

PREDICTION OF SMARTPHONE ADDICTION USING MACHINE LEARNING

A PROJECT PHASE I REPORT

Submitted by

PRATHIMA

212220040156

in partial fulfilment for the award of the degree of

**BACHELOR OF ENGINEERING
in
COMPUTER SCIENCE AND ENGINEERING**



SAVEETHA ENGINEERING COLLEGE (Autonomous)

ANNA UNIVERSITY :: CHENNAI- 600 025

DECEMBER 2023

ANNA UNIVERSITY, CHENNAI
BONAFIDE CERTIFICATE

Certified that this project report **“PREDICTION OF
SMARTPHONE ADDICTION USING MACHINE LEARNING”** is
the
bonafide work of **PRATHIMA(212220040156)** who carried out
the project work under my supervision.

SIGNATURE

Dr. ANITHA JULIAN, M.E., Ph.D
Professor

SUPERVISOR

Dept of Computer Science and
and Engineering,
Saveetha Engineering College,
College, Thandalam, Chennai 602105.

SIGNATURE

Dr.G.NAGAPPAN,M.E.,Ph.D.,
Professor

HEAD OF THE DEPARTMENT

Dept of Computer Science
Engineering,
Saveetha Engineering
Thandalam, Chennai 602105.

DATE OF THE VIVA VOCE EXAMINATION:

INTERNAL EXAMINER

ACKNOWLEDGEMENT

We express our deep sense of gratitude to our honourable and beloved Founder President Dr. N. M. Veeraiyan, our President Dr. Saveetha Rajesh, our Director Dr. S. Rajesh and other management members for providing the infrastructure needed.

We express our wholehearted gratitude to our Principal, Dr. N. Duraipandian, for his wholehearted encouragement in completing this project.

We convey our thanks to Dr. G. Nagappan, Professor and Head of the Department of Computer Science and Engineering, Saveetha Engineering College, for his kind support and for providing the necessary facilities to carry out the project work.

We would like to express our sincere thanks and deep sense of gratitude to our Supervisor and our Project Coordinator Dr. Anitha Julian, Professor, Department of Computer Science and Engineering for her valuable guidance, suggestions and constant encouragement that paved the way for the successful completion of the project work and for providing us necessary support and details at the right time and during the progressive reviews. We owe our thanks to all the members of our college, faculty, staff and technicians for their kind and valuable cooperation during the course of the project. We are pleased to acknowledge our sincere thanks to our beloved parents, friends and well-wishers who encouraged us to complete this project successfully.

ABSTRACT

Smartphone addiction has become a growing concern in recent years, with increasing numbers of people exhibiting symptoms such as excessive phone use, loss of productivity, and even physical and psychological health problems. As a result, there is a need to develop effective tools for predicting smartphone addiction and identifying those at risk.

In this study, we developed a machine learning model for predicting smartphone addiction using data collected from a survey of smartphone users. The survey included questions about demographics, phone use patterns, and various psychological factors such as anxiety, depression, and stress. a popular and effective machine learning method, to build our model. We preprocessed the data by encoding categorical variables and normalizing numerical variables to ensure the model could learn effectively. We then trained the model on a portion of the data and evaluated its performance on the remaining data using several metrics such as accuracy. Our results showed that the model achieved a high accuracy of in predicting smartphone addiction. The most important features for predicting addiction were phone use patterns such as the frequency of checking notifications, the number of hours spent on the phone each day, and the types of apps used most frequently. Other important factors included age, gender, and stress levels. The model we developed has several potential applications. It could be used by healthcare professionals to identify individuals who are at risk of developing smartphone addiction and provide appropriate interventions. It could also be used by app developers to design apps that are less addictive and promote healthier phone use habits. In conclusion, our study demonstrates the feasibility and effectiveness of using machine learning models for predicting smartphone addiction. Further research is needed to validate our findings on larger and more diverse datasets and to explore the potential applications of this model in different contexts.

TABLE OF CONTENTS

CHAPTER NUMBER	TITLE	PAGE NUMBER
	Acknowledgment	i
	Abstract	ii
	List of abbreviations	v
	List of figures	vi
1	INTRODUCTION	1
	1.1 Overview of the project	1
	1.2 Scope and Objective	2
2	LITERATURE SURVEY	3
	2.1 Introduction	3
	2.2 Literature Survey	4
3	SYSTEM DESIGN	9
	3.1 Introduction	9
	3.2 Existing System	10
	3.3 Proposed System	11
	3.4 Algorithm	

	3.5 System Architecture Diagram	19
	3.6 System Requirements	21
4	IMPLEMENTATION AND ANALYSIS	21
	4.1 Feasibility Study	23
	4.2 Dataset Description	24
	4.3 Module Description	27
	4.4 Results	29
5	CONCLUSION	34
	5.1 Conclusion	34
	5.2 Future Work	35
	Appendix A – Sample Coding	
	Appendix A – Sample Output	
	References	

LIST OF ABBREVIATIONS

SRC	Sport Related Concussion
OWI	Overall Wellness Index
ANN	Artificial Neural Network
CRT	Classification And Regression Trees
RAM	Random Access Memory

LIST OF FIGURES

Fig 1.1	Block Diagram
Fig 1.2	System Architecture
Fig 2.1	Home page
Fig 2.2	Registration Page
Fig 2.3	Login Page
Fig 2.4	Home Page
Fig 2.5	View Data Page
Fig 2.6	Model Page
Fig 2.7	Prediction Page
Fig 2.8	Accuracy Graph
Fig 3.1	Test Dataset
Fig 3.2	Train Dataset
Fig 3.3	Google Collaboratory Workspace

CHAPTER 1

INTRODUCTION

1.1 OVERVIEW OF THE PROJECT

Smartphones have become an integral part of our lives, and their usage has increased dramatically over the past decade. While smartphones offer numerous benefits, excessive smartphone usage can lead to addiction and have negative impacts on individuals' physical and mental health, social relationships, and productivity. Machine learning can be used to develop models that can predict smartphone addiction based on various features such as smartphone usage patterns, social media usage, demographic information, and psychological factors. These models can help identify individuals who are at risk of smartphone addiction and provide them with appropriate interventions and support. To develop a machine learning model for predicting smartphone addiction, one would typically start by collecting data from a large sample of individuals. This data would include information about their smartphone usage patterns, social media usage, demographic information such as age, gender, and psychological factors such as anxiety, depression, and stress levels. Once the data is collected, it is preprocessed and cleaned to remove any missing or irrelevant data points. Next, a suitable machine learning algorithm is selected, such as logistic regression, decision tree, or Random Forest, based on the nature of the data and the problem at hand. The data is then split into two sets, a training set and a testing set. The training set is used to train the machine learning model by feeding it with input features and corresponding output labels. The model learns to recognize patterns in the data and establish a relationship between the input features and the output labels. Once the model is trained, it is tested on the testing set to evaluate its performance. The performance of the model is measured using various metrics such

as accuracy. The model is further refined by tweaking its parameters or selecting different algorithms until satisfactory performance is achieved. Once the model is developed, it can be used to predict smartphone addiction in individuals by feeding their input features into the model. The model outputs a probability score indicating the likelihood of smartphone addiction. Based on this score, appropriate interventions and support can be provided to individuals at risk of addiction. In conclusion, machine learning models can be a valuable tool for predicting smartphone addiction and identifying individuals at risk. These models can help individuals and healthcare professionals take proactive measures to prevent addiction and mitigate its negative effects. However, it is essential to collect high-quality data and develop accurate and reliable models that can be effectively used in real-world settings.

1.2 SCOPE & OBJECTIVE

The scope of this research encompasses the development and evaluation of a machine learning model for the prediction of smartphone addiction. It includes the collection and analysis of diverse datasets comprising smartphone usage patterns, behavioral characteristics, and psychological indicators. The model's scope extends to assessing its accuracy, sensitivity, and specificity as a diagnostic tool for identifying smartphone addiction. Ethical considerations regarding data privacy and responsible use of predictive models are explored. Additionally, the study discusses potential interventions and support mechanisms based on the model's predictions to address smartphone addiction. The ultimate scope is to contribute to the mitigation of smartphone addiction and promote healthier digital lifestyles. The primary goal of this project is to determine the prediction of smartphone addiction and to know this we have used classification techniques.

