

DECISION SUPPORT SUPPORT FOR VENTURE CAPITAL INVESTMENT

Submitted in partial fulfillment of the requirements
for the award of
Bachelor of Engineering degree in Computer Science and Engineering
by

PUCHAKAYALA VAMSI KRISHNA(37110602)

RAYALA KRISHNA CHAITANYA (37110634)



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SCHOOL OF COMPUTING**

SATHYABAMA

**INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)**

**Accredited with Grade "A" by NAAC
JEPPIAAR NAGAR, RAJIV GANDHI
SALAI,
CHENNAI – 600 119**

MARCH - 2021



SATHYABAMA
INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)
Accredited with "A" grade by NAAC
Jeppiaar Nagar, Rajiv Gandhi Salai, Chennai – 600 119
www.sathyabama.ac.in



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this project report is the bonafide work of **Puchakayala Vamsi Krishna** (37110602) **Rayala Krishna Chaitanya** (37110634) who carried out the project entitled "**DECISION SUPPORT SYSTEM FOR VENTURE CAPITAL INVESTMENT**" under our supervision from November 2020 to March 2021

Internal Guide

Dr. R. AROUL CANESSANE, M.E., Ph.D.,

Head of the Department

Dr. S. VIGNESHWARI, M.E., Ph.D.,

Submitted for Viva voce Examination held on _____

Internal Examiner

External Examiner

DECLARATION

I **Rayala Krishna Chaitanya** here by declare that the Project Report entitled **“Decision Support For Venture Capital Investment Using Machine Learning”** is done by me under the guidance of Professor **Dr. R. AROUL CANESSANE, M.E., Ph.D.**, and **Dr. S. VIGNESHWARI, M.E., Ph.D.**, Head of the Department of CSE at Sathyabama Institute of Science and Technology is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering degree in Computer Science and Engineering.

DATE:

PLACE: CHENNAI

SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to Board of Management of **SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey our thanks to **Dr. T. Sasikala, M.E., Ph.D., Dean, School of Computing, Dr. S. Vigneswari, M.E., Ph.D.,** Head of the Department, Dept. of Computer Science and Engineering for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide **Dr. R. AROUL CANESSANE, M.E., Ph.D.,** for her valuable guidance, suggestions and constant encouragement paved way for the successful completion of my project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the Department of who were helpful in many ways for the completion of the project.

ABSTRACT

Decision model for venture capital provider and venture capitalist is established, and the effect of policy support on asset allocation and effort level of venture capitalist and the effect on asset allocation of venture capital provider is studied. The study shows that venture capitalist would enhance his effort level and venture capital company would increase the weight of investment to venture capital project because of policy support. The effect on allocation decision of venture capital provider depend on the correlation between venture capital projects and general private equity investments projects. The positive correlation will result in reducing the allocation of venture capital and negative correlation will increase the allocation of venture capital, but the amount of investment to venture capital project will increase in any case. The proposed system analysis the relationship between venture capital, corporate governance, firm capability and firm performance. Then we bring up a new path model of venture capital's impact on firm performance in the view of firm capability and corporate governance, and we use the model to describe the impact mechanism of venture capital on firm performance.

TABLE OF CONTENTS

CHAPTER No.	TITLE	PAGE No.
	ABSTRACT	v
	LIST OF FIGURES	vii
	LIST OF ABBREVIATIONS	viii
1	INTRODUCTION	1
	1.1 INTRODUCTION	1
2	LITERATURE SURVEY	3
	2.1 LITERATURE SURVEY	3
3	AIM AND SCOPE	9
	3.1 SCOPE	9
	3.2 OBJECTIVE	9
4	SYSTEM DESIGN AND METHODOLOGY	10
	4.1 EXISTING SYSTEM	10
	4.2 EXISTING SYSTEM DISADVANTAGES	10
	4.3 PROPOSED SYSTEM	11
	4.4 ADVANTAGES OF PROPOSED SYSTEM	11
	4.5 SYSTEM ARCHITECTURE	12
	4.6 ALGORITHM	12
	4.7 SYSTEM REQUIREMENTS	13
	4.8 TECHNOLOGY USED	13
	4.9 INTRODUCTION OF PYTHON	14
5	RESULTS AND DISCUSSION	21
6	CONCLUSION AND FUTURE WORK	35
	REFERENCES	36
	APPENDICES	38

LIST OF FIGURES

FIGURE No.	FIGURE NAME	PAGE No.
4.1	SYSTEM ARCHITECTURE	12
	TESTING PROCESS	23
	ACTIVITY DIAGRAM	25
	MODULE PROCESS	27
	PERFORMANCE TESTING	33

ABBREVIATIONS

Data Analysis - DA

ANN - Artificial Neural Network

SGD - Stochastic Gradient Descent

K-NN - K-Nearest Neighbors

SVM - Support Vector Machine

DT - Decision Tree

CLE - Collaborative Learning Environment

SJT - Situational Judgment task

CP - Computational Psychometrics

CNNs - Convolutional Neural Networks

CIA - Confidentiality, Integrity, Availability

IDE - Integrated Development Environment

TDD - Test Driven Development

CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION

Module 1 : Data Analysis

Data analysis (DA) is an investigative process in which you use summary statistics and graphical tools to get to know your data and understand what you can learn from it. With DA, you can uncover patterns in your data, understand potential relationships between variables, and find anomalies, such as outliers or unusual observations. The goal is to generate interesting questions or hypotheses that you can test using more formal statistical methods. It is always better to explore each data set using multiple exploratory techniques and compare the results. Once the data set is fully understood, it is quite possible that data scientist will have to go back to data collection and cleansing phases in order to transform the data set according to the desired business outcomes. The goal of this step is to become confident that the data set is ready to be used in a machine learning algorithm.

Module 2 : Data Pre-Handling

It is a data mining technique that transforms raw data into an understandable format. Raw data(real world data) is always incomplete and that data cannot be sent through a model. That would cause certain errors. That is why we need to preprocess data before sending through a model. Data cleaning refers to techniques to 'clean' data by removing outliers, replacing missing values, smoothing noisy data, and correcting inconsistent data. Many techniques are used to perform each of these tasks, where each technique is specific to user's preference or problem set. Below, each task is explained in terms of the techniques used to overcome it.

In order to deal with missing data, multiple approaches can be used. Let's look at each of them. Removing the training example: You can ignore the training example if the output label is missing (if it is a classification problem). This is usually discouraged as it leads to loss of data, as you are removing the attribute values that can add value to data set as well. Filling in missing value manually: This approach is time consuming, and not recommended for huge data sets. Using a standard value to replace the

missing value: The missing value can be replaced by a global constant such as 'N/A' or

'Unknown'. This is a simple approach, but not foolproof. Using central tendency (mean, median, mode) for attribute to replace the missing value: Based on data distribution, mean (in case of normal distribution) or median (for non-normal distribution) can be used to fill in for the missing value. Using central tendency (mean, median, mode) for attribute belonging to same class to replace the missing value: This is the same as method 4, except that the measures of central tendency are specific to each class. Using the most probable value to fill in the missing value: Using algorithms like regression and decision tree, the missing values can be predicted and replaced.

Module 3 : Feature Extraction

Feature Selection is one of the core concepts in machine learning which hugely impacts the performance of your model. The data features that you use to train your machine learning models have a huge influence on the performance you can achieve. Irrelevant or partially relevant features can negatively impact model performance. Feature selection and Data cleaning should be the first and most important step of your model designing. