I J C T A, 10(8), 2017, pp. 711-716 ISSN: 0974-5572

Battery Discharge Prediction Using support vector machine

Shweta Koparde*, Namrata Gawande*, Meghana Lokhande* and Sagar Salunke*

ABSTRACT

Mobile phone is the most widely used communication device in today's world. Mobile phones have become a part of our daily life. Since its invention mobile phones have developed from basic to smart. Today mobile phones are used not only for calling, but also for watching videos, listening to music, playing games , browsing the web and many more. But with the increase in features in the mobile phones, battery consumption of the phones has also increased. Today most of the mobile phones, especially Android phones faces the problem of battery power consumption. To reduce the power consumption, we are going to predict the battery discharge rate of the phones in order to increase the battery life of phones. Using data mining, we will collect the mobile phone data, user pattern of mobile usage and predict the discharge rate of the battery for the mobile phone. This data will help the user to change their pattern to increase their battery life. The mobile discharge rate prediction model makes use of Support Vector Model (SVM) and K-Fold Cross Validation.

Index Terms: Support Vector Model, K-Fold Cross validation, Discharge rate prediction, Energy consumption, Data Mining.

1. INTRODUCTION

In today's world mobile is a integral part of our life. The power required for running the phone is provided by the battery. But the power supplied by the battery is limited since mobile phone is a embedded device. When a user uses a phone device, the user does not know the actual battery lifetime. There are many applications available which predicts the battery lifetime based on the amount of power left in the battery. This method of prediction is a static prediction method. The applications predict battery lifetime based on the applications or processes running at that particular time, which we consider as a state. If there is increase or decrease in the number of applications or processes running which will change the state, the battery remaining time changes, which in turn contradicts the previous state. We can predict the battery lifetime dynamically by making use of the usage pattern of individual users, considering that the pattern of user remain more or less the same. This can help the user to analyze their usage pattern or make changes to their usage to suit their needs and increase the battery life.

1.1. K-Fold Cross Validation

K Fold Cross Validation divides data into k groups. Out of the k groups , one group of data is used as validation data while the remaining groups are used as training data. After the data has been the classified, a second group of data is used as validation data and the remaining are used as training data. The process is repeated for k iterations. In the final step, the results of the k iterations are averaged to find a single estimation. Using this approach, all data groups can be used for both validation and training.

1.2. Support Vector Machine

Support Vector Machine (SVM) is a method used for classi-fying patterns or groups of data. It is used for classification and regression analysis. SVM divides a given set of data into two categories. It is based on

* Assistant Professor, PCCOE, Pune, *Emails: ashwe04@gmail.com, gawande.namrata@gmail.com, meghana.ingole1983@gmail.com, sagargsalunke@gmail.com*

the concept of decision planes that define decision boundaries. A decision plane is a plane which separates sets of objects having different class membership. It performs linear as well as non-linear classification.

1.3. SQLite

SQLite is an Open Source database. Its is developed for embedded devices. SQLite is embedded into Android. It is relational database engine. SQLite supports standard relational databases features such as SQL syntax, transaction. The main feature of SQLite is that it requires less memory compared to other databases. It does not require the device to have a high memory configuration. It requires less memory during runtime (approx 250 Kbyte). It is the widely deployed SQL databaseengine in the world.

2. PROPOSED SYSTEM

Our system is mainly proposed for Android devices though this can be extended to other mobile OS devices. The system will collect usage pattern of a individual user. The data collected will include the list of running applications, start time, end time, battery level at the start of application and battery level at the end of the application, actually battery lifetime for the day. The data will be stored for each individual day separately. Each individual day's data will be subdivided into each hour. For a current day of the current day, the system will search the database for a usage pattern (previous days) matching closely with the current day's (today) usage pattern and will predict the battery lifetime with the help of actual battery lifetime of the matching using pattern (previous days).

