A First Step Towards Detecting Values-violating Defects in Android APIs

Conghui Li  
Faculty of Information Technology  
Monash University  
Melbourne, Australia  
clii0087@student.monash.edu

Humphrey O. Obie  
HumaniSE Lab  
Monash University  
Melbourne, Australia  
humphrey.obie@monash.edu

Hourieh Khalajzadeh  
HumaniSE Lab  
Monash University  
Melbourne, Australia  
Hourieh.Khalajzadeh@monash.edu

Abstract—Human values are an important aspect of life and should be supported in ubiquitous technologies such as mobile applications (apps). There has been a lot of focus on fixing certain kinds of violation of human values, especially privacy, accessibility, and security while other values such as pleasure, tradition, and humility have received little focus. In this paper, we investigate the relationship between human values and Android API services and developed algorithms to detect potential violation of these values. We evaluated our algorithms with a manually curated ground truth set resulting in a high performance, and applied the algorithms to 10,000 apps. Our results show a correlation between violation of values and the presence of viruses. Our results also show that apps with the lowest number of installations contain more violation of values and the frequency of the violation of values was highest in social apps.

I. INTRODUCTION

Mobile applications (apps) are widely used in the modern society to improve our productivity and to fulfil end-users purposes [1]. However, the use of technology can sometimes violate the values of end-users and in the worst case cause harm to them and society at large. For example, a recently publicised employment interview algorithms were shown to be flawed, when they scored a reporter 6 out of 9 points for English-language competency after the interview was conducted exclusively in German [2]. Such a flawed system could bar qualified candidates from getting jobs and cause employees to miss out on good candidates.

Human values are the guiding principles for what people consider important in life [3]. They serve as a guide for actions and a vehicle for expressing need. Human values such as privacy and security have garnered both academia and industry focus but other broader values including self-direction, benevolence, tradition, etc. have not received sufficient attention in the research community [4]. Although there have been some recent efforts in value-based software engineering, the field is still in its nascent stages [4].

One of the earliest attempts in describing human values at a low level is the recent work by Mougouei [5], which proposed a theoretical framework named, AIR (Annotation, Inspection, and Recommendation) to account for human values in source code. Specifically, this framework includes the following components: annotating Application Programming Interfaces (APIs) with human values based on the relationship between human values definition and the API definition; annotating code at class level or method level based on the identified values in the APIs; using the results to inspect values that may be potentially breached by the code, leading to a condition named value smell; and finally, recommending a possible fix for the potential values violation. Our work builds upon and extends this theoretical framework by concretising the detection of potential violation of values in source code.

In this paper, we investigate the relevant relationships between common API services and human values based on the literature and introduce multiple algorithms based on the Android API services to detect potential violation of values in Android apps. We evaluate our algorithms on an evaluation set; a manually annotated source code from 46 android apps with an average accuracy of 0.93 and recall of 0.84. Prompted by the promising evaluation results, we applied our models to 10,000 apps randomly curated from AndroZoo [6], covering multiple categories. Our results show a correlation between violation of values and the presence of viruses. Our results also show a correlation between the violation of human values and the presence of viruses in apps; apps with the lowest number of installations had more potential of value violations; and social apps had the highest rate of value violation.

II. BACKGROUND AND MOTIVATION

Human values determine behaviour and attitude. [7]. The concept of human values have been well explored and developed in the social sciences.

Human values play a critical role in our lives to explain the motivations, opinion types and actions for different cultural groups, and individuals [3]. The Schwartz theory of basic human values is the most widely accepted and adopted model of human values. This theory is grounded on surveys conducted in multiple countries and covers a wide range of different ages, cultural backgrounds, genders, and geography [3]. The Schwartz categorises 58 value items in 12 broad categories. These categories are described in Table I. Schwartz theory seen adoption in different areas including software engineering, e.g., the Values Q-sort has been used to understand the values opinion-types of software engineers [8]. A similar work adopted the Schwartz theory to understand the
values of Bangladeshi female farmers' for agriculture mobile applications development [9]. Recently, Obie et al. developed natural language processing (NLP) techniques to understand potential violation of human values in app reviews as reported by users [10].

