A New Hope: Contextual Privacy Policies for Mobile Applications and An Approach Toward Automated Generation

Shidong Pan and Zhen Tao, *CSIRO's Data61 and Australian National University;*
Thong Hoang, *CSIRO's Data61;* Dawen Zhang, *CSIRO's Data61 and Australian National University;* Tianshi Li, *Northeastern University;* Zhenchang Xing, *CSIRO's Data61 and Australian National University;* Xiwei Xu, Mark Staples, and Thierry Rakotoarivelo, *CSIRO's Data61;* David Lo, *Singapore Management University*

This paper is included in the Proceedings of the 33rd USENIX Security Symposium.

August 14–16, 2024 • Philadelphia, PA, USA

Open access to the Proceedings of the
33rd USENIX Security Symposium
is sponsored by USENIX.

A NEW HOPE: Contextual Privacy Policies for Mobile Applications and An Approach Toward Automated Generation

Shidong Pan[1,2,*], Zhen Tao[1,2], Thong Hoang[2,†], Dawen Zhang[1,2], Tianshi Li[3], Zhenchang Xing[1,2], Xiwei Xu[2], Mark Staples[2], Thierry Rakotoarivelo[2], and David Lo[4]

[1]*School of Computing, Australian National University*
[2]*Software and Computational Systems Research Program, CSIRO's Data61*
[3]*Khoury College of Computer Sciences, Northeastern University*
[4]*School of Computing and Information Systems, Singapore Management University*

Abstract

Privacy policies have emerged as the predominant approach to conveying privacy notices to mobile application users. In an effort to enhance both readability and user engagement, the concept of contextual privacy policies (CPPs) has been proposed by researchers. The aim of CPPs is to fragment privacy policies into concise snippets, displaying them only within the corresponding contexts within the application's graphical user interfaces (GUIs). In this paper, we first formulate CPP in mobile application scenario, and then present a novel multimodal framework, named **SEEPRIVACY**, specifically designed to automatically generate CPPs for mobile applications. This method uniquely integrates vision-based GUI understanding with privacy policy analysis, achieving 0.88 precision and 0.90 recall to detect contexts, as well as 0.98 precision and 0.96 recall in extracting corresponding policy segments. A human evaluation shows that 77% of the extracted privacy policy segments were perceived as well-aligned with the detected contexts. These findings suggest that SEEPRIVACY could serve as a significant tool for bolstering user interaction with, and understanding of, privacy policies. Furthermore, our solution has the potential to make privacy notices more accessible and inclusive, thus appealing to a broader demographic. A demonstration of our work can be accessed at https://cpp4app.github.io/SeePrivacy/

1 Introduction

Privacy policies have become the most prevalent privacy notice approach for mobile applications, with the aim of protecting users' privacy and digital security [26, 33, 49, 51, 52, 60, 66, 83]. On the contrary, application users often find themselves in a privacy paradox, specifically, *Digital Resignation*. This term refers to a state where users desire to understand and control the information that digital entities own about them, but feel powerless to do so [45]. A study has found that the average length of privacy policies is around 4,000 words for 75 prominent mobile applications and websites, requiring an estimated 16 minutes to read for an average adult [30]. Consequently, according to a survey conducted by Obar and Oeldorf-Hirsch [78], a significant portion of users, approximately 74%, opt for the "quick join" clickwrap option without engaging with privacy policies, primarily due to the lengthy and detailed nature of these documents. These numbers indicate that privacy policies have become increasingly difficult to read and understand in a timely manner, exacerbating the prevalence of digital resignation among users.

In addressing the pressing issue of digital resignation, the contemporary evolution of privacy policies has given rise to the concept of "just-in-time" privacy notices [12]. These notices furnish users with detailed information about data collection at the point of interaction. This practice has been recognized and included in the U.K. General Data Protection Regulation as a recommended approach for crafting privacy policies [12]. However, the dominant application of "just-in-time" notices is primarily limited to the installation phase of applications and first invocation of certain permissions. Figure 1a shows the "install-time" permission reminders that appear when a user installs a new Android app[1], delineating the permissions to be granted. Figure 1b presents another example, which illustrates a camera access reminder triggered at the initial "invoke-time". However, there are inherent limitations to current "just-in-time" notices. For one, while they provide a snapshot of when an app accesses specific data, it doesn't necessarily offer insights into the broader context – how this data might subsequently be transmitted off the device, shared with third parties, stored, or used for other purposes. Furthermore, current implementations of "just-in-time" reminders fall short in terms of efficacy, as security warnings regarding device permissions tend to be isolated from their immediate context, thus failing to adequately engage user awareness [47, 74].

---

*The author completed most of the research while being a visiting student at Singapore Management University
†Corresponding Author

[1]This feature was essentially deprecated after Android 6.0 (API level 23) since developers must request permissions at runtime.

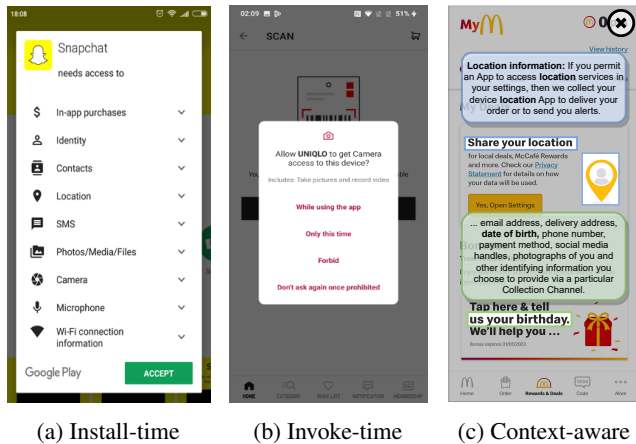(a) Install-time   (b) Invoke-time   (c) Context-aware

Figure 1: Figure 1a represents the install-time reminder of required permissions [20]; Figure 1b corresponds to the invoke-time reminder that appears when permission is first invoked; Figure 1c presents our proposed contextual privacy policy, specifically designed for mobile applications.

To counteract these deficiency, researchers proposed *Contextual Privacy Policies* (CPPs) [48, 94]. This innovative concept aims to deconstruct traditional privacy notices into shorter and digestible segments, displaying them only in relevant contexts in mobile applications. Privacy-related contexts refer to areas in which data practices are directly associated with graphical user interfaces (GUIs). These contexts on the screen are identified and the corresponding privacy policy segments are retrieved and displayed, with key information highlighted. This integration of privacy notices within the GUIs ensures a close and timely alignment with the context, thus conveying precise and succinct privacy information. Windl et al. [94] demonstrated that CPPs have the potential to address the usability issues in privacy policies, thereby enhancing user engagement and awareness of data practices. They also presented a framework for automatically generating CPPs for websites. Studies have shown that users spend 90% of their time using mobile apps rather than websites [5, 32, 62]. However, the prior [94] framework has the following limitations when applied to the mobile app context, all of which are successfully addressed in this work: 1) Their context selection is based on empirical observation of users' behaviour in website browsing scenarios, but lacks connection to privacy regulations and existing standards in practice. 2) The prior context detection method is based on ad-hoc HTML analysis, which has limited internal generalizability, especially for more compact and complicated user interfaces in mobile app platforms. Overall, while the concept of CPP has a great potential, generating accurate CPPs for mobile apps proposes unique challenges that cannot be addressed by the previous automated generation framework.

In this paper, we first formulate the CPPs in the mobile scenario, and then propose a novel multimodal framework, dubbed SeePrivacy, designed to automatically generate CPPs for mobile applications. Figure 1c presents an example of our CPP in mobile applications. Within this framework, computer vision techniques are incorporated with pretrained large language models (LLMs) to analyze mobile GUI screenshots, thus pinpointing privacy-related contexts. In addition, natural language processing techniques are used to dissect privacy policy texts, extracting the corresponding policy sections for identified contexts. To quantitatively evaluate performance, we establish our benchmark dataset, named Cpp4App, which covers more than 1,200 privacy-related contexts along with their matching privacy policy segments in mobile settings. The quantitative evaluation demonstrates the robust performance of our framework in multiple evaluation metrics. Regarding the identification of privacy-related contexts, SeePrivacy achieves accuracy, precision, and recall values of 0.81, 0.88, and 0.90, respectively. Regarding the extraction of the corresponding segments, the framework attains accuracy, precision, and recall rates of 0.94, 0.98, and 0.96, respectively. To further assess the functionality of SeePrivacy, we performed a human evaluation with 15 examiners on about 120 CPPs. The results are encouraging: 77% of the extracted privacy policy segments were perceived as elucidating the detected context.

We also discussed the applications, implications, and contributions of CPPs and SeePrivacy to the broader community. In the survey, participants display a significantly increased willingness to engage with CPPs, achieving an average score of 4.1 out of 5. In contrast, traditional privacy policies received an average score of just 2 out of 5. This difference suggests SeePrivacy could contribute meaningfully to improve user engagement with privacy policies, potentially mitigating the widespread challenge of digital resignation. Additionally, the overall usefulness rating for SeePrivacy is 4.53 out of 5, indicating strong user endorsement. Furthermore, we elaborate on two potential application scenarios of SeePrivacy: displaying CPPs in app markets and detecting "lack of disclosure" with exact problematic context. Lastly, we discuss how CPPs potentially make the privacy policies more inclusive and allow it to reach a larger audience such as children.

In summary, our proposed framework extends the automated generation of CPPs to mobile applications, which are the digital platforms users spend the majority of time on. The key contributions are:

- To our knowledge, this is the first research to define the concept of contextual privacy policies (CPPs) within mobile application scenarios.
- We have introduced a novel framework, named SeePrivacy, designed to automatically generate these contextual privacy policies for mobile applications.
- We have constructed the benchmark dataset, named Cpp4App, specifically to evaluate contextual privacy policies on mobile platforms and to quantitatively assess the functionality of the SeePrivacy framework.

Roadmap. The remainder of this paper is structured as follows: Section 2 introduces the development of transparent privacy notices and privacy-related context identification; Section 3 introduces our benchmark dataset, CPP4APP; Section 4 details the design of our proposed framework, i.e., SEEPRIVACY; and Section 5 presents the results of evaluating SEEPRIVACY on the benchmark dataset; Section 6 discusses the implications and contributions of our work to the broader community; and Section 7 concludes the paper.