2.1. Block Diagram

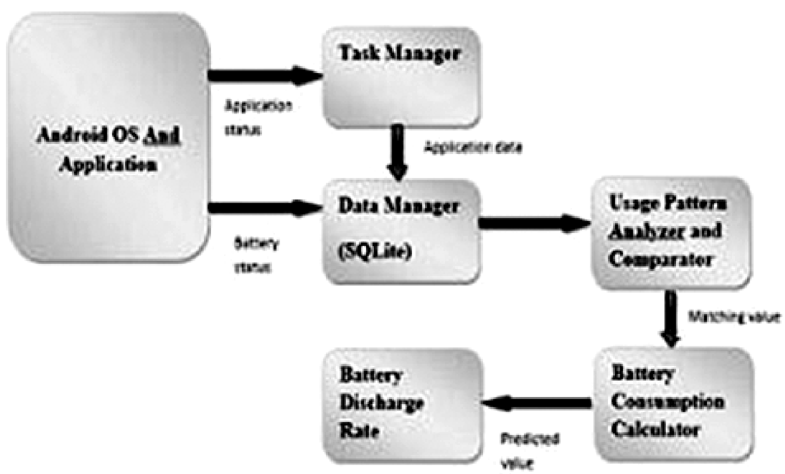


Figure 1: Block Diagram

2.2. Algorithms

K-Fold Cross Validation Algorithm:

1) Start
2) Make the previous days data as training data set.
3) Make the current day data (actual battery lifetime) as testing data.
4) After the end of the current day validate the training data set with the test data set.
5) If (change in training data and test data) train the training data set.
6) Else stop training
7) Stop

