

Decision Support for Mobile App Selection via Automated Privacy Assessment

Jens Wettlaufer¹ and Hervais Simo²

¹ Universität Hamburg, Vogt-Kölln-Straße 30, 22527 Hamburg, Germany
`jens.wettlaufer@uni-hamburg.de`

² Fraunhofer Institute for Secure Information Technology, Rheinstraße 75, 64295
Darmstadt, Germany
`hervais.simo@sit.fraunhofer.de`

Abstract. Mobile apps have entered many areas of our everyday life through smartphones, smart TVs, smart cars, and smart homes. They facilitate daily routines and provide entertainment, while requiring access to sensitive data such as private end user data, e.g., contacts or photo gallery, and various persistent device identifiers, e.g., IMEI. Unfortunately, most mobile users neither pay attention nor fully understand privacy indicating factors that could expose malicious apps. We introduce **APPA** (*Automated aPp Privacy Assessment*), a technical tool to assist mobile users making privacy-enhanced app installation decisions. Given a set of empirically validated and publicly available factors which app users typically consider at install-time, APPA creates an output in form of a personalized privacy score. The score indicates the level of privacy safety of the given app integrating three different privacy perspectives. First, an analysis of app permissions determines the degree of privateness preservation after an installation. Second, user reviews are assessed to inform about the privacy-to-functionality trade-off by comparing the sentiment of privacy and functionality related reviews. Third, app privacy policies are analyzed with respect to their legal compliance with the European General Data Protection Regulation (GDPR). While the permissions based score introduces capabilities to filter over-privileged apps, privacy and functionality related reviews are classified with an average accuracy of 79%. As proof of concept, the APPA framework demonstrates the feasibility of user-centric tools to enhance transparency and informed consent as early as during the app selection phase.

Keywords: Privacy assessment · Mobile apps · Permissions · Privacy policy · User reviews · Privacy perception · Decision support.

1 Introduction

While in 2013, only seven percent of companies had provided a corresponding app to their services, in 2017, 67 percent of small businesses offered mobile apps [40]. This trend, often called *appification*, even goes beyond smartphones. In 2015, Microsoft introduced a unified app marketplace for all Microsoft products.

Simultaneously, manufacturers released smart TVs on the basis of apps. Ap-pified systems are also deployed in the context of smart homes and connected vehicles. As such, apps play a major role in our daily routines. They run on devices that surround us all day long, even at night. From the moment they are installed they collect and process personal data, sometimes with little need for human intervention. For example, messenger apps have access to contact details, map apps track our location, and alarm clock apps know at least when users get up. We are motivated by the vision of empowering users with technology based decision support for mobile app selection. This work is a first step in this direction. Specifically, we focus on an automated and user-centric privacy assessment approach that aims at generating privacy recommendations for app users based on machine interpretation of various app attributes that are publicly available on app distribution markets like Google Play. These attributes are supposed to bring transparency and establish competition between apps. However, while users in the online context claim to care about their privacy, studies show that they mostly consider the more functionality informative parameters price, rating, number of installations, and user reviews during their installation decision [5, 22, 26, 9]. This phenomenon is known under the term "privacy paradox" [22, 43]. Existing explanations include the view that users often focus on simple parameters [6] because they are limited in time [17] or do not understand certain parameters [6, 17]. For example, privacy policies require legal knowledge to fully understand their implications [4, 30] and users cannot grasp the impact and consequences of granting certain permissions [7, 22, 45]. Moreover, observations show that obvious privacy related parameters are placed at disadvantageous positions, e.g., at the bottom of the app page. This is confirmed by research based on user studies revealing that the Google Play permission system is ineffective [36, 45, 7] mainly because requested permissions need to be granted after the installation decision [22, 14], which leads to a desensitization [18].

Efforts to improve this situation by introducing run-time permissions with Android 6.0 have not shown to be effective. For instance, security researchers found that more than 1,000 apps could access permissions that users had explicitly denied before [33, 37]. This results in users that can neither judge nor identify privacy-invasive, e.g., over-privileged, apps prior to the installation [19]. Moreover, although studies show that some apps do not comply with given privacy regulations [46, 4] and all apps actually need to provide a privacy policy due to the processing of any kind of personal data [10, 39], Google Play's privacy policy field is still optional. While the majority of users agree with the terms of services and privacy policies without reading them [3], studies showed that user perception of privacy differs [12, 31] and can be categorized into pre-defined privacy profiles [29]. These can help to present users more personalized privacy information.

With the aforementioned vision in mind, in this work, we introduce APPA, a mobile app vetting framework that aims at empowering users towards assessing the level of privacy intrusiveness of apps prior to its installation. APPA takes as input an app's set of attributes from the app market and various natural

language processing (NLP) techniques and quantitative models, and computes an overall privacy score for the given app. More specifically, the proposed framework considers three empirically validated app attributes which we hereafter refer to as installation factors: permissions, user reviews and privacy policy. However, note that while this work primarily focuses on these three factors, it is equally applicable to other installation factors deemed relevant to app users (cf. [5, 22, 9, 26, 25]).

The rest of this work is structured as follows. Section 2 introduces related work. Section 3 presents an overview of the APPA framework and defines key requirements for the related tool. While Section 4 discusses key components of our framework in detail, Section 5 presents the evaluation of APPA comparing it to existing work. Finally, Section 6 concludes this work and points to future directions.

