2019

AP CollegeBoard

# AP® Research Academic Paper

## Sample Student Responses and Scoring Commentary

**Inside:**

**Sample C**

- ☑ **Scoring Guideline**
- ☑ **Student Samples**
- ☑ **Scoring Commentary**

# AP® RESEARCH — ACADEMIC PAPER
# 2019 SCORING GUIDELINES

| The Response… | | | | |
|---|---|---|---|---|
| **Score of 1**<br>**Report on Existing Knowledge** | **Score of 2**<br>**Report on Existing Knowledge with Simplistic Use of a Research Method** | **Score of 3**<br>**Ineffectual Argument for a New Understanding** | **Score of 4**<br>**Well-Supported, Articulate Argument Conveying a New Understanding** | **Score of 5**<br>**Rich Analysis of a New Understanding Addressing a Gap in the Research Base** |
| Presents an overly broad topic of inquiry. | Presents a topic of inquiry with narrowing scope or focus, that is NOT carried through either in the method or in the overall line of reasoning. | Carries the focus or scope of a topic of inquiry through the method **AND** overall line of reasoning, even though the focus or scope might still be narrowing. | Focuses a topic of inquiry with clear and narrow parameters, which are addressed through the method and the conclusion. | Focuses a topic of inquiry with clear and narrow parameters, which are addressed through the method and the conclusion. |
| Situates a topic of inquiry within a single perspective derived from scholarly works **OR** through a variety of perspectives derived from mostly non-scholarly works. | Situates a topic of inquiry within a single perspective derived from scholarly works **OR** through a variety of perspectives derived from mostly non-scholarly works. | Situates a topic of inquiry within relevant scholarly works of varying perspectives, although connections to some works may be unclear. | Explicitly connects a topic of inquiry to relevant scholarly works of varying perspectives **AND** logically explains how the topic of inquiry addresses a gap. | Explicitly connects a topic of inquiry to relevant scholarly works of varying perspectives **AND** logically explains how the topic of inquiry addresses a gap. |
| Describes a search and report process. | Describes a nonreplicable research method **OR** provides an oversimplified description of a method, with questionable alignment to the purpose of the inquiry. | Describes a reasonably replicable research method, with questionable alignment to the purpose of the inquiry. | Logically defends the alignment of a detailed, replicable research method to the purpose of the inquiry. | Logically defends the alignment of a detailed, replicable research method to the purpose of the inquiry. |
| Summarizes or reports existing knowledge in the field of understanding pertaining to the topic of inquiry. | Summarizes or reports existing knowledge in the field of understanding pertaining to the topic of inquiry. | Conveys a new understanding or conclusion, with an underdeveloped line of reasoning **OR** insufficient evidence. | Supports a new understanding or conclusion through a logically organized line of reasoning **AND** sufficient evidence. The limitations and/or implications, if present, of the new understanding or conclusion are oversimplified. | Justifies a new understanding or conclusion through a logical progression of inquiry choices, sufficient evidence, explanation of the limitations of the conclusion, and an explanation of the implications to the community of practice. |
| Generally communicates the student's ideas, although errors in grammar, discipline-specific style, and organization distract or confuse the reader. | Generally communicates the student's ideas, although errors in grammar, discipline-specific style, and organization distract or confuse the reader. | Competently communicates the student's ideas, although there may be some errors in grammar, discipline-specific style, and organization. | Competently communicates the student's ideas, although there may be some errors in grammar, discipline-specific style, and organization. | Enhances the communication of the student's ideas through organization, use of design elements, conventions of grammar, style, mechanics, and word precision, with few to no errors. |
| Cites **AND/OR** attributes sources (in bibliography/ works cited and/or in-text), with multiple errors and/or an inconsistent use of a discipline-specific style. | Cites **AND/OR** attributes sources (in bibliography/ works cited and/or in-text), with multiple errors and/or an inconsistent use of a discipline-specific style. | Cites **AND** attributes sources, using a discipline-specific style (in both bibliography/works cited **AND** in-text), with few errors or inconsistencies. | Cites **AND** attributes sources, with a consistent use of an appropriate discipline-specific style (in both bibliography/works cited **AND** in-text), with few to no errors. | Cites **AND** attributes sources, with a consistent use of an appropriate discipline-specific style (in both bibliography/works cited **AND** in-text), with few to no errors. |

# AP® RESEARCH
# 2019 SCORING COMMENTARY

## Academic Paper

### Overview

This performance task was intended to assess students' ability to conduct scholarly and responsible research and articulate an evidence-based argument that clearly communicates the conclusion, solution, or answer to their stated research question. More specifically, this performance task was intended to assess students' ability to:

- Generate a focused research question that is situated within or connected to a larger scholarly context or community;
- Explore relationships between and among multiple works representing multiple perspectives within the scholarly literature related to the topic of inquiry;
- Articulate what approach, method, or process they have chosen to use to address their research question, why they have chosen that approach to answering their question, and how they employed it;
- Develop and present their own argument, conclusion, or new understanding while acknowledging its limitations and discussing implications;
- Support their conclusion through the compilation, use, and synthesis of relevant and significant evidence generated by their research;
- Use organizational and design elements to effectively convey the paper's message;
- Consistently and accurately cite, attribute, and integrate the knowledge and work of others, while distinguishing between the student's voice and that of others;
- Generate a paper in which word choice and syntax enhance communication by adhering to established conventions of grammar, usage, and mechanics.

# Quantitative Analysis of Android Permission Requests

Word Count: 4402

# Table of Contents

## Abstract

Android is currently the most popular mobile phone operating system worldwide (StatCounter, 2019). How does the system safeguard user private information? It employs a permission system to restrict hardware access and prevent malicious applications from obtaining sensitive data. Previous studies indicate that there may be an increased user awareness of this permission system when downloading mobile applications from 2012 to 2017, yet no existing research has explored this possible trend (Felt, 2012; Alani, 2017). This paper aims to address this gap by investigating how the permission requests of the top 100 google play store Android applications have varied in the United States from 2016 to 2018 via a quantitative analysis. Chart data were extracted from an online database, and processed by a custom program that parsed the values and generated graphs showing the relationship between time and the number of permission requests. This quantitative analysis was performed on overall permission requests, as well as individual requests, creating 193 graphs in total. These graphs were further grouped and the findings were then summarized. The study discovered a negative correlation between time and the total number of permission requests and identified three potential factors that led to this downward trend (user privacy awareness, the evolution of the Android permission system, and the end user demand for new features) by analyzing individual types of permissions. Overall, this study provides evidence for a downward trend of permission requests from Android applications in recent years and sheds lights on the potential contributing factors for this

trend. The findings can help application developers to better meet the needs of the end users; system programmers to further perfect the permission system, and researchers to understand and monitor user privacy awareness in a more quantifiable way.

# 1. Introduction

## 1.1 Context

Android, a mobile phone operating system developed by Google, was first released on September 23, 2008 (Morrill, 2008). It soon gained popularity among phone manufacturers due to its open source nature with no licensing fees and minimal cost. The adoption of the Android operating system has skyrocketed as more and more people use mobile phones. As of 2017, out of the 4.43 billion mobile phone users worldwide (eMarketer, n.d.), 73.54% of them are using the Android operating system (StatCounter, n.d.).