Ethical Considerations and Responsible Disclosure. Ethical approval for this research was secured from our institution's Institutional Review Board (IRB). In conducting our user study, we avoided collecting any personally identifiable information (PII). In line with our commitment to responsible research practices, any privacy discrepancies detected during the course of the study were promptly and transparently disclosed to both the study participants and the corresponding application developers.

2 Background

Mobile applications provide users with unparalleled convenience, multiple entertainment options, and various services. At the same time, they are responsible for the collection and exchange of substantial amounts of personal information, raising concerns surrounding data privacy. In response to the potential risks of privacy violations, the U.S. Department of Health, Education, and Welfare proposed a set of Fair Information Practice Principles (FIPP) in 1973 [9, 63]. Later, the Federal Trade Commission adopted these principles as "*the notice and choice privacy framework*" [78], which is commonly regarded as the foundation for modern privacy regulations such as the European General Data Protection Regulation (GDPR) [10], the California Consumer Privacy Act (CCPA) [6], and the Australian Privacy Principles (APP) [3]. In FIPP, the most fundamental principle is *"notice"* [23]. Consumers should be provided with a clear understanding of an entity's information practices before any personal data is collected from them (e.g., [GDPR Art. 12, GDPR Rectical 58]). In the absence of transparent and understandable notice, it becomes impossible for a consumer to make an informed decision regarding the extent of personal information they choose to disclose.

2.1 Related Work

Research efforts to improve the presentation and readability of privacy policies have been manifold and encompass various approaches. Kelly et al. [57, 58, 59] introduced an innovative method called privacy nutrition labels (PNLs), designed to facilitate consumers' understanding of how their information is collected and utilized in a structured table-like way. By employing this concise and organized framework, PNLs effectively aid in understanding the privacy practices

of software applications [40, 68, 70, 81]. In a related vein, Cranor [39] proposed a protocol, i.e., Platform for Privacy Preferences, that allows website organizations to declare their intended use of information collected from website users. Alongside these approaches, there has been a growing focus on visual aids such as privacy icons [46, 53, 54]. These small graphical symbols, symbolizing privacy concepts or practices, serve as visual shorthand, simplifying complex privacy policies. By standardizing these icons across various services and platforms, researchers aim to reduce confusion, thereby making privacy policies more readily accessible to the general populace [54].

The conceptualization of *Contextual Privacy Policies* emerged in 2004, marking a significant change in the approach to the presentation of privacy notices. Bolchini et al. [31] discerned that the conventional reading of privacy policies represented a monolithic block that hindered users from retrieving essential privacy details. In response, they invented a systematic method aimed at reorganizing privacy policies around specific user interaction contexts. Subsequent to this development, Feth [48] extended the concept by introducing *contextual privacy statements*. Rather than using a "one-size-fits-all" privacy policy that must suit every usage scenario, these contextual privacy statements offer precise privacy and data protection information more closely with the user's immediate context. In a recent advancement, Windl et al. [94] proposed the design and architecture of PrivacyInjector, a production AI tool that can automatically generate contextual privacy policies for websites. However, their scope of data practice identification is limited to six website-specific types, such as "Advertising" and "Cookies and Tracking Elements". Their identification procedure primarily considers which date type users are likely to encounter when browsing websites. In contrast, our approach is regulation-driven, examining which data practices should be transparently disclosed in privacy notices. Moreover, their context detection method is intrusive and based on ad-hoc HTML analysis, which is consequently not applicable to mobile applications. In contrast, the source code of mobile applications is typically inaccessible to end-users. This limitation led us to develop a purely vision-based context detection method, tailored to the unique challenges presented by mobile applications. Nonetheless, CPPs have been empirically shown to increase privacy notice transparency, constituting a progressive step toward more user-centric privacy communication [76, 79].

Contrasting with websites, mobile applications represent a unique landscape for personal information collection, often involving multifaceted approaches that include device sensors, user input, and third-party integrations [82]. These complex processes necessitate a precise and transparent articulation of how user information is acquired, employed, and disseminated—an issue of critical concern. Given the expansive and continually growing scale of mobile applications, there exists an emergent need for an automated framework capable

Table 1: Tallies of data types' existence.

| | Basic Personal Identifiable Information | | | | | | Other Personal Information | | | | | |
	Name	Birthday	Address	Phone	Email	Profile	Contacts	Location	Photos	Voices	Financial Info	Social Media
Administration												
GDPR [10]	●	○	●	●	●	●	○	○	○	○	○	○
CCPA [6]	●	○	○	○	●	●	○	●	○	○	●	○
CalOPPA [8]	●	○	●	●	●	○	○	●	○	○	●	●
COPPA [7]	●	○	●	●	●	●	○	●	●	●	○	○
APP [3]	●	○	●	●	●	●	○	●	●	○	●	○
Industry												
Google Play [25]	●	●	●	●	●	●	●	●	●	●	●	●
Apple App [17]	●	○	●	●	●	●	●	●	●	●	●	●
Huawei AppGallery [11]	●	●	●	●	●	●	●	●	●	●	●	●
Amazon Appstore [4]	○	○	○	●	●	○	●	●	●	●	●	●
Samsung Galaxy Store [22]	○	○	○	●	○	○	●	●	●	●	○	○

of generating CPPs tailored specifically to this domain. In pursuit of this objective, the present work attempts to establish such a framework. As a preliminary step, we recognize the imperative to define clear and comprehensive definitions for privacy-related contexts within the mobile environment, forming the foundation for our proposed framework.

2.2 Empirical Data Practice Identification

According to Windl et al. [94], privacy-related contexts are considered areas on graphical user interfaces (GUIs) where data practices might be applied. To rigorously define these practices, we formulate two guiding questions, each tailored to address a distinct facets of the issue:

1. What data practices should be included in privacy notices by regulations?

2. What data practices are of concern and desire for users?

These questions collectively serve to demarcate the scope of our investigation, guiding our analysis in a manner that encompasses regulatory compliance and user-centric considerations. To our knowledge, there are no existing studies that specifically address the issue of contextual privacy policies within the domain of mobile applications. Specifically, current research appears to be primarily concerned with improving the presentation and readability of privacy policies [37, 39, 57, 58, 59]. Therefore, we perform a rigorous examination of data practices as they relate to privacy regulations and industry market standards (see Table 1). We present each group as follows:

Administration. The administration often functions as the key regulator within the broader ecosystem of data privacy. A report by the United Nations Conference on Trade and Development reveals that 137 of 194 countries and regions have instituted data protection and privacy legislation [16]. To gain insight into the administrative perspective on these matters, we examine five representative privacy regulations, including the GDPR [10], CCPA [6], the California Online Privacy Protection Act (CalOPPA) [8], the Children's Online Privacy Protection Act (COPPA) [7], and APP [3]. These regulations serve as critical touchpoints in our exploration of the administration's stance on data practices.

Industry. Mobile application platforms typically impose particular privacy notice requirements on curated applications within their marketplaces, with special attention to the categories of data involved. For instance, the Google Play app store has delineated 14 primary data categories and 39 subcategories in their data safety reports, which all applications must adhere to [25]. These standards have been formulated through rigorous user studies and market tests, reflecting a comprehensive understanding of user concerns and expectations. Such market standards provide valuable insight into the alignment of privacy requirements with actual user needs and industry practices, thereby serving as a meaningful reference for privacy policy development in the mobile application domain.

2.3 Multimodal Privacy-related Contexts

Privacy-related contexts refer to areas within GUIs where data practices might be applied. In this investigation, we categorize two primary types of GUI components that indicatively collect personal information via human-device interactions: literal texts and graphical icons. Literal texts, often regarded as the foundational information sources in mobile application GUIs, are juxtaposed with graphical icons, which are ingeniously designed to communicate vital information with maximum efficiency. The growing adoption of graphical icons in place of literal texts is due to their concise nature, which provides users with the requisite information instantaneously, thereby substantially minimizing the cognitive effort required to interpret the accompanying textual descriptions [96].

Keyword collection. In the process of analyzing literal texts that describe data types, we snowballed a keyword list tailored to CPPs. This list was compiled by extracting keywords from academic research literature about privacy policies, including content categorization [56, 69, 93, 105], automated generation [89, 100, 106], and compliance analysis [29, 41, 98, 101, 104, 105]. These studies often present unique challenges due to differences in privacy policies, corpus selections, and specific tasks among various topics. Ambiguous terms, such as "address," which could refer to a residential or an email address, were deliberately excluded to

Table 2: Privacy-related contexts.

Data types	Description	Privacy-related keywords lists for textual GUI elements	Related RICO-icon class	Icon examples
Name	How a user refers to themselves	name, first name, last name, full name, real name, surname, family name, given name	n.a.	n.a.
Birthday	A user's birthday	birthday, date of birth, birth date, DOB, birth year	n.a.	n.a.
Address	A user's address	mailing address, physical address, postal address, billing address, shipping address, residential address, residence, personal address	n.a	n.a
Phone	A user's phone number	phone, phone number, mobile phone, mobile number, telephone, call, telephone number	[43] Call	
Email	A user's email address	email, e-mail, email address, e-mail address	[6] Email	
Profile	A user's account information	profile, account	[49] Avatar	
Contacts	A user's contact information, or the access to the contact permission	contacts, phone-book, phone book, device's address book	[68] Group, [3] Follow	
Location	A user's location information, or the access to the location permission	location, locate, geography, geo, geo-location, precision location	[40] Location crosshair, [72] Location	
Photos	A user's photos, videos, or the access to the camera permission	camera, photo, scan, album, picture, gallery, photo library, storage, image, video, scanner, photograph	[42] Photo, [56] Videocam, [82] Wallpaper	
Voices	A user's voices, recordings, or the access to the microphone permission	microphone, voice, mic, speech, talk, audio	[91] Microphone	
Financial info	Information about a user's financial accounts, purchases, or transactions	credit card, company, companies, organization, commercial, organizations, pay, payment, financial, bill, wallet, purchase	[61] Cart	
Social media	A user's social media information, or the access to social media accounts	social media, Facebook, Twitter, socialmedia, share	[77] Facebook, [89] Twitter	

avoid confusion. To further promote unification and consistency across contexts, we rendered the list case-insensitive. Table 2 presents a comprehensive display of the finalized keyword list of each data type.