2 Related Work

The inspection of apps in regard to security is well researched while app privacy analyses first received increased attention in recent years. For example, Kesswani et al. [23] analyze Android app privacy based on requested permissions. They divide permissions into generic and privacy-invasive permissions and classify the app’s privacy level respectively. Qu et al. [35] assume descriptions to consistently explain requested permissions and calculate the privacy risk based on the accordance between permissions and descriptions. To identify security and privacy related reviews, Nguyen et al. [34] utilized NLP in combination with machine learning (ML) techniques. They showed that such reviews can have an impact on app related privacy improvements by analyzing and correlating 4.5M historical reviews and app updates over time. Privacy policies are for example examined from Harkous et al. [17] applying self-trained word embeddings and convolutional neural networks in order to generate privacy grading icons. They trained their model on the manually annotated online privacy policy corpus OPP-115 [42]. These works introduce extraction mechanisms from different privacy indicators, but lack of a combined privacy understanding from different perspectives.

Further works intend to compare the described functionality with actual behavior. Therefore, Zimmeck et al. [46] contrast privacy policy statements with the results of a static code analysis. Furthermore, Yu et al. extend the privacy policy findings with description-to-permission fidelity and verify these with a bytecode analysis in order to examine the gap between said and done security and privacy practices. Both approaches take more perspectives into account to quantify the privacy violations of apps and thus the trustworthiness. However, they depend on source or byte code that need to be downloaded and analyzed, which takes more time contrary to a metadata analysis. Particularly, the need for source code to apply static code analysis is not possible in all cases, for example when priced apps are supposed to be investigated prior to the installation. In addition, they both analyze privacy practices on the basis of US regulations, e.g., the Californian Online Privacy Protection Act (CalOPPA), instead of the

European GDPR. A GDPR based and multi-source approach is presented by Hatamian et al. [20] suggesting to analyze user reviews, privacy policies, stated permissions and permissions usage. The risk level of ten apps is investigated based on an NLP and ML privacy threat model for user reviews, permission statistics, manual privacy policy analysis as well as a dynamic permission usage analysis over seven days. Thus, their approach lack of real-time app interaction to support users during the decision-making. Additionally, their privacy impact model only provides a subjective risk-perception without considering individual preferences. In contrast, Habib et al. [15] assess trustworthiness by incorporating user sensitivity in privacy issues into their trust score that also comprises the average rating, a sentiment analysis of user reviews, and additional static and dynamic code analysis. Familiarity as well as the posture to desensitization and advertisement frameworks are factors for the personalization. The information is taken by other apps installed on the user device, whether the app is over-privileged or uses an excessive amount of advertisement frameworks. While the first invades user privacy to a certain extent, the other information can only be retrieved using code analysis which introduces a lack of actual on-device real-time assessments. Different personalization techniques in form of privacy profiles are introduced by Liu et al. suggesting the trade-off between functionality and privacy preferences retrieved from app permissions as one measure [28] and learned profiles from a data set of permission settings retrieved by real users with rooted devices [27]. This work retrieves a privacy-to-functionality trade-off from user reviews, but incorporates a similar permission handling as the former. However, the latter is limited to the fact that their training data set was built upon users with rooted devices who are assumed to be more tech-savvy than general users, which affects the quality of the privacy profiles.

3 Approach Overview & Design Goals

3.1 Overview

Properly assessing the level of privacy safety of mobile apps at install-time is laborious and often requires considerable expertise. In many cases, users are not willing to spend substantial amounts of time required for vetting the app prior to its installation, i.e., determining if the later conforms to their personal privacy expectations and preferences. Moreover, due to technical design limitations in today’s app ecosystems, mobile users are not always able to make informed privacy decisions on whether a mobile app should be installed on their device [22], nor do they always fully understand implications of particular apps for their privacy [5]. This often results in curious or even malicious apps being installed on users’ mobile devices and their overall privacy undermined. Clearly, there is a need for a new privacy-enhancing approach for decision support in the context of app installation. What makes the situation even more challenging is that enhanced transparency and control for app installation decision support should consider the diversity of app users’ privacy preferences and the context-dependency of their decisions [24]. Therefore, we propose **APPA**, which is to the best of our

knowledge the first decision support tool for personalized app selection through a multi-factor privacy assessment. Our user-centered privacy assessment focuses on a set of empirically validated factors which app users consider before installation [5, 22, 9, 26, 25]. More precisely, the proposed solution considers the following three factors: app permissions, user reviews and the app’s privacy policy. For each factor, a score is computed. All three scores are subsequently combined into an overall privacy score which, along additional recommendations, is displayed to the app user. For the overall privacy score, optimal combination weights are computed based on empirical insights from [5, 22, 9, 26]. We therefore claim that the weights for the installation factors related scores allow the APPA framework to cover the diversity of app users’ privacy preferences. By providing users with the option to explicitly specify weight’s values within a pre-defined range, the APPA framework allows minimal user feedback while ensuring that algorithmic generated users’ privacy decisions remain context-dependent.