This large influx of Android users inevitably leads to a noticeable demand for Android applications. As the primary Android application store, Google Play store has become the largest mobile application store in the world with more than 2.8 million unique applications as of March 2017 (Loesche, 2018).

As mobile application development advances, applications are starting to utilize more and more features available on the phone. For example, a camera application would naturally utilize the camera hardware on the phone to achieve its purpose -- taking photos. On the other hand, malicious applications are also trying to access more and more private information on a phone. For example, a malicious flashlight application

might require the user's contact list and location, which are unnecessary. To limit the features a certain application can access, the Android operating system implements a permission system to regulate application access to the phone hardware and sensitive user information. An application has to clearly state the permissions it requires when being installed. Under this model, applications can only have access to permissions that they require and nothing more ("Permissions overview", 2018).

In order for the permission system to achieve its goal of regulating and restricting application access, applications need to follow the principle of least privilege first proposed by Jerome H. Saltzer and Michael D. Schroeder in 1975 (Saltier, 1975). The idea of the principle of least privilege is simple: each application should only declare permissions that it needs. This way, even if an application is compromised, the least amount of damage would be done, and it is up to the developers of the application to comply with this rule.

### 1.2 Past research

In the past ten years, there have been some published studies analyzing the Android permission system. However, due to its wide range of impacts and its complexity, the current paper is not able to cover all aspects of the subject. In this

section, I will review a few key studies regarding the Android permission system and discuss certain aspects that are worth exploring.

**Most of the current research focuses on the underlying design of the Android permission system** (Barrera, 2010; Felt, 2011; Au, 2012; Backes, 2016). In the study *Android Permissions Demystified*, Adrienne Porter Felt and her team developed the first novel approach of analyzing Android applications and how they interacted with the permission system. In their study, Felt's team used a computer algorithm to look through the underlying code of each application line by line and deduce a list of permissions that were necessary for the program to function. Then the researchers compared this list generated by their custom program and the actual permissions requested by the application which were specified by the developers. By comparing the two lists of permissions, Felt's team discovered a general trend of overprivileged applications where developers asked for permissions that were not being used by the application. This creates potential vulnerabilities for applications to access sensitive user data and may lead to compromised user privacy and security. Felt's team did a subsequent analysis on the reasons for these overprivileged applications and attributed it primarily to the lack of documentation and developers' mistakes. For example, instead of spending the time to make sure that the least amount of permissions are being requested, some developers might be lazy and request all of the permissions available. While applications created this way are still functional, they are more vulnerable to attacks and more likely to leak user information. One limitation of

this study, however, is that the computer software used to deduce the minimum list of permissions is not accurate enough, meaning that false-positives and false-negatives do exist (Felt, 2011).

To improve upon this and get a better idea of how developers adhere to the principle of least privilege, Michael Backes, Kathy Wain Yee Au, David Barrera, and others researchers have all made incremental improvements to the computer software for deducing the list of least permissions (Barrera, 2010; Felt, 2011; Au, 2012; Backes, 2016). These advancements include using dynamic analysis, which is to observe a program while it is running instead of just looking at its source code (Au, 2012) and code review, which is to analyze the source code for the Android operating system to better understand the application (Backes, 2016).

**On another front, researchers have also looked at how end users interact with the permission system and how well the permission system is doing its job of informing the end users.** In *Android Permissions: User Attention, Comprehension, and Behavior*, Adrienne Porter Felt and her team observed participants in a lab installing Android applications and found that only 17% of the participants looked at the permission requests while 42% were completely unaware of permission requests. After demonstrating the lack of end user attention to permission requests, the team went on to give recommendations regarding how the user interface of the permission system could be improved to increase user awareness (Felt, 2012). In 2017, Mohammed M.

Alani conducted a similar experiment to explore the relationship between end users and the permission system. Alani used the approach of a large-scale online survey instead of a small-scale in-person experiment that Felt employed. Alani was able to conclude that, as of 2017, 35.71% of the participants paid attention to permission requests all the time while only 11.40% never read the permissions (Alani, 2017).

Felt's and Alani's studies indicate that there is a noticeable change in the relationship between end users and the Android permission system from 2012 to 2017, yet there's no existing long-term research that has explored this trend of increasing user awareness. In this paper, I seek to fill this gap in the research using a technical approach similar to the Barrera, Felt, Au, and Backes' method. In other words, I would like to approach the topic of user awareness to permission requests using a quantitative analysis and answer the question: **Through a quantitative experimental analysis, how have the permission requests of the top 100 google play store Android applications varied in the United States from 2016 to 2018?**

Past studies implied that there's an increase in user privacy awareness from 2012 to 2017 (Felt, 2012; Alani, 2017). This increase means the end users are less likely to download applications with a lot of permission requests; therefore, the hypothesis is that there is **a negative correlation** between time and the average number of permission requests of the top 100 google play store Android applications in the United States from 2016 to 2018.

### 1.3 Significance

This research holds significance as it could present new insights regarding user awareness of the Android permission system. It serves the purpose of connecting the dots of separate studies (Felt, 2012; Alani, 2017) and revealing the long-term trend that is currently not clear.

From a social aspect, the long-term trend discovered could be used to measure and support current notions regarding the end user privacy awareness, which currently lacks quantifiable indicators. For example, if a decreasing trend of average permission requests is discovered, it could be used to indicate an increase in user privacy awareness in the past three years.

On the other hand, from a technical aspect, the result from the current study can demonstrate how applications on the Android platform have evolved and provide suggestions to developers regarding how an Android application should be designed to best adhere to the end user's demand.

## 2. Method

In order to discover long-term trends, the research queries and analyzes data from existing databases. This method is most appropriate for answering the research question as it provides various data points at different times in the past. To achieve the same amount of data points in a lab setting similar to Felt's and Alani's experiments (Felt, 2012; Alani, 2017), many years of repeated data collection would be required. Although such an approach would yield more accurate and consistent results, the time required for the study exceeds the scope of the current research.

This research uses *42matters*, a for-profit online service, as the primary source of data. This service is selected because no academic databases contain similar information required by the study. The *42matters* service offers a comprehensive list of information regarding the top 100 applications at any given day from Jan 1, 2016 (**Table 1**):

**Table 1**. List of information regarding the top 100 applications from *42matters*

| Name | description |
|---|---|
| package_name | The app package name (unique identifier) |
| title | App title |
| description | Full app description |
| category | The app category (human-readable string) |
| developer | App developer name |
| physical address | Physical address of the developer. |
| **permissions** | Each permission object contains a mapping 'id' => 'permission'. |
| privacy policy | A link to the app privacy policy |

Table taken from https://42matters.com/docs/app-market-data/android/apps/object

Of all the information offered, the researcher only focuses on the "**permissions**" attribute as it is the most relevant to this study. To aid the data extraction process, the research has developed a program in Javascript (see **Appendix A**) to interact with the *42matters* service and download the top 100 charts at 36 unique timestamps. The 36 unique timestamps chosen for this study are the first day of each month from Jan 2016 to Dec 2018. These timestamps are chosen as they are equally distributed throughout the three year period.