CHAPTER 2

LITERATURE SURVEY

1.1 INTRODUCTION

The literature review is one of the key components in keeping consistency. These are the essential actions that must be taken during the development process. This Literature survey aims to explore and analyze existing research and scholarly work on the prediction of smartphone addiction. By diving into the wealth of knowledge accumulated in this domain, we seek to identify key trends, methodologies, and findings that contribute to a comprehensive understanding of smartphone addiction prediction.

1.2 LITERATURE SURVEY

In Demir, K. & Akpinat, E. The effect of mobile learning applications on students' academic achievement and attitudes toward mobile learning. Malays. Online J. Educ. Technol. 6, 48–59 (2018) examines the effect of mobile learning applications on undergraduate students' academic achievement, attitudes toward mobile learning and animation development levels. Quasi-experimental design was used in the study. Participants of the study were students of the Buca Faculty of Education at Dokuz Eylul University in Turkey. The experiment was conducted during the first semester of 2013-2014 academic year. A mobile learning-based strategy was used in experimental group ($n = 15$), while the control group participated in a lecture-based classroom ($n = 26$). An attitude scale was used to measure the students' attitudes toward mobile learning, and achievement test was used to examine the effect of mobile learning applications on the students' achievement. In order to evaluate the animations developed by students, a rubric was used. For exploratory analysis, interviews were conducted with students. The

findings suggest that mobile learning may promote students' academic achievement. Both groups had significantly high attitude scores toward mobile learning. Furthermore, the students appreciated mobile learning as an approach that may significantly increase their motivation. Researchers and practitioners should take into consideration that mobile learning can create positive impact on academic achievement and performance and increase the motivation of students.

In Abadiyan, F., Hadadnezhad, M., Khosrokiani, Z., Letafatkar, A. & Akhshik, H. Adding a smartphone app to global postural re-education to improve neck pain, posture, quality of life, and endurance in people with nonspecific neck pain: A randomized controlled trial. *Trials* 22, 274 (2021) the effect of adding a smartphone app to an 8-week global postural reeducation (GPR) on neck pain, endurance, quality of life, and forward head posture (FHP) in patients with chronic neck pain and FHP was evaluated. Sixty male and female office workers (38.5 ± 9.1 years) with chronic neck pain were randomly assigned into three groups: group 1 (GPR+ a smartphone app, $n = 20$), group 2 (GPR alone, $n = 20$), and group 3 (the control group, $n = 20$). The primary outcome was pain and the secondary outcomes were disability, quality of life, endurance, and posture. Pain, disability, endurance, quality of life, and posture were evaluated using the visual analog scale (VAS), neck disability index (NDI), progressive iso-inertial lifting evaluation (PILE) test, quality of life questionnaire (SF-36), and photogrammetry, respectively, at pre-and post-8-week interventions. A one-way analysis of covariance (ANCOVA) has been conducted to statistically analyze the data.

In Osailan, A. The relationship between smartphone usage duration (using smartphone's ability to monitor screen time) with hand-grip and pinch-grip

strength among young people: An observational study. BMC Musculoskelet. Disord. 22, 186 (2021),The use of smartphones has become widely popular, especially among young people, for multiple purposes other than communication, including gaming and internet browsing. The hand and wrist weakness is one of the main complications associated with the increased use of smartphones. This weakness occurs due to the repetitive flexion and extension of the wrist, thumb, and fingers, leading to a significant musculoskeletal pathology. Little is known about the relationship between smartphone usage duration and hand-grip, pinch-grip strength. Therefore, the study was aimed to investigate the association between smartphone usage duration and hand-grip, pinch-grip strength among young people. One hundred young males volunteered to participate in the study. Participants were briefly examined for height and weight using a portable stadiometer and a digital scale. Hand-grip, pinch-grip strength measurement was performed using a hand-held dynamometer.

In Hitti, E., Hadid, D., Melki, J., Kaddoura, R. & Alameddine, M. Mobile device use among emergency department healthcare professionals: prevalence, utilization and attitudes. Sci. Rep. 11, 1917 (2021),Mobile devices are increasingly permeating healthcare and are being regularly used by healthcare providers. We examined the prevalence and frequency of mobile device use, and perceptions around clinical and personal usage, among healthcare providers in the Emergency Department (ED) of a large academic medical center in Lebanon. Half of the target population completed the cross-sectional electronic questionnaire. Mobile device usage for personal matters was uniform across all providers, with the highest usage reported by medical students (81.3%) and lowest by attendings (75.0%). Medical formulary/drug referencing applications were the most common application used by providers followed by disease diagnosis/management applications, 84.4% and 69.5% .Most respondents agreed that mobile devices enabled better-coordinated care among providers and were beneficial to patient care.

In Wilkerson, G. B. et al. Wellness survey responses and smartphone app response efficiency: Associations with remote history of sport-related concussion. *Percept. Mot. Skills* 128, 714–730 (2021).Recent research findings have strongly suggested that sport-related concussion (SRC) increases risk for subsequent injury of any type, as well as a potential for long-term adverse effects on neurological and psychological well-being. The primary purpose of this study was to explore the reliability and discriminatory power of clinical testing procedures for detecting persisting effects of SRC. We used a cross-sectional study design to assess both self-reported symptoms commonly associated with post-concussion syndrome, and the effects of mental or physical activity on metrics derived from a smartphone app designed to test perceptual-motor responses. Among 30 physically active college students, 15 participants reported a SRC occurrence prior to testing (M time- since-injury = 4.0 years, SD = 3.1, range = 5 months to 11 years). We found good test-retest reliability for key metrics derived from the smartphone app (ICC $\geq .70$); and the internal consistency for the Overall Wellness Index (OWI) for 10 categories of 82 post-concussion symptoms was ideal (Cronbach's $\alpha \geq .80$).

In Joo, E., Kononova, A., Kanthawala, S., Peng, W. & Cotten, S. Smartphone users' persuasion knowledge in the context of consumer mHealth apps: Qualitative study. *JMIR Mhealth Uhealth* 9, e16518 (2021). Persuasion knowledge, commonly referred to as advertising literacy, is a cognitive dimension that embraces recognition of advertising, its source and audience, and understanding of advertisers' persuasive and selling intents as well as tactics. There is little understanding of users' awareness of organizations that develop or sponsor mobile health (mHealth) apps, especially in light of personal data privacy. Persuasion knowledge or recognition of a supporting organization's presence, characteristics,

competencies, intents, and persuasion tactics are crucial to investigate because app

users have the right to know about entities that support apps and make informed decisions about app usage. The abundance of free consumer mHealth apps, especially those in the area of fitness, often makes it difficult for users to identify apps' dual purposes, which may be related to not only helping the public manage health but also promoting the supporting organization itself and collecting users' information for further consumer targeting by third parties.

In K. Yasudomi, T. Hamamura, M. Honjo, A. Yoneyama and M. Uchida, "Usage Prediction and Effectiveness Verification of App Restriction Function for Smartphone Addiction"(2020) IEEE International Conference on E-health Networking, Application & Services (HEALTHCOM),Shenzhen, China, 2021, pp. 1-8,there has been a growing problem of smartphone addiction. As the excessive use of smartphones has negatively impacted our daily lives, many apps for reducing smartphone addiction have been developed around the world. In this study, we focus on the app restriction function, which is one of the key features of digital medicines for smartphone addiction, and analyze the usage of the function and verify its effectiveness. The results showed significant differences in both psychological and behavioral aspects between those who used the app restriction function and those who did not. Specifically, we found that the app restriction function was more likely to be used by those who were more aware of their smartphone addiction. We also found that the app restriction function was effective in lessening smartphone usage time, especially when the smartphone addiction is relatively moderate.

In I. A. E. Zaeni, D. R. Anzani, Sudjiwanati, E. P. Kristianty and A. Q. Sheng, "Classification of the Smartphone Addiction using the Artificial Neural Network," 2021 International Research Symposium On Advanced Engineering And Vocational Education (IRSAEVE), Malang, Indonesia, 2021, pp. 14-18,Smartphones, as communication tools, have evolved and have a basic physical factor that makes them portable. Smartphones are fascinating devices that may quickly become addictive to their owners. Using an Artificial Neural Network

(ANN) algorithm, this study aims to diagnose smartphone addiction based on self-control. The classification findings can be used to decide who should participate in a self-control campaign. The findings of data collection yielded 168 participants who filled out the questionnaires that were distributed. These participants' responses were then collated and utilized as a dataset. The SAS response scores were then totaled and classified into low, medium, and high criteria. In this investigation, this criterion is chosen as the target class. The target class in this study was split into 54, 64, and 61 data points that were classified as low, medium, and high criteria, respectively. The accuracy of the algorithm on classifying the smartphone addiction is 85.29% and 81.81 % for the training and testing, respectively. This result can be categorized as a good result.