Mougouei argues that APIs such as those implemented in Android source code are not values-agnostic, and introduced the AIR framework as an approach to support value programming in software projects [5]. Mougouei proposed relationships between 9 common Android apps and human value categories (c.f. [5]). Based on these relationships, APIs can violate certain human values, e.g., the values of security and universalism can be violated by the Android Accessibility Service if not properly implemented [11]. Human values can be violated either through negligence or through deliberate abuse of API services. While [5] proposes a framework, our work is the first to design algorithms for detecting potential value-violations and conducting a large scale study to understand the prevalence of these violations in Android apps.

III. RULE-BASED VALUES VIOLATION DETECTORS

In this paper, we selected a subset of 6 Android APIs related to various human values. Our selection is based primarily on API usage frequency as documented by Wang and Godfrey [12]. Using the Android documentation[1] we also selected some APIs that support hardware interaction and user management. These include the following: android.animation, android.media, android.mtp, android.nfc, android.telephony and android.hardware. We show the selected API’s and their related values in Table I. We developed algorithms to detect potential violation of values in the use of selected APIs in Android apps.

A. Data Collection

We collected 46 Android apps from the Google Play store covering 10 categories and then manually annotated them with value violation – API tags. The first author did a round of annotation which was validated by the second author, and based on discussions during Zoom meetings between the first and second author, the annotations were finalised. Table III shows the application categories, APIs, and number of potential value violations. These annotated apps formed our truthset used for the evaluation of our algorithms.

B. Data Preparation

The preparation of our data for the detection of values-violation in the selected API’s is done in two simple phases: Android Application Package (APK) decomposition and Java Abstract Syntax Tree (AST) analysis.

1) APK Decomposition: the source in APK is sealed by android, hence it needs to be decompiled to be able to access the source code. We used the APKtool[2] in this phase.

2) AST Analysis: typically, a Java program without bug can be represented in an abstract syntax tree structure, and thus it is possible to traverse the tree. We traverse the AST using Depth-First Search and identify code that may be linked to values violation. An example detector for the violation of Universalism value in Media API is shown in Algorithm. 1

| VALUE CATEGORY | DESCRIPTION |
|----------------|-------------|
| Self Direction | Independent thought and action—choosing, creating, exploring |
| Stimulation    | Excitement, novelty, and challenge in life |
| Hedonism       | Pleasure and serious gratification for oneself |
| Achievement    | Personal success according to social standards |
| Power          | Social status and prestige, control or dominance over people and resources |
| Security       | Safety, harmony and stability of society, of relationships, and of self |
| Tradition      | Respect, commitment and acceptance of the customs and ideas that traditional culture or religion provide |
| Conformity     | Restraint of actions, inclinations, and impulses likely to upset or harm others and violate social expectations or norms |
| Benevolence    | Preservation and enhancement of the welfare of people with whom one is in frequent personal contact |
| Universalism   | Understanding, appreciation, tolerance and protection for the welfare of all people and for nature |
| Face           | Maintaining public image and avoiding humiliation |
| Humility       | Recognizing insignificance in the larger scheme of things |

Algorithm 1 Media API/Universalism value violation detector

INPUT: : java file path
OUTPUT: return a boolean value:

True: violation exist
False: violation does not exist

1: /* using deep first search to search all the AST nodes out*/
2: AstNodes ← search( javaPath )
3: ViolationList ← an empty list
4: VarNames ← an empty list
5: for all AstNode in AstNodes do
6: if AstNode is Import Decelator Node then
7: if “com.google.android.exoplayer” = AstNode.path then
8: append “Violation” in ViolationList
9: end if
10: end if
11: end for
12: for all ListElement in ViolationList do
13: if ListElement = “Violation” then
14: Return True
15: end if
16: end for
17: Return False

C. Values Violation Detection Methods

We introduce an overview of the APIs and algorithms for detecting potential violation of values. The complete set of algorithms and source codes are provided in a supplementary repository[3].

1) Media: The media API[4] provides the functionality of multimedia including audio, video, and manage media data for users. Based on [5], using the media API can potentially violate human values.

