names. It makes use of google search api to find out the privacy policy document from the web and then it is passed as an argument for further processing the data. Other user inputs present in the framework is a function to display the results of analysed dependencies in a table.

Data Processing Module: Extracts and pre-processes the privacy policy documents to convert them into a machine-readable format. This module is responsible for retrieving the privacy policy documents from the web. It parses HTML content to extract the key textual information based on keywords, ensuring that various formats and layouts are accommodated. Implements text cleaning, sentence segmentation, and tokenization.

Analysis and Training Module: A pre-trained BERT model is selected as the foundation for analysing and classifying the inputs. The filtered privacy policy texts are passed one by one to the model. The BERT model is fine-tuned with a dataset comprising annotated privacy policies, which allows the model to adapt to the specific language and terms used in privacy policy documents.

Evaluation and Scoring Module: Analyses the processed text to assign PIA scores based on predefined criteria and scores. It uses an algorithm that maps model outputs to a quantifiable score reflecting the overall value of the privacy of the third party dependency.

Output Module: Presents the PIA score and insights through a command line user interface or as an API service. It allows for result analysis, including aspects such as data security, consent management, and GDPR compliance.

5 Implementation

The implementation section describes the last phase of developing the Privacy Impact Assessment (PIA) framework. It provides information about the results achieved and the tools and programming languages used to create the solution. The final stage of the implementation focused on integrating the PIA Framework into the organization's Vulnerability management dashboards. The primary outputs produced during this stage include transformed data, code for the integration, and the deployment of the scoring framework.

5.1 Transformed Data

The structured data was subjected to a comprehensive analysis, utilizing natural language processing techniques to extract key insights regarding privacy practices associated with the third-party dependencies. This in-depth examination provided valuable information on data handling practices, user consent mechanisms, and security measures implemented by the third-party services. The raw text from privacy policies was normalized to maintain consistency. This involved converting all text to a uniform case, removing special characters, and standardizing terminologies. Post-transformation, the data was inputted into the BERT NLP model. The model, pre-trained on a language model, was fine-tuned with this annotated, transformed data. This fine-tuning enabled the model to better understand the context and nuances specific to privacy policies.

5.2 Codebase Development

In the development Privacy Impact Assessment (PIA) framework, it is carefully built a thorough and flexible codebase. This acts as the main working part of the framework, containing all the necessary functions such as collecting and preparing data, training models, and displaying results. The Python programming language was chosen for its extensive ecosystem of libraries and tools, which are especially conducive to machine learning and data processing tasks.

SQL was used for database management to store and retrieve transformed data and results using the model. In case of Libraries and Frameworks TensorFlow and PyTorch were Utilized for machine learning model development, training, and evaluation. Flask was used to create the API service. Postman was used for assistance in API development by testing API endpoints.

The codebase is structured to support scalability and maintainability, the codebase is organized into distinct modules, each responsible for a specific aspect of the PIA process. These components work smoothly together, making sure that information moves seamlessly from one step to another. This ultimately results in the creation of PIA scores and detailed analytical reports. Throughout the development process, best practices in software engineering were adhered to, including the use of version control with Git to manage the codebase evolution. The final code reflects the use of advanced NLP methods in analysing privacy policies. It demonstrates the application of modern and efficient software principles.

5.3 Models Developed

The model was iteratively trained and validated on the annotated dataset to ensure high accuracy and reliability in privacy policy assessment. The final output was a text classification model using BERT for automated privacy policy analysis. Testing and validation procedures were conducted with various documents to verify the algorithm's effectiveness in identifying potential privacy violations.



Figure 5: BERT Model workflow

5.4 CLI Interface

The Command-Line Interface (CLI) tool developed as part of the PIA framework is a Python-based application that integrates the BERT model for analysing and giving insights for third-party dependencies. When executed, the CLI tool presents users with a menu offering two primary options: firstly, the ability to analyse a new dependency, which initiates the fetching and assessment of a privacy policy document for a specified third-party service; secondly, the option to view a dashboard of previously analysed dependencies, including details such as the PIA score for each. To include scenarios where the privacy policy is not readily retrievable via the Google Search API, the tool also provides the functionality to manually input a specific policy URL. This feature ensures that users have the control over the source of the privacy policy being analysed, thereby enhancing the tool's flexibility.

5.5 Outputs in Vulnerability Management Dashboard

The Privacy Policy Analyzer results, including NLP insights and identified vulnerabilities was converted into a privacy impact assessment score and successfully displayed in the Dependency Management System within the Vulnerability Management Dashboards.

The scoring framework outputs, based on predefined criteria, provided a clear visualization of the privacy impact assessment for each third-party dependency.

