Student Research Abstract: "Hard to Understand, Easy to Ignore": An Automated Approach to Predict Mobile App Permission Requests

Majid Hatamian*
Chair of Mobile Business & Multilateral Security
Goethe University Frankfurt
Frankfurt am Main, Germany
hatamianm@acm.org

ABSTRACT

In this paper, we propose a novel automated approach to predict the potential privacy sensitive permission requests by mobile apps. Based on machine learning (ML) and natural language processing (NLP) techniques, personal data access and collection practices mentioned in app privacy policy text are analyzed to predict the required permission requests. Further, the predicted list of permission requests is compared with the real permission requests to check whether there is any mismatch. We further propose user interface designs to map mobile app permission requests to understandable language definitions for the end user. The combination of these concepts provides users with special knowledge about data protection practice and behavior of apps based on the analysis of privacy policy text and permission declaration which are otherwise difficult to analyze. Initial results demonstrate the capability of our approach in prediction of app permission requests. Also, by exploiting our already proposed app behavior analyzer tool, we investigated the correlation between what mobile apps do in reality and what they promise in their privacy policy text resulting in a positive correlation.

KEYWORDS

mobile app, privacy policy, permission, risk, supervised learning.

1 PROBLEM AND MOTIVATION

After releasing Android 6.0, users are given control over permission requests, and they are able to restrict the requested permissions even at run-time [7]. Although such enhancement enables users to better preserve their privacy, prior studies showed that few users are aware of it, hence permissions are often ignored even though they might appear irrelevant to the real functionality of the app [8] because many users do not understand the technical and sometimes ambiguous definitions of permissions [5]. E.g. the basic permission `READ_PHONE_STATE` enables an invader to gain access to multiple sensitive resources such as phone number, cellular network information, ongoing calls, etc. But how an ordinary user is able to infer this information by only knowing the permission technical name? Additionally, users mostly value the use of the apps more than their personal data, despite the fact that the apps collect large amounts of personal data, for various purposes ranging from functionality to empower their ads mechanisms [4].

Our Work: In this paper, we focus on app privacy policy texts as an important source containing information about the data types that the developers access and collect. Given this, we propose an automated approach to predict mobile apps permission requests by analyzing their privacy policy text. Our approach is mainly based on ML and NLP techniques. To the best of our knowledge, we are the first proposing an approach to analyze mobile app privacy policies to predict permission requests with a special focus aimed at easing the understanding of app permission requests for users. That is to say, we propose a new model of privacy indicators that can translate the technical (and ambiguous) definitions of permissions into understandable language definitions for the end user. By investigating the correlation between what apps are actually doing in reality and what they promise in their privacy policy texts, we reveal interesting contradictions concerning these two concepts (behavior analysis and privacy policy text analysis).

The rest of this paper is organized as follows. In Section 2 we review the existing relevant work. Section 3 introduces our proposed approach for the prediction of app permission requests considering its privacy policy. In Section 4 we show the experiments and initial results. Finally Section 5 concludes this paper.

2 RELATED WORK

Rosen et al. [12] studied the ineffectiveness of permissions helping users to understand app behavior. They developed a framework for profiling mobile apps to help user for making informed choices. Complementary to this, Felt et al. [4] performed a user study by surveying 3,115 smartphone users about 99 risks associated with 54 app permissions. They asked users to rate how upset they would be if given risks occurred and used this data to rank risks by levels of user concern. Pandita et al. [10] tried to close the gap between user expectations and app characteristics to specify why access to specific resources on the device is needed. They applied NLP techniques for evaluating app descriptions available on the app market and identified justifications for permission requirements despite the remaining risk of inaccurate app descriptions. This is an interesting work, however, justification of permission requests considering apps' description may not necessarily lead to a concert and fine-grained conclusion. By contrast, in our work we focus on app privacy policy text as a legislation requirement that claims the main data types collected from the users. Lie et al. [9] proposed an approach to learn privacy profiles for permission settings. They

