Android Battery Saver

Dr. Devi Kannan[1], Shyamal Tripathi[2], Vikash Kumar[3], Vishwajeet Singh[4]
[1, 2, 3, 4] Dept of Computer Science and Engineering
[1, 2, 3, 4] Atria Institute of Technology

Abstract- *Over the last two decades, the evolution of mobile technologies has led to an unprecedented adoption of cellphones in mainstream society. Consumers have moved from having the ability to make telephone calls from anywhere at any time using traditional cellphones to having an all-access pass to cyberspace using today's smartphones. This portable device has become a reliable support system for the user because of the accessibility and flexibility it offers to maintain the users' daily routine. Although the functionality of the smartphone is ever-increasing, AI has not advanced enough to tailor its functionality and adapt to the user's unique requirements automatically. In addition, developments in portable devices are dependent upon advances in portable battery technologies which are still lagging. To address this issue, portable electronic system designers have been implementing hardware and software-based power optimization techniques to improve the battery efficiency of smartphones. The authors in this chapter have chosen software optimization techniques to increase battery efficiency because these techniques are more robust. This chapter introduces a novel idea of an automated system for smartphones that prioritize application access based on the owner's usage patterns and daily routine to conserve battery life. This system will serve two purposes: save battery power and improve the smartphone's artificial intelligence.*

Keywords- Smartphones, AI, Battery technology.

I. INTRODUCTION

With the advancement of smartphones' technology functionality has been increasing. Smartphone battery life and its longevity have been a challenge. These smartphones are featured with a large screen, multiple core processors, RAMs comparable to computers, ROMs in the scale of Gigabytes, and a variety of sensors. As the time is now for versatile usage of smartphones, the power consumption of this portable device becomes one of the most important issues. If youknow that, where your phone's battery consume, then you can get a solution to solve these types of problems. The applications start automatically due to two reasons as it can be seen in the permission of the applications as for the reason of network access many applications report constant usage and error so they are run in the background for the developer to improve the features and functions of the application shortly.

Most of the end-users have difficulty determining which applications are more energy-efficient application designers have an incentive to develop energy-efficient smartphones software. Their main barrier is the difficulty of determining the impact of software design decisions on system energy consumption, but that barrier can be overcome. We are also providing a real power estimation method called Battery Viewer that uses as a function to determine system-level power consumption. We have noticed that the different phones have different power consumption. For such mobile devices, Android1 is a very popular open-source mobile platform and has dominated the smartphone market. As the functionality of these Android applications becomes more and more powerful, their power consumption increases too. Battery usage has become a very important quality metric for Androidapps.

Thus, power management optimization is an important goal for Android app developers. Some researchers propose useful tools to help developers to gain insights into the energy usage patterns of their applications. Additionally, many energy- efficient techniques have been proposed and they work at the operating system level and application level.

II. RELATED WORK

Consumers have transitioned from the ability to make telephonic calls from anywhere to anywhere using traditional and old technology to today's cellphones and smartphones. These easily accessible and portable devices have become a reliable system for the users because of the flexibility and less complexity they offer to the users' daily routine [1].

Improving the synchronization of smartphones by using Cloud computing and virtualization techniques to shift the workload from merely a smartphone to a resource-rich computational Cloud environment. The data will be automatically synchronized to the user's devices when they are connected to the internet [2].

Android devices are being activated per day and power management of these devices is becoming an issue. The one problem is common that is Low battery life of the device. It is not a common thing in smartphones. In this approach, we do not require client-server architecture. We will include a learning engine. This learning will monitor the user behaviorin

terms of apps used, battery consumption, and contexts, and then we will collect the information for some time and then fed it to a learning engine [3].

An empirical study of power management commits in Android applications. Our study extends that of Moura et al. who perform an empirical study on energy-aware commits; however they do not focus on Android applications and only a few of the commits that they study come from Android applications [3].

With mobile phones becoming a basic and important part of life in all ways, the recharging of mobile phone batteries has always been a problem. We propose a device where all the users get to see a clear example of automation. The device uses an Android application for monitoring the battery percentage and to notify the ESP [4].

This paper aims at analyzing and evaluating the consumption of energy of Android power-saving applications; however, the study has been done in a qualitative manner. The study highlights an issue that the notifications regarding the power saving shown on the screen seem to exploit a lot of battery. Therefore, the study has been done to reflect the ways that could help the users to save the phone battery without using any power from the same battery in an efficient manner [5].

III. OBJECTIVE

The developed application objective is to

1) To enlist the apps that are consuming power and their consumption rate.
2) To help handle these battery consuming apps by performing certain actions.
3) To improve the battery life of the android device.

IV. METHODOLOGY

We had used battery management feature, App Standby Buckets. App Standby Buckets helps the system prioritize apps' requests for resources based on how recently and how frequently the apps are used. Based on the app usage patterns, each app is placed in one of five priority buckets. The system limits the device resources available to each app based on which bucket the app is in.

