Smart Notifier

Kulkarni Atharva
MS Computer Engineering
Stonybrook University
Stony brook, NY
atharva.kulkarni@stonybrook.edu

Utkarsha Jadhav
MS Computer Engineering
Stonybrook University
Stony brook, NY
utkarsha.jadhav@stonybrook.edu

Shivani Pande
MS Computer Engineering
Stonybrook University
Stony brook, NY
shivani.pande@stonybrook.edu

***Abstract*—In today's fast-paced world, the quest for productivity and mental well-being is paramount. Research indicates that task completion not only clears mental space, allowing for enhanced creativity and problem-solving, but also significantly boosts confidence and overall well-being [3]. Furthermore, the act of crossing off items from a to-do list has been linked to the release of dopamine, the "feel-good" neurotransmitter, which reinforces the sense of achievement and motivation [4]. However, despite the proliferation of productivity tools, their effectiveness can be inconsistent, with some studies suggesting a potential for these tools to backfire if not used judiciously [5].**

Enter the Smart Notifier app, a solution designed to bridge the gap between intention and action. By leveraging advanced machine learning algorithms, the app intelligently recognizes user activities and suggests related tasks from their to-do list, facilitating a seamless integration into daily life. This approach not only ensures that both major and minor tasks are completed but also enhances mental well-being by providing users with a structured method to manage their responsibilities. The Smart Notifier app stands as a beacon of innovation, offering a practical solution to the modern dilemma of task management and productivity.

I. Introduction

Persistent thoughts about unfinished tasks can lead to a state where the mind becomes more preoccupied with these tasks rather than focusing on formulating effective strategies to address them. This cognitive preoccupation often results in increased mental load and stress, which can hinder overall productivity and problem-solving abilities. Instead of devising solutions or plan steps to complete the tasks, individuals may find themselves stuck in a cycle of rumination and anxiety. This can exacerbate feelings of being overwhelmed, reducing the capacity for efficient and creative thinking.

The app's design is informed by the Zeigarnik effect, which suggests that uncompleted tasks linger in our memory more persistently than those we have finished [6]. By leveraging this cognitive bias, the Smart Notifier app ensures that users are reminded of pending tasks at the most opportune moments, thus facilitating task completion and enhancing overall productivity. This smart integration of psychological insights with advanced technology positions the Smart Notifier app as an essential tool for anyone looking to optimize their schedule and improve their mental well-being.The application employs a Human Action Recognition model to identify completed tasks and prompt users to fulfill remaining tasks in an efficient and intelligent manner.

Identify applicable funding agency here. If none, delete this.

II. Approach

The application addresses a common issue where individuals focus primarily on major tasks, often losing track of smaller, yet important, tasks. By leveraging this smart approach, users are more likely to complete their entire to-do list, leading to a more productive and balanced day. Research has shown that completing tasks on a to-do list can significantly enhance mental well-being and productivity [2]. This method not only streamlines task management but also provides a structured way to ensure all tasks, big and small, are addressed, ultimately yielding positive outcomes.
Our application utilizes a machine learning model to recognize user actions and relay this information intelligently. By identifying the activities being performed, the system can prompt users about existing, similar tasks on their to-do list that can be completed after the current activity. This approach offers users an efficient opportunity to complete tasks without the need for extensive planning, thereby optimizing their schedule.

A. Machine/ Deep learning model

Recognizing human activities using smartphone sensors is crucial for various applications, including health monitoring and behavior analysis. Leveraging the "Human Activity Recognition Using Smartphones" dataset from the UCI repository, our approach aims to develop a robust machine learning model capable of accurately identifying different activities performed by users.

TABLE I
Statistics of the UCI Human Activity Recognition Dataset

Subjects	30 volunteers
Age Range	19-48 years
Activities Recorded	Walking, Walking Upstairs, Walking Downstairs, Sitting, Standing
Equipment Used	Device: Samsung Galaxy S II
Sensors	Accelerometer, Gyroscope
Sampling Rate	50 Hz

The model architecture consists of an LSTM (Long Short-Term Memory) layer with 32 units, tailored to capture temporal dependencies inherent in the activity data. Following the LSTM layer is a Dropout layer, strategically incorporated to address overfitting concerns by selectively deactivating neurons during training, thereby enhancing the model's capacity

for generalization. Subsequently, a Dense layer with 6 units, mirroring the number of distinct activity classes intended for recognition, constitutes the final layer of the model. With a total of 5574 trainable parameters, this architecture is meticulously optimized for processing time-series data, a pivotal aspect in accurately classifying human activities based on sensor inputs.

While the model may seem straightforward, its simplicity is deliberate, driven by the primary objective of seamless integration into native smartphone environments. This design choice aims to mitigate potential performance issues such as latency and dependence on internet connectivity, rendering the application fully operational offline when necessary.