TABLE I
VALUE CATEGORIES AND DESCRIPTION [3]

| Value Category | Description |
|----------------|-------------|
| Self Direction | Independent thought and action—choosing, creating, exploring |
| Stimulation    | Excitement, novelty, and challenge in life |
| Hedonism       | Pleasure and serious gratification for oneself |
| Achievement    | Personal success according to social standards |
| Power          | Social status and prestige, control or dominance over people and resources |
| Security       | Safety, harmony and stability of society, of relationships, and of self |
| Tradition      | Respect, commitment and acceptance of the customs and ideas that traditional culture or religion provide |
| Conformity     | Restraint of actions, inclinations, and impulses likely to upset or harm others and violate social expectations or norms |
| Benevolence    | Preservation and enhancement of the welfare of people with whom one is in frequent personal contact |
| Universalism   | Understanding, appreciation, tolerance and protection for the welfare of all people and for nature |
| Face           | Maintaining public image and avoiding humiliation |
| Humility       | Recognizing insignificance in the larger scheme of things |

1https://developer.android.com/reference/packages
2https://ibotpeaches.github.io/Apktool/
3https://github.com/awesomehumphrey/Values-violating-defects-in-android-apis
4https://developer.android.com/reference/android/media/package-summary
Firstly, multimedia advertisements (ads) can lead to the violation of self-direction, especially when the ads are forced on the users without an option to close the ad. Also, in cases where the presented ads lack diversity, representing the different background of users, this can constitute a violation of the value of universalism. To detect this potential violation, our algorithm flags import statements related to advertisements, as shown in Algorithm 1.

Secondly, automatic background music without controls for stopping or pausing might violate the values of pleasure and independence of a user. To detect this, we check the code stopping or pausing might violate the values of pleasure and

### TABLE II

**RELATIONSHIP BETWEEN SELECTED APIs AND HUMAN VALUES. ADAPTED FROM MOUGOUÉI [5]**

| API name     | API description                                                                 | Self-direction | Stimulation | Hedonism | Face | Tradition | Humility |
|--------------|---------------------------------------------------------------------------------|----------------|-------------|----------|------|-----------|----------|
| Android.animation | These classes provide functionality for the property animation system, which allows you to animate object properties of any type. | ✓              | ✓           | ✓        |      |           |          |
| Android.media | Provides classes that manage various media interfaces in audio and video. | ✓              | ✓           |          |      | ✓         |          |
| Android.mtp  | Provides APIs that let you interact directly with connected cameras and other devices, using the Picture Transfer Protocol (PTP) |                | ✓           |          |      |           |          |
| Android.nfc  | Provides access to Near Field Communication (NFC) functionality, allowing applications to read NDEF message in NFC tags. | ✓              | ✓           |          |      |           |          |
| Android.telephony | Provides APIs for monitoring the basic phone information, plus utilities for manipulating phone number strings. | ✓              | ✓           |          |      |           |          |
| Android.hardware | Provides support for hardware features, such as the camera and other sensors. |                |             |          |      | ✓         |          |

### TABLE III

**OVERVIEW OF EVALUATION TRUTHSET DATA**

| Human value | API name | communication | finance | education | fitness | game | health | photography | productivity | social | travel |
|-------------|----------|---------------|---------|-----------|---------|------|--------|-------------|--------------|--------|--------|
| Self Direction | mtp | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | media | 9 | 5 | 3 | 3 | 2 | 1 | 1 | 2 | 0 | 0 |
| | nfc | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Security | hardware | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | telephony | 3 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | nfc | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| hedonism | media | 0 | 4 | 3 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | animation | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Universalism | media | 0 | 4 | 3 | 0 | 2 | 0 | 1 | 1 | 0 | 0 |
| Conformity | telephony | 3 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Total number | 14 | 7 | 7 | 7 | 2 | 3 | 1 | 3 | 1 | 1 | 1 |

3) **MTP**: The MTP API[6] provides the functionality of interaction among user’s device, camera and other devices with the Picture Transfer Protocol (PTP). The MTP API is related to the value of self-direction (i.e., independence). To detect potential violation of the value of self-direction, our algorithm checks to see whether the data transmission is bi-directional; if the MtpDevice objects do not call the input and output data methods in the same class, it is flagged as a potential value violation.