*This is a post-peer-review, pre-copyedit version of an article published in the proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing (ACM SAC 2019). The final authenticated version is available online at: https://doi.org/10.1145/3297280.3297660. This research work has received funding from the H2020 Marie Sklodowska-Curie EU project "Privacy&Us" under the grant agreement No 675730.

interviewed 84 Android users who were receiving privacy nudges designed to motivate them to interact with their permission settings. Afterwards, they found similar users (in terms of privacy settings) and generated recommendation for their permission settings. Neither of these works consider the importance of permission request summarization for mobile users and positive consequences that it might have to reduce mobile app privacy risks. As writing a proper app privacy policy text is not an easy task that needs high level of technical and legal knowledge, the authors in [13] proposed a system to facilitate the generation of app privacy policy text for developers. They mapped permissions to private information to infer what should be written in the policy text. By contrast, we are interested to predict apps' permission requests by analyzing their already written policy texts.

3 PROPOSED APPROACH

In this section, we propose a new two-pillar approach over the prediction and summarization of mobile app permission requests. Fig. 1 shows an overview of our two-pillar approach. A more detailed architecture for *Pillar I* and *Pillar II* is illustrated by Fig. 2. The following subsections elaborate on the main components.

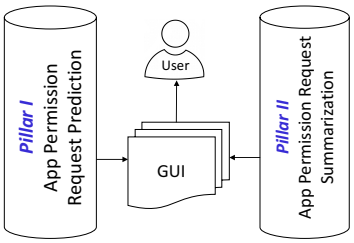


Figure 1: A high level overview of the proposed approach.

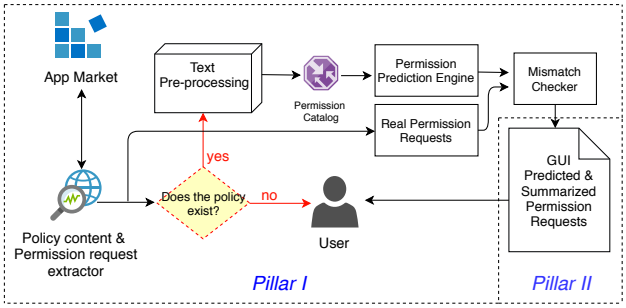


Figure 2: A high level architecture of *Pillar I* and *Pillar II*.

3.1 Pillar I: App Permission Request Prediction

The goal of *Pillar I* is to analyze the mobile app privacy policy text to predict app permission requests. As the nature of mobile app privacy policies is not static and their texts vary from one privacy policy to another one, we do need an intelligent mechanism to mine the policy texts. For this purpose, we exploit supervised learning methods.

3.1.1 Policy Text Finder. This component is responsible to find the privacy policy link for a desired app. It gets as an input the app's url and goes through its web page to check whether the app has a proper link to its privacy policy. We exploited a modified version of the scraper in [1] to crawl the policy page.

3.1.2 Text Pre-processing. It gets as an input the text extracted by the previous component and it uses NLTK [3] based on Python, and performs the following standard techniques of NLP: (1) Tokenization, where each privacy policy text is split into several tokens to later ease the process of stemming and removing stop words. (2) Removing stop words (e.g., "the", "on", "is", etc.) is done to increase the quality of data. (3) Stemming is applied on all texts to reduce the number of words and improve the results of NLP processes.

3.1.3 Permission Catalog. To make our scope as narrow as possible, we focus on those permissions which have a direct relation to the users' privacy as defined by Android [1]. A list of the identified sensitive permissions and their descriptions can be seen in Table 1.

Table 1: Permission catalog

#	Permission	Description
1	READ_CALENDAR	Allows to read the user's calendar data.
2	CAMERA	Allows access to the camera.
3	READ_CONTACTS	Allows to read the user's contacts data.
4	GET_ACCOUNTS	Allows access to the list of accounts in the Accounts Service.
5	LOCATION	Allows access to the user's location.
6	RECORD_AUDIO	Allows to record audio.
7	READ_PHONE_STATE	Allows read only access to phone state.
8	BODY_SENSORS	Allows access to data from sensors used by the user such as heart rate.
9	READ_SMS	Allows to read SMS messages.
10	INTERNET	Allows access to the Internet.
11	BLUETOOTH	Allows connection to Bluetooth devices.
12	NFC	Allows I/O operations over NFC.