For deployment in our application, we leveraged TensorFlow Lite, a framework specifically designed for deploying machine learning models on mobile and embedded devices. TensorFlow Lite offers optimized performance and reduced model size, facilitating efficient execution on resource-constrained platforms such as smartphones. Integrating the aforementioned model into our application enabled real-time activity recognition directly on the device, eliminating the need for continuous internet connectivity and ensuring seamless operation even in offline scenarios. This utilization of TensorFlow Lite not only enhances the user experience by minimizing latency but also underscores our commitment to delivering a versatile and responsive mobile application tailored to diverse usage environments.

The activities identified by this model predominantly comprise fundamental actions, constrained to the capabilities of sensors integrated within smartphones and demonstrated with a high degree of precision. Yet, it is essential to recognize that these foundational activities can be systematically aggregated, offering valuable indications that facilitate approximations of broader user behaviors.

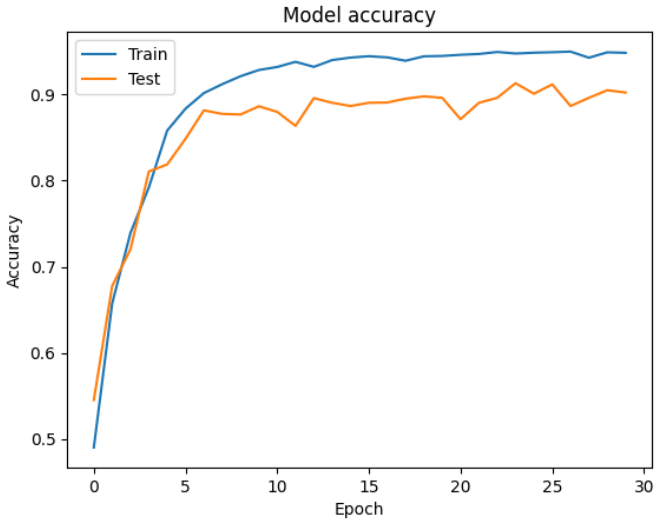

Fig. 1. Training accuracy over different epochs.

TABLE II
TRAINING CONFIGURATION AND PERFORMANCE

Configuration	Details
Training Time (Local Machine)	More than 2 hours
Epochs	30
Hardware Specs (Local)	Dell Latitude, 16 GB RAM
Training Time (Cloud Kaggle TPU)	40 minutes

B. LLM

We employ ChatGPT LLM (Large Language Model) to extract a to-do card from the user and subsequently deconstruct the associated task into its constituent fundamental actions. This process involves breaking down the approximate actions required to accomplish the task into elemental components. Following this breakdown, the model is prompted to recognize these fundamental actions, discerning whether the user is engaged in executing the specified task or a related task that shares similar action sequences. By employing this methodology, our system aims to facilitate task recognition by identifying common action patterns, thereby enhancing the accuracy of task completion assessments and enabling proactive assistance in handling both specified tasks and analogous activities within the user's workflow

C. Amazon RDS

In our implementation, we leverage Amazon RDS (Relational Database Service) to store individual user tasks within a SQL database. This approach enables the seamless synchronization of to-do lists across multiple devices, ensuring that users can access and manage their tasks irrespective of the device they are using. By centralizing task storage in a relational database hosted on Amazon RDS, we ensure data consistency and reliability while accommodating the dynamic needs of users across various platforms. This architecture not only enhances user convenience but also facilitates efficient task management and collaboration in multi-device environments.

III. PROJECT EVOLUTION AND METHODOLOGY OVERVIEW

A. Algorithm Approach for HAR and Prompting Mechanism

In the initial phase of the project, the primary focus was on formulating an effective algorithmic approach for Human Activity Recognition (HAR) while concurrently developing a prompting mechanism. This phase involved rigorous exploration and testing of various HAR algorithms to ensure accurate classification of human activities based on sensor data inputs. Additionally, a prompting mechanism was conceptualized and designed to facilitate user interaction by guiding them through tasks or activities based on contextual cues or user input.

B. Using Smartwatch Sensors and Cloud for Predictions

As the project progressed, a pivotal transition occurred towards the utilization of smartwatch sensors for data acquisition and leveraging cloud infrastructure for predictive analytics. This phase involved the collection and processing of data streams from smartwatch sensors, including accelerometers and gyroscopes, to facilitate the training of machine learning models. Subsequently, trained models were deployed and executed on cloud platforms, harnessing the scalability and computational capabilities of cloud resources to make accurate predictions based on the sensor data received from the smartwatches.

C. Using Cloud for Model Generation on Native Android Device

A significant evolution in the project's trajectory was witnessed as it transitioned towards leveraging cloud resources for the generation of machine learning models optimized specifically for deployment on native Android devices. This advanced phase involved utilizing cloud infrastructure for training and optimizing machine learning models tailored to the constraints and requirements of native Android environments, such as memory and processing power limitations. By generating models on the cloud and deploying them on native Android devices, the project achieved efficient and real-time predictions without heavy reliance on internet connectivity or external cloud services during inference, thus enhancing the applicability and usability of the solution for end-users.