2.3. Architectural Design

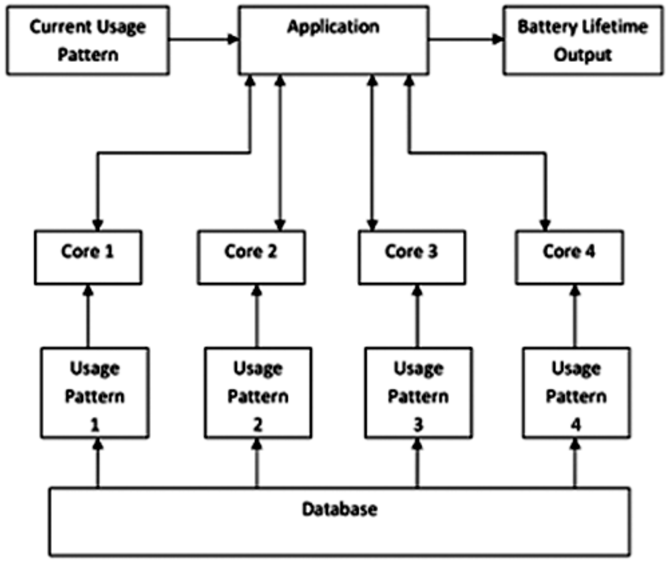


Figure 2: Architectural Diagram

2.4. Activity Diagram

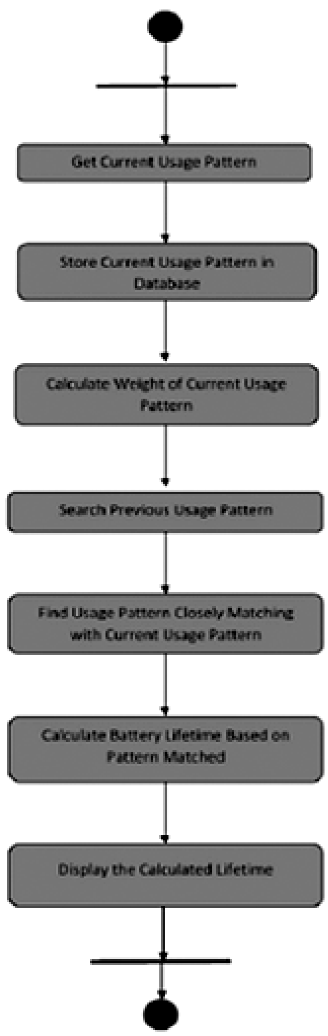


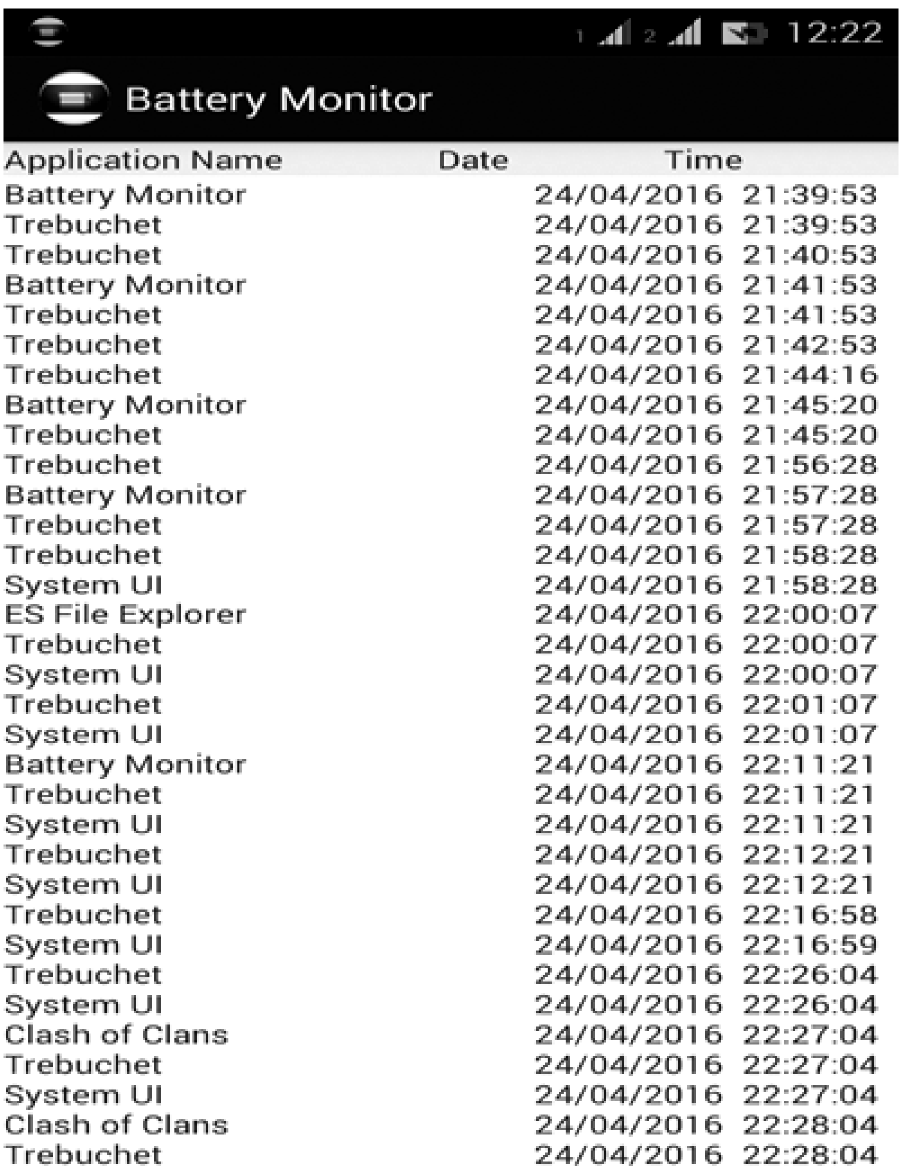
Figure 3: Activity Diagram

2.5. System Modules

1) Task Collector: It will collect all the details of running application and processes details such as timestamp, memory consumed.
2) Database Manager: SQLite database will be used which will store the information collected by the Task Collector. This data will be used for prediction.
3) Usage Pattern Comparator: SVM will be used to compare data so that they will belong to a particular class (matching and non matching pattern) of our consideration.
4) Battery Lifetime Calculator: This will calculate the average discharge rate and predict battery lifetime in real time.

3. COMPARLSON RESULTS

The Following Figure shows the results of prediction model of discharge rate using K-fold cross validation and SVM.