3.1.4 Permission Prediction Engine. This analyzes each privacy policy text with respect to those sensitive permissions defined in Table 1. As we are solving a supervised learning task, the ultimate goal is to train a classifier to classify the privacy policy text into these distinct permission classes. This enables us to infer whether or not the app developers have already made the need of requesting certain permissions clear in the privacy policy text.

3.1.5 Mismatch Checker. This is responsible to compare the predicted list of permission requests (generated by the ML algorithm) with the real permission requests (extracted from the app's url). Once a mismatch is flagged, it will be communicated to the user (*Pillar II*).

3.2 Pillar II: App Permission Summarization

To summarize and communicate the information extracted concerning the permission prediction, we designed dedicated icons for each

[1] https://developer.android.com/guide/topics/permissions/overview

Table 2: Performance measures of the classification.

Classes	Recall	Precision	F-score
READ_CALENDAR	N/A	N/A	N/A
CAMERA	N/A	N/A	N/A
READ_CONTACTS	0.5105	0.8311	0.6884
GET_ACCOUNTS	0.6908	0.7145	0.7039
LOCATION	0.5012	0.6650	0.6550
RECORD_AUDIO	0.7118	0.8717	0.8513
READ_PHONE_STATE	0.7092	0.8493	0.8342
BODY_SENSORS	0.6638	0.7816	0.7365
READ_SMS	0.6711	0.7957	0.7717
INTERNET	0.9113	0.9518	0.9453
BLUETOOTH	0.6882	0.8441	0.7994
NFC	N/A	N/A	N/A
Overall	0.7310	0.7981	0.7879

permission request to ease the understanding of each individual permission request for the user. This would help users to know what is the permission about in a glance. As a further step, if users would be interested to know more about each permission in detail and to read the relevant sensitive information that might be revealed through accessing that specific permission, they can simply click on each dedicated icon. Fig. 3 demonstrates the different screens regarding the proposed interfaces.

4 RESULTS

4.1 Data Collection and Preparation

We collected the privacy policy text of the top 3 apps within 10 popular app categories from the Google Play Store (in total 30 apps) [2]. To provide our classifier with training and testing data, two experts went through the collected privacy policy texts and labeled them manually. We used *CountVectorizer* and *TfidfTransformer* packages in scikit-learn open source library for Python [11] for feature extraction. We then split the data set into training (65%) and testing data (35%). Using scikit-learn we exploited several classification algorithms, e.g. *Random Forest, Decision Trees, SVMs, Logistic Regression (LR)*, etc. We observed that *LR* outperforms others, therefore, we only show the results related to this classifier.

4.2 Performance Evaluation

We used recall, precision and F-score metrics to evaluate the performance of the classifier. Table 2 shows the values for the aforementioned metrics for each predicted permission. The overall recall, precision and F-score values are of 73.10%, 79.81% and 78.79%, respectively.

4.3 Reality Vs. Promise

We performed empirical experiments to examine the correlation between what mobile apps do in reality (behavior analysis) and what they promise (privacy policy analysis). Therefore, we firstly analyzed the extent to which the mobile app privacy policies are relevant to the permission requests to infer whether app developers

Table 3: Purpose specification of permission requests in policy text of health-based apps: No specification (×), specification (✓), no request (N).

App #	CAMERA	READ_SMS	READ_CONTACTS	LOCATION	PHONE_STATE	RECORD_AUDIO	GET_ACCOUNT	BODY_SENSOR
Lifesum	×	N	×	N	N	N	✓	×
Endomondo	N	N	✓	✓	✓	N	✓	✓
30dayFitnessChallenge	N	N	×	N	N	N	×	N
Runkeeper	×	N	×	✓	N	N	✓	N
Pedometer	×	N	✓	✓	×	N	N	✓

Table 4: Sensitive permission access by health-based apps: No access (×), access (✓), no request (N).