After downloading all 36 copies of the top 100 app chart at a given timestamp, the data are then condensed and filtered where only the "**permissions**" attribute remains for each app, and the final data are stored as JSON files for analysis. The

JSON file format is used here due to its compatibility with different programming languages which is proven to be an advantage during the analysis phase.

After obtaining all the data, the research aims to analyze the correlation between time and the number of permission requests. In other words, the researcher seeks to investigate how popular Android applications have evolved from the permission perspective. This goal is achieved through a linear correlation test where the average number of permission requests (inclusive of all default Android permissions and exclusive of custom app permissions) of the top 100 applications at a given time is plotted on a scattergram, and the line of best fit is drawn over the scattergram. The line of best fit, in essence, shows whether a correlation exists between the two quantities plotted on the x-axis (time) and y-axis (number of requests).

Although the graph of average number of permission requests overall does display the large picture with regard to the evolution of the Android applications, it, however, lacks the specificity to answer question such as: Did the request for all types of permission increase/decreased, or what type of permission experience the most amount of changes in the three-year period? To address these questions and arrive at a more comprehensive finding, the researcher also plots the number of permission requests with respect to time for each individual request, such as requests for camera access, location access, and etc. This generates an additional 191 graphs resulting in 192 graphs in total.

Such analysis would necessarily demand a heavy workload if performed manually with software such as Microsoft Excel or Logger Pro; therefore, to save time and encourage repeated iteration, the researcher chooses to create the graphs through programming using Python, Numpy, and Matplotlib (see **Appendix B**). The finished program would digest JSON files generated during the data collection phase to produce all the necessary graphs and calculate the corresponding slopes and r-values. In addition, graphs with an absolute r-value greater than 0.5 and a sufficient amount of data points are filtered out separately to signal the researcher about potential significant results.

## 3. Results and Analysis

The experiments described in the section above are performed and the results will be shared in this section.

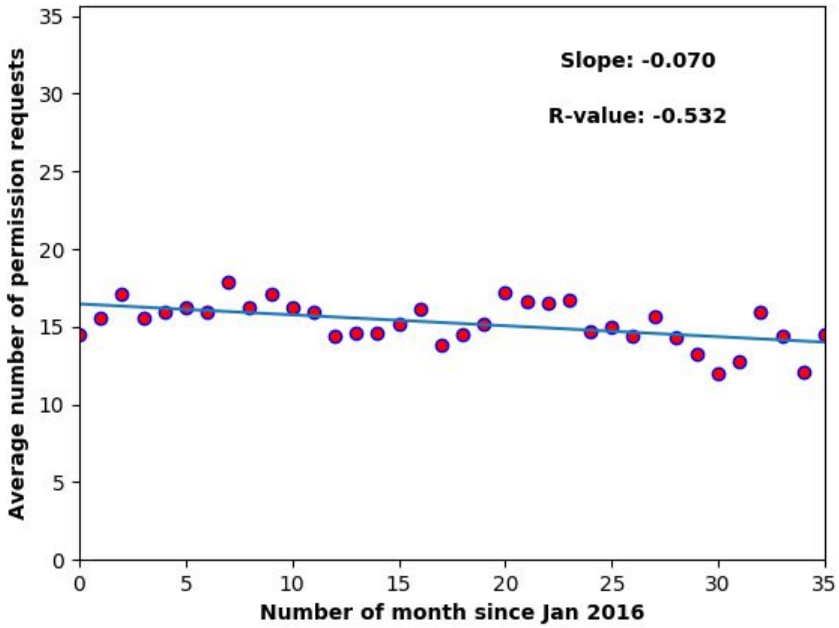



**Figure 1**. Changes of average numbers of permission requests of top 100 applications in the US from Jan 2016 to Dec 2018.

As shown in **Figure 1**, **there is a moderate negative correlation between the time since Jan of 2016 and the average number of permission requests ($p < 0.05$).** This suggests that between 2016 and 2018, popular Android applications on average requested fewer permissions. This could be a result of better developer education and better user privacy awareness in the US (Felt, 2011). However, the research cannot

eliminate the possibility that the downward trend could also be the result of a shift in the type of applications that are popular. For example, a social media app would certainly request more features than a simple mini-game. Such a shift in the type of applications that are popular is not accounted for in the research and future research would be needed.

In addition to the holistic graph, a few insightful graphs of certain permission requests will be presented below.

Supporting the overall trend, 23 individual permissions also showed a noticeable decrease in usage during this time period (**Table 2**, also see corresponding graphs in **Appendix C**):

**Table 2**. List of individual permission requests that exhibit a significant decrease in usage.

| Name | Description |
|---|---|
| BATTERY_STATS | Allows an application to collect battery statistics |
| BROADCAST_STICKY | Allows an application to broadcast sticky intents. |
| CHANGE_NETWORK_STATE | Allows applications to change network connectivity state. |
| CHANGE_WIFI_STATE | Allows applications to change Wi-Fi connectivity state. |
| CLEAR_APP_CACHE | Allows an application to clear the caches of all installed applications on the device. |
| DOWNLOAD_WITHOUT_NOTIFICATION | App can download content without alerting the user. |
| EXPAND_STATUS_BAR | Allows an application to expand or collapse the status bar. |
| FOREGROUND_SERVICE | Allows a regular application to use Service.startForeground. |
| GET_ACCOUNTS | Allows access to the list of accounts in the Accounts Service. |
| GET_PACKAGE_SIZE | Allows an application to find out the space used by any package. |
| GET_TASKS | Allows an application to get information about the currently or recently running tasks. [deprecated] |
| KILL_BACKGROUND_PROCESSES | Allows an application to call ActivityManager.killBackgroundProcesses(String). |
| MODIFY_AUDIO_SETTINGS | Allows an application to modify global audio settings. |
| PACKAGE_USAGE_STATS | Allows an application to collect component usage statistics. |
| READ_CALL_LOG | Allows an application to read the user's call log. |
| READ_PHONE_STATE | Allows read only access to phone state, including the phone number of the device, current cellular network information, the status of any ongoing calls, and a list of any PhoneAccounts registered on the device. |
| READ_SMS | Allows an application to read SMS messages. |
| RECORD_AUDIO | Allows an application to record audio. |
| USE_CREDENTIALS | Allows an application to request authentication tokens. |

| | |
|---|---|
| | [deprecated] |
| WAKE_LOCK | Allows using PowerManager WakeLocks to keep processor from sleeping or screen from dimming. |
| WRITE_CONTACTS | Allows an application to write the user's contacts data. |
| WRITE_EXTERNAL_STORAGE | Allows an application to write to external storage. |
| WRITE_SETTINGS | Allows an application to read or write the system settings. |

Content retrieved from https://developer.android.com/reference/android/Manifest.permission.html