CHAPTER 3

SYSTEM DESIGN

3.1 INTRODUCTION

In system design, Input and Output Design are critical components shaping the functionality and user experience. Input design meticulously crafts forms and screens, prioritizing purposeful effectiveness, accuracy, and user-centric simplicity. It adheres to fundamental principles, understanding the required inputs and user interactions. Objectives span designing efficient data entry procedures, reducing input volume, creating source documents, and implementing validation checks. Output design is equally pivotal, involving the identification of output types, incorporation of output controls, and prototyping report layouts. This phase ensures processed information is presented comprehensively to meet user needs.

Harmonizing these elements creates a robust information processing environment. The system becomes user-friendly, minimizing errors while maximizing usability. The strategic interplay between input and output components is foundational, fostering a seamless flow of information within the system. Ultimately, this comprehensive system design enhances efficiency, user satisfaction, and overall effectiveness, vital in the development of successful information systems in a dynamic technological landscape.

3.2 EXISTING WORK

In the existing system, implementation of machine learning algorithms is bit complex to build due to the lack of information about the data visualization. Mathematical calculations are used in existing system for Logistic Regression model building this may takes the lot of time and complexity. To overcome all this, we use machine learning packages available in the scikit-learn library.

Disadvantages:

1. Requires more time
2. Difficult to handle.

3.3 PROPOSED SYSTEM

Many machine learning algorithms are available for prediction of smartphone addiction Some of the machine learning algorithm are Decision Tree, Random Forest We used proposed and compute best method for diagnosis a comparative study of machine learning techniques for smartphone addiction detection In this stage we have first implement these dataset and the implement algorithm individual then we are combine these results and an compute the Accuracy.

Advantages:

1. Requires less time
2. Good score
3. Easy to handle

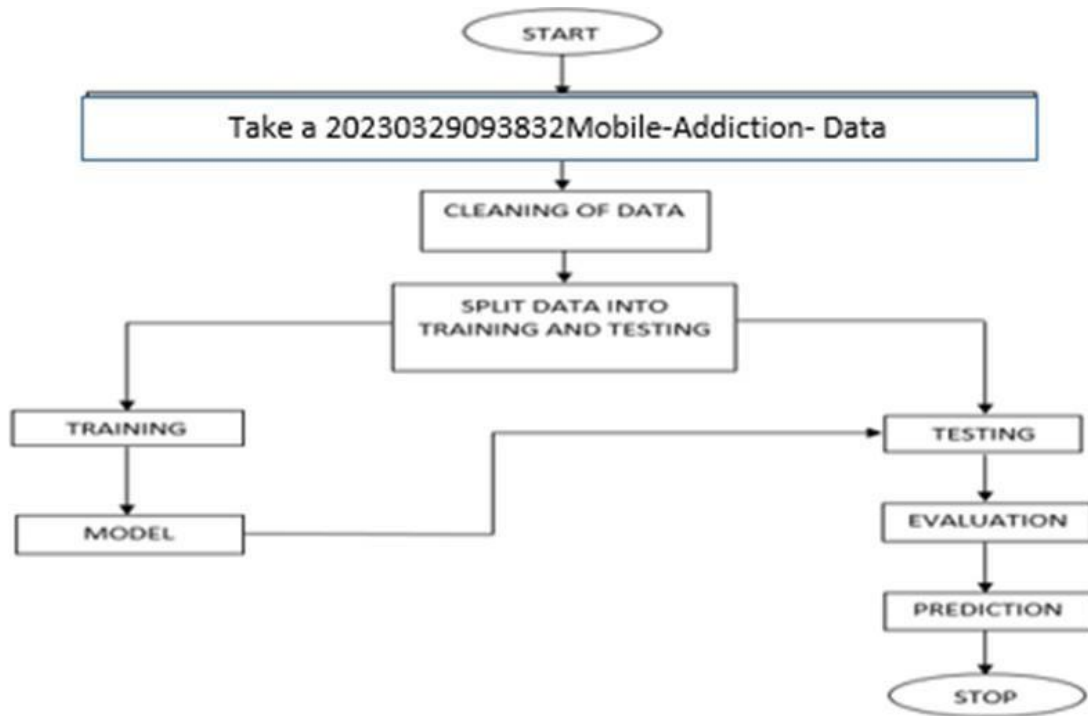


Fig 1.1:Block diagram

3.4 METHODOLOGY

1. Decision Tree:

A tree has many analogies in real life, and turns out that it has influenced a wide area of machine learning, covering both classification and regression. In decision analysis, a decision tree can be used to visually and explicitly represent decisions and decision making. As the name goes, it uses a tree-like model of decisions. Though a commonly used tool in data mining for deriving a strategy to reach a particular goal.

A decision tree is drawn upside down with its root at the top. In the image on the left, the bold text in black represents a condition/internal node, based on which the tree splits into branches/ edges. The end of the branch that doesn't split anymore is the decision/leaf, in this case, whether the passenger died or survived, represented as red and green text respectively.

Although, a real dataset will have a lot more features and this will just be a branch in a much bigger tree, but you can't ignore the simplicity of this algorithm. The feature importance is clear and relations can be viewed easily. This methodology is more commonly known as learning decision tree from data and above tree is called Classification tree as the target is to classify passenger as survived or died. Regression trees are represented in the same manner, just they predict continuous values like price of a house. In general, Decision Tree algorithms are referred to as CART or Classification and Regression Trees.

So, what is actually going on in the background? Growing a tree involves deciding on which features to choose and what conditions to use for splitting, along with knowing when to stop. As a tree generally grows arbitrarily, you will need to trim it down for it to look beautiful. Let's start with a common technique used for splitting.

2. Random Forest :

A random forest is a machine learning technique that's used to solve regression and classification problems. It utilizes ensemble learning, which is a technique that combines many classifiers to provide solutions to complex problems.

A random forest algorithm consists of many decision trees. The 'forest' generated by the random forest algorithm is trained through bagging or bootstrap aggregating. Bagging is an ensemble meta-algorithm that improves the accuracy of machine learning algorithms.

The (random forest) algorithm establishes the outcome based on the predictions of the decision trees. It predicts by taking the average or mean of the output from various trees. Increasing the number of trees increases the precision of the outcome.

A random forest eradicates the limitations of a decision tree algorithm. It reduces the over fitting of datasets and increases precision. It generates predictions without requiring many configurations in packages (like Scikit-learn).

Features of a Random Forest Algorithm:

- It's more accurate than the decision tree algorithm.
- It provides an effective way of handling missing data.
- It can produce a reasonable prediction without hyper-parameter tuning.
- It solves the issue of over fitting in decision trees.
- In every random forest tree, a subset of features is selected randomly at the node's splitting point.

Decision trees are the building blocks of a random forest algorithm. A decision tree is a decision support technique that forms a tree-like structure. An overview of decision trees will help us understand how random forest algorithms work.

A decision tree consists of three components: decision nodes, leaf nodes, and a root node. A decision tree algorithm divides a training dataset into branches, which further segregate into other branches. This sequence continues until a leaf node is attained. The leaf node cannot be segregated further.

The nodes in the decision tree represent attributes that are used for predicting the outcome. Decision nodes provide a link to the leaves. The following diagram shows the three types of nodes in a decision tree.

The information theory can provide more information on how decision trees work. Entropy and information gain are the building blocks of decision trees. An overview of these fundamental concepts will improve our understanding of how decision trees are built.

Entropy is a metric for calculating uncertainty. Information gain is a measure of how uncertainty in the target variable is reduced, given a set of independent variables.

The information gain concept involves using independent variables (features) to gain information about a target variable (class). The entropy of the target variable (Y) and the conditional entropy of Y (given X) are used to estimate the information gain. In this case, the conditional entropy is subtracted from the entropy of Y.

Information gain is used in the training of decision trees. It helps in reducing uncertainty in these trees. A high information gain means that a high degree of uncertainty (information entropy) has been removed. Entropy and information gain are important in splitting branches, which is an important activity in the construction of decision trees.