3.2 Requirements

Designing a suitable technology that allows mobile app users to assess the level of privacy safety of any given app and hence answer the question to which extent the app is trustworthy, presents a number of difficult challenges. We argue that APPA regarded as transparency enhancing solution should at least satisfy the following requirements:

R.0 Functionality. The envisaged system shall be able to capture relevant app’s metadata from the app market. APPA should be able to autonomously assess the app level of privacy safety, i.e., the app privacy score, in order to link it with privacy recommendations.

R.1 Data Minimization and Privacy-by-Design. The overall APPA framework has to be designed and implemented according to the Privacy-by-Design [16] principle of data minimization. The framework should only store users’ digital footprint that is absolutely necessary for analysis and visualization purposes. The framework should not leak any sensitive data to any third party. Access to sensitive data by our framework should require explicit user consent. The confidentiality of the metadata and inferred knowledge, e.g., insights from the associated analysis results, has to be ensured.

R.2 Usability. The APPA framework should not degrade the mobile user’s experience. APPA should be implemented as a mobile Android app that does not require root permissions. Users should be able to specify and manage their own privacy and security policies, i.e., rules governing the handling of metadata and functionalities of the framework in a fine-grained manner. The configuration and administration of the app should not require the user to hold specific knowledge about security and privacy. All framework related processes should be mostly automated and require minimal user intervention. Especially the installation process and the specification of privacy preferences and controls should be as simple and unobtrusive as possible. Moreover, the user should have the possibility to realize that she is leaving digital footprints behind while using smartphones as well as the privacy implications of these disclosures. Users should be provided

with details about which particular information the APPA framework accesses and which part of these information is stored or further processed. Additionally, the user should be able to modify and delete already collected metadata. To satisfy these usability objectives, three additional challenges have to be met: personalization, minimal overhead and comprehensible visualization. *R.2.1 Personalization.* Given the fact that users perceive privacy differently, personalized recommendations and notices are required. *R.2.2 Minimal Overhead.* The APPA framework should operate with minimal overhead and as efficient as possible. Especially the communication overhead, the battery consumption overhead and the use of computational resources like memory consumption and CPU processing time should be minimal. Unreasonable overheads should be prevented, since they may eventually lead to a decreased user experience and users entirely removing the APPA-app from their devices. *R.2.3 Comprehensible Visualization.* In order for the user to better understand its device’s network interactions, APPA should enable a comprehensible visualization of both raw metadata and analysis results. All visualizations should be intuitive to the user, meaning that the user should immediately at first glance be able to grasp important aspects about the data that is presented.

R.3 Extensibility. The APPA framework should be extensible. Components of APPA should be designed and implemented in a way that enables other developers to build upon the framework for future work. For instance, future developers and researchers should be able to leverage our framework to build new, platform-agnostic privacy-enhancing prototypes to be deployed in large scale user behavioral studies.

4 System Design

The APPA framework consists of five main components, as depicted in Figure 1: App Isolation Module, Metadata Downloader, Parameters Inspection Modules, Personalized Privacy Assessment, Visualization Engine and Graphical User Interface (GUI).

4.1 App Isolation Module

Leveraging the public Virtual Private Network (VPN) API³ provided by the Android OS (since version 4.0+), we design this component as a firewall that prevent any app under consideration from accessing critical on-device resources or interacting with any remote entities up until the vetting by APPA is completed. Upon analysis by our tool, the user is presented with an overall score and a set of recommendations. Based on this information, the user can either revoke the firewall’s pre-defined isolation rules and hence allow the installation of the vetted app to be finalized, or reject the app altogether.

³ <https://developer.android.com/reference/android/net/VpnService.html>

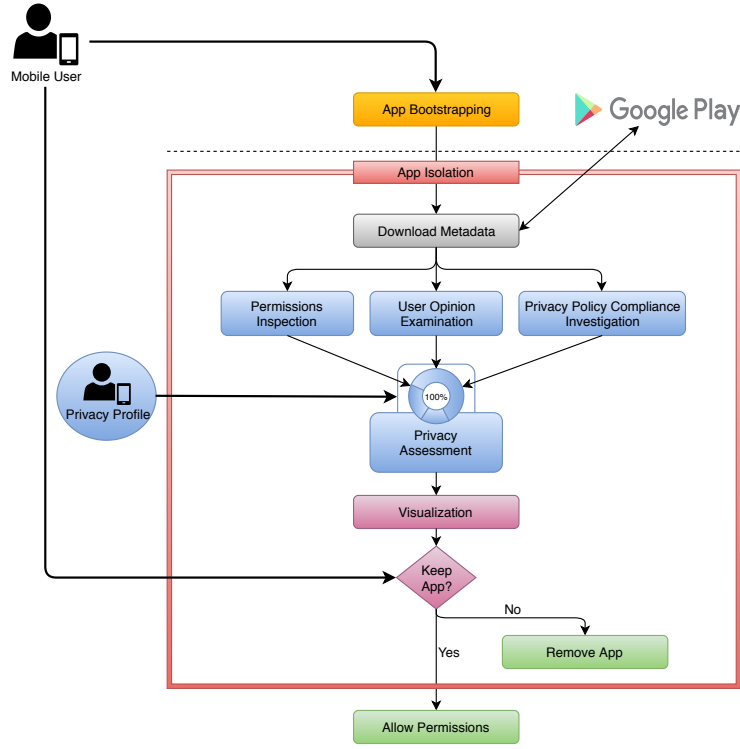


Fig. 1. Privacy-preserving procedure to support users during their app installation decision using an automated and personalized privacy assessment.