Permissions in applications, such as “READ_SMS”, “RECORD_AUDIO”, and “READ_CALL_LOG” are clearly linked to user privacy suggesting how **user privacy awareness could be playing a role in the downward trend observed.**

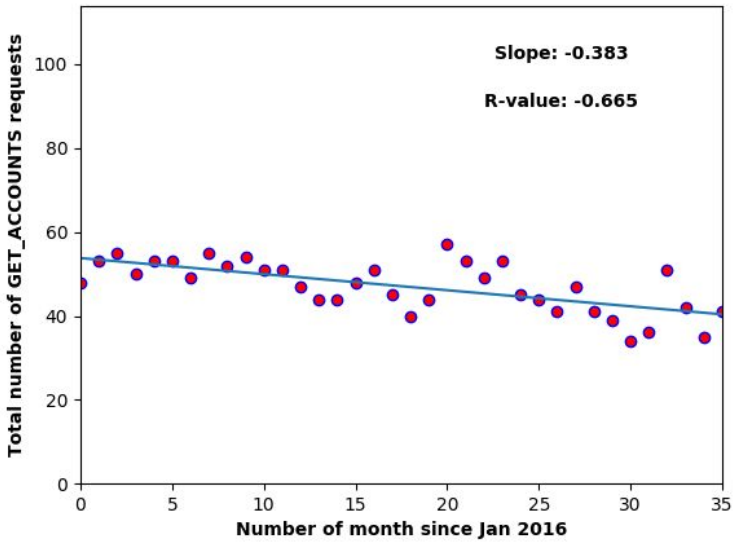



**Figure 2.** Significant decrease of GET_ACCOUNTS permission requests from Jan 2016 to Dec 2018.

Furthermore, as shown in **Figure 2**, the single permission request that experienced the largest decrease in usage is the GET_ACCOUNTS permission. It is

noted that some permissions do have a steeper downward slope, but they don't have an r-value greater than 0.5 to suggest at least a moderate correlation. The GET_ACCOUNTS permission request "allows [an application] access to the list of accounts in the Accounts Service." ("Manifest.permission," n.d.). In other words, this permission let an app see if the user has logged into different accounts such as Google or Facebook. Although its decrease in usage could be a result of increasing user privacy awareness, it is more likely the result of the evolution of the Android permission system. According to the official Android developer documentation, "beginning with Android 6.0 (API level 23), if an app shares the signature of the authenticator that manages an account, it does not need "GET_ACCOUNTS" permission to read information about that account." ("Manifest.permission," n.d.) What this means is that apps such as Facebook would no longer need to request this permission in order to see if the user has logged into its own service, in this case, Facebook. This would most likely explain the downward trend as Android phone users upgrade their phones to the newer versions. Similarly, permissions that experienced a decrease in usage such as "USE_CREDENTIALS" and "GET_TASKS" are also deprecated by the Android permission system enforcing the idea that **the evolution of the system itself plays a role in the downward trend.**

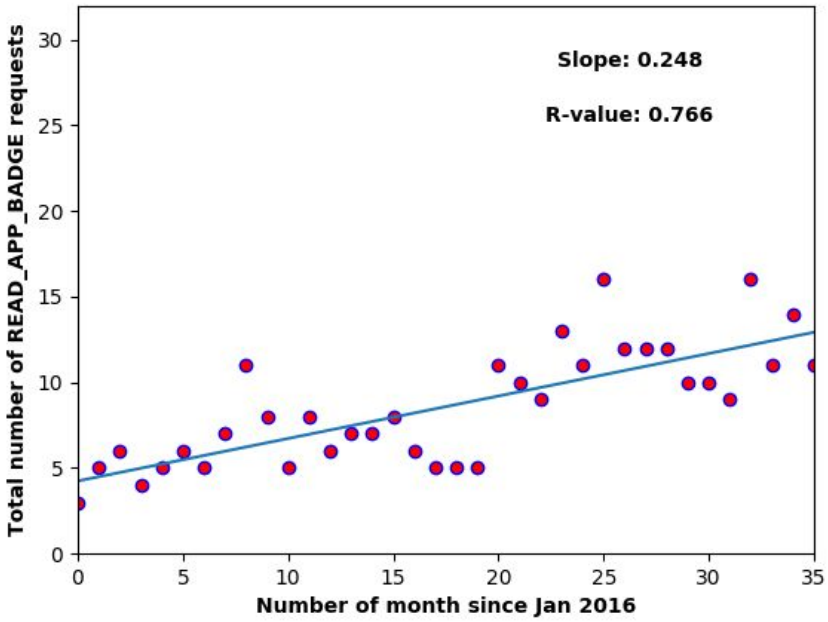



**Figure 3.** Significant increase of READ_APP_BADGE permission requests from Jan 2016 to Dec 2018.

The READ_APP_BADGE permission request, on the other hand, is the only request that suggests a statistically significant increase in usage from 2016 to 2018 (**Figure 3**). This permission allows developers to display a notification badge on the corner of the app icon. This permission is not a feature of the official Android permission system instead device manufacturers (like Xiaomi) or third-party developers (like Nova Launcher) implement it and allow app developers to use (“Badge on App icon,” 2016). **The finding here could indicate a larger adoption of such standard and a demand from users to see such a feature being implemented.**

## 4. Discussion

In summary, the current study discovered an overall negative correlation between the number of permission requests and time in the period from 2016 to 2018. By examining the trends of individual permission requests, I am able to identify a few potential causes that may explain the trend including 1. **user privacy awareness**; 2. **the evolution of the Android permission system**; and 3. **end user feature demand**. In this section, each of the three potential factors would be explored and discussed.

The idea that user privacy awareness would play a role in this research is first addressed by Felt's and Alani's study which indicates that there is a noticeable change in the relationship between end users and the Android permission system from 2012 to 2017 (Felt, 2012; Alani, 2017). This notion of increasing privacy awareness is further supported by other research. A research conducted by Campbell at Miami University finds an increase in awareness of privacy issues in college students from 1999 to 2000 (Campbell, 2001), and the 2017-2018 UK Information Commissioner's Annual Report has stated a "significant increase" in data protection complaints (up 15%), self-reported breaches (up 30%) and freedom of information complaints (up 5%) (Ashford, 2018; "Information Commissioner's Annual Report", 2018). All of these studies agree that user privacy awareness is indeed prevalent and support the hypothesis that user privacy awareness plays a role in affecting the number of permission requests. Although the study is not able to prove direct causation between user privacy awareness and

permission requests, it does, however, raise the possibility of such a relationship, which would prompt future research in this direction. If such relationship is to be identified, the average number of permission requests of the top 100 Android applications measured in this research can become a more generalized indicator for quantifying user privacy awareness, which up to this point cannot be measured quantitatively.

Similarly, there have been researches that suggest the evolution of the Android permission system would play a role in affecting the number of permission requests (Felt, 2011; Backes, 2016). The reasoning for this notion is simple: as the permission system evolves, it would likely make changes to the permissions that are available and regroup them in different ways. Just as a hypothetical example, the system might discover that application that needs access to the camera usually needs access to past photos as well, and applications do not usually request access to past photos without requesting access to the camera. In this case, the permission system might decide to deprecate the two individual permission requests and create a new permission request that will grant access to both the camera and past photos. If this notion is correct, it could mean that the popular applications may not be requesting fewer features; instead, the same number of features are being bundled into fewer permissions by the system. Therefore, the downward slope discovered could also imply a larger change in the design of the permission system and the way it bundles features into permissions.