The five buckets prioritize apps into groups by the following characteristics:

- Active
- Working Set
- Frequent
- Rare
- Never

1. Active

An app is in the active bucket if the user is currently using the app, for example:

The app has launched an activity
The app is running a foreground service
The app has a sync adapter associated with a content provider used by a foreground app
The user clicks on a notification from the app
If an app is in the active bucket, the system does not place any restrictions on the app's jobs, alarms, or FCM messages.

2. Working set

An app is in the working set bucket if it runs often but it is not currently active. For example, a social media app that the user launches most days is likely to be in the working set. Apps are also promoted to the working set bucket if they're used indirectly.

If an app is in the working set, the system imposes mild restrictions on its ability to run jobs and trigger alarms.

3. Frequent

An app is in the frequent bucket if it is used regularly, but not necessarily every day. For example, a workout-tracking app that the user runs at the gym might be in the frequent bucket.

If an app is in the frequent bucket, the system imposes stronger restrictions on its ability to run jobs and trigger alarms, and also imposes a cap on high-priority FCM messages.

4. Rare

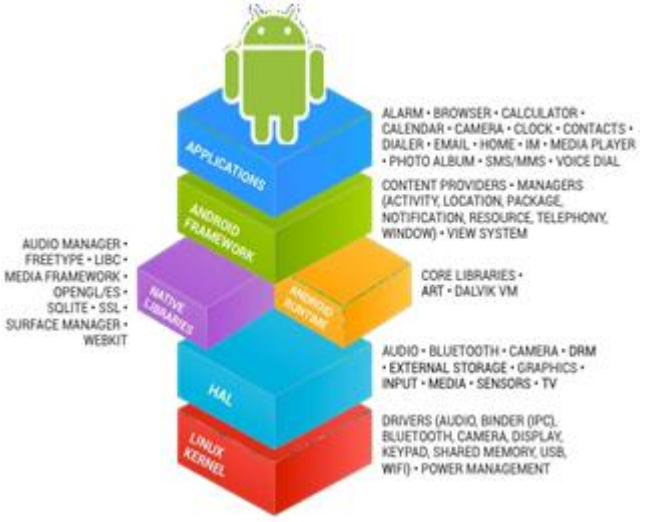
An app is in the rare bucket if it is not often used. For example, a hotel app that the user only runs while they're staying at that hotel might be in the rare bucket.

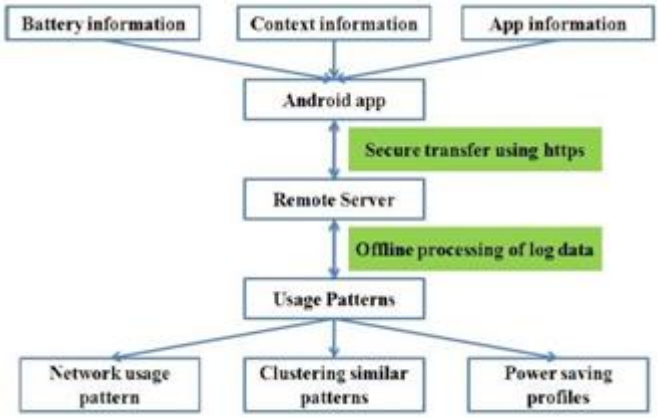
If an app is in the rare bucket, the system imposes strict restrictions on its ability to run jobs, trigger alarms, and receive high-priority FCM messages. The system also limits the app's ability to connect to the internet.

5. Never

Never Apps that have been installed but never run are assigned to the never bucket. The system imposes severe restrictions on these apps.

V. VISUALISATION





VI. CONCLUSION

The application developed considers all the information provided to optimize the power of the battery and to activate the power saving mode to reduce loss and increase power duration. The power duration of any battery is proportional to the device configurations and background running applications. Their uses, running cycle, application authorization, interlinking and screen action time.

REFERENCES

[1] Guo Y, Wang C, Chen X. Understanding Application-Battery Interactions on Smartphones: A Large-Scale Empirical Study. IEEE Access. 2017;5:13387-400.

[2] Zhang, Y., Huang, G., Liu, X., Zhang, W., Mei, H., and Yang, S. Refactoring android java code for on-demand computation o_oading. In ACM SIGPLAN Notices (2012), vol. 47, ACM, pp.233{248.

[3] Navjeet Kumar, Dorathy R., Shruthi M., Dr. Anusaya S. "IoT based Smart Charger: An ESP8266 based Automatic Charger" International Journal of Pure and Applied Mathematics Vol.115, No.8, 2017,565-570.

[4] Pathak, A., Hu, Y.C. and Zhang, M. (2012). Where is the energy spent inside my app? Proceedings of the 7th ACM european conference on Computer Systems - EuroSys'12.