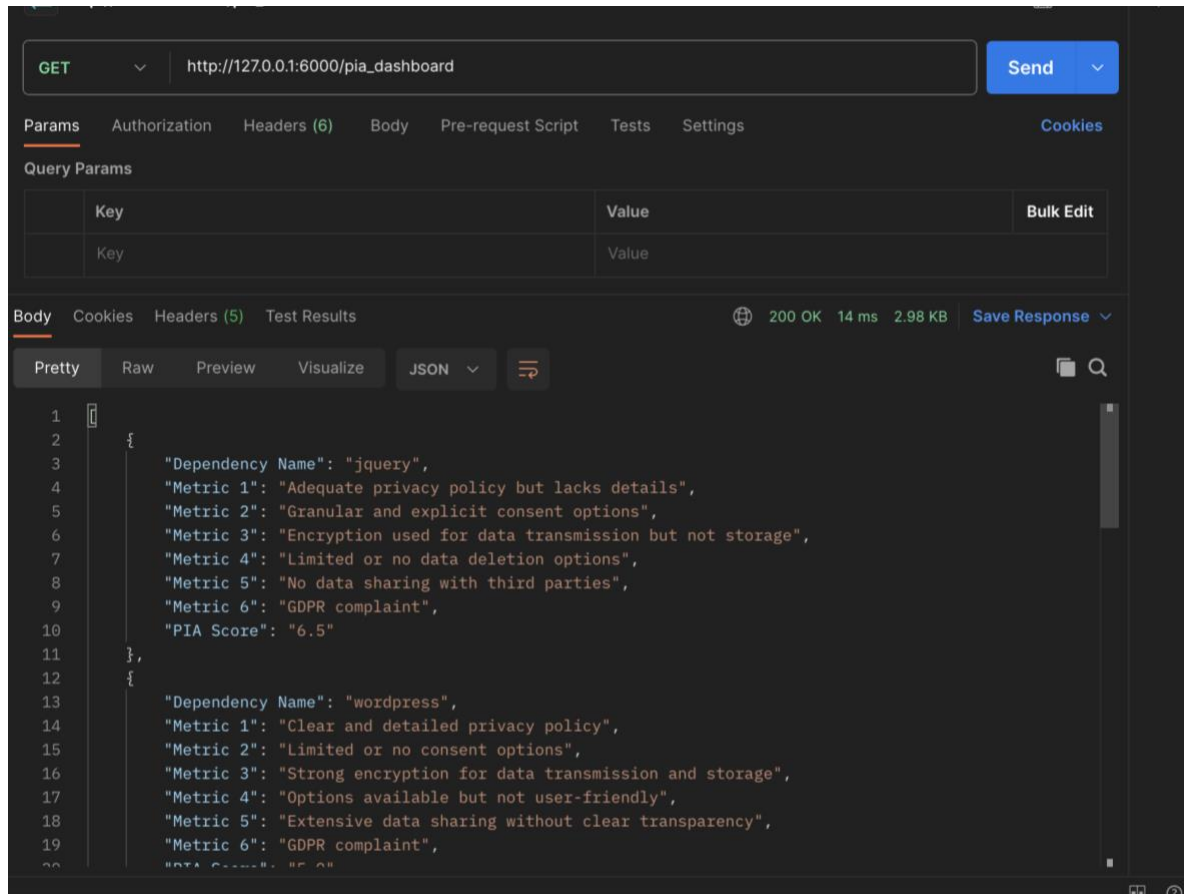


Figure 6: PIA framework hosted as an API service.

6 Evaluation

The evaluation section rigorously examines the results and main findings of the study, assessing the efficacy and implications of the developed Privacy Impact Assessment (PIA) program. The investigation focuses on the most pertinent outcomes that reinforce the research questions and objectives. This analytical discourse offers a synthesis from both academic and practical standpoints. Applying it to three real-world third-party dependency privacy policies: WordPress, Woebot, and LoadLash. These dependencies were selected based on their varied privacy impact quality good, vague, and mediocre, respectively offering a broad spectrum for analysis.