In addition, end user demand might also play a part in effecting the trend. As shown in the results section, the READ_APP_BADGE permission is the only one to experience an increase in usage from 2016 to 2018, and it is primarily fueled by end users demanding the feature from applications. As previously mentioned, this permission is not defined by the official Android standard; instead, it's only supported by third-party developers who realize the demand for such a feature. This connects with the evolution of the Android permission system as mentioned where the direction in which the platform will evolve would very much be influenced by the demand from end users. This notion of end user demand, however, does come in conflict with the notion of increasing user privacy to a certain extent as more features access can lead to a higher risk of application abusing its permissions and harming the end user's privacy. The downward trend discovered might also be viewed as an indicator that user privacy might be playing a larger role in impacting the overall trend compared to the demand for new features. This is also reflected in the latest version of Android - Android pie where privacy and security are main features being advertised. On the official Android website, it states "Android 9 safeguards privacy in a number of new ways. Now, Android will restrict access to your phone's microphone, camera, or other sensors when an app is idle or running in the background. (If an app does need to access a sensor, it will show a persistent notification on your phone.) Android 9 also brings important improvements that protect all web communications and offer private web surfing." ("Android 9 Pie", n.d.) This suggests that there is a heavy focus on privacy compared to new features, which echoes with the findings in this study.

## 5. Conclusion

In conclusion, this study has bridged the gap in previous research and showed how the average number of permissions requests of the top 100 google play store applications has changed from 2016 to 2018. The study finds a negative correlation between time and the number of permission requests matching the initial hypothesis. To further investigate the driving factors behind the decreasing trend, individual types of permissions are being analyzed and three primary factors are identified: user privacy awareness, the evolution of the Android permission system, and the end user demand for new features.

While the study did reveal the general trend of the number of permission requests, it still has a lot of limitations. For example, the study is not able to clearly identify the root cause of such trend nor is it able to gather more data for a longer period of time. Therefore, future research should focus on gathering more data to see if such a decreasing trend holds true for a longer period of time, and perhaps conduct lab experiments with participants to obtain data in a more standard and controlled fashion. The three driving factors identified in this study should also be investigated individually.

Overall, this study is able to shed light on a topic regarding the Android permission system that has not been previously explored and offer new insights to the

scientific community such as the decreasing trend of permission requests, and its potential driving factors. Such research would be valuable for researchers trying to understand the evolution of the Android permission system. More specifically, how application developers interact with the system. In addition, application developers can also find trend through this research to see how popular applications have changed from 2016 to 2018, which would help them to develop better applications for the end users. Last but not least, end users can understand through this study that factors such as user privacy awareness and end user feature demand do play a part in shaping the Android permission system and Android ecosystem at large. This study serves as the first step toward a deeper understanding regarding the long-term trend of permission request changes of Android applications.

**6. Work cited**

Alani, M. M. (2017). Android Users Privacy Awareness Survey. International Journal of Interactive Mobile Technologies (iJIM), 11(3), 130. doi:10.3991/ijim.v11i3.6605

Android 9 Pie. (n.d.). Retrieved March 19, 2019, from https://www.android.com/versions/pie-9-0/

Ashford, W. (2018, July 20). Uptick in UK privacy awareness, says ICO. Retrieved from https://www.computerweekly.com/news/252445266/Uptick-in-UK-privacy-awareness-says-ICO

Au, K. W., Zhou, Y. F., Huang, Z., & Lie, D. (2012). PScout. Proceedings of the 2012 ACM Conference on Computer and Communications Security - CCS 12. doi:10.1145/2382196.2382222

Backes, M., Bugiel, S., Derr, E., McDaniel, P., Octeau, D., & Weisgerber, S. (2016). On Demystifying the Android Application Framework: Re-Visiting Android Permission Specification Analysis. Usenix.

Badge on App icon. (2016, November). Retrieved March 18, 2019, from https://stackoverflow.com/questions/40148979/badge-on-app-icon

Barrera, D., Kayacik, H. G., Oorschot, P. C., & Somayaji, A. (2010). A methodology for empirical analysis of permission-based security models and its application to android. Proceedings of the 17th ACM Conference on Computer and Communications Security - CCS 10. doi:10.1145/1866307.1866317

Campbell, J., Sherman, R. C., Kraan, E., & Birchmeier, Z. (2001). Internet Privacy Awareness and Concerns among College Students. *APS Annual Convention, Toronto.*

eMarketer. (n.d.). Number of mobile phone users worldwide from 2015 to 2020 (in billions). In Statista - The Statistics Portal. Retrieved December 3, 2018, from https://www.statista.com/statistics/274774/forecast-of-mobile-phone-users-worldwide/.

Felt, A. P., Chin, E., Hanna, S., Song, D., & Wagner, D. (2011). Android permissions demystified. Proceedings of the 18th ACM Conference on Computer and Communications Security - CCS 11. doi:10.1145/2046707.2046779

Felt, A. P., Ha, E., Egelman, S., Haney, A., Chin, E., & Wagner, D. (2012). Android permissions. Proceedings of the Eighth Symposium on Usable Privacy and

Security - SOUPS 12. doi:10.1145/2335356.2335360

*Information Commissioner's Annual Report and Financial Statements 2017-18*. (2018). UK Information Commissioner's Office.

Loesche, D. (2018, January 09). Infographic: The Biggest App Stores. Retrieved from https://www.statista.com/chart/12455/number-of-apps-available-in-leading-app-stores/.

Manifest.permission. (n.d.). Retrieved from https://developer.android.com/reference/android/Manifest.permission.html

Morrill, D. (2008, September 23). Announcing the Android 1.0 SDK, release 1. Retrieved from https://android-developers.googleblog.com/2008/09/announcing-android-10-sdk-release-1.html

Permissions overview | Android Developers. (2018, November 20). Retrieved from https://developer.android.com/guide/topics/permissions/overview

Saltier, J. H., & Schroeder, M. P. (1975). Protection of information in computer systems. *IEEE CSIT Newsletter, 3*(12), 19-19. doi:10.1109/csit.1975.6498831

StatCounter. (2019). Mobile operating systems' market share worldwide from January 2012 to December 2017. In Statista - The Statistics Portal. Retrieved December 3, 2018, from https://www.statista.com/statistics/272698/global-market-share-held-by-mobile-operating-systems-since-2009/.