Icon collection. Icons frequently serve as symbols for specific categories of privacy information. One common example is that the camera-like icons typically correspond to the authorization of camera access by mobile applications or access to the device photo gallery; and a location crosshair icon often represents the access to location permission or users physical location information. Liu et al. [73] developed a comprehensive icon taxonomy, comprising 99 common icon classes in mobile applications. This research effort also yielded one of the largest icon datasets, consisting of over 118,000 distinct icons. Subsequently, we performed a semantic alignment between the RICO-icon classes proposed by Liu et al. and the data types that were identified through our survey. By doing so, the textual information (privacy-related keywords) and visual information (icons) on GUI are well aligned with regulatory requirements (data types), as detailed in Table 2.

3   Cpp4App: The Benchmark Dataset

In our study, the construction of a benchmark dataset is essential to quantitatively assess the efficacy of the proposed framework in generating contextual privacy policies (CPPs). Although existing datasets are available for privacy policy analysis [27, 29, 93, 105], and for understanding mobile graphical user interfaces (GUIs) [34, 42, 64], we identified two major shortcomings within current collections. These limitations, as detailed below, necessitate the compilation of a new dataset.

- **Temporal inconsistency between applications, policies, and GUIs.**

With mobile applications undergoing frequent changes in privacy practices, there is a concomitant need to update their corresponding privacy policies. In addition, the GUIs of these apps are subject to rapid evolution. Consequently, it becomes imperative to maintain alignment between the screenshots (representing the GUIs) and their temporally corresponding privacy policies for CPP generation. This alignment ensures the precision and dependability of quantitative evaluations.

- **No annotations about privacy-related contexts in GUIs.** To the best of our knowledge, we are the first to introduce the concept of CPPs specific to mobile applications. For the purpose of evaluating the performance of the automated generation framework, benchmark datasets should ideally incorporate two essential features: 1) the annotated regions (i.e., bounding boxes) representing privacy-related contexts within GUI screenshots, and 2) the mapping relations that connect these privacy-related contexts to corresponding segments of the privacy policy. Unfortunately, as of the time of this study, no existing dataset encompasses both of these vital characteristics.

In the following subsections, we present steps employed to create a benchmark dataset, thereby addressing the aforementioned issues.

3.1   Mobile Apps Selection

The Android platform is often considered transparent and favorable for academic research, with numerous apps markets readily available [67, 92]. Among these, the Google Play app store stands out as the most expansive and accessible, hosting more than two million applications [75, 91]. In particular, Google Play is a leader in requiring mobile applications to

provide a link to their privacy policy on their homepages [19]. As a result, users downloading Android mobile applications from the Google Play app store can directly access the respective privacy policy links.

Consistent with previous research, widely used mobile applications tend to adhere to more stringent privacy practices [61, 71, 105]. In addition, less popular applications often have poorly written privacy policies [81, 82], making it meaningless to generate CPPs for them. In this study, we leveraged market data from AppBrain [24], one of the largest third-party platforms monitoring app statistics for the Google Play app store, to pinpoint the 200 most popular mobile applications. Our study collection method includes four specific strategies. First, recognizing the variations in GUIs, functions, and privacy practices across different application categories (i.e., genre), we endeavored to ensure a diverse representation by selecting the most popular mobile applications from 17 distinct categories. Second, applications published by the same parent company normally share the same privacy policy, so we try to select apps from different developers for better representation. Third, we excluded any mobile applications with privacy policies that were less than 200 words or smaller than 2 KB in size, as these may constitute invalid or insufficient privacy policies [61, 72]. Finally, we limited our selection to those applications that offered privacy policies in English, thus narrowing our candidate pool to 50 mobile applications for subsequent fine-grained annotation.

3.2 Context Annotation Strategy

To create a benchmark dataset, it is imperative to ascertain that vital privacy-related GUI elements are incorporated into our dataset. Therefore, drawing on established guidelines about mobile application GUI design [1, 2], we have systematically gathered specific pages from each selected application. The collection process adheres to the categorizations delineated below:

- *Registration/Login page.* Mobile applications frequently gather user registration information on this page to facilitate account creation. This may encompass both personal and other private details. Certain applications require new users to register before accessing specific features, making it infeasible for these users to circumvent the registration or login process.
- *Home page.* This serves as the initial interface upon accessing a mobile application. It contains essential features and information related to its app's functionality, which may necessitate the provision of personal data or the granting of device permissions. Furthermore, screenshots of adjacent main pages are also collected for analysis.
- *Profile page.* This section may retain users' personal information. For example, some mobile applications might request access to the camera or photo roll to furnish customization options related to public profile appearance.

Table 3: Basic statistic of CPP4APP.

No. Mobile applications & Privacy policies	50
No. Screenshots	402
Screenshots per Mobile application	8.04
No. Privacy-related contexts	1,217
Contexts per Mobile application	24.34
No. Words in Privacy policies	297,010
Words per Privacy policy	5,940

- *Setting page.* Often associated with privacy customization, this page may govern permissions related to device sensors and access to personal information.
- *Onboarding page.* It serves as a tutorial or introduction. Moreover, this page elucidates a mobile application's salient features and benefits.
- *Map page.* Common to various application categories such as travel, posting, food delivery, and shopping, this page is often utilized to amass users' location data.
- *Essential pages for specific categories.* This includes specialized interfaces such as the product card pages in shopping apps, the post-editing interface in forum applications, and the feed page within social media platforms.

We follow these categorizations to systematically collect relevant pages from each selected mobile applications, ensuring that essential privacy-related GUI elements are represented in our dataset. Notably, we have opted for manual collection of GUI pages as opposed to relying on static analysis for several reasons. Accurate static analysis is a challenging task in mobile program analysis, particularly for popular apps that are not only complexly programmed but also commonly employ code obfuscation. Furthermore, iterating through all possible activities to capture their corresponding GUIs is a non-trivial task, and outdated or restricted GUIs may not be accessible through regular user interactions. Manual collection allows us to naturally avoid the aforementioned problems.

After the data collection, the annotators need to manually examine the text in each screenshot, cross-referencing it with our CPP-related keyword list, as mentioned in Table 2. If the text contains any of these CPP-related keywords, it is labeled as the corresponding data type and designated with a bounding box. Subsequently, the annotators are tasked with detecting all icons in the screenshot to determine if any belong to a privacy-related data type. If an icon meets this criterion, it is labeled with the corresponding class and marked with a bounding box. For these purposes, the *OpenCV-Python* [13] library is employed to draw bounding boxes and document coordinates. Two annotators were asked to execute the annotation process independently. For any disagreement, they discussed and agreed on the same opinion, and if the disagreement persisted, a senior annotator joined the discussion to facilitate a resolution. The percent agreement (inter-rater reliability) for initial annotation is about 84%.
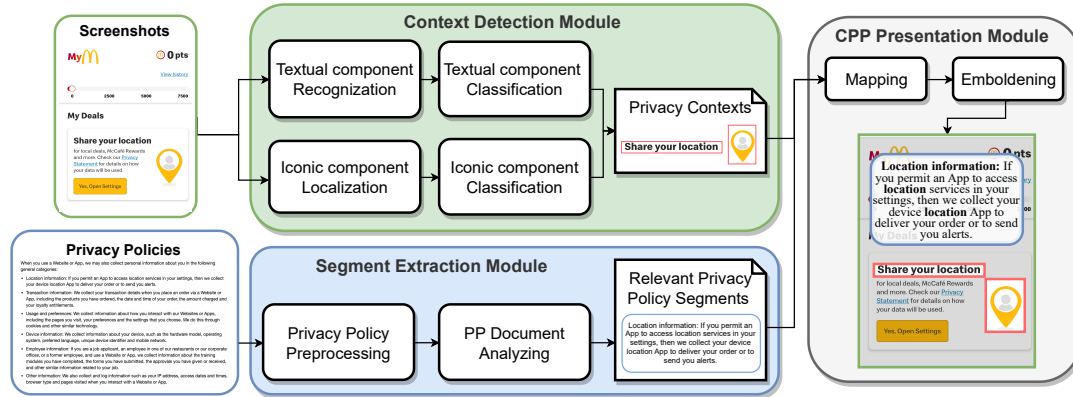
Figure 2: An overview of SEEPRIVACY.

3.3 Segment Annotation Strategy

Initially, if annotators are unfamiliar with the mobile application of the policy they need to annotate, they are mandated to explore and familiarize themselves with its features for a duration of 10 minutes. Then, two annotators independently scrutinize each data privacy-related context within the privacy policies and extract pertinent segments. To ensure the integrity and quality of the annotation process, we establish a reading speed parameter ranging from 250 to 300 words per minute (the reading speed of an average adult [30]) for the annotators, implementing countdown timers in accordance with the policy lengths. If the *segment_sim* (See the Formula 5.1.2) is above 0.8, two annotations are counted as an agreement. The mechanism for resolving disagreements is consistent with the process outlined in the previous paragraph. In cases where both annotators are unable to identify any relevant segments within the privacy policies, the segment is explicitly labeled as "No relative information is found in the privacy policy."

3.4 Setups and Statistics

All annotators involved in this study are research scientists with a minimum of two years of experience in privacy research for mobile applications. To ensure the alignment between the GUIs and privacy policies, an Android mobile phone, i.e., the OnePlus 7 Pro, Android 11, and Hydrogen OS 11.0.9.1.GM21, is employed to download all the selected mobile applications. The corresponding privacy policies are downloaded simultaneously. To preserve the status quo of the apps and policies, the auto-update features within the Google Play app store, individual mobile apps, and the system's general settings on the phone (including auto-update over Wi-Fi) are all deactivated. In total, this effort resulted in the collection of 402 screenshots from 50 diverse mobile apps, accompanied by 1,217 labeled privacy-related contexts, each correlating to specific segments within privacy policies. We have designated this benchmark dataset as CPP4APP, and the statistical

analysis is provided in Table 3. As the inaugural dataset of its nature, we are confident that it is sufficient for evaluating our automatic generation framework and positioned to serve as a benchmark for future research in this field.

4 SEEPRIVACY: The Framework

In this section, we introduce our multimodal framework SEEPRIVACY, designed to automatically generate CPPs for a given screenshot and a given privacy policy, as illustrated in Figure 2. The framework accepts privacy policies and screenshots from a given mobile application as input parameters. These inputs are processed through three primary modules within the framework to autonomously build CPPs for the designated mobile application. The three main modules integral to our framework are detailed as follows:

- *Context Detection Module:* This module is designed to identify specific contexts within a mobile application's screenshots where CPPs should be displayed.
- *Segment Extraction Module:* This module focuses on extracting pertinent segments from the privacy policy of the given mobile application.
- *CPPs Presentation Module:* This module is responsible for linking privacy policy segments to their corresponding contexts and articulating the essential information, thereby facilitating the generation of CPPs for mobile applications.