Let's take a simple example of how a decision tree works. Suppose we want to predict if a customer will purchase a mobile phone or not. The features of the phone form the basis of his decision. This analysis can be presented in a decision tree diagram.

The root node and decision nodes of the decision represent the features of the phone mentioned above. The leaf node represents the final output, either *buying* or *not buying*. The main features that determine the choice include the price, internal storage, and Random Access Memory (RAM). The decision tree will appear as follows.

Applying decision trees in random forest

The main difference between the decision tree algorithm and the random forest algorithm is that establishing root nodes and segregating nodes is done randomly in the latter. The random forest employs the bagging method to generate the required prediction.

Bagging involves using different samples of data (training data) rather than just one sample. A training dataset comprises observations and features that are used for making predictions. The decision trees produce different outputs, depending on the training data fed to the random forest algorithm. These outputs will be ranked, and the highest will be selected as the final output.

Our first example can still be used to explain how random forests work. Instead of having a single decision tree, the random forest will have many decision trees. Let's assume we

have only four decision trees. In this case, the training data comprising the phone's observations and features will be divided into four root nodes.

The root nodes could represent four features that could influence the customer's choice (price, internal storage, camera, and RAM). The random forest will split the nodes by selecting features randomly. The final prediction will be selected based on the outcome of the four trees.

The outcome chosen by most decision trees will be the final choice. If three trees predict *buying*, and one tree predicts *not buying*, then the final prediction will be *buying*. In this case, it's predicted that the customer will buy the phone.

LOGISTIC REGRESSION

Logistic Regression was used in the biological sciences in early twentieth century. It was then used in many social science applications. Logistic Regression is used when the dependent variable (target) is categorical.

For example,

To predict whether an email is spam (1) or (0)

Whether the tumor is malignant (1) or not (0)

Consider a scenario where we need to classify whether an email is spam or not. If we use linear regression for this problem, there is a need for setting up a threshold based on which classification can be done. Say if the actual class is malignant, predicted continuous value 0.4 and the threshold value is 0.5, the data point will be classified as not malignant which can lead to serious consequence in real time.

From this example, it can be inferred that linear regression is not suitable for classification problem. Linear regression is unbounded, and this brings logistic regression into picture. Their value strictly ranges from 0 to 1.

Purpose and examples of logistic regression:

Logistic regression is one of the most commonly used machine learning algorithms for binary classification problems, which are problems with two class values, including predictions such as “this or that,” “yes or no” and “A or B.”

The purpose of logistic regression is to estimate the probabilities of events, including determining a relationship between features and the probabilities of particular outcomes.

One example of this is predicting if a student will pass or fail an exam when the number of hours spent studying is provided as a feature and the variables for the response has two values: pass and fail.

Organizations can use insights from logistic regression outputs to enhance their business strategies so they can achieve their business goals, including reducing expenses or losses and increasing ROI in marketing campaigns, for example.

An e-commerce company that mails expensive promotional offers to customers would like to know whether a particular customer is likely to respond to the offers or not. For example, they’ll want to know whether that consumer will be a “responder” or a “non responder.” In marketing, this is called propensity to respond modeling.

Likewise, a credit card company develops a model to decide whether to issue a credit card to a customer or not will try to predict whether the customer is going to default or not on the credit card based on such characteristics as annual income, monthly credit card payments and number of defaults. In banking parlance, this is known as default propensity modeling.

Uses of logistic regression:

Logistic regression has become particularly popular in online advertising, enabling marketers to predict the likelihood of specific website users who will click on particular advertisements as a yes or no percentage.

- Logistic regression can also be used in:
- Healthcare to identify risk factors for diseases and plan preventive measures.
- Weather forecasting apps to predict snowfall and weather conditions.
- Voting apps to determine if voters will vote for a particular candidate.

Insurance to predict the chances that a policy holder will die before the term of the policy expires based on certain criteria, such as gender, age and physical examination.

Banking to predict the chances that a loan applicant will default on a loan or not, based on annual income, past defaults and past debts.

Logistic regression vs. linear regression:

The main difference between logistic regression and linear regression is that logistic regression provides a constant output, while linear regression provides a continuous output.

In logistic regression, the outcome, such as a dependent variable, only has a limited number of possible values. However, in linear regression, the outcome is continuous, which means that it can have any one of an infinite number of possible values.

Logistic regression is used when the response variable is categorical, such as yes/no, true/false and pass/fail. Linear regression is used when the response variable is continuous, such as number of hours, height and weight.

For example, given data on the time a student spent studying and that student's exam scores, logistic regression and linear regression can predict different things.

With logistic regression predictions, only specific values or categories are allowed. Therefore, logistic regression can predict whether the student passed or failed. Since linear regression predictions are continuous, such as numbers in a range, it can predict the student's test score on a scale of 0 -100.

INPUT DESIGN:

In an information system, input is the raw data that is processed to produce output. During the input design, the developers must consider the input devices such as PC, MICR, OMR, etc. Therefore, the quality of system input determines the quality of system output. Well- designed input forms and screens have following properties –

- It should serve specific purpose effectively such as storing, recording, and retrieving the information.
- It ensures proper completion with accuracy.
- It should be easy to fill and straightforward.
- It should focus on user's attention, consistency, and simplicity.

All these objectives are obtained using the knowledge of basic design principles regarding –

- o What are the inputs needed for the system?
 - o How end users respond to different elements of forms and screens.
- Objectives for Input Design:

The objectives of input design are

- To design data entry and input procedures
- To reduce input volume
- To design source documents for data capture or devise other data capture methods
- To design input data records, data entry screens, user interface screens, etc.
- To use validation checks and develop effective input controls.

OUTPUT DESIGN:

The design of output is the most important task of any system. During output design, developers identify the type of outputs needed, and consider the necessary output controls and prototype report layouts.

Objectives of Output Design:

The objectives of input design are:

- . To develop output design that serves the intended purpose and eliminates the production of unwanted output.
- . To develop the output design that meets the end user's requirements.
- . To deliver the appropriate quantity of output.
- . To form the output in appropriate format and direct it to the right person.
- . To make the output available on time for making good decisions.

3.5 SYSTEM ARCHITECTURE

The dataset we use for the proposed project has been taken from Kaggle. But this data set is in raw format. The data set is a collection of values. The initial step is to convert raw data into processed data. Which is done by feature extraction, since the raw data collected have multiple attributes but only some of those attributes are needed for the prediction. Feature extraction is a reduction process. The structure, behavior and views of a system is given by structural model.