4.2 Metadata Downloader

This component handles all tasks related to the interception and aggregation of the app’s metadata. In order to successfully complete this task, we designed the Metadata Downloader to be able to query the app market and all related third party domains. As such, it includes three specific sub-components, each focusing on one of the app metadata being considered in this work: A *Permission Collector*, a *User Review Crawler* and a *Privacy Policy Crawler*. Using the unique identifier of the app to be vetted, the Permission Collector fetches all intended permissions as declared by the app developer in the app manifest; the User Review Crawler crawls the top 200 user reviews from the app web page; and the Privacy Policy Crawler searches the app web page for the URL to the privacy policy, follows all subsequent links and extract the policy text. Given the diversity of formats in which privacy policies are displayed, our Privacy Policy Crawler leverages tools such as *textract* [8] and *jusText* [2] to extract text from images to pdf files. The set of metadata capture by all three sub-components of the Downloader is stored on device and made available to other components of APPA for further processing.

4.3 Parameters Inspection Modules

Currently, the APPA framework independently assesses three privacy-indicating app parameters including permissions, user reviews, and the privacy policy in order to inform users about their privateness preservation, i.e., the amount of private mobile information users preserve after the installation, the privacy-to-functionality trade-off, and the app’s GDPR compliance, respectively.

Permissions: Privateness Preservation Modern operating systems such as Android builds upon permission-based security mechanisms to restrict access to sensitive users’ data and critical device resources. In Android, app developers must declare all the permissions required by the app to access resources or data outside of the app’s own sandbox. Permissions are listed in a so called Manifest file. For a specific subset of the declared access permissions, the normal permissions, the Android OS automatically grants the app access to the related resources or data at install-time. The remaining permissions, the dangerous permissions, are prompted to the user at run-time, requesting her to approve or reject access to sensitive information or resources. However, research has shown that a significant portion of apps overuse access permissions [13] while most people do not fully pay attention nor comprehend permission requests [11]. As component of the APPA framework, the Permission Inspection provides means to quantify possible risks associated with specific permissions. Existing permissions analysis approaches can be compared by three consecutive aspects. First, the set of permissions is determined for the calculation, e.g., all permissions or only a critical subset. Second, the scoring algorithm is defined, e.g., absolute percentage or an ML approach. Finally, each score has a certain interpretation. For example, Kesswani et al. [23] utilize a custom set of privacy invasive permissions, while Hatamian et al. [20] rely on Google’s pre-defined permissions with the protection level *dangerous*. These so called *dangerous* permissions have access to personal data such as camera or contacts, according to Google. Finally, they measure the privacy invasiveness and privacy gap, respectively, on the basis of a percentile of app-specific requested permissions.

Our approach extends the use of *dangerous* permissions, based on the fact that Google organizes them in groups and as soon as one permission of a *dangerous* group is granted, all permissions of this group are granted. Therefore, our *advanced dangerous* method counts the number of app specific requested permissions by also taking implicitly granted permissions into account due to the *dangerous* group coherence. Furthermore, we add further permissions to the set that can be hacked as identified by security researchers [44], i.e., `READ_EXTERNAL_STORAGE`. Our score is calculated relatively to the average of *dangerous* permissions. A representative average was calculated by leveraging the permissions of the 40,332 top selling apps of Google Play across 55 categories from March 2019. There are 29 *dangerous* permissions in total. The averages result in $\mathcal{O}_{dangerous} = 5$ and $\mathcal{O}_{adv.dangerous} = 7$, which is used as threshold Q to compute the relative permission score R as follows:

$$R_{\in[0,1]} = \begin{cases} 1 - \frac{\#Permissions}{2*Q} & \text{if } \#Permissions \leq 2 * Q \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

Essentially, the permission score is worst in case of at least twice the threshold requested *advanced dangerous* permissions. Furthermore, the score is framed positively, meaning that the score can be interpreted as privateness preservation. Hence, the higher the score the better. The effect of this relative approach (bold line) in contrast to related work (dotted line) can be seen in Figure 2, which shows the permission scores of 60 randomly chosen apps from a data set of benign and malicious apps [32] separated by their type.

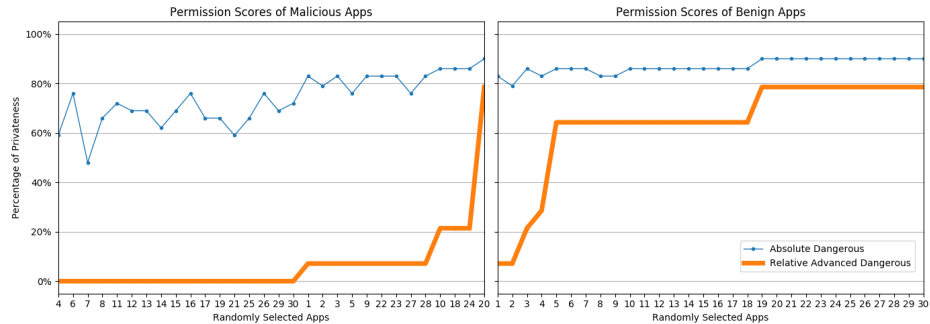


Fig. 2. Comparison of the permission score of related work (dotted) and this work (bold) in regard to malicious and benign apps randomly selected from [32]. The classification of malicious apps can easily be determined with our novel permission score.

