Optimizing Battery Life in Android Applications with AI Driven Energy Management

Pankaj Chavada, Mohib Thasariya, Niraj Dineshkumar Bhagchandani

1. Student, Department of Information Technology, Atmiya University, Rajkot, Gujarat, India
2. Student, Department of Information Technology, Atmiya University, Rajkot, Gujarat, India
3. Assistant Professor, Department of Information Technology, Atmiya University, Rajkot, Gujarat, India

Abstract - With apps using up to 45% of battery life on new devices, phone users are very concerned about battery drain. We require intelligent energy management systems as Android phones become more sophisticated and powerful. Clever power management tools made possible by recent advances in AI can significantly extend battery life without reducing the phone's usability. Google's addition of Adaptive Battery to Android P revolutionized power conservation by extending battery life by up to 30%. This study examines the models, techniques, and approaches used to integrate AI-powered energy management into Android applications. It provides developers with helpful advice on how to create apps that consume less power.

Index Terms - Adaptive battery, CRAM, power management, real-time monitoring, AI in mobile computing, resource allocation, artificial intelligence, machine learning, energy-efficient mobile apps, and Android battery optimization.

I. INTRODUCTION

Smartphones have merged into our lives as a necessity rather than just devices for communication; they are multi-functional gadgets we depend on for almost everything, from organizing our schedules and finances and tracking our health to navigation, entertainment, and information, thus defining our day-to-day operations. However, while the functionality that they offer continues to increase, the energy demand also increases as people need more from these gadgets. Modern Android apps are progressively rich in functionality, but at the same time, consuming more power to run all these features and hence the price. It is indeed true that high-performance processors, good displays, background activity, GPS and map data access, and online/offline synchronization surely do take their toll on whatever little battery is provided by most modern phones. Although the technology in batteries has certainly improved, most people still have to fight just to keep their phones running all day long. This leads us to one of the most common annoyances: having a great device that always needs charging. Adding to the mix is the fact that conventional battery-saving techniques are simply not always sufficient. Although they can be of help at times, they predominantly follow hard-coded rules and do not lend themselves well to varying phone usage patterns between different users. Thus, while one user may be all-day video-streaming, another may mostly require it for navigation or social media applications. This is where AI and ML come in Introducing a new perspective. Rather than blanket energy-saving rules for all, AI can ascertain individual energy interaction patterns with their applications and then optimize energy use based on that. For instance, if AI learns that a user usually checks emails only in the morning, it may not sync much at other times thereby lowering battery consumption. It may also learn to identify apps that are senselessly draining power and intervene before the user even ‘feels’ the problem. This research studies how we can leverage AI to build smarter energy management systems for our Android applications — systems that learn and adapt live to help to continue extending the battery life, all without compromising performance or user experience. The idea is a candidate for a future where phones last longer, work better and adjust intelligently to each user’s individual habits.

II. METHODOLOGY

This research in Android applications follows well-defined procedures to enhance battery life. The methodology undertaken identifies portions of an application that are power-intensive and makes attempts to optimize them. This will be laid out in the following steps:

A. Research Planning

Battery optimization strategies are the focal point of this study which start with extensive reviewing. Studing existing literature crafted handy step-by-step solutions to the problems else where. For addressing the research gap, this study developed specific objectives which formed the scope of this research.

B. Tool Selection and Setup

Android Studio was selected as the Integrated Development Environment (IDE) for building and testing the application due to its comprehensive features. Its associated programming language Kotlin was the primary choice because it integrates seamlessly with Android APIs. For the purpose of using Android Studio, multiple was set, Android Profiler for monitoring active processes within the app in real time, Battery Historian for analyzing recorded usage of the battery over time, and PowerProfile APIs for providing detailed energy information at the level of the OS.

C. Application Development

An Android application example simulating real-world usage has been created. The app was made of background services, usage of sensors, and regular data synchronization, all of these being the components capable of very heavy battery consumption. This application was used as a testbed to get the power needs in different erynergy levels.

D. Profiling and Monitoring

As the app was operational, the logic built-in inside was executed to monitor the resource use by the app. This included checking the tasks done by the central processing unit, what area of memory was accessed, what was the network traffic, and what was the state of wake lock in the device. This knowledge helped detect exactly what functions of the app were involved in the power drain.

E. Applying Optimization Strategies

The data obtained from the profiling of the app was the basis for the energy-saving optimizations. Among them, of the utmost importance was to change the frequency of background services, re-optimizing the most power-consuming components of the system, and to improve wakelocks control in such a way that sleep would be allowed when the device is not working. It was endeavored the app be no less power-efficient while still being of use.

F. Evaluation and Analysis

The app was retested after the settings have been changed to the most energy-efficient options and run under the same conditions. The results were compared with the initial test results to quantify the level of improvement. The survey was done to illustrate the reduction in consumption and to see if the optimizations had any impact on user satisfaction.

G. Documentation and Reporting

A detailed account of all the main findings, such as tools and methods used, as well as the achieved results, was carefully documented. Moreover, charts and tables were used as visual aids for the better explanation of power consumption patterns. The study conclusion was drawn from this interpretation, and it serves as a strong platform for future enhancements and ongoing research in the area.

III. Modeling And Analysis

1 Adaptive Resource Allocation

The adaptive resource allocation based on application priority is at the heart of the modeling framework for AI-driven energy management. Google's Adaptive Battery function serves as an example of how this strategy tailors phone performance to commonly used apps, preserving the most power for the apps that customers depend on the most. The system may efficiently put less crucial apps into a dormant state to lower their battery consumption by prioritizing and managing resources like CPU performance. This prioritizing technique limits the power consumption of less frequently used apps while guaranteeing that key programs receive sufficient resources by establishing a dynamic hierarchy of applications based on usage patterns. The machine learning technology gradually identifies unique usage patterns, leading to longer periods of more constant battery life.