The details of each module are explained in the following subsections.

4.1 Context Detection Module

The source code of mobile apps is typically inaccessible to end-users. This limitation led us to develop a purely vision-based Context Detection Module tailored to the unique challenges presented by mobile apps. Specifically, in our approach to GUI understanding, we opt for visual components as atomic elements rather than function-level GUI widgets. This choice

is driven by the observation that certain widgets (e.g., a navigation bar) may be too coarse-grained to adequately represent a data practice.

Textual GUI components. To detect and localize privacy-related texts from a given screenshot, we apply a state-of-the-art ultra-lightweight OCR system, named PaddlePaddle-OCR (PP-OCR) [65]. After obtaining these texts, we employ a pre-trained large language model, i.e., GPT-3.5[2], to classify their data types. If the answer contains a data type, we will regard the *{detected text}* as relevant to this data type; otherwise, it does not belong to any data type.

Iconic GUI components - Localization. An empirical study [35] demonstrated that rule-based object detection methods on GUIs outperform learning-based methods, such as YOLO [86] series. In addition, rule-based methods have better explainability. To localize iconic GUI components from their screenshots, we adapt previous work [99] and propose an updated set of detection rules to filter out irrelevant objects. Specifically, we have four rules presented as follows: (a) Filtering out elements whose area is greater than 10% of the total area of the screenshot, in order to remove the bounding boxes and large images; (b) Filtering out elements whose area is less than 5% of the total area of the screenshot, aiming to remove the small noises; (c) Removing rectangular elements whose aspect ratio (width to height) is less than 0.6, ruling out control bars and top/bottom banners; (d) Removing elements that area is overlapped with OCR results since some letters in large font-size are also detected as potential icons. By applying these rules to our dataset, we can obtain the position of the iconic GUI components on their screenshots.

Iconic GUI components - Classification. CNN-based neural networks have a strong capability of recognizing patterns on GUIs [36, 73, 95]. In this paper, we train two models to classify the detected graphical icons from the screenshots, including a ViT-based model and a ResNet-based model. ViT stands for Vision Transformer [44]. Specifically, we employ a pre-trained vanilla ViT and fine-tune it on ImageNet [43]. Regarding the ResNet-based model, we employ ResNet-18 as the backbone because of its tiny size and decent efficiency. We further add dropout layers to every ResNet block to enhance its generalizability, preventing the potential over-fitting issue. We then trained both models on the RICO-icon dataset with the default train-test split. We use the ViT-based model as the default for Contextual Detection Model as it achieves the best performance.

4.2 Segment Extraction Module

The objective of this module is to obtain privacy policy segments with their corresponding privacy-related contexts. Mainstream mobile application markets, such as the Google Play app store [19] and the Apple App Store [17], require

developers to provide a privacy policy link on the application's homepage. This requirement allows us to access those links based on the name of a mobile application. To capture these privacy policies in HTML format, we leverage two Python libraries, *Selenium* [14] and *BeautifulSoup* [18], which fetch the information via the corresponding privacy policy links. Recognizing the global market trend and the need to cater to multilingual users, we note that an increasing number of privacy policies are available in multiple languages. To maintain consistency and clarity in our analysis, we deploy *langdetect* [21] library to filter out non-English texts in privacy policies. Note that our framework has the flexibility to be easily extended to accommodate other languages.

Previous studies [90, 98] have discussed that only using sentence-level privacy policies processing and analysis can lead to contradictory conclusions, since the same or similar sentences in different sections may have different implications. Therefore, to retrieve the relevant sentence-based segments, we adopt the multi-level privacy policies processing method proposed in [28, 98]. For privacy policy documents whose HTML structure follows the $(\langle\text{Heading}\rangle\langle\text{Paragraph}\rangle^+)^+$ format, we reuse the pre-trained Bayesian multi-label classifier model [98] to classify privacy policy paragraphs based on their headings since headings usually contain words that can be used as identifiers. The classifier is claimed to achieve 0.85 accuracy and can classify the majority of the documents based on their headings. The paragraphs classified as *Types* are related to the "types of personal data collected by the current app". Those paragraphs are highly related to specific data practices, so we only conduct sentence-level analysis on them. For privacy policy documents whose HTML structure does not follow the $(\langle\text{Heading}\rangle\langle\text{Paragraph}\rangle^+)^+$ format, we follow previous works [52, 94] to classify paragraphs on specific data practices. Specifically, we train a CNN-based multi-label classification model on a large privacy policy dataset [93] with 23K fine-grained data practice annotations. Note that this dataset contains 12 high-level categories for data practices, including *First-Party Collection/Use* and *Third-Party Sharing/Collection*. The results show that we can achieve a 0.84 top-1 precision for classifying the privacy segments with their related data practices.

After obtaining paragraph-level privacy segments, we tokenize them into sentences using the *stanza* [85] Python library. We then perform a two-stage sentence-level analysis to extract the sentences for each data type. First, we perform a keyword search based on our collected keyword list (see Table 2). If a keyword exists, the sentence will be added to the corresponding privacy policy segments of its data type. Second, for sentences without any keywords, we further employ a Bayesian binary classifier to determine whether the sentence is related to a data type. The classifier is trained on the dataset from [98], reaching an accuracy of 0.98. If yes, we use *SpaCy* [15] Python library to obtain noun chunks from the sentence. A noun chunk is a phrase that includes a noun and

---

[2] https://platform.openai.com/docs/models/gpt-3-5

Table 4: Performance of SeePrivacy's modules. "CDM" stands for Context Detection Module. "R-i Class" stands for Rico-icon Class as per Table 2.

(a) CDM - Textual GUI components

Category	Accuracy	Precision	Recall
Name	0.98	0.98	1.00
Birthday	1.00	1.00	1.00
Address	0.38	0.41	0.86
Phone	0.87	0.90	0.96
Email	0.61	0.97	0.62
Profile	0.72	0.77	0.92
Contacts	0.98	1.00	0.98
Location	0.92	0.96	0.96
Photos	0.87	1.00	0.87
Voices	0.91	0.91	1.00
Financial info	0.93	0.93	0.93
Social media	0.67	0.75	1.00
Average	**0.82**	**0.87**	**0.93**

(b) CDM - Iconic GUI components

R-i Class	Category	Accuracy	Precision	Recall
Call	Phone	0.96	1.00	0.96
Email	Email	0.96	0.95	1.00
Avatar	Profile	0.92	0.96	0.96
Follow	Contacts	0.89	0.94	0.94
Group	Contacts	0.73	0.73	1.00
Location	Location	0.93	0.98	0.95
Crosshair	Location	0.91	0.94	0.97
Photo	Photos	0.92	0.92	1.00
Wallpaper	Photos	0.94	0.94	1.00
Videocam	Photos	0.83	1.00	0.83
Microphone	Voices	1.00	1.00	1.00
Cart	Financial Info	0.91	0.95	0.95
Facebook	Social media	0.96	1.00	1.00
Twitter	Social media	1.00	1.00	1.00
Average		**0.92**	**0.95**	**0.96**

(c) Segments Extraction Module

Category	Accuracy	Precision	Recall
Name	1.00	1.00	1.00
Birthday	0.96	1.00	0.95
Address	0.52	0.82	0.50
Phone	1.00	1.00	1.00
Email	0.99	0.99	1.00
Profile	0.99	0.99	1.00
Contacts	0.79	1.00	0.78
Location	0.94	0.95	0.98
Photos	0.95	0.97	0.97
Voices	0.86	0.76	1.00
Financial info	1.00	1.00	1.00
Social media	1.00	1.00	1.00
Average	**0.94**	**0.98**	**0.96**

(d) Overall Context Identification

Category	Accuracy	Precision	Recall
Name	0.98	0.98	1.00
Birthday	1.00	1.00	1.00
Address	0.38	0.41	0.86
Phone	0.88	0.91	0.97
Email	0.66	0.96	0.68
Profile	0.70	0.74	0.93
Contacts	0.75	0.94	0.78
Location	0.94	0.97	0.96
Photos	0.86	0.98	0.88
Voices	0.91	0.91	1.00
Financial info	0.81	0.93	0.87
Social media	0.80	0.89	0.89
Average	**0.81**	**0.88**	**0.90**

any connected words such as its adjectives. We calculate the similarity of the phrases *phrase_sim*( ) between noun chunks and keywords as follows:

$$phrase\_sim(p_1, p_2) = \frac{2 \times path\_similarity(p_1, p_2)}{word\_count(p_1) + word\_count(p_2)}$$

where the p_1 and p_2 are a noun chunk and a keyword, respectively; *path_similarity*( ) is the function from WordNet [77], reflecting the similarity based on parsing their semantic constituency trees' structure; *word_count*( ) is to count the number of words. If the similarity between a noun chunk and a keyword is higher than 0.8, an empirically set threshold, then we regard that the sentence that the noun chunk belongs to is the data type of the keyword.

Above all, we obtain the sentence-level policy segments related to each data type and group them as privacy policy segments. We also record the positions of keywords or noun chunks in the resulting sentences. For data types that do not have any match, we simply add "No relative information is found in the privacy policy." to notify users. For less popular applications, such notifications are likely to appear frequently, as they tend not to be stringent to disclosed privacy practices [61, 71, 105].

4.3 CPPs Presentation Module

For each mobile application, privacy policy, and its screenshots, the *segment extraction module* and *context detection module* allow us to obtain the specific privacy policy segments and associated privacy contexts. We further process this information to ensure a more coherent and user-friendly presentation.

Initially, we group the detected contexts of the same data types and map them to their corresponding privacy policy segments. As illustrated in Figure 1c, the SEEPRIVACY framework detects three contexts, including two textual GUI components, i.e., *"Share your location"* and *"use your birthday,"* and one iconic GUI component, such as a location mark icon in the middle right of the screenshot. These are classified into *Location*, *Location*, and *Birthday* data types, respectively. The

text *"Share your location"* and the icon of the location mark are coupled with the segment about location information. The text *"use your birthday"* only corresponds to the segment about *Birthday* data type. Drawing on existing research that emphasizes the improvement of readability through the strategic use of typography [80], we render specific keywords or noun phrases within the privacy policy segments in bold font. This design choice, grounded in empirical findings, aims to accentuate essential information and thereby augment user comprehension.