User Reviews: Privacy-to-Functionality Trade-off User reviews are a means for app users to share their experience and opinion regarding apps in natural language with the community. Previous work shows that reviews contain functionality and privacy related content [34], and can therefore be exploited to understand the community opinion. Related work concentrates on identifying security and privacy related comments [34, 20], and consecutively, extracting privacy threat information [20]. These approaches have the disadvantage that users often communicate privacy concerns with emotions instead of technical descriptions. Additionally, user reviews will often contain only such privacy violations that are obvious to app users.

In contrast, our approach bases on the findings of Wottrich et al. [43] that users are confronted with a cost-benefit trade-off, which we interpret as privacy-to-functionality trade-off. Our analysis outweighs community feelings of privacy costs against functionality benefits. For this, we first divide privacy (P) from non-privacy ($nonP$) related reviews using a neural net classifier consisting of two dense layers with 128 and 64 neurons with ReLU and a dropout rate of

0.75. The feature set includes character n-grams to circumvent typos. It was trained on a rather small training set, i.e., 200 privacy and 230 non-privacy reviews, 10-fold cross-validated and scores a mean accuracy of 79%. Second, we use sentiment analysis on each review utilizing SentiStrength, which provides human-level accuracy on short informal texts [41]. This results a sentiment score per review. Third, review related Helpfuls, similar to Likes, are leveraged to calculate a weighted average over all sentiment scores for privacy and non-privacy reviews, respectively. Finally, Equation 2 computes a trade-off that emphasizes the privacy opinion over functionality.

$$T(P, nonP)_{\in[-1,1]} = \begin{cases} P - nonP & \text{if } P < nonP \\ P & \text{if } P \geq nonP \end{cases} \quad (2)$$

$$S_{\in[0,1]} = \frac{T(P, nonP) + 1}{2} \quad (3)$$

If the privacy sentiment trumps the functionality community opinion, the privacy sentiment value is adopted. However, when functionality exceeds privacy, the distance between both values is negated. To be in line with the other scores, the privacy-to-functionality trade-off is normalized as depicted in Equation 3. Figure 3 exemplary illustrates the behavior of this score. The graph is sorted by the green line representing the privacy-to-functionality trade-off score. From Pokémon GO to Santander Mobile Banking, the privacy sentiment is lower than the functionality opinion. Therefore, the resulting score is always below 50% because the initially scaled values are below zero. As soon as privacy exceeds functionality, the score makes use of the privacy value and is always greater than 50% due to normalization.

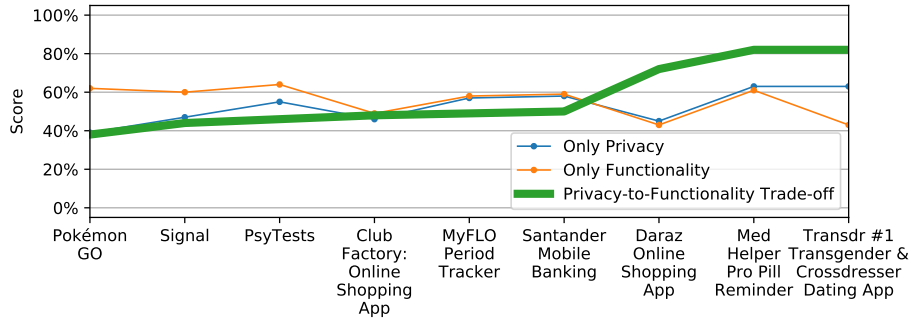


Fig. 3. Comparison of the privacy-to-functionality trade-off score regarding exemplary Android apps. Note that the score differs from the actual privacy values because it is normalized from $[-1, 1]$ to $[0, 1]$.

Privacy Policy: GDPR Compliance Privacy policies are app corresponding legal documents that are supposed to fully disclose any collection and processing of personal user data. Data protection regulations, such as the CalOPPA in the US and the GDPR in Europe, oblige app developers to have a privacy policy as soon as they process any user data. This work focuses on the European GDPR requiring in Article 5.1 (a) GDPR [10] to fulfill the main principles lawfulness, fairness and transparency in relation to the data subject. As of today, we are only able to assess the transparency of privacy policies. We have already built a fully functional learning based model to cover lawfulness and fairness, but an extensive training set in regard to the GDPR is still work in progress. We measure the readability of policy texts in order to investigate the "concise, transparent, intelligible and easily accessible form, using clear and plain language" as demanded in Article 12.1 GDPR [10]. Consistent to previous work [38], we average the Gunning Fog, Flesch-Kincaid, and SMOG readability metrics to compute a score between 6 and 17 directly mapping to education levels. While 6 corresponds to the sixth grade and 17 to a college graduate, our score defines GDPR compliant readability between 8 and 13. The minimum is set to the age of children where they do not necessarily need a parental guardian anymore, while the maximum corresponds to a college freshman which is still acceptable. Exemplary results of the transparency measurements transformed into GDPR compliance scores can be viewed in Table 1. Privacy policies with scores close to 100% could easily be understood by children in the eighth grade, while scores around 0% indicate that their policies can only be grasped after at least 14 years of education.