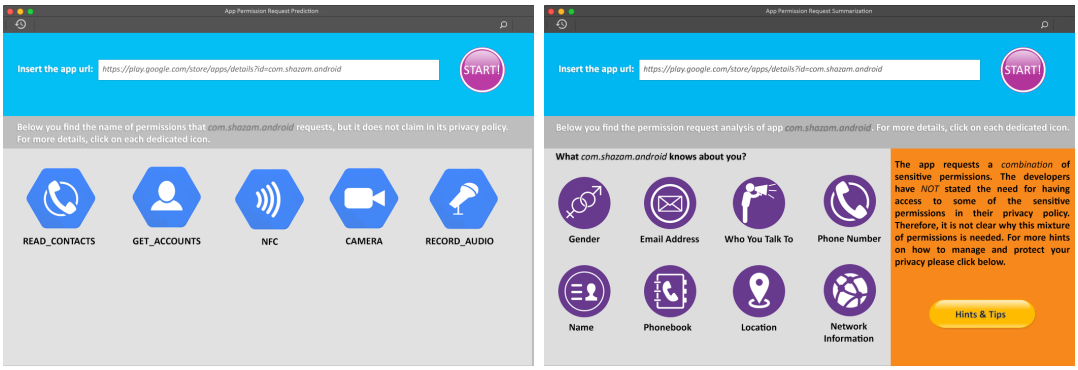
App #	CAMERA	READ_SMS	READ_CONTACTS	LOCATION	PHONE_STATE	RECORD_AUDIO	GET_ACCOUNT	BODY_SENSOR
Lifesum	×	N	✓	N	N	N	✓	✓
Endomondo	N	N	✓	✓	✓	N	✓	×
30dayFitnessChallenge	N	N	✓	N	N	N	✓	N
Runkeeper	✓	N	✓	✓	N	N	✓	N
Pedometer	✓	N	✓	✓	✓	N	N	×

claim in their privacy policies that they are going to use a certain permission. Because of space limitation, we only focus on top five health-based apps (we chose such apps, as we assume they are directly dealing with users' highly sensitive data) in our data set as shown in Table 3. Surprisingly, there is a significant number of incidents (shown by ×) where the developers failed to clarify the need of requesting and accessing certain sensitive data types.

To analyze the resource access pattern behavior of the studied apps, we exploited our already proposed tool [6]. We installed the apps on an Android device and monitored their behavior for a period of one week. It is worth mentioning that we were interested to see whether the apps access sensitive resources while there is no interaction between device and user. Thus, we did not interact with the device during this period. Table 4 shows the results of our analysis. As can be seen, all the apps accessed all the studied sensitive resources, shown by ✓(except one).

4.4 Discussion

Needless to say, our results confirmed that there is still a significant gap between what mobile apps are doing in reality and what they promise in their privacy policies. This signals that models on data protection aspects of mobile apps are problematic and call for actions become necessary when users do not have the chance to get proper and concrete information about data collection practices of



(a) (b)

Figure 3: The propose GUI: (a) The identified permission requests that have not been already clarified in the privacy policy text, (b) the information that the app might know about user.

apps. Thus, we argue that app developers should carefully clarify the needs of requesting sensitive permissions in privacy policy text. We also highlight that app privacy policies need to be severely revisited by their developers as we observed that there is a substantial number of privacy policies that do not focus on the app data access and collection practices itself, but unrelated content to the app's privacy practices.