5 Evaluation

In this section, we evaluate SEEPRIVACY both objectively and subjectively. Some visualization examples are displayed in Figure 3.

5.1 Quantitative Evaluation

Our framework, consisting of three integral modules, is systematically evaluated for its performance on our specially curated benchmark dataset, named CPP4APP.

5.1.1 Context Detection Module

In assessing whether a detected GUI component aligns with the ground truth, we utilize Intersection over Union (IoU) [87], a prevalent evaluation metric in object detection models. If the IoU between the ground truth and the detected component area exceeds a threshold β, the match is considered correct. Consistent with standard practice in object detection tasks [84], we empirically establish β = 0.5.

Textual GUI components. Table 4a presents the results related to the detection of textual GUI components. Our method exhibits a strong capability to recognize and categorize texts within mobile GUI screenshots, achieving an accuracy of 0.82, a precision of 0.87, and a recall of 0.93.

Iconic GUI components. Table 5 shows the comparison of the icon localization capability between our method and previous methods. The results show that our proposed method

Table 5: Performance comparison between iconic GUI components localization methods.

Method	Precision (mislabel)	Recall (missing label)
UIED [99]	0.78	0.44
Liu et al. [73]	0.79	0.86
Ours	**0.84**	**0.88**

Table 6: Performance and efficiency comparison for iconic GUI elements classification on the RICO-icon dataset. "kNN" stands for k-nearest neighbors algorithm, and we use the normalized pixel values of the grayscale icon images as features. "Acc.", "Prec." and "Rec." represents Accuracy, Precision and Recall, respectively. Time(s) denotes the elapsed seconds to process a thousand icons.

Model	Acc.	Prec.	Rec.	Size(MB)	Time(s)
kNN [38]	0.75	0.61	0.52	**0**	6.78
Liu et al. [73]	0.90	**0.90**	0.79	293	5.02
MobileNetV2 [88]	0.86	0.73	0.73	9	0.56
MobileNetV3 [55]	0.79	0.62	0.66	6	0.52
Ours (ViT-based)	**0.97**	**0.90**	**0.90**	328	11.28
Ours (ResNet-based)	0.91	0.80	0.81	43	**0.37**

performs better in identifying the icons curated on GUI screenshots. Table 4b shows the class-wise breakdown for Iconic GUI components, achieving an accuracy of 0.92, a precision of 0.95, and a recall of 0.96. We then compare our icon classification models with several baselines on the RICO-icon dataset. Table 6 shows that our ViT-based model achieves the best performance, but has the largest model size of 328 MB and the slowest time efficiency of 11.28 seconds. The ResNet-based model provides a good balance of performance and efficiency, with the fast time efficiency, decent performance metrics, and a much smaller model size compared to others.

Overall context identification. We further assess the overall performance of our context detection module by considering both textual and iconic GUI components. Table 4d shows the overall context identification results considering textual and iconic GUI components. Specifically, SEEPRIVACY exhibits commendable results, achieving accuracy, precision, and recall rates of 0.81, 0.88, and 0.90, respectively.

5.1.2 Segment Extraction Module

First, to determine whether the retrieval is successful, we introduce a similarity metric, denoted by *segment_sim*( ), which quantifies the similarity between the retrieved segments and the manually annotated ground truth. For a given ground truth s_{gt} and a retrieved segments s_{ret}, our segment similarity

Table 7: Statistic results of our human evaluation. SD stands for standard deviations.

Topic	Mean	Median	SD
Data type & Context	4.22	5	1.14
Data type & Policy segment	3.95	4	1.28
Policy segment & Context	3.75	4	1.40

segment_sim( ) is calculated as follows:

$$segment\_sim(s_{ret}, s_{gt}) = \frac{1}{\text{MIN}(n,m)} \sum_{i=1}^{n} \sum_{j=1}^{m} \overline{lcs}(s_i^{ret}, s_j^{gt})$$

$$\overline{lcs}(s_i^{ret}, s_j^{gt}) = \frac{lcs(s_i^{ret}, s_j^{gt})}{\text{MIN}(\text{LEN}(s_i^{ret}), \text{LEN}(s_j^{gt}))}$$

where $lcs($) stands for the function to calculate longest common string; s_j^{ret} are phrases separated by punctuation of retrieved segments and s_i^{gt} are labelled phrases of ground truth segments. If it is greater than 0.8, an empirically set threshold, then the segment pair is regarded as a successful match. Table 4c shows the results of the segment extraction module, SEEPRIVACY achieves 0.94, 0.98, and 0.96 in terms of accuracy, precision, and recall, respectively. There are some categories, such as *Address*, which are more complex than the other categories due to relatively semantically ambiguous keywords. In addition, there are relatively less data for *Voices* and *Contacts*, thus the limited performance of this category could be caused by the randomness.

5.2 Human Evaluation

To further validate the functionality of SEEPRIVACY, we conduct a human evaluation to examine users' perceptions of its capacity for generating accurate CPPs for mobile applications. This investigation seeks to provide empirical evidence on how SEEPRIVACY performs in real-world scenarios from users' perspective, thereby offering insights into its effectiveness and relevance.

5.2.1 Evaluation Study Design

For recruitment, we recruited examiners based on the following criteria: professional fluency in English reading and be capable of using mobile apps without accessibility tools (e.g., VoiceOver on iOS), via mailing lists of authors' institution without offering monetary incentives. For the human evaluation, we randomly sampled 30 screenshots with their privacy policies from the benchmark dataset and employ SEEPRIVACY to generate CPPs. The manual examination involves heavy privacy policy reading and comprehending tasks, and we aim to reduce the session length to keep the examiners focused and increase the result validity. The average duration of

evaluation session is about 30 minutes. To guarantee the validity of evaluation, we then divide the generated CPPs into three folds and compose the questionnaire for each fold, i.e., ten screenshots in each questionnaire. For each CPP, examiners need to rate the following statements:

- The data type matches with the detected context.
- The data type matches with the displayed privacy policy segment.
- The displayed privacy policy segment explains the detected context on the screenshot.

For all the ten allocated screenshots, they are asked to rate the agreement level on a 5-point Likert scale ranging from five (strongly agree) to one (strongly disagree). In addition to providing a general introduction to the user study and obtaining consent, we also provide examiners with detailed descriptions of the data types, explicit instructions, and a concrete example to guide them prior to the beginning of our study.
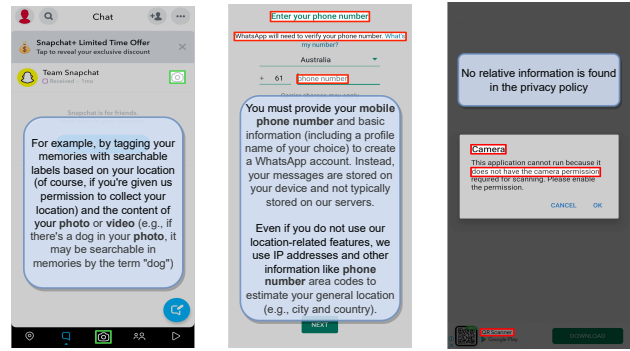
5.2.2 Evaluation Results

In total, we recruited 15 people for the evaluation (6 females, 8 males, and 1 who preferred not to be mentioned). Each fold of screenshots was evaluated by five different examiners. The average duration for evaluation sessions is about 30 minutes. The results are detailed in Table 7. For the functionality evaluation pertaining to the matching between data types and detected contexts, the ratings were highly favorable, with more than 81% of the responses falling within the categories of "Agree" or "Strongly Agree." The mean and median values for this aspect were also substantial at 4.22 and 5, respectively. Concerning the alignment between data types and privacy policy segments, over 72% of examiners assigned positive ratings, yielding a mean value of 3.95 and a median value of 4. Additionally, more than 77% of examiners concurred that the retrieved privacy policy segments help explain the detected contexts, evidenced by a mean rating of 3.75 and a median rating of 4. Overall, these results demonstrate that SEEPRIVACY successfully fulfills its intended purpose by offering an efficient and effective solution for delivering CPPs to users.

6 Discussion

In this section, we delve into the implications, potential applications, and contributions of our framework, elucidating its significance both for individual mobile application users and the wider community.

6.1 Privacy Concern and Reading Willingness

Before and after the human evaluation, we also asked several questions about privacy concerns and their privacy policy reading habits to examiners. Table 8 lists the specific questions (Q_1 and Q_2 are pre-evaluation; Q_3 and Q_4 are post-evaluation)

(a) Sceenshot 6-1   (b) Screenshot 11-9   (c) Screenshot 16-1

Figure 3: Three examples appeared in the human evaluation.

and results. It is evident that privacy concerns are a prominent issue for users while using mobile applications, with a mean concern rating of 4.13 and a median value of 4 for question Q_1. The utility of SEEPRIVACY in furnishing relevant privacy information for mobile applications was highly acclaimed by the participants. Specifically, 14 out of 15 participants evaluated it as either "Very useful" or "Useful."

In addition, we compare the reading willingness between traditional privacy policies and the CPPs generated by SEEPRIVACY. The results demonstrate a substantial increase in the willingness of participants to engage with CPPs, as reflected by a mean score that surged from 2 to 4.1. Furthermore, among 8 participants who are "very concerned" about their privacy information while using mobile apps, 6 select that they will always read CPPs, and 2 select often read CPPs. These findings collectively underscore the potential of SEEPRIVACY. By enhancing the transparency, accessibility, and digestibility of privacy policies, our proposed framework plays a vital role in engaging users to read privacy statements, potentially mitigating the prevalence of "digital resignation". Although the diversity of participants is slightly under representative, we believe those preliminary findings are still valuable to be reported in this paper.

6.2 Potential Adoption Scenario

6.2.1 CPP in App Market

Our framework has the potential to be applied in various real-world scenarios. One plug-and-play application is displaying CPP in the app market, as illustrated in Figure 4. Research has shown that users' app downloading decisions are often influenced by curated market information such as ratings and reviews [50]. On the apps' homepage (①, ②) in app markets (e.g., the Google Play app store), developers usually curate some showcase screenshots (③) of their apps to exhibit some characteristics. Those screenshots often include the actual GUIs that appear during the users' actual usage. As a privacy policy link (④) is also commonly mandated by app markets,

Table 8: Pre- and post-questions of the functionality evaluation about the general usability of SeePrivacy.