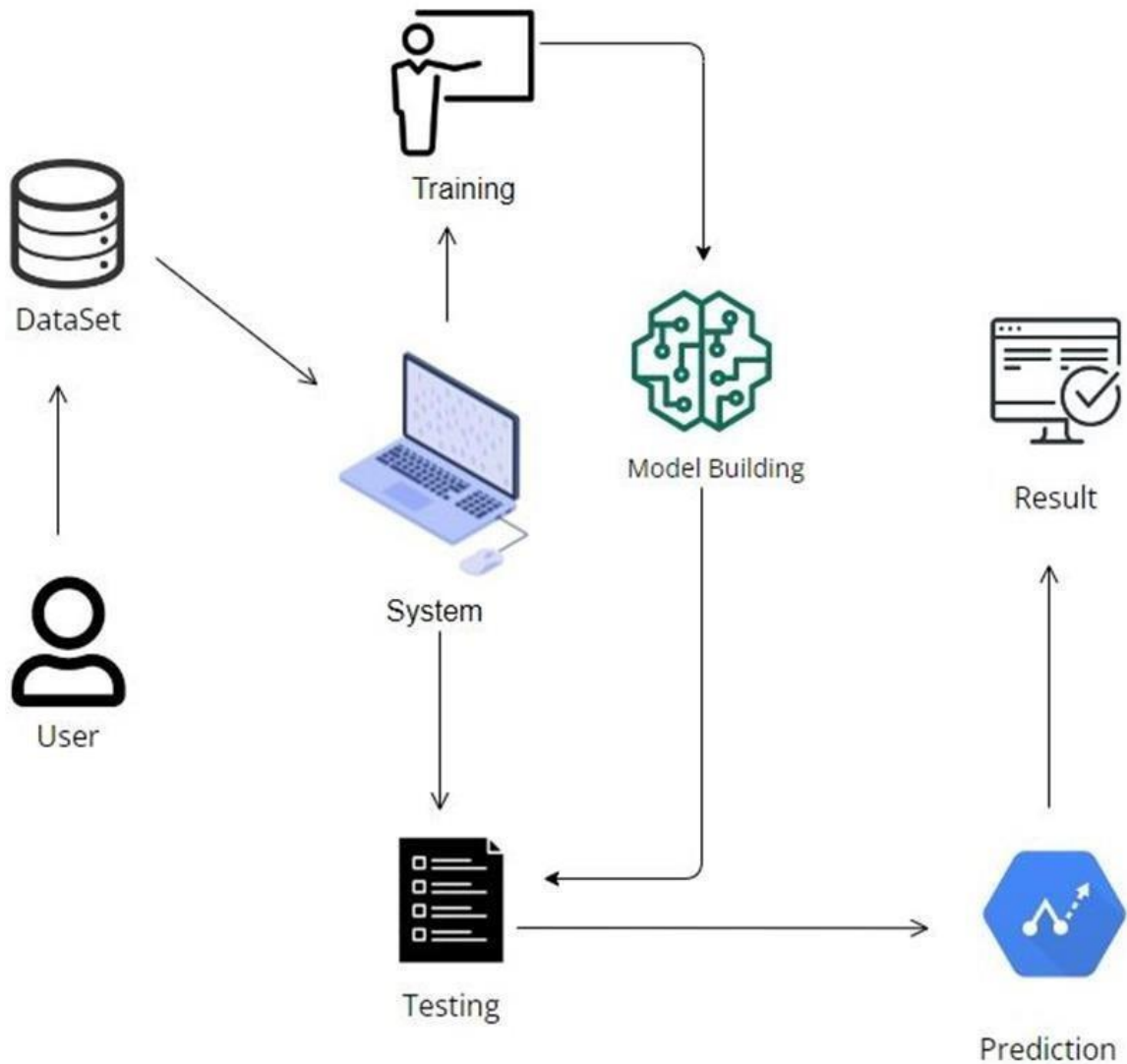


Fig 1.2 System Architecture

The above figure gives the demonstration on the dataset extraction and refining the raw dataset by categorizing into two phases of training and testing data.

From the given dataset a well modified categorization is extracted and a graph set is plotted to gain the required output which gives the addiction of smartphone prediction range.

3.6 SYSTEM REQUIREMENTS

H/W Configuration:

- Processor - I3/Intel Processor
- Hard Disk - 160 GB
- RAM - 8 GB

S/W Configuration:

- Operating System : Windows 7/8/10 .
- Server side Script : HTML, CSS & JS.
- IDE : Pycharm.
- Libraries Used : Numpy, IO, OS, Django, keras.
- Technology : Python 3.6+.

FUNCTIONAL REQUIREMENTS

These are the requirements that the end user specifically demands as basic facilities that the system should offer. All these functionalities need to be necessarily incorporated into the system as a part of the contract. These are represented or stated in the form of input to be given to the system, the operation performed and the output expected. They are basically the requirements stated by the user which one can see directly in the final product, unlike the non-functional requirements.

Examples of functional requirements:

- 1) Authentication of user whenever he/she logs into the system

- 2) System shutdown in case of a cyber-attack
- 3) A verification email is sent to user whenever he/she register for the first time on some software system.

NON-FUNCTIONAL REQUIREMENTS

These are basically the quality constraints that the system must satisfy according to the project contract. The priority or extent to which these factors are implemented varies from one project to other. They are also called non-behavioral requirements. They basically deal with issues like:

- Portability
- Security
- Maintainability
- Reliability
- Scalability
- Performance
- Reusability
- Flexibility

Examples of non-functional requirements:

- 1) Emails should be sent with a latency of no greater than 12 hours from such an activity.
- 2) The processing of each request should be done within 10 seconds
- 3) The site should load in 3 seconds whenever of simultaneous users are > 10000

CHAPTER 4

IMPLEMENTATION AND ANALYSIS

4.1 FEASIBILITY STUDY

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub-assemblies, assemblies and/or a finished product. It is the process of exercising software with the intent of ensuring that the

Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

4.2 DATASET DESCRIPTION

This is the fundamental module before starting of the project. The dataset is a group of data that are mended together to show the data variations in a time span to undergo further estimation and the source of the resources and its outcome for the later time of evaluation. It generates the result optimization and gives a feasible time period to customize and get the flow to the derivation.

This increases and are used in the level of research and finding the best suitable resource out of the same the resources have to be finely estimated and derived for the best possible outcome and the finest the value become the better is the level of extraction and closure is the best yield values that needs to be considered.

DATA COLLECTION :

Data collection is one of the important and basic things in our project. The right dataset must be provided to get robust results. We will be taking and analyzing data from Kaggle. After that, seeing the accuracy, we will use the data in our model. The size of overall dataset is 80.0 KB (81,920 bytes).

DATA PREPROCESSING:

In preprocessing first of all we will check whether there is any Nan values. If any Nan values is present we will fill the Nan values with different fillna techniques like bfill, ffill, mode, and mean. Here we used the ffill (front fill) technique on our project.

DATA ABSTRACTION:

Abstraction is the finding of the resource to its best to categorized the above dataset and learning the best out of it. Abstraction of the data is the integral part to the flow. All the data are a huge set of chunks which on processing can limited the yield result and the computational mean too. Thus, with the available resources the data yield had to be derivative. Abstraction of the dataset is to customize the data set and finding the best suitable constraints to take into consideration and the unwanted resources are the dump which will be dumped and the supreme cluster is created with the valuable constrains and a pattern is needed to be derived from the same.

Data are cleared on this level for the beginning of the process. The valuable data are the set that brings the value to the data set for a better

understanding and giving a better yield and production by evaluating the same.

This is a feature abstraction module to extract the featurings of the dataset. This is a feature model process where all the feasible resources are categorized and the same will be in use for the featurings.

TRAINING DATASET:

Irrespective of the algorithm we select the training is the same for every algorithm. Given a dataset we split the data into two parts training and testing, the reason behind doing this is to test our model/algorithm performance just like the exams for a student the testing is also exam for the model. We can split data into anything we want but it is just good practice to split the data such that the training has more data than the testing data, we generally split the data. And for training and testing there are two variables X and Y in each of them, the X is the features that we use to predict the Y target and same for the testing also. Then we call the `.fit ()` method on any given algorithm which takes two parameters i.e., X and Y for calculating the math and after that when we call the `.predict ()` giving our testing X as parameter and checking it with the accuracy score giving the testing Y and predicted X as the two parameters will get our accuracy score and same steps, these are just checking for how good our model performed on a given dataset.

TEST DATASET:

These are the sets of data that gives the result after learning from the data. This is the test generation with the output result. Results are generated in each phase of testing. This is also termed as the testing phase. Now a new set of datasets are passed which are deliberately like the training dataset and the efficiency of the same is calculated.

Over-Fitting of the dataset. Validation of the same with the effective constraints and hyper parameters are checked. This phase is training and the output is evaluated with the set of training. After each process of computation, the set of data are trained and efficiency of the same is measured and is evaluated with the others.

Various batches of the test is implemented to get to the level of accuracy and derive result to fetch and yield for the best performance and to be true to the effectiveness of the data which is not biased with any constraints available. This determines the efficiency of the system which is must for the predictions.

4.3 MODULE DESCRIPTION

Table 2: List of modules with explanation

<pre>import pandas as pd</pre>	pandas (all lowercase) is a popular Python-based data analysis toolkit which can be imported using <code>import pandas as pd</code> . It presents a diverse range of utilities, ranging from parsing multiple file formats to converting an entire data table into a NumPy matrix array. The <code>as pd</code> portion of the code then tells Python to give pandas the alias of <code>pd</code> . This allows you to use pandas functions by simply typing <code>pd</code> .
<pre>import seaborn as sns</pre>	The <code>import seaborn</code> portion of the code tells Python to bring the Seaborn library into your current environment. The <code>as sns</code> portion of the code then tells Python to give Seaborn the alias of <code>sns</code> .
<pre>import LabelEncoder</pre>	The <code>LabelEncoder</code> class is commonly used for encoding categorical labels with numerical values, which is often necessary for certain machine learning algorithms.
<pre>from google.colab import files</pre>	IMPORTING FILES USING IMPORT COMMAND IN GOOGLE COLABORATORY
<pre>from sklearn.preprocessing</pre>	The <code>sklearn.preprocessing</code> package provides several common utility functions and transformer classes to change raw feature vectors into a representation that is more suitable for the downstream estimators. In general, learning algorithms benefit from standardization of the data set.