4) **NFC**: Near Field Communication (NFC) is a technique widely applied in our daily life. For instance, with the NFC users’ phones can be used as digital wallets. The NFC tag consists of 2 types of the data format: NFC Data Exchange Format (NDEF) or Non-NDEF. Moreover, to distribute the NFC tag data to the correct activities precisely, the tag dispatch system provides 3 NFC action intents, including NDEFDISCOVERED, TECH DISCOVERED and TAG DISCOVERED, for different distribution priority. The priority structure is shown [Figure 4](#). Also, the intent NDEFDISCOVERED can only be manipulated by NDEF data.

Typically, a well-operated system only distributes the required resource or authority to an activity. As such, our algorithm flags the potential violation of the value of self-direction if the developer does not set the intents in the order of intent priority in the hierarchical structure, e.g., if the NDEF tag is first distributed to the TAG DISCOVERED

---

[1]https://developer.android.com/reference/android/animation/package-summary

[2]https://developer.android.com/reference/android/nfc/package-summary

[3]https://developer.android.com/reference/android/mtp/package-summary

[4]https://developer.android.com/reference/android/nfc/package-summary

---
instead of *NDEFDISCOVERED*. After inspecting all intents in the manifest file, the violation of the self-direction value would be flagged if intent with low priority supercedes intent with higher priority.

In general, when applications write information into NFC tags, developers may need to ensure that the tags have security mechanisms to prevent issues like privacy leaks. One effective solution is to insert the Android Application Records (AAR). To check for a potential violation of privacy, our algorithm checks whether the method call `createApplicationRecord()` from `NdefRecord` object exists.

![Diagram](image.png)

**Fig. 1. NFC Tag Dispatch System**

5) Telephony: The telephony API\(^{9}\) provides essential information and status of cell phone, and also serves as the communication monitor. Because the function of sending an SMS message to other users through their phone numbers is prone to eaves-dropping and possible manipulation, there exist the potential violations of the values of security and conformity. Taking the method `SendMultipleMessage()` as an example, sending and being called by `SmsManager` object will be queried over all java files of the application by our algorithm.

6) Hardware: The hardware API\(^{9}\) support developers in manipulating sensors, camera, and other components in devices. The human value associated with the hardware API is privacy and security. Using hardware API, the most significant privacy and security issue is that an app can collect photographs or other media data without users’ consent. Since photographs are one of the most important private and personal identifying information of users,\(^{10}\) our algorithm detects automatic photograph capture and invisible camera layout. Our algorithm tracks the method `takePicture()` called by Camera object and `createCaptureSession()` called by `CameraDevice` object, as well as the surface layouts with a small size.

**D. Evaluation Results**

In total we developed 10 algorithms for detecting potential violation of human values in 6 Android APIs. We evaluated our algorithms on our truthset consisting of 46 apps from the Google Play store. Our algorithms detected potential violation of human values with high accuracy and recall performance as shown in Table V.

| API name | violation       | Accuracy | Recall |
|----------|-----------------|----------|--------|
| animation | hedonism        | 1.0      | 1.0    |
| media    | hedonism        | 0.761    | 0.808  |
|          | self direction  | 0.761    | 0.808  |
|          | Universialism   | 0.913    | 0.818  |
| mtp      | self direction  | 1.0      | 1.0    |
|          | security        | 1.0      | 1.0    |
| nfc      | security        | 1.0      | 1.0    |
| nfc      | Conformity      | 0.857    | 0.5    |
| telephony| security        | 1.0      | 1.0    |
| hardware | security        | 1.0      | 1.0    |

**IV. LARGE SCALE ANALYSIS OF ANDROID APPS**

Based on the promising evaluation results, we carried out a large scale analysis of Android apps using our algorithms to understand potential violation of human values in these different apps and their categories.

**A. Dataset**

We randomly selected 10,000 APK files from AndroZoo for our large scale analysis. AndroZoo is a very rich APK dataset collected from Google Play by Allix et al., AppChina and Anzhi\(^{6}\). In addition to the APK files, we also collected other information from the Google Play store including user comments, and number of installations.

In our analysis, we consider the root APIs and their sub-APIs and even APIs encapsulated by third-party libraries, e.g., the Google Video Ads API extends the Media API. While some generic root APIs such as android.app and android.animation are widely used by developers, some functional APIs such as android.mtp, android.gesture only exist in an APP that has specific functions, so are used less frequently.

**B. Results and Discussion**

Below we highlight and discuss the results of our Android apps analysis.