Snapchat	86%
Signal	83%
Telegram	70%
Pinterest	54%
WhatsApp, Facebook, Facebook Messenger, Instagram, Google Search, YouTube, Tinder, Twitter, Tumblr, reddit, Pokémon Go, Candy Crush Saga, Spotify, Jodel, Threema	0%

Table 1. GDPR compliance scores of 20 popular apps measuring the readability of privacy policies in the above mentioned GDPR compliant range.

4.4 Personalized Privacy Assessment

This module of APPA aggregates all independent results of the parameter assessments. As shown in Figure 4 and 5, our visualization repertoire introduces a detailed triangle scheme that represents the scores independently as well as a combined privacy score. While the former on the one hand informs about each score but also reveals the overall privacy impact, the latter enables a personalized privacy score by adopting the weights in accordance to user preferences. It

expresses positively framed privacy safety visualized in five circles with a white plus as studies resulted in it as the most intuitive pattern [6, 7].

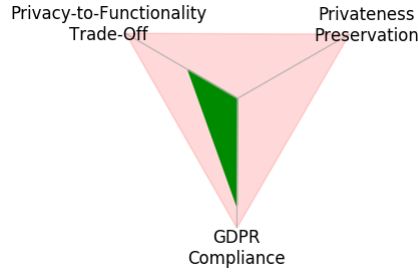


Fig. 4. Visualization of independent results by introducing the overall privacy impact.



Fig. 5. Intuitive privacy safety score that allows for personalization. This score shows the unpersonalized default option in regard to Figure 4, i.e., an unweighted average.

4.5 Visualization Engine (VE) and Graphical User Interface (GUI)

The VE is a generic component leveraging free and open source data visualization libraries. Relying on this engine, we implemented a modularized front-end interface, the APPA’s GUI. The latter includes a plethora of options for menus, settings and views on details over various app details and the assessment metrics and results. More specifically, the GUI enables visualization of the app’s overall privacy score and provides the end user with options to interact with other components of the APPA framework, including activating or deactivating the Isolation Module.

5 Evaluation & Discussion

While the parameter specific assessments were already evaluated in the previous section, the overall outcome of the APPA framework is therefore compared to existing privacy scoring systems as shown in Table 2. *AppCensus*⁴ [1] focuses on informing users about which sensitive permissions are accessed and what personal data is shared using a dynamic run-time analysis. *PrivacyGrade*⁵ [21] intends to grade the gap between user expectation and the actual app behavior.

Although *AppCensus* states actually used permissions and shared data through a dynamic analysis, users need to have a certain expertise to identify the privacy risks or functionality benefits. The dynamic analysis approaches come with the drawback of high false negative rates, because it might miss certain functionality that uses further permissions or data. In contrast, APPA computes the privacy

⁴ <https://www.appcensus.mobi/>

⁵ <http://privacygrade.org/>

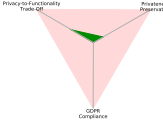
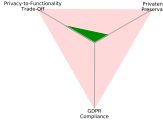
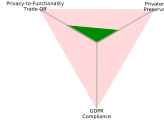
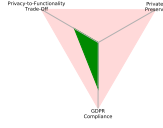
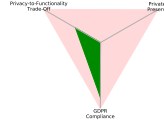
Pokémon GO	Jodel	Tinder	Telegram	Signal
<i>AppCensus</i> (used sensitive permissions / shared sensitive data)				
1 / 2	0 / 2	1 / 2	3 / 0	1 / 0
<i>PrivacyGrade</i> (#requested sensitive permissions)				
A 9	A 5	A 7	A 20	A 30
Our Score (option: average) *				
0.9	1.2	1.4	1.9	2.1
++++	++++	++++	++++	++++
				

Table 2. Exemplary comparison of privacy related scoring schemes in regard to five popular apps. *Triangle scheme labels are identical to Figure 4.

score by intentionally assuming the most privacy-invasive app state. While *PrivacyGrade*'s grading system might mislead users into thinking that the number of permissions does not affect their privacy, our score precisely indicates that no privateness is preserved when apps use more than 14 sensitive permissions. Similar to *AppCensus*, our score enables users to interpret the independent results based on their personal privacy perception. This can even be automatized in the final privacy safety score.

6 Conclusion & Future Work

The APPA framework achieves an automatic privacy assessment of apps prior to the installation by solely relying on publicly available parameters to lead users to more informed app installation decisions. Therefore, it leverages permissions as indication for the preservation of private information, extracts the user community opinion about the cost-benefit trade-off between privacy and functionality, and investigates the app's legal compliance to the GDPR. The result is visualized in a both detailed and combined privacy safety score that allows for an optional and thus privacy-preserving personalization by weighing the final score in accordance to the user perception. APPA's construction equally brings extensibility of privacy assessment methods as well as transferability to further app based platforms. For example, the GDPR compliance assessment still lacks of justification regarding the main columns fairness and lawfulness. For this, we will extend the assessment with a learning based algorithm that recognizes whether a privacy policy is complete and fair according to the GDPR. On the other hand, future directions of the APPA's privacy safety score include a validating user study and its use as an app wide quality label in regard to privacy due to the fact that only publicly available metadata is needed. In conclusion, the proposed

APPA framework enables users to better inform themselves about the privacy safety level of an app without the need to risk their own privacy.