<pre>from imblearn.over_sampling</pre>	Imblearn library is specifically designed to deal with imbalanced datasets. It provides various methods like undersampling, oversampling, and SMOTE to handle and removing the imbalance from the dataset.
<pre>import RandomOverSampler</pre>	Class to perform random over-sampling. Object to over-sample the minority class(es) by picking samples at random with replacement. Random Oversampling includes selecting random examples from the minority class with replacement and supplementing the training data with multiple copies of this instance.
<pre>import sklearn.model_selection</pre>	model_selection is used to split our data into train and test sets where feature variables are given as input in the method. test_size determines the portion of the data which will go into test sets and a random state is used for data reproducibility.
<pre>import train_test_split</pre>	train_test_split splits the data and returns a list which contains four NumPy arrays, while train_size = .75 puts 75 percent of the data into a training set and the remaining 25 percent into a testing set.
<pre>import sklearn.metrics</pre>	The sklearn. metrics module implements several loss, score, and utility functions to measure classification performance.

<pre>import accuracy_score</pre>	<p>The accuracy_score function of the sklearn.metrics package calculates the accuracy score for a set of predicted labels against the true labels.</p>
----------------------------------	--

4.4 RESULTS:

FRONT END :

Home Page: User can view the Home page.



Fig2.1: Home Page

About: This is The small information about project

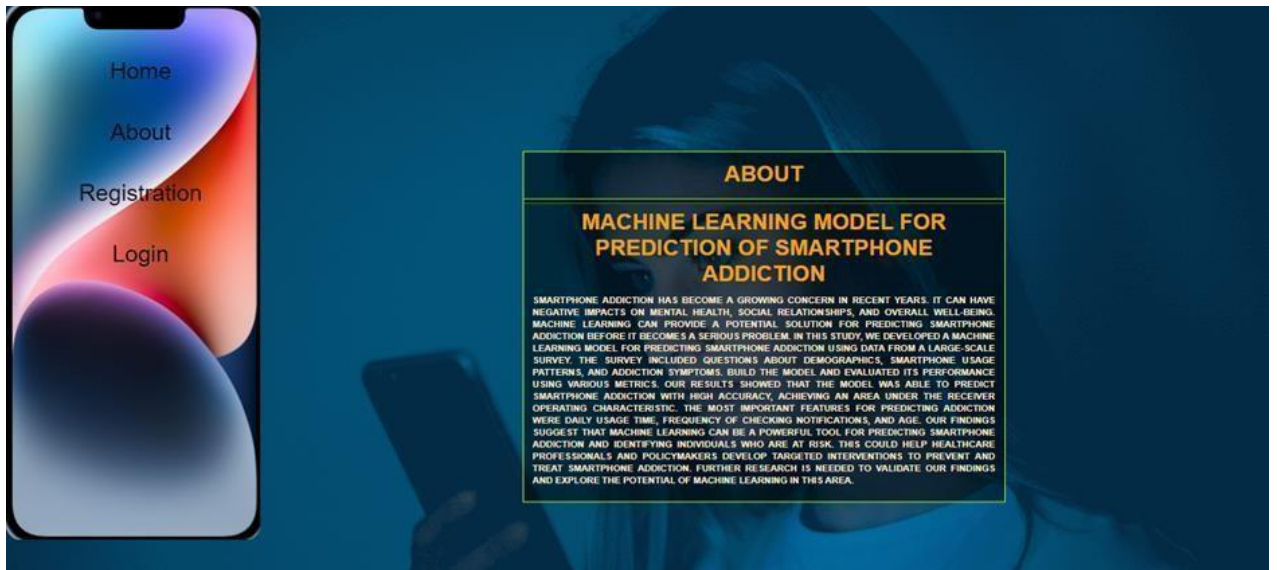


Fig2.2: About Page.

User Registration page: User can register with required details.

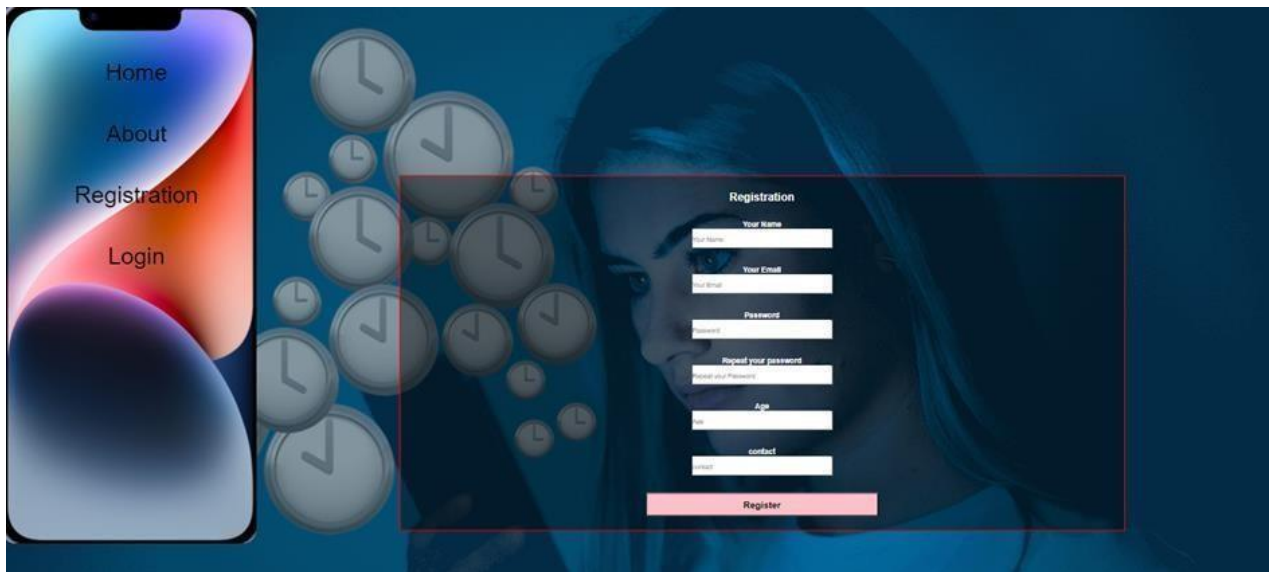


Fig 2.3: registration page

User Login: User can login with valid credentials.

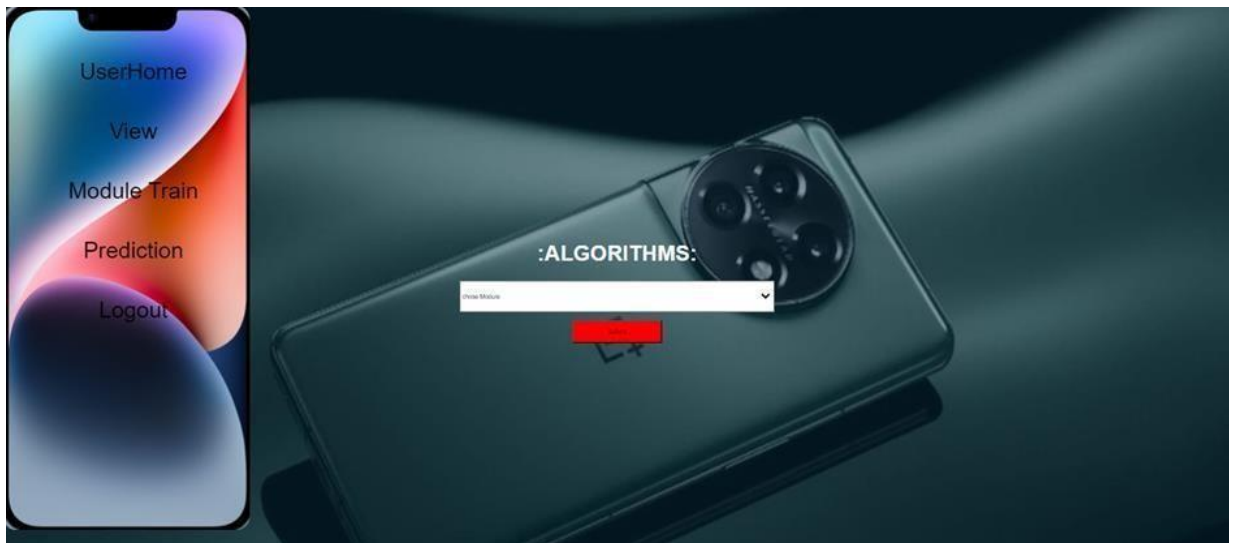


Fig 2.6: Model Page.