The PIA framework was systematically applied to the privacy policies of WordPress, Woebot, and LoadLash. Each policy was subjected to the same analysis process, evaluating key areas such as Data Security, Consent Management, Data Encryption, Data Retention and Deletion, Data Sharing Practices, and GDPR Compliance.


```

1. Display PIA Dashboard
2. Analyze a New Dependency
3. Exit

Select an option (1, 2, or 3): 2
Enter the Dependency Name:: Woebot
Searching for Privacy policy Document....
Privacy Policy for Woebot found at: https://woebothealth.com/privacy-webview/
Analyzing the Privacy Policy Document....
1. Adequate privacy policy but lacks details
2. Limited or no consent options
3. Limited or no encryption
4. Limited or no data deletion options
5. Extensive data sharing without clear transparency
6. GDPR complaint

PIA Score for Woebot: 2
Dependency analyzed and saved successfully.

1. Display PIA Dashboard
2. Analyze a New Dependency
3. Exit

Select an option (1, 2, or 3): █

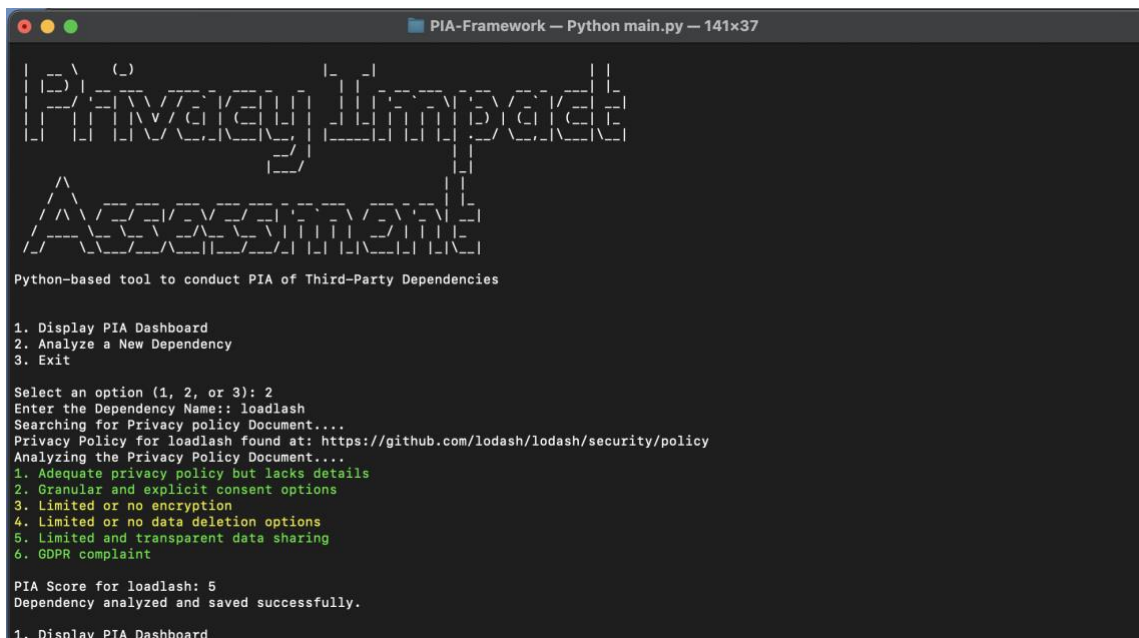
```

Figure 8: Analysing Woebot privacy document and providing Insights and PIA score

6.3 Case Study 3

LoadLash Privacy Policy Evaluation⁹: As a third-party dependency software, Lodlash is a JavaScript utility library that delivers consistency, customization, and performance in coding.

- Loadlash’s policy yielded a mediocre PIA score, indicating ambiguity in certain sections.
- Framework highlighted areas requiring clarity, like Data Retention and Consent Management.



```

PIA-Framework — Python main.py — 141x37

FIVEGIGS
ANALYZE

Python-based tool to conduct PIA of Third-Party Dependencies

1. Display PIA Dashboard
2. Analyze a New Dependency
3. Exit

Select an option (1, 2, or 3): 2
Enter the Dependency Name:: loadlash
Searching for Privacy policy Document....
Privacy Policy for loadlash found at: https://github.com/loadash/loadash/security/policy
Analyzing the Privacy Policy Document....
1. Adequate privacy policy but lacks details
2. Granular and explicit consent options
3. Limited or no encryption
4. Limited or no data deletion options
5. Limited and transparent data sharing
6. GDPR complaint

PIA Score for loadlash: 5
Dependency analyzed and saved successfully.

1. Display PIA Dashboard

```

Figure 9: Analysing loadlash privacy document and providing Insights and PIA score.

⁹ Loadlash Privacy Policy: <https://github.com/loadash/loadash/security/policy>

6.4 Discussion

The evaluation of the Privacy Impact Assessment framework through the analysis of three distinct privacy policies from WordPress, Woebot, and LoadLash which has mixed results in terms of accuracy and reliability. While the framework demonstrated proficiency in categorizing and assessing privacy policies with clear and structured language, it occasionally produced false insights when analysing vague policy statements. This variability in performance highlights the challenges in automating the task of privacy document analysis.