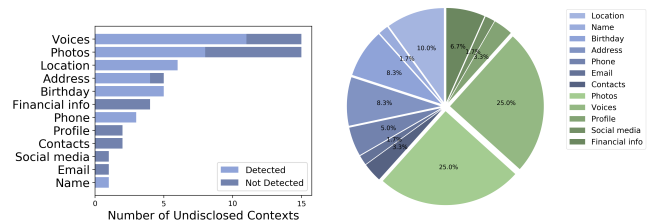| No.	Pre- and post-questions	Scale (5-Likert)	Mean	Median	SD
Q_1	How concerned are you about your privacy information while using mobile apps?	5 for very concerned, 1 for very unconcerned	4.13	4	1.09
Q_2	Do you read mobile app's privacy policies when you encountered?	5 for always read, 1 for never read	2.00	2	1.03
Q_3	What do you think the usefulness of this tool in terms of providing privacy information for mobile apps?	5 for very useful, 1 for very useless	4.53	5	0.62
Q_4	Will you read contextual privacy policies when you encountered in future?	5 for always read, 1 for never read	4.07	4	0.85

Figure 4: An example of the application scenario of our framework. (1) The homepage of the app at the Google Play app store. (2) Basic information of the app. (3) The showcase screenshots provided by the app developer. (4) The link of privacy policy. (5) The CPPs generated by SEEPRIVACY.

(a) Detection Result of Contexts (b) Distribution of Contexts

Figure 5: The left figure shows the detection result of contexts that do not have information in privacy policies. The right figure presents the data type distribution of those contexts.

our framework will be able to generate the CPPs (⑤) by utilizing the showcase screenshots and the corresponding privacy policy on the homepage. By leveraging SEEPRIVACY, users can gain contextualized insights into potential data practices and pertinent policy statements of the mobile application prior to installation. This application allows users to comprehend data practices with real-use scenarios, enabling them to make informative decisions and establish better expectations before downloading the app.

6.2.2 Lack of Disclosure: Contextual Detection

Numerous prior studies [97, 98, 104, 105] have highlighted the inconsistency between statements articulated in privacy policies and an application's actual behaviors. Specifically, the absence of proper disclosure concerning data practices may place users at considerable risk concerning digital security. In the current landscape of privacy policy analysis, we identify two critical shortcomings. First, in the case of essential applications, users often find themselves in a position where they must accept the terms, even if they are made aware of any lack of disclosure within the policies. There is no space

to negotiate with developers for specific functions or certain GUI components under the "all-or-nothing" model. Second, existing methodologies fail to link the noted inconsistencies to the real contexts with which users may engage on GUIs. This omission presents a significant gap in understanding how policy misalignment may manifest in a user's interaction with an application. The lack of notice and choice inevitably leads to a digital resignation phenomenon, where users acquiesce to potentially problematic terms out of necessity, rather than informed consent.

To address the aforementioned challenges, SEEPRIVACY is proposed to provide the affected GUI elements, i.e., the context, allowing users to make informed decisions about their interactions. For example, in Figure 3c, SEEPRIVACY detects three contexts related to *"Photos"* in red bounding boxes, but does not obtain any relevant statements in the privacy policy of the app. Therefore, "No relative information is found in the privacy policy" is displayed to warn users about the potential risk. Users can simply choose not to interact with those *"Photos"*-related GUI elements as a band-aid solution, meanwhile "safely" using other features of the app. In total, we found that 60 contexts do not have corresponding privacy policy segments in CPP4APP. The 60 contexts are distributed among 17 mobile applications in CPP4APP and cover all 12 data types. Figure 5 illustrates the specific numbers and the distribution for lack of disclosure. Among the various data

types, *Photos* and *Voices* combined constitute 50% of the data. Interestingly, the *Voices* data type seems to be frequently overlooked. Contexts for this data type typically involve requests to record audio (e.g., a microphone icon), or the conversion of voice to text for search functionality (e.g., a phone icon). In total, SEEPRIVACY successfully identified 63.3% (38/60) lack of disclosure contexts.

6.3 Broader Impacts

Legal Impacts. Privacy regulations frequently stipulate specific requirements regarding the completeness of privacy practices within policies. For instance, GDPR mandates that privacy notices disclose the categories of personal data concerned [GDPR, Art. 14(1)(d)] and recipients or categories of recipients of the personal data [Art. 13(1)(e), 14(1)(e)]. Likewise, the CCPA prescribes that privacy notices should enumerate the categories of personal information collected in the preceding 12 months [CCPA §1798.130(a)(5)(B), §1798.110(c), Regs §999.308(c)(1)(d)]. While such requirements are often adhered to by mobile application developers and can be directly assessed by regulators, **the challenges of formulating best practices to achieve transparency, and evaluating the transparency level of a privacy statement, persist**. In this context, our innovative definition of CPPs within the mobile scenario emerges as a potential metric for evaluating transparency. By linking the stated privacy practices to the actual GUI elements with which users interact, SEEPRIVACY offers an empirical means of ensuring that privacy statements are not only compliant with legal requirements but also meaningfully transparent to end-users. This approach aligns with the broader legal and ethical imperatives of user-centric privacy and could contribute significantly to the development of transparent and accountable privacy practices in the mobile application ecosystem.

Accessibility and Inclusion. By transforming privacy policies into a contextual and understandable format, our approach potentially broadens their accessibility to a more diverse range of users. This includes individuals who may not possess advanced literacy skills as well as those who face challenges related to reading disorders. Such inclusivity not only democratizes the understanding of privacy policies but also aligns with broader societal goals of ensuring that vital information is comprehensible to a wide audience. In this regard, SEEPRIVACY contributes to creating a more equitable digital environment where awareness and control over personal data are not confined to a specific subset of users but are extended to encompass various demographic groups. The potential reach and impact of this approach underscore the need for continued research and innovation in the domain of user-centered privacy communication.

Children and Education. The GDPR Recital 58 (The Principle of Transparency) states "... *any information and communication, where processing is addressed to a child, should be in such a clear and plain language that the child can easily understand.*" This principle acknowledges the unique challenge of making legal and privacy-related texts accessible to young audiences. The SEEPRIVACY framework aligns with this directive by transforming privacy policies into potentially engaging and comprehensible formats to children. By connecting formal legal terms with real privacy use cases, it can facilitate a more straightforward understanding of the implications of online activities. While SEEPRIVACY strives to comply with regulatory requirements, its broader goal is to contribute to the digital literacy of disadvantage groups. This effort seeks to provide them with a basic understanding of online privacy, aiming to equip the next generation with the knowledge needed to approach the digital world with informed caution and responsibility. SEEPRIVACY aligns with the growing emphasis on the creation of tools and resources specifically designed to meet the developmental needs and competencies of young internet users.

6.4 *Notice-and-Choice Privacy Framework* for Mobile Apps: Is This the Way?

The *notice-and-choice* framework for mobile application ecosystem is fundamentally challenged by two critical issues. First, there is a significant disconnection between privacy notices and user-oriented interactions, leading to ineffective engagement of users in meaningful "notice" processes and subsequent privacy decision-making. Second, as illustrated in Figure 1a and 1b, the current designs of privacy controls often present users with an all-or-nothing choice, leaving them without genuine control over their data privacy. Therefore, this imbalance between utility and privacy highlights the need for administrative action through the regulation establishment and enforcement to safeguard consumer privacy rights.

While SEEPRIVACY is not a "panacea" for these challenges, it enlightens "a new hope" in this landscape. It suggests a paradigm shift in the privacy *notice* (what users are agreeing to) and *choice* (to agree or not) from a generic application level to specific functions or certain GUI components. It is critical to proactively elevate privacy considerations from non-functional to functional requirements in software design and development. Our framework aims to actively prompt users with timely and contextually relevant privacy information, enabling the potential for more granular control in future. In addition, exploring the practice of Right to be Forgotten can sufficiently supplement this approach, empowering users with more control over their choices [102, 103].

6.5 Limitations.

Our current human evaluation focuses more on the functionality of SEEPRIVACY, i.e., the level of matching between contexts and retrieved policy segments, thus, all examiners are relatively well-educated in technology. Thus, the preliminary

findings of reading willingness and popularity of CPPs among them may be biased due to the small and non-representative sample. We plan to further conduct a large-scale user study that includes people from diverse backgrounds in the future. In addition, the exploration of fatigue and habituation within human-computer interaction and its impact on interaction design have not been fully developed. These factors may profoundly affect user engagement and interface efficacy. Furthermore, the time and storage efficiency of SEEPRIVACY were evaluated theoretically and may not reflect real-world performance in different application scenarios. Regarding generalizability, our CPP detection is entirely vision-based, allowing for straightforward adaptation to other platforms, such as iOS mobile applications.

7 Conclusion and Future Work

Privacy policies have long been criticized for their complexity and poor readability, leading to a growing interest in contextual privacy policies (CPPs) as user-focused alternatives. In this paper, we introduce the concept of CPPs specifically within the domain of mobile applications and propose a novel framework, SEEPRIVACY, designed to automatically generate these privacy policies. We rigorously evaluated SEEPRIVACY on our benchmark dataset, CPP4APP, which consists of over 1,200 privacy-related contexts and corresponding privacy policy segments. In the quantitative evaluation, our framework demonstrates 0.88 precision and 0.90 recall to detect contexts; as well as 0.98 precision and 0.96 recall to extract corresponding policy segments. Furthermore, a human evaluation was conducted to validate the functionality our approach, reflected by a median score of 4 on a 5-point Likert scale. Our pre- and post-survey revealed that people exhibit much greater willingness to engage with CPPs, scoring 4.1 out of 5, compared to a mere 2 out of 5 for traditional privacy policies. We also discussed the potential adoption scenarios and implications of SEEPRIVACY to the wider community.

Our promising results identify two potential research directions. First, while our current efforts emphasize methodological development and evaluation, we recognize the importance of exploring real-world deployment scenarios. The user experience, particularly how users interact with contextual privacy policies, is vital and could significantly impact the system's effectiveness. Future work may include integrating SEEPRIVACY into specific human-device interactions, such as live applications on smartphones or sandbox environments within which applications operate. Second, although we aim to deliver the original privacy policy content without any modification, we notice that some retrieved policy segments are too long to comprehend in a timely manner. Thereby, it is worthwhile to evaluate how to digest lengthy policy segments with minimum information loss and distortion to maintain their integrity and practicability simultaneously. Overall, we believe this work provides a substantial step forward in deliv-

ering transparent privacy notices, paving the way for future research and practical implementation in this crucial area.

Data Availability. Benchmark dataset CPP4APP and the replication package are available at: https://github.com/Cpp4App/Cpp4App.