1) Relationship Between Violation of Values and Malware: VirusTotal\(^{11}\) is a free analysis service for viruses, worms, Trojans, and all kinds of malware that can quickly detect suspicious files and web addresses. The difference between VirusTotal and traditional anti-virus software is that it scans files through various anti-virus engines. Using a variety of anti-virus engines allows users to determine whether an uploaded file is a malicious software based on the detection results of each anti-virus engine. In the AndroZoo dataset, the query results of VirusTotal exist as a feature of the data, so we can measure the performance of our algorithms in terms of security vulnerabilities by comparing value violations with the presence of viruses.

\(^{9}\)https://developer.android.com/reference/android/telephony/package-summary

\(^{10}\)https://developer.android.com/reference/android/hardware/package-summary

\(^{11}\)https://www.virustotal.com/gui/
### Table IV

| app categories | count | violation rate | Hedonism | Self Direction | Universalism | Security | Conformity |
|----------------|-------|----------------|----------|----------------|--------------|----------|------------|
| ART            | 17    | 5.8%           | 0        | 1              | 1            | 0        | 0          |
| AUTO           | 31    | 6.4%           | 0        | 2              | 2            | 0        | 0          |
| BEAUTY         | 17    | 5.8%           | 0        | 0              | 0            | 1        | 1          |
| BOOKS          | 17    | 6.7%           | 1        | 11             | 1            | 0        | 0          |
| BUSINESS       | 262   | 8.3%           | 1        | 18             | 17           | 5        | 3          |
| COMICS         | 7     | 14.3%          | 0        | 0              | 0            | 1        | 1          |
| COMMUNICATION  | 80    | 11.2%          | 0        | 8              | 7            | 3        | 2          |
| DATING         | 20    | 5.0%           | 0        | 5              | 0            | 0        | 0          |
| EDUCATION      | 279   | 13.1%          | 0        | 48             | 46           | 9        | 4          |
| ENTERTAINMENT  | 176   | 15.9%          | 0        | 23             | 23           | 6        | 2          |
| EVENTS         | 9     | 22.2%          | 0        | 2              | 2            | 0        | 0          |
| FINANCE        | 156   | 8.3%           | 1        | 12             | 11           | 1        | 1          |
| FOOD           | 93    | 11.8%          | 0        | 8              | 8            | 2        | 1          |
| GAME           | 641   | 13.2%          | 2        | 79             | 77           | 6        | 2          |
| HEALTH         | 167   | 20.3%          | 0        | 33             | 30           | 2        | 2          |
| HOUSE          | 34    | 5.9%           | 0        | 2              | 2            | 0        | 0          |
| LIBRARIES      | 8     | 0%             | 0        | 0              | 0            | 0        | 0          |
| LIFESTYLE      | 229   | 13.1%          | 0        | 26             | 26           | 0        | 2          |
| MAPS           | 74    | 9.4%           | 0        | 7              | 7            | 0        | 0          |
| MEDICAL        | 59    | 5.0%           | 0        | 3              | 3            | 0        | 0          |
| MUSIC          | 217   | 30.8%          | 0        | 64             | 60           | 7        | 2          |
| NEWS           | 101   | 27.2%          | 0        | 29             | 28           | 0        | 0          |
| PARENTING      | 7     | 0%             | 0        | 0              | 0            | 0        | 0          |
| PERSONALIZATION| 135   | 13.3%          | 0        | 17             | 16           | 1        | 1          |
| PHOTOGRAPHY    | 81    | 14.8%          | 0        | 12             | 12           | 0        | 0          |
| PRODUCTIVITY   | 198   | 11.0%          | 0        | 17             | 17           | 5        | 5          |
| SHOPPING       | 125   | 12.0%          | 0        | 14             | 14           | 5        | 5          |
| SOCIAL         | 104   | 24.0%          | 0        | 23             | 23           | 1        | 1          |
| SPORTS         | 106   | 14.1%          | 1        | 15             | 14           | 0        | 0          |
| TOOLS          | 283   | 7.7%           | 1        | 16             | 15           | 6        | 3          |
| TRAVEL         | 151   | 6.6%           | 0        | 10             | 10           | 0        | 0          |
| VIDEO          | 33    | 18.1%          | 0        | 6              | 6            | 0        | 0          |
| WEATHER        | 20    | 5.0%           | 0        | 1              | 1            | 0        | 0          |