2 Architecture of Computational Memory

Computational random-access memory (CRAM) is a promising architectural model for lowering energy usage in AI-driven applications. A cutting-edge hardware device that processes data wholly within the memory array without moving it between logic and memory components has been developed by researchers at the University of Minnesota. This gets rid of the continuous transfer of data between processor and storage units, which is one of the most energy-intensive parts of computing.

With the potential to reduce energy usage by a factor of at least 1,000 for AI computing applications, the CRAM model marks a significant leap in energy-efficient computing architecture. Although still in the research phase, this strategy provides insightful information on how basic design adjustments might significantly increase the energy efficiency of next Android apps and devices.

IV. Results And Discussion

Battery Life Improvements

Significant increases in battery life have been seen for Android devices with AI-driven energy management algorithms installed. With its clever resource allocation strategy, Google's Adaptive Battery function promises a 30% increase in battery life. One of the most frequent consumer complaints regarding smartphone performance is directly addressed by this significant gain in battery life.

These methods not only provide the immediate advantage of extending battery life between charges, but they also enhance battery health in general. Over time, fewer battery recharge cycles are required as customers need to charge their devices less regularly. Beyond daily performance gains, this decrease in charging frequency might increase the battery's total lifespan and yield long-term advantages.

There is a lot of promise for future applications of research into specialized hardware for AI-driven energy management. Researchers from the University of Minnesota showed how to use computational random-access memory (CRAM), which might cut energy usage for AI.

User Experience Enhancements

The advantages of AI-driven energy management go beyond straightforward battery life indicators and encompass more extensive enhancements to the user experience. Applications with reduced power consumption exhibit superior thermal performance, which enhances user happiness and device responsiveness overall. These programs aid in preventing thermal throttling, which can impair performance over time, by avoiding excessive power usage.

Furthermore, depending on current usage patterns, AI-powered predictive algorithms can calculate the remaining battery life, giving users more precise information about the condition of their device. By assisting users in making well-informed decisions regarding the use of their devices, this predictive capability lessens the worry that comes with unplanned battery drain.

Environmental Impact

Beyond the performance of individual devices, there may be environmental advantages to the broad use of AI-driven energy management in mobile applications. These technologies help to lessen the carbon footprint of charging gadgets by lowering overall energy use. Due to the widespread use of mobile technology, even small decreases in energy use per device can have a big impact on the environment over time.

V. Conclusion

Battery optimization for Android apps has undergone a paradigm leap thanks to AI-driven energy management. These systems can greatly increase battery life while preserving or enhancing application performance by utilizing machine learning algorithms to monitor resource consumption, evaluate usage trends, and adaptively allocate system resources. With quantifiable gains in both short-term battery life and long-term battery health, the incorporation of features like Google's Adaptive Battery highlights the useful advantages of this strategy.

Future efficiency benefits could be much larger because to the development of specialized hardware architectures like computational random-access memory (CRAM), which could cut the amount of energy used by AI applications by orders of magnitude. Although these technologies are still in their infancy, they show how much space exists for advancement in the methods of energy management that are now in use.

Instead of being viewed as an optional optimization, developers should view the integration of AI-driven energy management as a crucial component of program design. Energy efficiency can now be measured and enhanced through methodical testing and optimization thanks to tools like Android Studio's Power Profiler. Developers may produce apps that not only work effectively but also increase user pleasure and device longevity by putting an emphasis on energy conservation and utilizing AI capabilities.

The significance of intelligent energy management will only grow as mobile devices continue to get more potent and feature-rich. One of the most promising approaches to resolving the ongoing problem of battery life in mobile computing is the combination of AI capabilities and energy management strategies.

VI. Acknowledgements

The examination of publicly accessible data from IT firms, academic institutions, and developer resources allowed for the conduct of this study. The continuous efforts of Google's Android team to create and improve energy management features and tools for the Android platform are especially noteworthy. Furthermore, recognition is given to the University of Minnesota researchers for their innovative work on computational memory structures, which could influence the direction of energy-efficient computing in the future.

VII. REFERENCES

[1] Android Developers. Save battery with Adaptive Battery, 2023.
[2] Google. Battery Historian, GitHub, 2023.
[3] Android Studio. Power Profiler Overview, 2023.
[4] Lee et al., 2022. Compute-in-memory for energy-efficient AI: A CRAM-based model. IEEE Micro, 42(1), pp. 48–55.
[5] Kejriwal & Bhatia, 2021. AI-Based Optimizations in Mobile Applications. International Journal of Computer Applications, 179(7), pp. 1–5.
[6] Gupta & Deshpande, 2021. Role of Artificial Intelligence in Power Management of Android Applications. Journal of Software Engineering and Applications, 14, pp. 185–194.
[7] University of Minnesota. CRAM: Computational RAM for Energy-Efficient AI Systems, 2022.
[8] Kim et al., 2018. Low Power Application Design Techniques for Mobile Devices. ACM TECS, 17(2), pp. 1–21.
[9] Das & Banerjee, 2021. Energy-Efficient Mobile Applications: Challenges and Future Directions. IEEE TMC, 20(7), pp. 2341–2355.
[10] Satyanarayana, 2017. The Emergence of Edge Computing. Computer, 50(1), pp. 30–39.