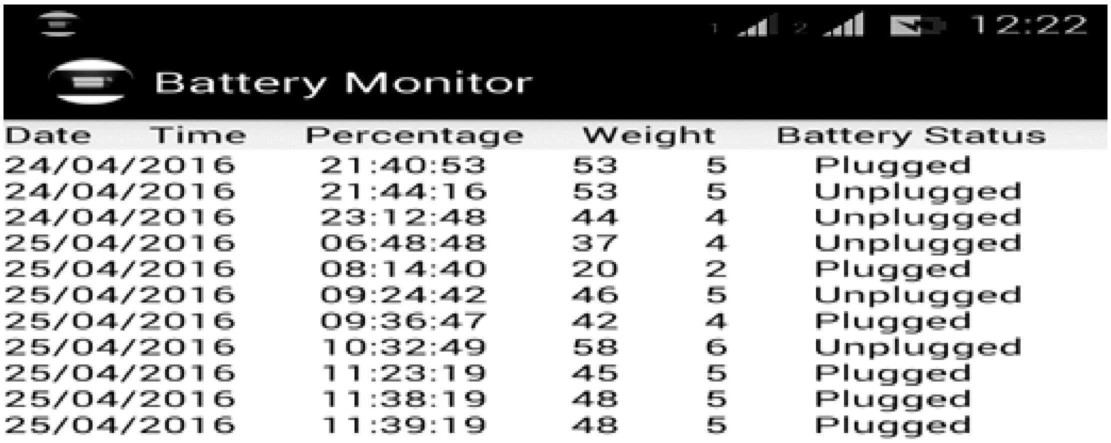


Application Name	Date	Time
Battery Monitor	24/04/2016	21:39:53
Trebuchet	24/04/2016	21:39:53
Trebuchet	24/04/2016	21:40:53
Battery Monitor	24/04/2016	21:41:53
Trebuchet	24/04/2016	21:41:53
Trebuchet	24/04/2016	21:42:53
Trebuchet	24/04/2016	21:44:16
Battery Monitor	24/04/2016	21:45:20
Trebuchet	24/04/2016	21:45:20
Trebuchet	24/04/2016	21:56:28
Battery Monitor	24/04/2016	21:57:28
Trebuchet	24/04/2016	21:57:28
Trebuchet	24/04/2016	21:58:28
System UI	24/04/2016	21:58:28
ES File Explorer	24/04/2016	22:00:07
Trebuchet	24/04/2016	22:00:07
System UI	24/04/2016	22:00:07
Trebuchet	24/04/2016	22:01:07
System UI	24/04/2016	22:01:07
Battery Monitor	24/04/2016	22:11:21
Trebuchet	24/04/2016	22:11:21
System UI	24/04/2016	22:11:21
Trebuchet	24/04/2016	22:12:21
System UI	24/04/2016	22:12:21
Trebuchet	24/04/2016	22:16:58
System UI	24/04/2016	22:16:59
Trebuchet	24/04/2016	22:26:04
System UI	24/04/2016	22:26:04
Clash of Clans	24/04/2016	22:27:04
Trebuchet	24/04/2016	22:27:04
System UI	24/04/2016	22:27:04
Clash of Clans	24/04/2016	22:28:04
Trebuchet	24/04/2016	22:28:04

Figure 4: Screenshot of Running Applications Collector

Here, we estimate battery discharge by 3 usages:

- Current state with use or running any applications. (Standby)
- State with running application not related to network.
- State with running application related to network.



Date	Time	Percentage	Weight	Battery Status
24/04/2016	21:40:53	53	5	Plugged
24/04/2016	21:44:16	53	5	Unplugged
24/04/2016	23:12:48	44	4	Unplugged
25/04/2016	06:48:48	37	4	Unplugged
25/04/2016	08:14:40	20	2	Plugged
25/04/2016	09:24:42	46	5	Unplugged
25/04/2016	09:36:47	42	4	Plugged
25/04/2016	10:32:49	58	6	Unplugged
25/04/2016	11:23:19	45	5	Plugged
25/04/2016	11:38:19	48	5	Plugged
25/04/2016	11:39:19	48	5	Plugged

Figure 5: Screenshot of Running Applications Collector

In Figure 5, we present the sample graph for battery discharge rate . We use the sampling data for every 5minute for each state. The battery discharge rate is based on the coefficients calculated from time and battery flow.