### Table VI

**Comparison of Value Violations and Security Vulnerabilities**

| Human value | API name | Virus Total | Values violation | Overlapping rate |
|-------------|----------|-------------|------------------|------------------|
| Self Direction |          |             |                  |                  |
| mtp          | 0        | 1           | 0.0              |                  |
| media        | 157      | 127         | 16.0%            |                  |
| nfc          | 2        | 14          | 14.29%           |                  |
| Security     |          |             |                  |                  |
| hardware     | 0        | 0           | NA               |                  |
| telephony    | 74       | 117         | 63.25%           |                  |
| nfc          | 34       | 53          | 64.13%           |                  |
| media        | 3        | 15          | 20.0%            |                  |
| animation    | 0        | 0           | NA               |                  |
| Universalism |          |             |                  |                  |
| media        | 155      | 962         | 63.25%           |                  |
| Conformity   | telephony| 74          | 16.11%           |                  |

As summarized in [Table VI], there is a high degree of correlation between viruses and the potential violation of security and conformity values i.e., about 63 per cent of the APK files that contain potential values violation of security and conformity also have viruses. In addition, the degree of correlation between other violations and viruses is minimal in the other value categories. This shows that the violation of specific human values in Android apps, such as security, is often accompanied by security vulnerabilities.

2) User Perception: Users may show concern towards violation of their values, e.g., when there are too many security issues in an app or cluttered ads that interfere with the user experience. These concerns are reflected in the reviews and downloads of the app. Hence, we also evaluated additional information of the apps including comments and frequency of installation. **Table VII** and **Table VIII** summarize user comments and frequency of app installation under different type of violations. In order to explore the perceptions of users in the comments, we used the overall star rating of an app to represent users’ overall satisfaction of the app. Since the lowest star a user can assign to an app is 1 in the Google Play store, in places where the user score was 0, the app did not receive any feedback whatsoever, i.e., there is no star rating or user comment.

As shown in [Table VIII] in most APIs, apps with the lowest number of installations had more potential violation of human values. Violations such as NFC and telephony are often found in apps with 50,000 or fewer downloads. A possible
explanation is that developers would fix these violations when an app is popular because of profits, reputation, and their responsibility to improve the user experience [13]. However, in the case of media, the delivery of video ads mainly leads to the violation of the value of pleasure. While advertising can constitute a violation of value because it is such an effective way to monetize, developers usually will not turn down this revenue stream, no matter the size of the downloads.

3) App Categories: Different types of apps may have different emphasis on their functionalities, which leads them to use different APIs. For example, Table VIII summarizes the total number of different types of apps and corresponding values violation. Even though different apps make use of different API services, this does not mean that value violation frequency of apps necessarily relates to the frequency of their API use, e.g., although financial and business apps use many APIs, the value violation ratio for high-security-requirement software remains relatively low.

Furthermore, social apps such as music, and dating apps show high rates of values violation, i.e., 25% to 30%. It can be noted that video advertising makes up the majority of violations. Because of the free nature of these apps, advertising can be highly profitable, it also significantly reduces the user experience.

**CONCLUSION**

In this paper, we explored the relationships between human values and Android API services based on the literature. We presented algorithms to detect potential violation of human values in 6 API services. The algorithms were evaluated using a manually annotated set of Android apps. Based on the high performance of the algorithms, we carried out a large scale analysis of 10,000 apps covering multiple categories. The results of our analysis show that there is a correlation between the violation of human values and the presence of viruses in apps; apps with the lowest number of installations had more potential of value violations; and social apps had the highest rate of values violation.

Our future work entails conducting empirical studies to understand in finer details how end user values are violated by software artefacts from the perspectives of end users and also the awareness and perception of developers in dealing with these kind of challenges.

**ACKNOWLEDGMENT**

Support for this work from ARC Laureate Program FL190100035 is gratefully acknowledged.

**REFERENCES**

[1] R. K. Bellamy, K. Dey, M. Hind, S. C. Hoffman, S. Houde, K. Kannan, P. Lohia, S. Mehta, A. Mojsilovic, S. Nagar, K. N. Ramamurthy, J. Richards, D. Saha, P. Sattigeri, M. Singh, K. R. Varshney, and Y. Zhang, “Think your artificial intelligence software is fair? think again,” IEEE Software, vol. 36, no. 4, pp. 76–80, 2019.