The experiments were conducted with the intent of providing a rigorous test of the PIA framework's capabilities. In instances of clear and well-defined privacy policies with less privacy impact, such as that of WordPress, the framework's accuracy was good enough. However, the some challenges were introduced when the framework encountered policies with not a well-defined language and less detailing, as was the case with Woebot and LoadLash. This underscores a limitation in the model's ability to deal with the complexity and variability of natural language used in privacy policies.

One of the other challenges faced while the evaluation was during the parsing of privacy policy documents from the web using python libraries the segmentation of sentences was based on the keywords defined in the policy extraction scripts. Sometimes it resulted in extraction of wrong data since the same keywords are found in many other parts of a privacy policy document of a third party dependency. It also highlights the need for more accurate text extraction before passing it to the model for analysis.

To enhance the PIA framework's accuracy, several modifications are proposed:

- **Multi-Model Integration:** Incorporating a blend of NLP models and techniques could provide a more robust analysis, capturing nuances that a single model might miss.
- **Expand the Dataset:** Enriching the dataset with a wider array of privacy policies, especially those with ambiguous phrasing, could improve the model's learning and adaptability.
- **Iterative Training:** Implementing a continuous learning loop where the model is regularly updated with new data and expert feedback can help in refining its predictive capabilities.

The results of the assessment demonstrate that the analysis of legal documents by automated systems is a complex task, as supported by existing literature. Previous studies have frequently emphasized the challenges machines face in fully understanding legal terminology and the inherent ambiguity found in policy documents. The performance of the PIA framework reflects these difficulties, indicating that although progress has been made, there is an ongoing requirement for enhancements in this field.

Compared to the tools and frameworks mentioned in the literature review, the PIA framework is notable because it employs an advanced and latest BERT model. However, it encounters similar challenges, especially when handling documents that are unclear or have varying language. This finding indicates a shared obstacle in the field of automated privacy policy assessment and points towards a definite path for future research and development.

7 Conclusion and Future Work

This research paper began with the question: How can a Privacy Impact Assessment (PIA) framework be developed to evaluate third-party dependencies in software development? The objectives were to create a model capable of analysing privacy policies and outputting a clear PIA score, thereby aiding developers, compliance, and legal teams in assessing third-party services.

The PIA framework developed through this study successfully addresses the research question. It utilizes a BERT-based model to evaluate the privacy policies of third-party dependencies, offering a quantifiable PIA score. The framework demonstrates a high level of accuracy in identifying key privacy policy attributes such as data security, consent management, and GDPR compliance. The research yielded several significant findings:

- The PIA framework can recognise a broad spectrum of privacy policy quality, evidenced by testing with different level grade of policies.
- It provides a scalable solution for assessing complex legal documents like privacy policies.
- The PIA scores are indicative of the clarity and robustness of privacy policies.

The implications of this research are substantial for both academic circles and industry practices academically, it contributes to the field of automated policy analysis using advanced NLP techniques. Practically, it offers a tool that can significantly streamline the privacy policy assessment process for various project teams and legal teams in Cubic Telecom.

The efficacy of the research is clear from the accurate assessment of privacy policies, although it acknowledges the limitations inherent in the current scope of data and potential biases in the model.

The study's limitations include:

- The need for a larger dataset to improve the model's robustness.
- A reliance on manual annotation, which requires considerable human effort.

Future research could explore several avenues including integrating alternative NLP methods or emerging AI techniques to complement BERT's capabilities, applying the model to other domains of legal compliance beyond privacy policies and developing a more interactive tool that allows for real-time feedback and learning from end-users. There is significant commercial potential for this PIA framework, especially as companies seek to fortify their data privacy practices considering increasing regulatory pressures.

The intention to enhance the model with a more extensive dataset is clear, as is the commitment to human-powered annotation to ensure the quality of the data. The future publication of the tool on GitHub as an open-source resource underscores the project's dedication to community engagement and transparency.

In conclusion, the research presents a valuable contribution to the field of privacy impact assessment and opens the door to numerous possibilities for further exploration and enhancement. The planned release of the model as an open-source tool not only promises continued improvement through community involvement but also stands as a testament to the collaborative spirit of innovation in the digital age.