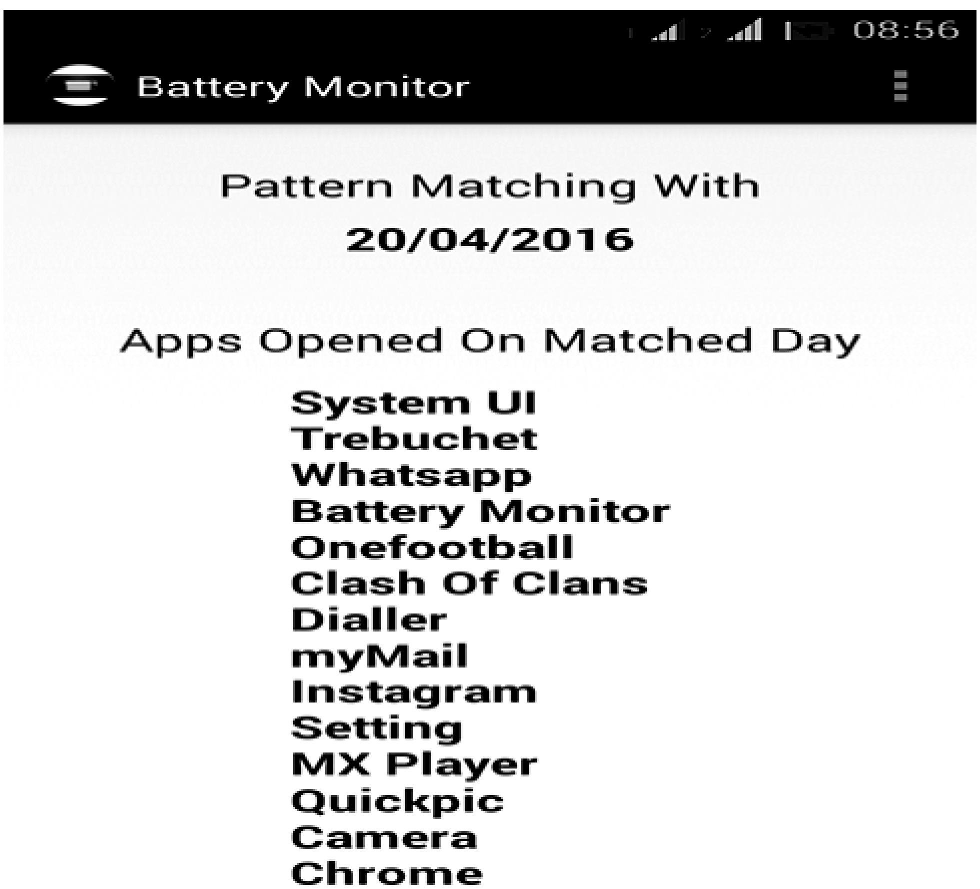


Figure 6: Matching Pattern Details

In this research uses the data mining to create model

for prediction of the energy in mobile phone’s battery using k-fold cross validation and SVM.

Figure 7 shows mobile battery percentage and battery discharge rate per hour.

Figure 7: Battery Percentage And Predicted Discharge Rate

4. CONCLUSION

Today’s android phones have high battery consumption. Bypredicting the discharge rate and power consumption by each application we can help the user in helping choose to run the applications that are necessary so as to increase the battery life. This will also lead to saving the energy.

REFERENCES

[1] Joon-Myung Kang, Sin-seok Seo and James Won-KiHong, “Personalized Battery Lifetime Prediction for Mobile Devices based on Usage Pat-terns.”Journal of Computing Science and Engineering, Vol. 5, No. 4, December 2011, pp. 338-345.

[2] P. Nusawat S. Adulkasem and C. Chantrapornchai, “Battery Discharge Rate Prediction Model for Mobile Phone Using Data Mining”, 20 14 6th International Conference on Knowledge and Smart Technology (KST)

[3] Chandra Krintz, Ye Wen, “Rich Wolski, Application-level Prediction Of Battery Dissipation”, Proceedings of the International Symposium on Lower Power Electronics and Design, Newport Beach, CA, 2004

[4] www.sqlite.com