Prediction: User can give a input and view The Predicted Result

Fig 2.7: Prediction Page.

BACK END:

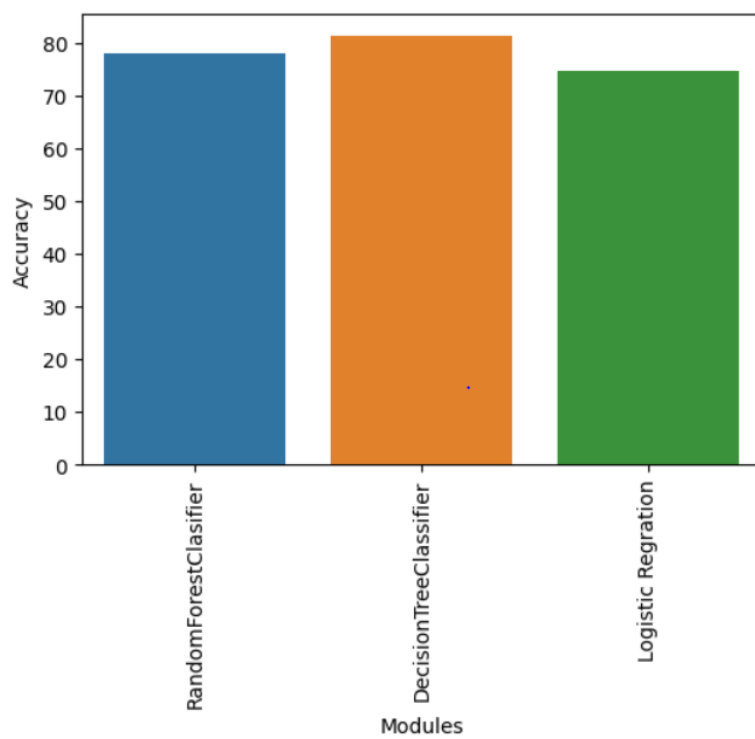


fig 2.8. accuracy graph

Random Forest Classifier	77.966102
Decision Tree Classifier	81.355932
Logistic Regression	74.576271

CHAPTER 5

CONCLUSIO

N

5.1 CONCLUSION

To conclude stock is an unpredictable mechanism which follows the segments of chain and the dependencies of the same are unpredictable. It is defined to be a curve which keeps on changing and turning the price from low to high and vice-versa.

As the integration of the same is higher with other dependencies so leaving one dependency compromises the level of accuracy. Accuracy is not the term used over in stock as the actual prediction is not possible for any fiscal days it keeps on changing and turning the tables day and night. Having higher component assets and the dependencies makes it more feasible and flexible in nature causing it even harder to predict. The approx. value are taken into consideration and the hit or profit or the gain rate is calculated for the same.

In the project various high level machine learning algorithms are implemented and integrated and the output is generated from the same making a user visible with the outputs in the form of graph which makes it easier for them to see and interpret what's the scenario and they can decide on the same to invest and get the benefit out of it.

The proposed software takes the raw set of data from the dataset or the .csv file and process it. The cleaning and cleansing of data are done and then

further processed to gain the effective outcomes. After the computational mean the output is displayed in the screen in the form of graph.

In present, there are several models to predict the stock market but they are less accurate. We proposed a model that uses RNN and LSTM to predict the trend in stock prices that would be more accurate. LSTM introduces the memory cell, a unit of computation that replaces traditional artificial neurons in the hidden layer of the network. In this work by increasing the Epochs and batch size, the accuracy of prediction is more.

In proposed method, we are using a test data that is used to predict which gives results that are more accurate with the test data. The proposed method is capable of tracing and prediction of stock market and the prediction will produce higher and accurate results. In our above model we are getting accurate results which will be more useful to stock analysts, Business analysts, Stock Market Investors.

5.2 FUTURE ENHANCEMENTS

Future enhancements for our smartphone addiction prediction application could encompass several avenues of development. Firstly, we can refine the model by incorporating more diverse and real-time data sources, such as GPS location and user interaction patterns, to further improve accuracy. Additionally, integrating behavioral therapy and mental health resources within the application can offer immediate support to individuals identified as potentially addicted. This could include personalized recommendations for reducing screen time and encouraging healthy digital habits. Collaboration with mental health professionals and institutions could facilitate more comprehensive assessments and interventions. Moreover, continuous model monitoring and updates would ensure its relevance in an ever-evolving digital landscape. Ultimately, our application can evolve into a holistic platform for promoting digital well-being, with features to prevent, diagnose, and support users in managing smartphone addiction effectively.

APPENDIX A – SAMPLE CODING

```
import pandas as pd

from matplotlib import pyplot as plt

import seaborn as sns

from sklearn.preprocessing import LabelEncoder

from imblearn.over_sampling import RandomOverSampler

from sklearn.model_selection import train_test_split

from sklearn.metrics import accuracy_score
```

```
df = pd.read_csv("20230329093832Mobile-Addiction-.csv")

df.head()

df.tail()

df.info()

df.shape

df.isnull().sum(

)

df.duplicated()

df.describe()

df.isnull().sum().sum()

df.columns

df['whether you are addicted to phone?'].value_counts()

df.info()

for i in col:
```

```

df['Can you live a day without phone ? '].fillna(method='bfill',inplace=True)
df['whether you are addicted to phone?'].fillna(method='bfill',inplace=True)
df.isnull().sum()
df.head()
le = LabelEncoder()
for i in col:
    df[i]=le.fit_transform(df[i])
df.head()
df['whether you are addicted to phone?'].value_counts()
df1=df.select_dtypes(exclude=['object', 'bool'])
for column in df1:
    plt.figure(figsize=(17,1))
    sns.histplot(data=df1,
                 x=column)
plt.show()
df1=df.select_dtypes(exclude=['object', 'bool'])
for column in df1:
    plt.figure(figsize=(7,1))
    sns.countplot(data=df1, x=column)
plt.show()
df.info()
sns.lineplot(df["whether you are addicted to phone?"])
sns.histplot(df["whether you are addicted to phone?"])
sns.kdeplot(df["whether you are addicted to phone?"])
sns.pairplot(df)
df.drop('Timestamp', axis = 1,inplace = True)
x = df.drop(['whether you are addicted to phone?'], axis = 1)
y = df['whether you are addicted to phone?']
Oversample = RandomOverSampler(random_state=72)
x_sm, y_sm =
Oversample.fit_resample(x[:100],y[:100])
x_train, x_test, y_train, y_test = train_test_split(x_sm, y_sm, test_size
= 0.3, random_state= 72)
x_train.shape
e
x_test.shape
y_train.shape
e
y_test.shape
from sklearn.ensemble import RandomForestClassifier
re = RandomForestClassifier(random_state=72)
re.fit(x_train,y_train)
re_pred = re.predict(x_test)
ac = accuracy_score(y_test,re_pred)

```

```
ac
from sklearn.tree import DecisionTreeClassifier
de = DecisionTreeClassifier()
```



```

de.fit(x_train,y_train)
de_pred = de.predict(x_test)
ac1 = accuracy_score(y_test,de_pred)
ac1
from sklearn.linear_model import LogisticRegression
le = LogisticRegression()
le.fit(x_train,y_train)
le_pred = le.predict(x_test)
ac2 = accuracy_score(y_test,le_pred)
ac2
graph =
pd.DataFrame({'Modules': ['RandomForestClasifier', 'DecisionTreeClassifier', 'Lo
gistic Regration'],
              'Accuracy': [ac*100, ac1*100, ac2*100]})

graph
plt.figure(figsize=(6,4))
sns.barplot(y=graph.Accuracy,x=graph.Modules
) plt.xticks(rotation='vertical')
plt.show()