Acknowledgments

References

[1] 10 Essential Types of App Screens in Mobile UI Design. cadabra.studio/blog, Accessed: 2023-02-21.

[2] App Design 9 Main Types of App Screens. decode.agency/article, Accessed: 2023-02-21.

[3] Australian Privacy Principles (APP). oaic.gov.au/privacy, Accessed: 2022-05-03.

[4] Permissions for Apps on the Amazon Appstore. amazon.com/gp/help/customer, Accessed: 2023-05-10.

[5] A Complete Understanding of Mobile App vs. Mobile Website. appvelocity.ca/blog, Accessed: 2023-10-17.

[6] California Consumer Privacy Act of 2018 (CCPA). oag.ca.gov/privacy/ccpa, Accessed: 2022-04-25.

[7] Children's Online Privacy Protection Rule (COPPA). ftc.gov/legal-library, Accessed: 2023-02-20.

[8] The Online Privacy Protection Act of 2003 (CalOPPA). leginfo.legislature.ca.gov, Accessed: 2023-02-20.

[9] Fair Information Practice Principles (FIPPs). fpc.gov/resources/fipps/, Accessed: 2023-04-05.

[10] General Data Protection Regulation (GDPR). https://gdpr-info.eu/, Retrieved: 2022-04-25.

[11] Selecting Collection Service Scenarios and Data Items to Be Collected upon App Release. developer.huawei.com/consumer, Accessed: 2023-02-20.

[12] Guide to the General Data Protection Regulation. ico.org.uk/for-organisations/, Accessed: 2023-04-05.

[13] OpenCV. pypi.org/project, Accessed: 2023-07-24.

[14] Selenium. selenium.dev/, Accessed: 2022-10-01.

[15] SpaCy. github.com/spaCy, Accessed: 2023-05-01.

[16] UNCTAD. unctad.org/page, Accessed: 2022-08-26.

[17] App Privacy Details App Store. developer.apple/app-store/, Accessed: 2023-02-20.

[18] Beautifulsoup. software/bs4/, Accessed: 2022-10-01.

[19] Android Developers, Google Play Console Help Center. android-developer/answer, Accessed: 2022-04-30.

[20] Stack Overflow - How to request permissions ... apps in Google Play? stackoverflow.com/questions/44985078/, Accessed: 2023-04-20.

[21] langdetect. pypi/langdetect/, Accessed: 2022-10-01.

[22] Samsung Galaxy Store. galaxystore.samsung.com.

[23] Fair Information Practice Principles, 2010. web.archive/www.ftc.gov/reports, Archived March 9, 2010, at the Wayback Machine.

[24] AppBrain, 2023. www.appbrain.com/.

[25] Understand Google Play's Data Safety Section, Accessed: 2023-02-20. support.google.com/.

[26] Paul C Adams. Agreeing to surveillance: Digital news privacy policies. *Journalism & Mass Communication Quarterly*, 97(4):868–889, 2020.

[27] Ryan Amos et al. Privacy policies over time: Curation and analysis of a million-document dataset. In *Proceedings of the Web Conference 2021*, 2021.

[28] Benjamin Andow, Samin Yaseer Mahmud, Wenyu Wang, Justin Whitaker, William Enck, Bradley Reaves, Kapil Singh, and Tao Xie. Policylint: Investigating internal privacy policy contradictions on google play. In *USENIX Security Symposium*, pages 585–602, 2019.

[29] Benjamin Andow, Samin Yaseer Mahmud, Justin Whitaker, William Enck, Bradley Reaves, Kapil Singh, and Serge Egelman. Actions speak louder than words:{Entity-Sensitive} privacy policy and data flow analysis with {PoliCheck}. In *29th USENIX Security Symposium*, pages 985–1002, 2020.

[30] Jarni Blakkarly and Daniel Graham. Privacy policy comparison reveals half have poor readability, 2022. choice.com.au/consumers-and-data/.

[31] Davide Bolchini, Qingfeng He, Annie I Antón, and William Stufflebeam. "I Need It Now": Improving Website Usability by Contextualizing Privacy Policies. In *Web Engineering: 4th International Conference, ICWE 2004*, pages 31–44. Springer.

[32] Andrew Buck. Mobile Apps vs Mobile Websites: Which is Best for 2023?, 2023. mobiloud.com/blog.

[33] Duc Bui, Brian Tang, and Kang G Shin. Detection of inconsistencies in privacy practices of browser extensions. In *2023 IEEE Symposium on Security and Privacy (SP)*, pages 2780–2798. IEEE, 2023.

[34] Sara Bunian, Kai Li, et al. Vins: Visual Search for Mobile User Interface Design. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, pages 1–14, 2021.

[35] Jieshan Chen, Mulong Xie, Zhenchang Xing, Chunyang Chen, Xiwei Xu, Liming Zhu, and Guoqiang Li. Object detection for graphical user interface: Old fashioned or deep learning or a combination? In *In proceedings of the 28th ACM joint meeting on ESEC/FSE*, pages 1202–1214, 2020.

[36] Jieshan Chen, Amanda Swearngin, Jason Wu, Titus Barik, Jeffrey Nichols, and Xiaoyi Zhang. Towards complete icon labeling in mobile applications. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*, pages 1–14, 2022.

[37] Corey A Ciocchetti. The future of privacy policies: A privacy nutrition label filled with fair information practices. *J. Marshall J. Computer & Info. L.*, 26:1, 2008.

[38] Thomas Cover and Peter Hart. Nearest neighbor pattern classification. *IEEE transactions on information theory*, 13(1):21–27, 1967.

[39] Lorrie Cranor. *Web privacy with P3P*. 2002.

[40] Lorrie Faith Cranor. Necessary but not sufficient: Standardized mechanisms for privacy notice and choice. *J. on Telecomm. & High Tech. L.*, 10:273, 2012.

[41] Hao Cui, Rahmadi Trimananda, Athina Markopoulou, and Scott Jordan. {PoliGraph}: Automated privacy policy analysis using knowledge graphs. In *32nd USENIX Security Symposium*, pages 1037–1054, 2023.

[42] Biplab Deka, Zifeng Huang, Chad Franzen, et al. Rico: A mobile app dataset for building data-driven design applications. In *Proceedings of the 30th annual ACM symposium on UIST*, pages 845–854, 2017.

[43] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on CVPR*, pages 248–255.

[44] Alexey Dosovitskiy et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[45] Nora A Draper and Joseph Turow. The corporate cultivation of digital resignation. *New media & society*, 21 (8):1824–1839, 2019.

[46] Zohar Efroni, Jakob Metzger, Lena Mischau, and Marie Schirmbeck. Privacy icons: a risk-based approach to visualisation of data processing. *Eur. Data Prot. L. Rev.*, 5:352, 2019.

[47] Adrienne Porter Felt, Kate Greenwood, and David Wagner. The effectiveness of install-time permission systems for third-party applications. *University of California at Berkely, Electrical Engineering and Computer Sciences, Technical report*, 2010.

[48] Denis Feth. Transparency through contextual privacy statements. In *Mensch und Computer 2017 - Workshopband*, 2017. doi: 10.18420/muc2017-ws05-0406.

[49] Carlos Flavián and Miguel Guinalíu. Consumer trust, perceived security and privacy policy: three basic elements of loyalty to a web site. *Industrial management & data Systems*, 106(5):601–620, 2006.

[50] Necmiye Genc-Nayebi and Alain Abran. A systematic literature review: Opinion mining studies from mobile app store user reviews. *Journal of Systems and Software*, 125:207–219, 2017.

[51] Hamza Harkous, Sai Teja Peddinti, Rishabh Khandelwal, Animesh Srivastava, and Nina Taft. Hark: A deep learning system for navigating privacy feedback at scale. In *2022 IEEE Symposium on S&P*.

[52] Hamza Harkous, Kassem Fawaz, Rémi Lebret, Florian Schaub, Kang G Shin, and Karl Aberer. Polisis: Automated analysis and presentation of privacy policies using deep learning. In *27th USENIX security symposium*, pages 531–548, 2018.

[53] Leif-Erik Holtz, Katharina Nocun, and Marit Hansen. Towards displaying privacy information with icons. In *Privacy and Identity Management for Life: 6th IFIP WG, PrimeLife International Summer School.*, pages 338–348. Springer, 2011.

[54] Leif-Erik Holtz, Harald Zwingelberg, and Marit Hansen. Privacy policy icons. In *Privacy and Identity Management for Life*, pages 279–285. Springer, 2011.

[55] Andrew Howard, Mark Sandler, Vijay Vasudevan, et al. Searching for mobilenetv3. In *Proceedings of the IEEE/CVF ICCV*, 2019.

[56] Jasmin Kaur, Rozita A Dara, Charlie Obimbo, Fei Song, and Karen Menard. A comprehensive keyword analysis of online privacy policies. *Information Security Journal: A Global Perspective*, 27(5-6):260–275, 2018.

[57] Patrick Gage Kelley, Joanna Bresee, Lorrie Faith Cranor, and Robert W Reeder. A" nutrition label" for privacy. In *Proceedings of the 5th Symposium on Usable Privacy and Security*, pages 1–12, 2009.

[58] Patrick Gage Kelley, Lucian Cesca, Joanna Bresee, and Lorrie Faith Cranor. Standardizing privacy notices: an online study of the nutrition label approach. In *Proceedings of the SIGCHI Conference on Human factors in Computing Systems*, pages 1573–1582, 2010.

[59] Patrick Gage Kelley, Lorrie Faith Cranor, and Norman Sadeh. Privacy as part of the app decision-making process. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pages 3393–3402, 2013.

[60] Katharine Kemp. Concealed data practices and competition law: why privacy matters. *European Competition Journal*, 16(2-3):628–672, 2020.

[61] Renuka Kumar, Apurva Virkud, Ram Sundara Raman, Atul Prakash, and Roya Ensafi. A large-scale investigation into geodifferences in mobile apps. In *31st USENIX Security Symposium*.

[62] AMPLITUDE LABS. 2022 App vs. Website Report. amplitude.com/guides/2022-app-vs-website-report, Accessed: 2023-10-17.

[63] Martha K Landesberg, Toby Milgrom Levin, Caroline G Curtin, and Ori Lev. Privacy online: A report to congress. *NASA*, (19990008264), 1998.