Acknowledgment. This work has been supported in part by the German Federal Ministry of Education and Research (BMBF) within the project "Forum Privatheit und selbstbestimmtes Leben in der Digitalen Welt".

References

1. AppCensus: Appcensus app search. Online (2019), <https://search.appcensus.io/>, [Accessed on 2019-07-20]
2. Belica, M.: jusText 2.2.0 . Python Software Foundation, <https://pypi.org/project/jusText/>, [Accessed on 2019-04-21]
3. Board, T.E.: Opinion: How silicon valley puts the ‘con’ in consent. Online (Feb 2019), <https://www.nytimes.com/2019/02/02/opinion/internet-facebook-google-consent.html>, [Accessed on 2019-07-20]
4. Brandtzaeg, P.B., Pultier, A., Moen, G.M.: Losing control to data-hungry apps - a mixed-methods approach to mobile app privacy. *Social Science Computer Review* (May 2018)
5. Chin, E., Felt, A.P., Sekar, V., Wagner, D.: Measuring user confidence in smart-phone security and privacy. In: *Proceedings of the Eighth Symposium on Usable Privacy and Security* (2012)
6. Choe, E.K., Jung, J., Lee, B., Fisher, K.: Nudging people away from privacy-invasive mobile apps through visual framing. In: *Human-Computer Interaction – INTERACT 2013* (2013)
7. Chong, I., Ge, H., Li, N., Proctor, R.W.: Influence of privacy priming and security framing on android app selection. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting* (2017)
8. deanalmgren: textract. GitHub.com (2014), <https://textract.readthedocs.io/en/stable/>, [Accessed on 2019-02-23]
9. Dogruel, L., Joeckel, S., Bowman, N.D.: Choosing the right app: An exploratory perspective on heuristic decision processes for smartphone app selection. *Mobile Media & Communication* (2014)
10. European Parliament and Council of the European Union: Regulation (eu) 2016/679 (general data protection regulation). *Official Journal of the European Union* (May 2018), <https://eur-lex.europa.eu/eli/reg/2016/679/2016-05-04>, [Accessed on 2019-05-06]
11. Felt, A.P., Ha, E., Egelman, S., Haney, A., Chin, E., Wagner, D.: Android permissions: User attention, comprehension, and behavior. In: *SOUPS. ACM* (2012)
12. Fogg, B.J., Iizawa, D.: Online persuasion in facebook and mixi: A cross-cultural comparison. In: *Persuasive Technology* (2008)
13. Gorla, A., Tavecchia, I., Gross, F., Zeller, A.: CHABADA: Checking app behavior against app descriptions. In: *Proceedings of the 36th International Conference on Software Engineering - ICSE 2014. ACM Press* (2014)
14. Gu, J., Xu, Y.C., Xu, H., Zhang, C., Ling, H.: Privacy concerns for mobile app download: An elaboration likelihood model perspective. *Decision Support Systems* (2017)

15. Habib, S.M., Alexopoulos, N., Islam, M.M., Heider, J., Marsh, S., Muehlhaeuser, M.: Trust4app: Automating trustworthiness assessment of mobile applications. In: 2018 17th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/ 12th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE). pp. 124–135 (Aug 2018)
16. Hansen, M.: Data protection by design and by default à la european general data protection regulation. In: IFIP International Summer School on Privacy and Identity Management. pp. 27–38. Springer (2016)
17. Harkous, H., Fawaz, K., Leuret, R., Schaub, F., Shin, K.G., Aberer, K.: Polisis: Automated analysis and presentation of privacy policies using deep learning. CoRR (2018)
18. Harris, M., Brookshire, R., Patten, K., Regan, E.: Mobile application installation influences: Have mobile device users become desensitized to excessive permission requests? Americas Conference on Information Systems (2015)
19. Harris, M.A., Brookshire, R., Chin, A.G.: Identifying factors influencing consumers' intent to install mobile applications. International Journal of Information Management (2016)
20. Hatamian, M., Momen, N., Fritsch, L., Rannenber, K.: A multilateral privacy impact analysis method for android apps. In: Naldi, M., Italiano, G.F., Rannenber, K., Medina, M., Bourka, A. (eds.) Privacy Technologies and Policy. pp. 87–106. Springer International Publishing, Cham (2019)
21. Hong, J.: Privacygrade: Grading the privacy of smartphone apps. Online (2014), <http://privacygrade.org/home>, [Accessed on 2019-07-20]
22. Kelley, P.G., Cranor, L.F., Sadeh, N.: Privacy as part of the app decision-making process. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (2013)
23. Kesswani, N., Lyu, H., Zhang, Z.: Analyzing android app privacy with gp-pp model. IEEE Access (2018)
24. Knijnenburg, B.: A User-Tailored Approach to Privacy Decision Support. Master's thesis, University of California, Irvine (07/2015 2015), <http://www.ics.uci.edu/~kobsa/phds/knijnenburg.pdf>
25. Kulyk, O., Gerber, P., Marky, K., Beckmann, C., Volkamer, M.: Does this app respect my privacy? design and evaluation of information materials supporting privacy-related decisions of smartphone users. In: NDSS Symposium 2018 (USEC), San Diego, CA, February 18-21, 2019 (2019)
26. Lim, S.L., Bentley, P.J., Kanakam, N., Ishikawa, F., Honiden, S.: Investigating country differences in mobile app user behavior and challenges for software engineering. IEEE Transactions on Software Engineering (2015), data: http://www0.cs.ucl.ac.uk/staff/S.Lim/app_user_survey/
27. Liu, B., Andersen, M.S., Schaub, F., Almuhiemedi, H., Zhang, S.A., Sadeh, N., Agarwal, Y., Acquisti, A.: Follow my recommendations: A personalized privacy assistant for mobile app permissions. In: 12th Symposium on Usable Privacy and Security 2016. USENIX Association, Denver, CO (2016)
28. Liu, B., Kong, D., Cen, L., Gong, N.Z., Jin, H., Xiong, H.: Personalized mobile app recommendation: Reconciling app functionality and user privacy preference. In: Proceedings of the Eighth ACM International Conference on Web Search and Data Mining. WSDM '15, ACM, New York, NY, USA (2015)
29. Liu, B., Lin, J., Sadeh, N.: Reconciling mobile app privacy and usability on smartphones: Could user privacy profiles help? In: Proceedings of the 23rd International Conference on World Wide Web (2014)