## 7. Appendix A

### Javascript program used to extract data from *42matters*

```
const fs = require('fs');
const axios = require('axios');

const country = 'US';
const basePath = `./data/${country}/`;
const access_token = ‘TOKEN’;

(async () => {
  for (let year = 2016; year <= 2018; year++) {
    for (let month = 1; month <= 12; month++) {
      const date = `01-${(month+'').padStart(2,'0')}-${year}`;
      console.log(date);
      const r = await
axios.get('https://data.42matters.com/api/v2.0/android/apps/top_google_charts.json', {
        params: {
          list_name: 'topselling_free',
          cat_key: 'OVERALL',
          limit: '100',
          country,
          access_token,
          date,
        },
      });
      fs.writeFileSync(basePath+date+'.json', JSON.stringify(r.data));
    }
  }
})();
```

## 8. Appendix B

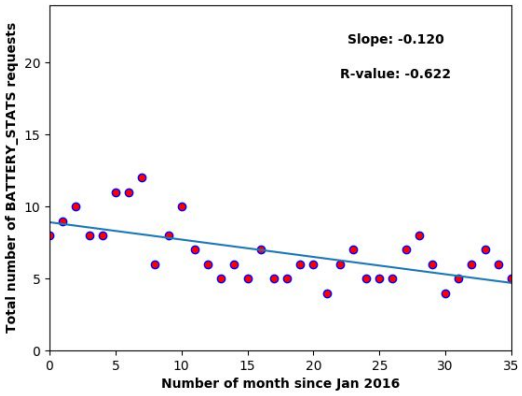
### Python program used to generate graphs

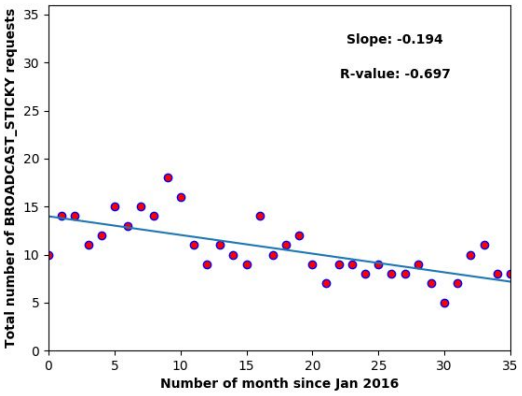
```
import matplotlib.pyplot as plt
import numpy as np

def graphLine(points, titles=('x', 'y'), filename=None):
  x, y = points
  xt, yt = titles
  fig, ax = plt.subplots()

  ax.scatter(x=x, y=y, marker='o', c='r', edgecolor='b')

  # ax.set_title('Scatter: $x$ versus $y$')
  ax.set_xlabel(xt, weight = 'bold')
  ax.set_ylabel(yt, weight = 'bold')

  ax.set_xlim(left=0, right=max(x))
  ax.set_ylim(bottom=0, top=max(y)*2)


  s = np.polyfit(x, y, 1)
  hy = s[0]*x+s[1]
  r = np.corrcoef(x, y)[0,1]
  ax.plot(x, hy)
  ax.text(0.75,0.9,'Slope: {:.3f}'.format(s[0]), horizontalalignment='center',
verticalalignment='center', transform=ax.transAxes, weight = 'bold')
  ax.text(0.75,0.8,'R-value: {:.3f}'.format(r), horizontalalignment='center',
verticalalignment='center', transform=ax.transAxes, weight = 'bold')

  if filename is not None and abs(r) > 0.5:
    fig.savefig(filename)
  else:
    fig.show()

  plt.close(fig)

x = np.array(list(range(36)))
y = [14.51, 15.57, 17.12, 15.57, 15.99, 16.26, 15.9, 17.84, 16.28, 17.06, 16.25, 15.94,
14.42, 14.55, 14.59, 15.13, 16.15, 13.85, 14.48, 15.21, 17.17, 16.65, 16.49, 16.76,
14.65, 14.98, 14.37, 15.63, 14.26, 13.22, 12, 12.81, 15.92, 14.39, 12.08, 14.53]
```

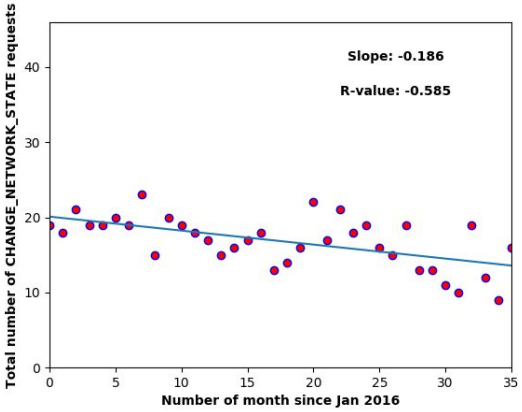
```
graphLine((x, y), ('Number of month since Jan 2016', 'Average number of permission
requests'), './useful_graph/_main.png')

with open('./final_permission_data') as f:
  data = map(lambda x: x.split(': '), f.read().strip().split('\n'))
  data = map(lambda x: (x[0], [int(e) for e in x[1].split(',')]), data)

for name, values in data:
  if max(values) > 10:
    graphLine((x, values), ('Number of month since Jan 2016', 'Total number of {}
requests'.format(name)), './useful_graph/{}.png'.format(name))
    print '{} graph done'.format(name)
```

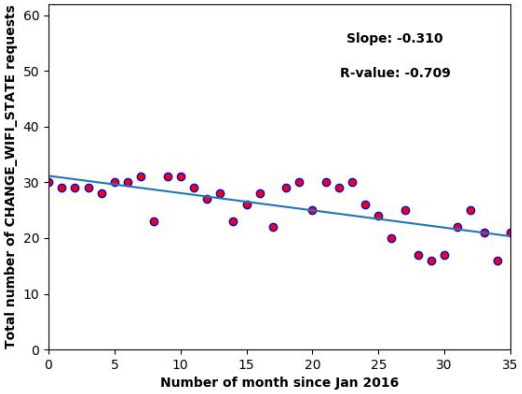
## 9. Appendix C

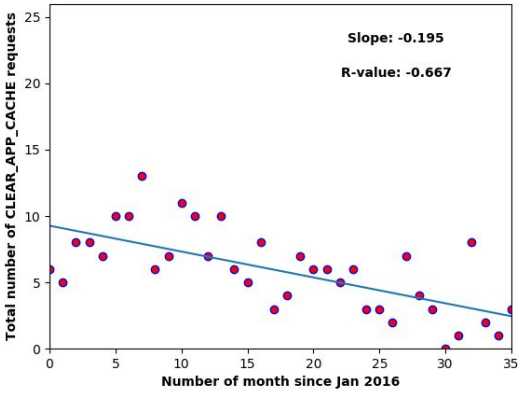
## 23 individual permission requests demonstrating a significant decrease in usage from 2016 to 2018