IV. User Interface

The application encompasses a dynamic user interface featuring familiar mechanisms for creating, updating, and deleting user to-do list cards in real-time. These cards are securely stored within an Amazon RDS SQL database infrastructure.

To realize this functionality in the Android Studio development environment, we employ a layered approach. The user interface (UI) is crafted using XML layout files, incorporating interactive elements such as buttons and input fields. The Java programming language facilitates the backend logic, orchestrating communication with the Amazon RDS SQL database via established APIs.

The user interface (UI) depicted in Figure 2 comprises several interactive elements designed to facilitate efficient list management. It features:

- *Navigation options, such as tabs or a sidebar menu, for seamless access to different sections of the application.*
- *Input fields for creating new to-do list cards, allowing users to enter task details such as title, description, and due date.*
- *Buttons or icons for performing actions like saving, updating, or deleting tasks.*
- *A list or grid layout displaying existing to-do list cards, showcasing task titles and relevant details.*

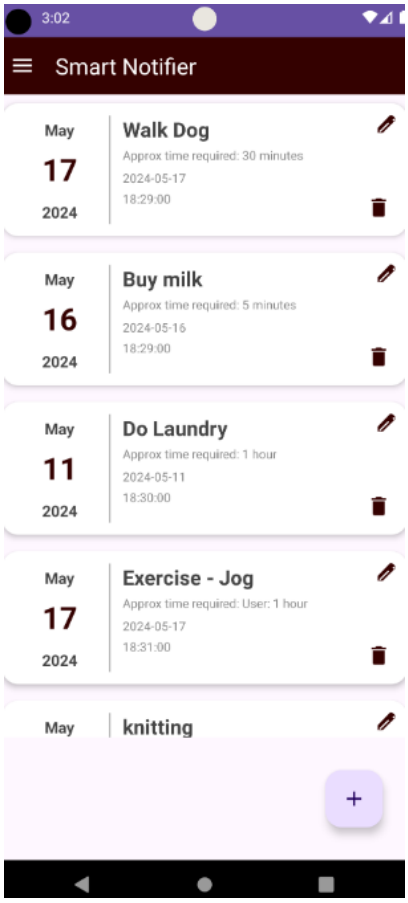

Fig. 2. User Interface Snapshot

This UI design aims to provide users with a seamless and intuitive experience for managing their tasks effectively.

This UI design aims to provide users with a seamless and intuitive experience for managing their tasks effectively.

Within Android Studio, we leverage the robust capabilities of the Android SDK to handle user interactions seamlessly. For instance, to create a to-do list card, the application captures user input through UI components and sends a structured request to the backend. Upon receipt, the backend processes the request and stores the new entry in the Amazon RDS SQL database.

Real-time updates are facilitated through efficient data synchronization techniques. The application employs asynchronous tasks to periodically fetch updates from the database, ensuring that the user's interface remains current at all times. Additionally, we implement push notifications to alert users of any changes made to their to-do list cards, further enhancing the real-time experience.

To enable efficient data management, the application incorporates functionality for updating and deleting to-do list cards. Users can seamlessly modify existing entries by interacting with intuitive UI controls, triggering backend operations to reflect these changes in the database. Similarly, the deletion

process seamlessly removes unwanted entries from both the UI and the database, ensuring data integrity.

V. Conclusion

The Smart Notifier app represents a significant advancement in personal productivity and task management. By intelligently recognizing user actions and suggesting related tasks from their to-do list, the app seamlessly integrates into daily routines, promoting efficiency without the need for meticulous planning. This innovative approach not only aids in the completion of both major and minor tasks but also contributes to enhanced mental well-being by reducing the cognitive load associated with task organization. The app's ability to remind users of pending tasks at opportune moments ensures a more balanced and productive day. Ultimately, the Smart Notifier app stands as a testament to the power of technology in augmenting human capabilities, enabling users to achieve a more organized, stress-free, and fulfilling lifestyle.

References

[1] Masicampo EJ, Baumeister RF. Consider it done! Plan making can eliminate the cognitive effects of unfulfilled goals. J Pers Soc Psychol. 2011 Oct;101(4):667-83. doi: 10.1037/a0024192. PMID: 21688924.

[2] https://www.theguardian.com/lifeandstyle/2017/may/10/the-psychology-of-the-to-do-list-why-your-brain-loves-ordered-tasks

[3] https://www.maketimecount.com/how-completion-helps-students-build-confidence/

[4] https://www.psychologytoday.com/us/basics/productivity/mental-health-and-productivity

[5] https://www.bbc.com/worklife/article/20210208-why-relying-on-productivity-tools-can-backfire

[6] https://en.wikipedia.org/wiki/Zeigarnikeffect

Please find the application code in this repository :