```

APPENDIX A – SAMPLE OUTPUT

TEST DATASET:

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
Timestamp	Full Name	Gender :	Do you use	Do you bur	Does your	When you	Do you wc	Do you tak	Do you use	Do you oft	Do you ch	Do you ke	Do you ch	Do y
2023/02/1	Dabbeta G	MALE	Yes	Yes	Yes	Yes;No	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
2023/02/2	Mir Junaid	MALE	Yes	No	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No	Yes
2023/02/2	Dev	MALE	Yes	No	No	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No	Yes
2023/02/2	A.Ruthvik	MALE	Yes	No	Yes	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes	Yes
2023/02/2	Vishnu	MALE	Yes	Yes	Yes	Yes	Yes	No	Yes	Yes	Yes		No	No
2023/02/2	Sathvik	MALE	No	No	No	Yes	Yes	No	No	No	Yes	No	No	Yes
2023/02/2	K Ruthvekl	MALE	No	No	Yes	No	No	Yes	Yes	Yes	Yes	Yes	Yes	Yes
2023/02/2	Karthik Ko	MALE	Yes	No	Yes	No	Yes	No	Yes	Yes	Yes	No	No	Yes
2023/02/2	Kojaaa	Prefer not	No	No	No	No	No	Yes	No	No	No	No	No	No
2023/02/2	M Venkat	MALE	Yes	Yes	Yes	Yes	No	No	No	Yes	Yes	Yes	Yes	Yes
2023/02/2	N.Jashwar	MALE	Yes	No	Yes	Yes	Yes	No	Yes	Yes	Yes	No	Yes	No
2023/02/2	Kavva Akh	MALE	Yes	No	Yes	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes	Yes
2023/02/2	Akula Teje	MALE	Yes	Yes	No	No	No	Yes	No	Yes	Yes	Yes	Yes	No
2023/02/2	Ram	MALE	Yes	No	Yes	No	Yes	Yes	No	No	No	No	Yes	No
2023/02/2	Muni	MALE	Yes	Yes	No	No	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes
2023/02/2	V Ravi Kun	MALE	No	No	No	No	No	Yes	No	No	No	No	No	No
2023/02/2	Naga Mun	MALE	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No	Yes
2023/02/2	Srisai Mad	MALE	Yes	No	Yes	Yes	Yes	No	Yes	Yes	Yes	Yes	No	Yes
2023/02/2	Prudhvitha	Female	Yes	Yes	Yes	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes	Yes
2023/02/2	Sree Naya	MALE	Yes	No	Yes	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes	Yes
2023/02/2	GIRIBOIN	MALE	Yes	No	Yes	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes	Yes

Fig 3.1: Test dataset

TRAIN DATASET:

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
2023/03/0	Sai	MALE	Yes	Yes	No	Yes	Yes	No	No	Yes	No	No	Yes	No	No	No	N
2023/03/0	Sanju	Female	Yes	No	Yes	No	No	No	Yes	No	No	No	No	No	Yes	No	N
2023/03/0	Sathvika B	Female	Yes	Yes	Yes	Yes	No	No	Yes	No	Yes	Yes	Yes	Yes	No	No	N
2023/03/0	Vaishnavi	Female	Yes	No	Yes	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	N
2023/03/0	Pranay	MALE	Yes	No	No	Yes	Yes	Yes	Yes	Yes	No	No	Yes	No	No	Yes	N
2023/03/0	Sharath	MALE	No	No	No	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No	Yes	Yes	N
2023/03/0	Shiva koyir	Female	Yes	No	Yes	No	Yes	No	Yes	No	Yes	No	No	Yes	Yes	Yes	N
2023/03/0	Eshaa	Female	Yes	No	No	Yes	Yes	No	No	Yes	Yes	Yes	Yes	Yes	Yes	Yes	N
2023/03/0	Chukka	Female	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No	Yes	Yes	Yes	Y
2023/03/0	Akshaya	Female	Yes	No	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No	Yes	Yes	Yes	
2023/03/0	Sahasra	Female	Yes	Yes	No	Yes	Yes	No	Yes	Yes	Yes	Yes	No	Yes	Yes	No	Y
2023/03/0	Rishika	Female	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Y
2023/03/0	Preethi	Female	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Y

Fig 3.2: Train dataset

GOOGLE COLLABRATORY WORKSPACE:

Project.ipynb ☆

File Edit View Insert Runtime Tools Help Last edited on November 16

Comment Share Settings

les

+ Code + Text

```
[ ] import pandas as pd
from matplotlib import pyplot as plt
import seaborn as sns
from sklearn.preprocessing import LabelEncoder
from imblearn.over_sampling import RandomOverSampler
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score
```

```
[ ] # read The csv file
df = pd.read_csv("20230329093832Mobile-Addiction-.csv")
```

```
[ ] # Show The First Five Row
df.head()
```

Timestamp	Full Name	Gender	Do you use your phone to click pictures of class notes?	Do you buy books/access books from your mobile?	Does your phone's battery last a day?	When your phone's battery dies out, do you run for the charger?	Do you worry about losing your cell phone?	Do you take your phone to the bathroom?	Do you use your phone in any social gathering (parties)?	...	Do you check your phone just before going to sleep/just after waking up?	Do you keep your phone right next to you while sleeping?	Do you check emails, missed calls, texts during class time?
2023/02/17 2:21:53 PM GMT+5:30	Dabbeta Ganesh Kumar	MALE	Yes	Yes	Yes	Yes/No	Yes	Yes	Yes	...	Yes	Yes	Yes

sk 80.83 GB available

Fig 3.3 : Google Collaboratory Workspace

REFERENCES

1. Demir, K. & Akpinat, E. The effect of mobile learning applications on students' academic achievement and attitudes toward mobile learning. *Malays. Online J. Educ. Technol.* 6, 48–59 (2018).
2. Abadiyan, F., Hadadnezhad, M., Khosrokiani, Z., Letafatkar, A. & Akhshik, H. Adding a smartphone app to global postural re-education to improve neck pain, posture, quality of life, and endurance in people with nonspecific neck pain: A randomized controlled trial. *Trials* 22, 274 (2021).
3. Osorio-Molina, C. et al. Smartphone addiction, risk factors and its adverse effects in nursing students: A systematic review and meta-analysis. *Nurse Educ. Today* 98, 104741 (2021).
4. Osailan, A. The relationship between smartphone usage duration (using smartphone's ability to monitor screen time) with hand-grip and pinch-grip strength among young people: An observational study. *BMC Musculoskelet. Disord.* 22, 186 (2021).
5. Hitti, E., Hadid, D., Melki, J., Kaddoura, R. & Alameddine, M. Mobile device use among emergency department healthcare professionals: prevalence, utilization and attitudes. *Sci. Rep.* 11, 1917 (2021).
6. Shaygan, M. & Jaber, A. The effect of a smartphone-based pain management application on pain intensity and quality of life in adolescents with chronic pain. *Sci. Rep.* 11, 6588 (2021).
7. Sohn, S. Y., Krasnoff, L., Rees, P., Kalk, N. J. & Carter, B. The association between smartphone addiction and sleep: A UK cross-sectional study of young adults. *Front. Psych.* 12, 629407 (2021).
8. Wilkerson, G. B. et al. Wellness survey responses and smartphone app response efficiency: Associations with remote history of sport-related concussion. *Percept. Mot. Skills* 128, 714–730 (2021).
9. Thornton, L. et al. A multiple health behavior change, self-monitoring mobile app for adolescents: Development and usability study of the Health4Life App. *JMIR Format. Res.* 5, e25513 (2021).
10. Joo, E., Kononova, A., Kanthawala, S., Peng, W. & Cotten, S. Smartphone users' persuasion knowledge in the context of consumer mHealth apps: Qualitative study. *JMIR Mhealth Uhealth* 9, e16518 (2021).

[10] Joo, E., Kononova, A., Kanthawala, S., Peng, W. & Cotten, S. Smartphone users' persuasion knowledge in the context of consumer mHealth apps: Qualitative study. JMIR Mhealth Uhealth 9, e16518 (2021).

S. Prathima
<<PRATHIMA
CSE
Saveetha Engineering College
Chennai – 602105 India>>

Anitha Julian
12/03/2024
<<Dr. Anitha Julian, M.E., Ph.D.,
CSE
Saveetha Engineering College
Chennai – 602105 India>>