5 CONCLUSION AND FUTURE WORK

In this paper, we demonstrated the applicability of app permission request prediction by using ML and NLP techniques. We proposed a new system to predict app sensitive permission requests by analyzing its privacy policy text. Our approach is based on supervised learning methods having a deep emphasize on proper interface designs for privacy indicator communication. Our observations showed that it is quite feasible to formulate the problem of app permission prediction considering its privacy policy text using an intelligent approach with acceptable accuracy. Also we inferred that the actual behavior of mobile apps and their permission requests do not match the promises and claims in the privacy policy text that requires regulators' attention to overcome this issue. We believe studies like ours can help to improve mobile users' privacy by enhancing awareness and advancing the understanding about privacy threats. Additionally, our results can be expanded through further lines of research. User studies can further enhance the insights in the topic of user privacy views by considering an explanatory study investigating the role of our novel system in the causal relationship of privacy attitude/behavior change. As a claim regarding the limitation of our work, during the manual labeling step, we did not find any information regarding three permissions (shown by N/A in Table 2). Therefore, providing more labeled data can further improve the discrimination power of the classifier.

REFERENCES

[1] Google Play Scraper. https://github.com/facundoolano/google-play-scraper/.

[2] Most popular Google Play app store categories. https://www.statista.com/statistics/256772/most-popular-app-categories-in-the-google-play-store/, Accessed Sep 10, 2018.

[3] Natural language toolkit. " https://www.nltk.org/.

[4] A. P. Felt, S. Egelman, and D. Wagner. 2012. I've got 99 problems, but vibration ain't one: A survey of smartphone users' concerns. In *the Proceedings of the 2nd ACM Workshop on Security and Privacy in Smartphones and Mobile Devices (SPSM'12), New York, NY, USA*. 33–44.

[5] A. P. Felt, E. Ha, S. Egelman, A. Haney, E. Chin, and D. Wagner. 2012. Android permissions: User attention, comprehension, and behavior. In *the Proceedings of the 8th ACM Symposium on Usable Privacy and Security (SOUPS'12), New York, NY, USA*. 1–3.

[6] M. Hatamian, A. Kitkowska, J. Korunovska, and S. Kirrane. 2018. "It's Shocking¡': Analysing the Impact and Reactions to the A3: Android Apps Behaviour Analyser. In *Data and Applications Security and Privacy XXXII*. Springer International Publishing, Cham, 198–215.

[7] M. Hatamian, J. Serna, K. Rannenberg, and B. Igler. 2017. FAIR: Fuzzy alarming index rule for privacy analysis in smartphone apps. In *the Proceedings of the 14th International Conference on Trust and Privacy in Digital Business (TrustBus), Lyon, France*. 3–18.

[8] J. Lin, S. Amini, J. I. Hong, N. Sadeh, J. Lindqvist, and J. Zhang. 2012. Expectation and purpose: understanding users' mental models of mobile app privacy through crowdsourcing. In *the Proceedings of ACM Conference on Ubiquitous Computing (UbiComp'12), New York, NY, USA*. 501–510.

[9] B. Liu, M. S. Andersen, F. Schaub, H. Almuhimedi, S. Zhang, N. Sadeh, A. Acquisti, and Y. Agarwal. 2016. Follow my recommendations: A personalized privacy assistant for mobile app permissions. In *the Proceedings of the 12th Symposium on Usable Privacy and Security (SOUPS 2016), Denver, CO, USA*. 27–41.

[10] R. Pandita, X. Xiao, W. Yang, W. Enck, and T. Xie. 2013. WHYPER: towards automating risk assessment of mobile applications. In *the Proceedings of the 22nd USENIX Conference on Security, Washington, D.C., USA*. 527–542.

[11] F. Pedregosa, G. Varoquaux, A. Gramfort, B. Thirion V. Michel, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, JA. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research* 12 (2011), 2825–2830.

[12] S. Rosen, Z. Qian, and Z. M. Mao. 2013. AppProfiler: a flexible method of exposing privacy-related behavior in android applications to end users. In *the Proceedings of the 3rd ACM Conference on Data and Application Security and Privacy, San Antonio, Texas, USA*. 221–232.

[13] Le Yu, Tao Zhang, Xiapu Luo, and Lei Xue. 2015. 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)*. ACM, New York, NY, USA, 39–50. https://doi.org/10.1145/2808117.2808125