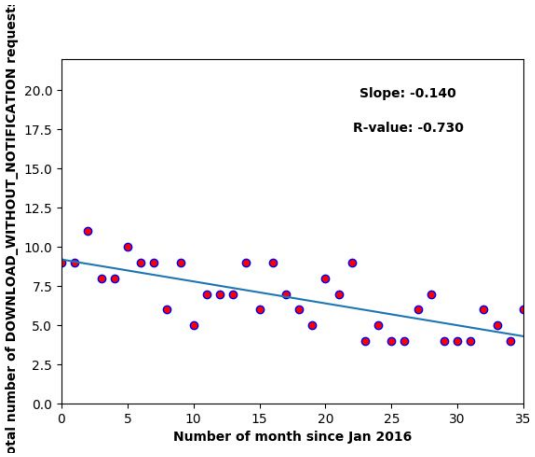


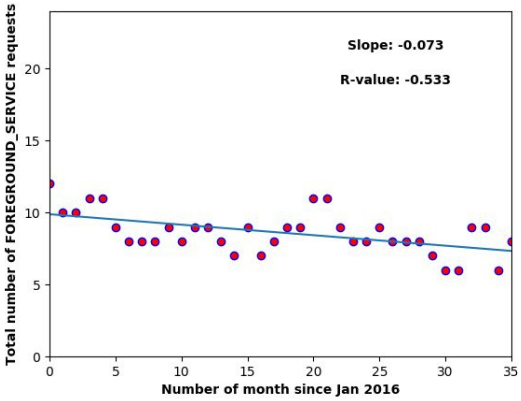



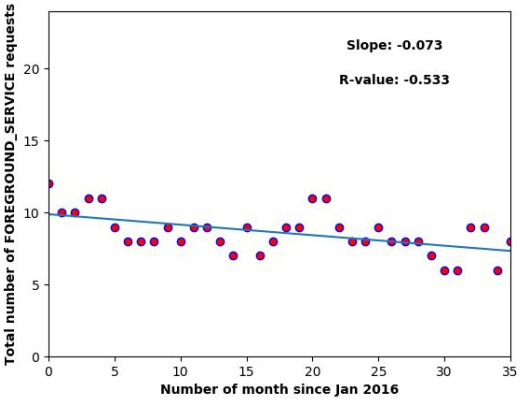



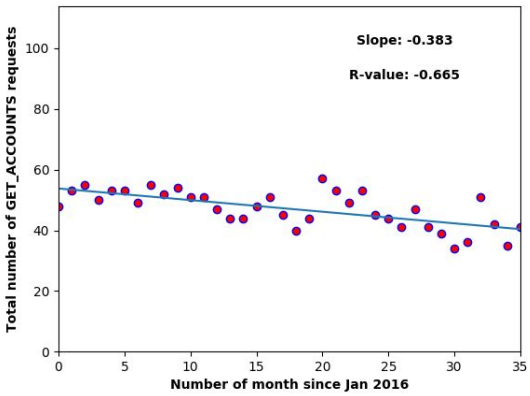



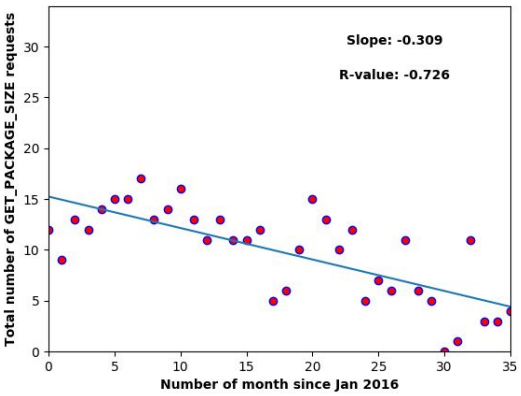



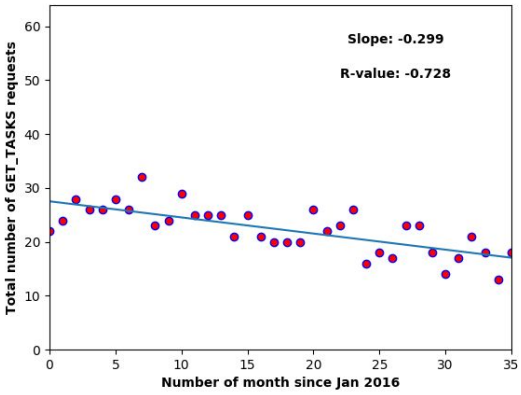



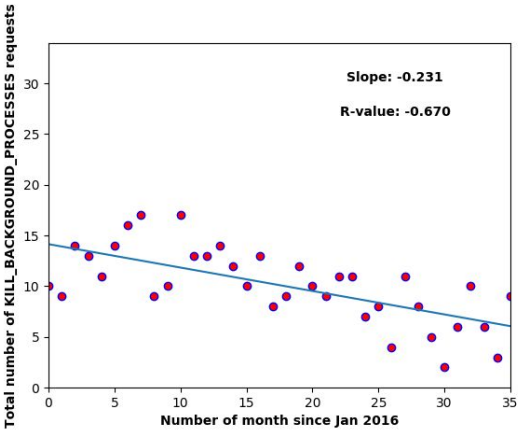


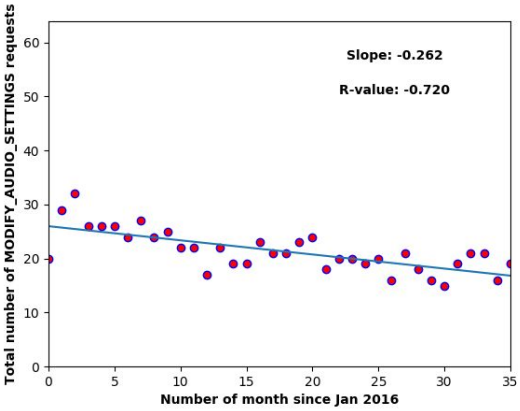



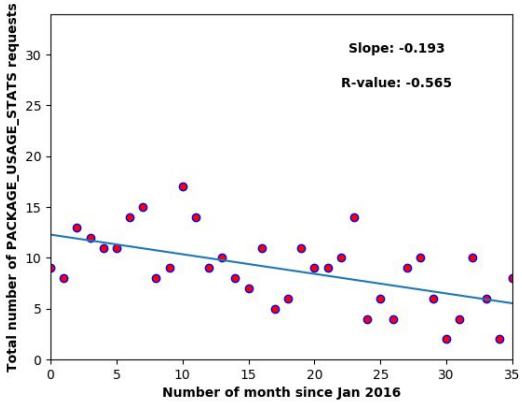



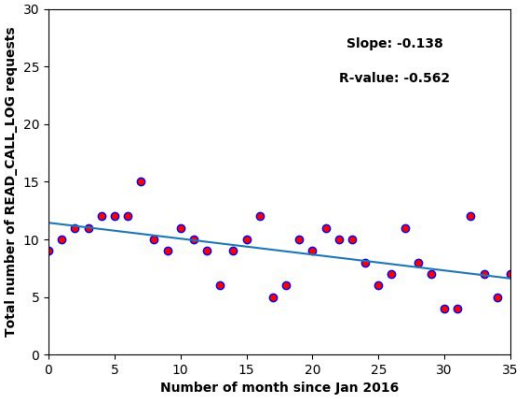



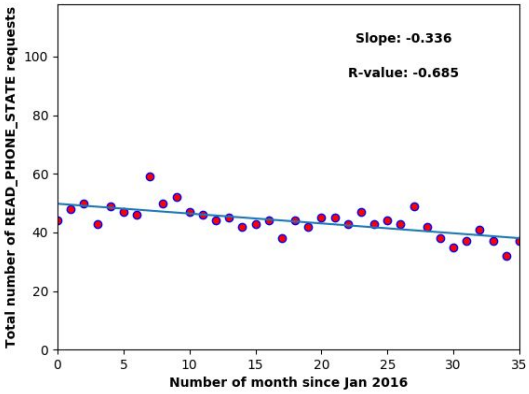



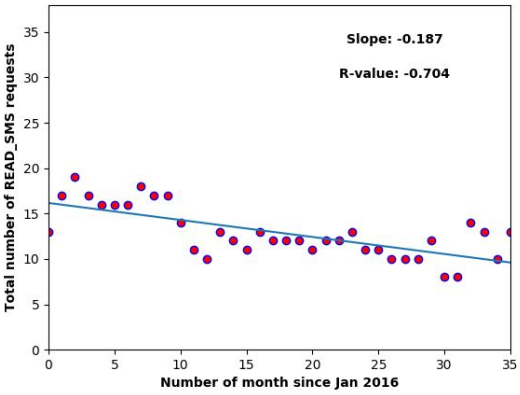



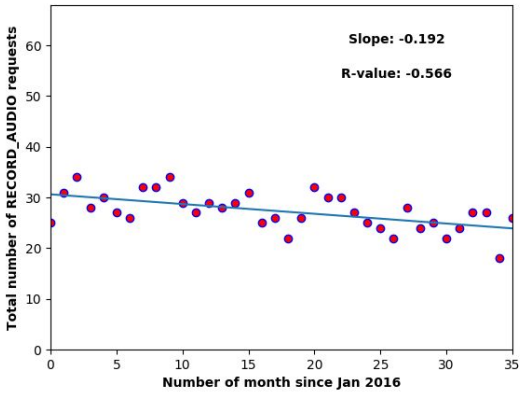


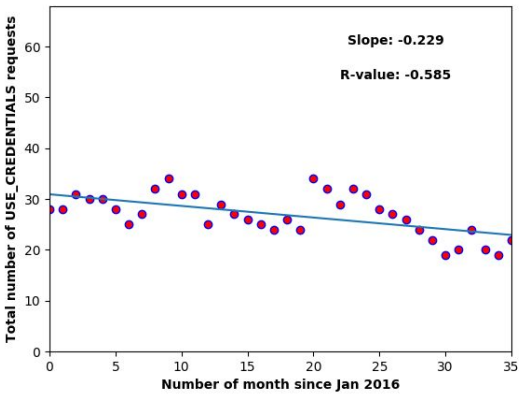



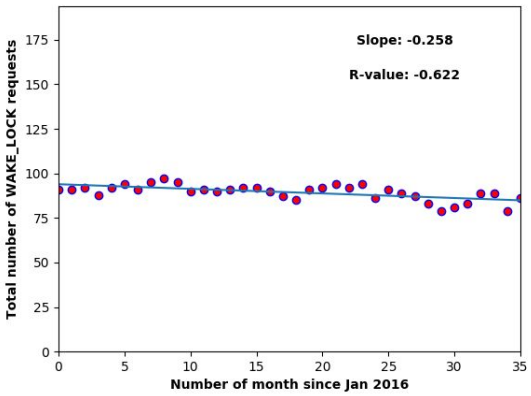



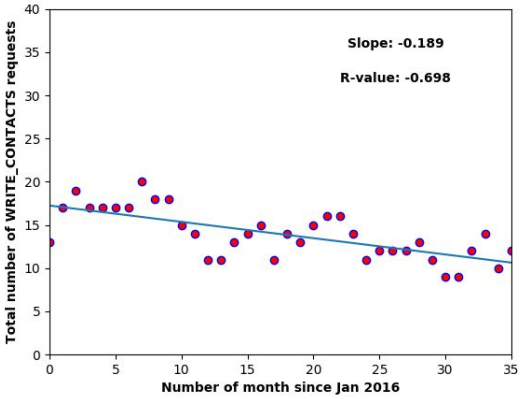



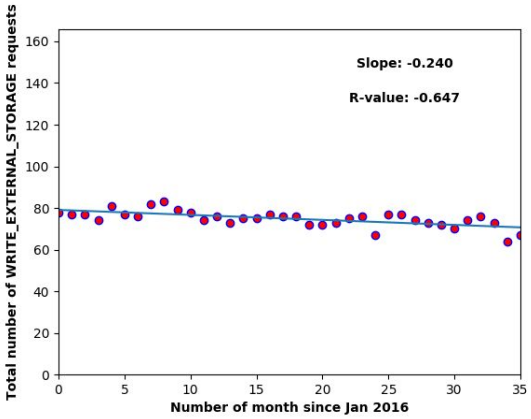



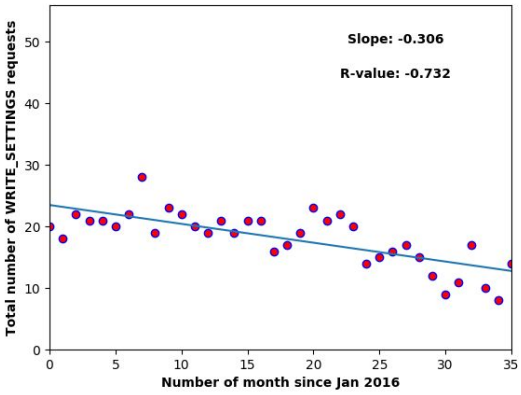





# AP® RESEARCH
# 2019 SCORING COMMENTARY

## Academic Paper

**Note:** Student samples are quoted verbatim and may contain spelling and grammatical errors.

**Sample: C**
**Score: 4**

This paper earned a score of 4 because the paper clearly identifies a gap that is situated in the literature, fully explicates the methods used, and identifies the limitations of the study. On page 9 the paper discusses previous literature and identifies a clear gap with a focused question. On page 10 the student explicitly states their hypothesis and discuss the significance of the research question. On pages 9 and 11 the student defends their method. The description of the method makes it replicable; pages 7–13 explain the process for gathering news stories, analyzing the sentiment of each article using TextBlob, and rationale for the use of a classification algorithm. On page 21, the student presents a new understanding: "In summary, the current study discovered…"

This paper did not earn a score of 3 because the student explicitly connects their topic within the scholarly conversation; for example (page 7), the student's discussion of the scholarly literature on Android permissions is both sophisticated and detailed. The paper logically defends their method, and the conclusion is supported by the evidence.

This paper did not earn a score of 5 because the student does not provide a substantive analysis of their limitations. The implications are present but underdeveloped. The writing is competent but not elegant (e.g., there are awkward sentence structures and changes of tense within a single paragraph). The new understanding is supported but not well-justified. The paper does not discuss, for example, the different ways the negative associations could be interpreted and how to distinguish in future research between those possibilities (e.g., user concern about permissions vs. app developers finding ways to have fewer permissions). Weak limitations are consistent with the score of a 4. On page 24, the paper notes, "it still has a lot of limitations" and then offers only one sentence. These held the score to a 4 where implications are stated versus explained.