[64] Luis A Leiva, Yunfei Xue, Avya Bansal, Hamed R Tavakoli, Tuðçe Köroðlu, Jingzhou Du, Niraj R Dayama, and Antti Oulasvirta. Understanding visual saliency in mobile user interfaces. In *22nd International conference on human-computer interaction with mobile devices and services*, pages 1–12, 2020.

[65] Chenxia Li, Weiwei Liu, Ruoyu Guo, Xiaoting Yin, Kaitao Jiang, Yongkun Du, Yuning Du, Lingfeng Zhu, Baohua Lai, Xiaoguang Hu, et al. Pp-ocrv3: More attempts for the improvement of ultra lightweight ocr system. *arXiv preprint arXiv:2206.03001*, 2022.

[66] Jingjie Li, Kaiwen Sun, Brittany Skye Huff, Anna Marie Bierley, Younghyun Kim, Florian Schaub, and Kassem Fawaz. "It's up to the Consumer to be Smart": Understanding the Security and Privacy Attitudes of Smart Home Users on Reddit. In *IEEE Symposium on Security and Privacy)*, pages 380–396, 2023.

[67] Li Li, Tegawendé F Bissyandé, and Jacques Klein. Rebooting research on detecting repackaged android apps: Literature review and benchmark. *IEEE Transactions on Software Engineering*, 47(4):676–693, 2019.

[68] Tianshi Li, Kayla Reiman, Yuvraj Agarwal, Lorrie Faith Cranor, and Jason I Hong. Understanding challenges for developers to create accurate privacy nutrition labels. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*.

[69] Tianshi Li, Elizabeth Louie, Laura Dabbish, and Jason I Hong. How developers talk about personal data and what it means for user privacy: A case study of a developer forum on reddit. *Proceedings of the ACM on Human-Computer Interaction*, 4(CSCW3):1–28, 2021.

[70] Yucheng Li, Deyuan Chen, Tianshi Li, Yuvraj Agarwal, Lorrie Faith Cranor, and Jason I Hong. Understanding ios privacy nutrition labels: An exploratory large-scale analysis of app store data. In *CHI Conference on Human Factors in Computing Systems Extended Abstracts*, pages 1–7, 2022.

[71] Shuang Liu, Baiyang Zhao, Renjie Guo, Guozhu Meng, Fan Zhang, and Meishan Zhang. Have you been properly notified? automatic compliance analysis of privacy policy text with gdpr article 13. In *Proceedings of the Web Conference 2021*, pages 2154–2164, 2021.

[72] Shuang Liu, Fan Zhang, Baiyang Zhao, Renjie Guo, Tao Chen, and Meishan Zhang. Appcorp: a corpus for android privacy policy document structure analysis. *Frontiers of Computer Science*, 17(3):173320, 2023.

[73] Thomas F. Liu, Mark Craft, Jason Situ, Ersin Yumer, Radomir Mech, and Ranjitha Kumar. Learning design semantics for mobile apps. In *The 31st Annual ACM Symposium on User Interface Software and Technology*, 2018.

[74] Zhongmin Ma. *Android application install-time permission validation and run-time malicious pattern detection*. PhD thesis, Virginia Tech, 2014.

[75] Ivano Malavolta, Stefano Ruberto, Tommaso Soru, and Valerio Terragni. End users' perception of hybrid mobile apps in the google play store. In *2015 IEEE International Conference on Mobile Services*, pages 25–32. IEEE, 2015.

[76] Mariavittoria Masotina and Anna Spagnolli. Transparency of privacy notices and contextualisation: effectively conveying information without words. *Behaviour & Information Technology*, 41(10), 2022.

[77] George A Miller. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41, 1995.

[78] Jonathan A Obar and Anne Oeldorf-Hirsch. The biggest lie on the internet: Ignoring the privacy policies and terms of service policies of social networking services. *Information, Communication & Society*, 23(1):128–147, 2020.

[79] Anna-Marie Ortloff, Maximiliane Windl, Valentin Schwind, and Niels Henze. Implementation and in situ assessment of contextual privacy policies. In *Proceedings of the 2020 ACM Designing Interactive Systems Conference*, pages 1765–1778, 2020.

[80] Hilary Palmén, Michael Gilbert, and David Crossland. How bold can we be? the impact of adjusting font grade on readability in light and dark polarities. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*, pages 1–11, 2023.

[81] Shidong Pan, Thong Hoang, Dawen Zhang, Zhenchang Xing, Xiwei Xu, Qinghua Lu, and Mark Staples. Toward the cure of privacy policy reading phobia: Automated generation of privacy nutrition labels from privacy policies. *arXiv preprint arXiv:2306.10923*, 2023.

[82] Shidong Pan, Dawen Zhang, Mark Staples, Zhenchang Xing, Jieshan Chen, Xiwei Xu, and James Hoang. A large-scale empirical study of online automated privacy policy generators for mobile apps. *arXiv preprint arXiv:2305.03271*, 2023.

[83] Alfredo J Perez, Sherali Zeadally, and Jonathan Cochran. A review and an empirical analysis of privacy policy and notices for consumer internet of things. *Security and Privacy*, 1(3):e15, 2018.

[84] Pushkar Pushp. Understanding the IoU Metric in Object Detection. pub.towardsai.net/understanding-iou-metric-in-object-detection, Accessed: 2023-07-24.

[85] Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. Stanza: A Python natural language processing toolkit for many human languages. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, 2020.

[86] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016.

[87] Hamid Rezatofighi, Nathan Tsoi, JunYoung Gwak, Amir Sadeghian, Ian Reid, and Silvio Savarese. Generalized intersection over union: A metric and a loss for bounding box regression. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 658–666, 2019.

[88] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4510–4520, 2018.

[89] Ruoxi Sun and Minhui Xue. Quality assessment of online automated privacy policy generators: an empirical study. In *Proceedings of the Evaluation and Assessment in Software Engineering*, pages 270–275. 2020.

[90] Damiano Torre, Sallam Abualhaija, Mehrdad Sabetzadeh, Lionel Briand, et al. An ai-assisted approach for checking the completeness of privacy policies against gdpr. In *2020 IEEE 28th International Requirements Engineering Conference (RE)*.

[91] Haoyu Wang, Zhe Liu, Jingyue Liang, Narseo Vallina-Rodriguez, Yao Guo, Li Li, Juan Tapiador, Jingcun Cao, and Guoai Xu. Beyond google play: A large-scale comparative study of chinese android app markets. In *Proceedings of the Internet Measurement Conference 2018*, pages 293–307, 2018.

[92] Xuetao Wei and Michael Wolf. A survey on https implementation by android apps: issues and countermeasures. *Applied Computing and Informatics*, 13(2): 101–117, 2017.

[93] Shomir Wilson, Florian Schaub, Aswarth Abhilash Dara, Frederick Liu, Sushain Cherivirala, Pedro Giovanni Leon, Mads Schaarup Andersen, Sebastian Zimmeck, et al. The creation and analysis of a website privacy policy corpus. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*.

[94] Maximiliane Windl, Niels Henze, Albrecht Schmidt, and Sebastian S Feger. Automating contextual privacy policies: Design and evaluation of a production tool for digital consumer privacy awareness. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*, pages 1–18, 2022.

[95] Shengqu Xi, Shao Yang, Xusheng Xiao, Yuan Yao, Yayuan Xiong, Fengyuan Xu, Haoyu Wang, et al. Deepintent: Deep icon-behavior learning for detecting intention-behavior discrepancy in mobile apps. In *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*.

[96] Xusheng Xiao, Xiaoyin Wang, Zhihao Cao, Hanlin Wang, and Peng Gao. Iconintent: automatic identification of sensitive ui widgets based on icon classification for android apps. In *2019 IEEE/ACM 41st International Conference on Software Engineering (ICSE)*, pages 257–268. IEEE, 2019.

[97] Yue Xiao, Zhengyi Li, Yue Qin, Xiaolong Bai, Jiale Guan, Xiaojing Liao, and Luyi Xing. Lalaine: Measuring and characterizing non-compliance of apple privacy labels.

[98] Fuman Xie, Yanjun Zhang, Chuan Yan, Suwan Li, Lei Bu, Kai Chen, Zi Huang, and Guangdong Bai. Scrutinizing privacy policy compliance of virtual personal assistant apps. In *37th IEEE/ACM International Conference on Automated Software Engineering*.

[99] Mulong Xie, Sidong Feng, Zhenchang Xing, Jieshan Chen, and Chunyang Chen. Uied: a hybrid tool for gui element detection. In *Proceedings of the 28th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering*, pages 1655–1659, 2020.

[100] Le Yu, Tao Zhang, Xiapu Luo, and Lei Xue. Autoppg: Towards automatic generation of privacy policy for android applications. In *Proceedings of the 5th Annual ACM CCS Workshop on Security and Privacy in Smartphones and Mobile Devices*, SPSM '15.

[101] Le Yu, Xiapu Luo, Chenxiong Qian, Shuai Wang, and Hareton K. N. Leung. Enhancing the description-to-behavior fidelity in android apps with privacy policy. *IEEE Transactions on Software Engineering*, 2018.

[102] Dawen Zhang, Pamela Finckenberg-Broman, Thong Hoang, Shidong Pan, Zhenchang Xing, Mark Staples, and Xiwei Xu. Right to be forgotten in the era of large language models: Implications, challenges, and solutions. *arXiv preprint arXiv:2307.03941*, 2023.

[103] Dawen Zhang, Shidong Pan, Thong Hoang, Zhenchang Xing, Mark Staples, Xiwei Xu, Lina Yao, Qinghua Lu, and Liming Zhu. To be forgotten or to be fair: Unveiling fairness implications of machine unlearning methods. *AI and Ethics*, pages 1–11, 2024.

[104] Xueling Zhang, Xiaoyin Wang, Rocky Slavin, Travis Breaux, and Jianwei Niu. How does misconfiguration of analytic services compromise mobile privacy? In *Proceedings of the ACM/IEEE 42nd International Conference on Software Engineering*.

[105] Sebastian Zimmeck, Peter Story, Daniel Smullen, Abhilasha Ravichander, Ziqi Wang, Joel Reidenberg, N Cameron Russell, and Norman Sadeh. Maps: Scaling privacy compliance analysis to a million apps. *Proceedings on Privacy Enhancing Technologies*, 2019(3): 66–86, 2019.

[106] Sebastian Zimmeck, Rafael Goldstein, and David Baraka. Privacyflash pro: Automating privacy policy generation for mobile apps. In *NDSS*, 2021.