References

- Del Alamo, J.M. *et al.* (2022) ‘A systematic mapping study on automated analysis of privacy policies’, *Computing*, 104(9), pp. 2053–2076. Available at: <https://doi.org/10.1007/S00607-022-01076-3/FIGURES/5>.
- Amos, R. *et al.* (no date) ‘Privacy Policies over Time: Curation and Analysis of a Million-Dataset; Privacy Policies over Time: Curation and Analysis of a Million-Dataset’, 12(21). Available at: <https://doi.org/10.1145/3442381.3450048>.
- Andrade, V.C. *et al.* (2022) ‘Privacy by design and software engineering a systematic literature review’, *ACM International Conference Proceeding Series* [Preprint]. Available at: <https://doi.org/10.1145/3571473.3571480>.
- Castellanos-Ardila, J.P., Gallina, B. and Governatori, G. (2021) ‘Compliance-aware engineering process plans: the case of space software engineering processes’, *Artificial Intelligence and Law*, 29(4), pp. 587–627. Available at: <https://doi.org/10.1007/S10506-021-09285-5>.
- Chaw, C.Y. and Chua, H.N. (2021) ‘A Framework System Using Word Mover’s Distance Text Similarity Algorithm for Assessing Privacy Policy Compliance’, *Lecture Notes in Electrical Engineering*, 782, pp. 79–89. Available at: https://doi.org/10.1007/978-981-16-4118-3_8.
- Cheng, H. *et al.* (2023) ‘Detecting third-party libraries for privacy leakage in packed android applications’, pp. 5053–5058. Available at: <https://doi.org/10.1109/CAC57257.2022.10054907>.
- Chhetri, T.R. *et al.* (2022) ‘Data Protection by Design Tool for Automated GDPR Compliance Verification Based on Semantically Modeled Informed Consent’, *Sensors 2022, Vol. 22, Page 2763*, 22(7), p. 2763. Available at: <https://doi.org/10.3390/S22072763>.
- Clarke, R. (2009) ‘Privacy impact assessment: Its origins and development’, *Computer Law & Security Review*, 25(2), pp. 123–135. Available at: <https://doi.org/10.1016/J.CLSR.2009.02.002>.
- van Daalen, O.L. (2023) ‘The right to encryption: Privacy as preventing unlawful access’, *Computer Law & Security Review*, 49, p. 105804. Available at: <https://doi.org/10.1016/J.CLSR.2023.105804>.
- Koroteev, M. V. (2021) ‘BERT: A Review of Applications in Natural Language Processing and Understanding’. Available at: <https://arxiv.org/abs/2103.11943v1> (Accessed: 26 December 2023).
- Mannan, M. *et al.* (2019) ‘Final Report for OPC Contributions “Privacy Report Card for Parental Control Solutions”’.
- Nazir, F. *et al.* (2017) ‘The applications of natural language processing (NLP) for software requirement engineering - A systematic literature review’, *Lecture Notes in Electrical Engineering*, 424, pp. 485–493. Available at: https://doi.org/10.1007/978-981-10-4154-9_56.

Qi, P.G. *et al.* (2023) *{PoliGraph}: Automated Privacy Policy Analysis using Knowledge Graphs*. Available at: <https://sites.google.com/view/> (Accessed: 26 December 2023).

Ravichander, A. *et al.* (2021) ‘Breaking Down Walls of Text: How Can NLP Benefit Consumer Privacy?’, *ACL*, pp. 4125–4140. Available at: <https://doi.org/10.18653/V1/2021.ACL-LONG.319>.

Tesfay, W.B. *et al.* (2018) ‘I Read but Don’t Agree: Privacy Policy Benchmarking using Machine Learning and the EU GDPR’, *The Web Conference 2018 - Companion of the World Wide Web Conference, WWW 2018*, pp. 163–166. Available at: <https://doi.org/10.1145/3184558.3186969>.

Tulili, T.R., Capiluppi, A. and Rastogi, A. (2023) ‘Burnout in software engineering: A systematic mapping study’, *Information and Software Technology*, 155, p. 107116. Available at: <https://doi.org/10.1016/J.INFSOF.2022.107116>.

Wagner, T.J. and Ford, T.C. (2020) ‘Metrics to Meet Security Privacy Requirements with Agile Software Development Methods in a Regulated Environment’, *2020 International Conference on Computing, Networking and Communications, ICNC 2020*, pp. 17–23. Available at: <https://doi.org/10.1109/ICNC47757.2020.9049681>.

Wilson, S. *et al.* (2016) ‘The Creation and Analysis of a Website Privacy Policy Corpus’, *54th Annual Meeting of the Association for Computational Linguistics, ACL 2016 - Long Papers*, 3, pp. 1330–1340. Available at: <https://doi.org/10.18653/V1/P16-1126>.

Wright, D. and De Hert, P. (2012) ‘Introduction to Privacy Impact Assessment’, *Law, Governance and Technology Series*, 6, pp. 3–32. Available at: https://doi.org/10.1007/978-94-007-2543-0_1.