30. Meineck, S.: Komplizierter als Kant: Nerd erstellt Ranking der furchtbarsten AGB. Online (2019), <https://www.vice.com/de/article/5974vb/datenschutz-ranking-der-schlimmsten-agb-facebook-airbnb-google-dsgvo>, [Accessed on 2019-07-28]
31. Mylonas, A., Theoharidou, M., Gritzalis, D.: Assessing privacy risks in android: A user-centric approach. In: Risk Assessment and Risk-Driven Testing (2014)
32. Navarro, C.U.A.: Dataset malware/benign permissions android (2016). <https://doi.org/10.21227/H26P4M>
33. Ng, A.: More than 1,000 android apps harvest data even after you deny permissions. Online (2019), <https://www.cnet.com/news/more-than-1000-android-apps-harvest-your-data-even-after-you-deny-permissions/>, [Accessed on 2019-07-20]
34. Nguyen, D.C., Derr, E., Backes, M., Bugiel, S.: Short text, large effect: Measuring the impact of user reviews on android app security & privacy. In: Proceedings of the IEEE Symposium on Security & Privacy, May 2019. IEEE (May 2019)
35. Qu, Z., Rastogi, V., Zhang, X., Chen, Y., Zhu, T., Chen, Z.: AutoCog: Measuring the Description-to-permission Fidelity in Android Applications. In: Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security - CCS '14. ACM Press (2014). <https://doi.org/10.1145/2660267.2660287>
36. Rajivan, P., Camp, J.: Influence of privacy attitude and privacy cue framing on android app choices. In: 12th Symposium on Usable Privacy and Security (2016)
37. Reardon, J., Álvaro Feal, Wijesekera, P., On, A.E.B., Vallina-Rodriguez, N., Egelman, S.: 50 ways to leak your data: An exploration of apps' circumvention of the android permissions system. In: 28th USENIX Security Symposium (2019)
38. Robillard, J.M., Feng, T.L., Sporn, A.B., Lai, J.A., Lo, C., Ta, M., Nadler, R.: Availability, readability, and content of privacy policies and terms of agreements of mental health apps. Internet Interventions (2019)
39. State of California Department of Justice: Privacy laws. State of California Department of Justice (2003), <https://oag.ca.gov/privacy/privacy-laws>, [Accessed on 2019-05-06]
40. The Realtime Report: How appification is transforming the internet. Online (2017), <https://therealtime.com/2017/11/01/appification-transforming-internet/>, [Accessed on 2019-07-26]
41. Thelwall, M., Buckley, K., Paltoglou, G., Cai, D., Kappas, A.: Sentiment strength detection in short informal text. *J. Am. Soc. Inf. Sci. Technol.* (2010)
42. Wilson, S., Schaub, F., Dara, A.A., Liu, F., Cherivirala, S., Leon, P.G., Andersen, M.S., Zimmeck, S., Sathyendra, K.M., Russell, N.C., Norton, T.B., Hovy, E.H., Reidenberg, J.R., Sadeh, N.M.: The creation and analysis of a website privacy policy corpus. In: ACL (2016)
43. Wottrich, V.M., van Reijmersdal, E.A., Smit, E.G.: The privacy trade-off for mobile app downloads: The roles of app value, intrusiveness, and privacy concerns. *Decision Support Systems* (2017)
44. Yin, S.: What can a zero-permissions android app do? Online (Apr 2012), <http://securitywatch.pcmag.com/none/296635-what-can-a-zero-permissions-android-app-do>, [Accessed on 2019-06-16]
45. Zhang, B., Xu, H.: Privacy nudges for mobile applications: Effects on the creepiness emotion and privacy attitudes. In: Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work & Social Computing - CSCW '16 (2016)
46. Zimmeck, S., Wang, Z., Zou, L., Iyengar, R., Liu, B., Schaub, F., Wilson, S., Sadeh, N., Bellovin, S., Reidenberg, J.: Automated analysis of privacy requirements for mobile apps. In: The 2016 AAAI Fall Symposium Series: Privacy and Language Technologies (2016)