[2] S. Soper, “Fired by bot at amazon: ‘it’s you against the machine’,” [Online.] Available: https://www.bloomberg.com/news/features/2021-06-28/fired-by-bot-amazon-turns-to-machine-managers-and-workers-are-losing-out?ref=EJ35iF5v, accessed June, 2021.

[3] S. H. Schwartz, J. Ciecich, M. Vecchione, E. Davidov, R. Fischer, C. Beierlein, A. Ramos, M. Verkasalo, J.-E. Lönqvist, K. Demiruktu, O. Dirilen-Gumus, and M. Konty, “Refining the theory of basic individual values,” Journal of personality and social psychology, vol. 103, no. 4, pp. 663–688, 2012.

[4] H. Perera, A. Nurwidiantoro, W. Hussain, D. Mougouei, J. Whittle, R. Shams, and G. Oliver, “A study on the prevalence of human values in software engineering publications, 2015 – 2018,” in ICSE, 2020.

[5] D. Mougouei, “Engineering human values in software through value programming,” p. 133–136, 2020.

[6] K. Allix, T. F. Bissyandé, J. Klein, and Y. L. Traon, “Androzoo: Collecting millions of android apps for the research community,” in IEEE/ACM 13th Working Conference on Mining Software Repositories, 2016, pp. 468–471.

[7] A.-S. Cheng and K. R. Fleischmann, “Developing a meta-inventory of human values,” in ASDS&T, vol. 47, 2010.

[8] E. Winter, S. Forshaw, and M. A. Ferrario, “Measuring human values in software engineering,” in ACM/IEEE 12th International Symposium on Empirical Software Engineering and Measurement, 2018, pp. 1–4.

[9] R. A. Shams, W. Hussain, G. Oliver, A. Nurwidiantoro, H. Perera, and J. Whittle, “Society-oriented applications development: Investigating users’ values from bangladeshi agriculture mobile applications,” 2020.

[10] H. O. Obwe, W. Husssain, X. Xia, J. Grundy, L. Li, B. Turhan, J. Whittle, and M. Shaheen, “A first look at human values-violation in app reviews,” in 2021 IEEE/ACM 43rd International Conference on Software Engineering: Software Engineering in Society (ICSE-SEIS), 2021, pp. 29–38.

[11] M. Naseri, N. Borges, A. Zeller, and R. Rouvoy, “Accessileaks: Investigating privacy leaks exposed by the android accessibility service,” Proceedings on Privacy Enhancing Technologies, vol. 2019, pp. 291 – 305, 2019.

[12] W. Wang and M. W. Godfrey, “Detecting api usage obstacles: A study of ios and android developer questions,” in 2013 10th Working Conference on Mining Software Repositories (MSR), 2013, pp. 61–64.

[13] J. Bûller, A. Baur, M. Bick, and J. Shi, “Big data, big opportunities: Revenue sources of social media services besides advertising,” in 14th Conference on e-Business, e-Services and e-Society (I3E), 2015, pp. 183–199.

**TABLE VIII**

| Human value       | API name | 0 - 100 | 100 - 1000 | 1000 - 10000 | 10000 - 50000 | 50000 - 1e+05 | 1e+05 - 1e+06 | higher 1e+06 |
|-------------------|---------|---------|------------|--------------|---------------|---------------|---------------|--------------|
| Self Direction    | mtp     | 0       | 0          | 0            | 0             | 0             | 0             | 0            |
|                   | media   | 117     | 86         | 74           | 22            | 36            | 79            | 63           |
|                   | telephony | 4      | 2          | 2            | 0             | 0             | 0             | 0            |
| Security          | hardware | 0     | 0          | 0            | 0             | 0             | 0             | 0            |
|                   | telephony | 7      | 5          | 9            | 6             | 3             | 3             | 2            |
| Hedonism          | media   | 13     | 8          | 5            | 2             | 0             | 1             | 0            |
|                   | animation | 0    | 0          | 0            | 0             | 0             | 0             | 0            |
| Conformity        | telephony | 10    | 7          | 9            | 3             | 3             | 3             | 2            |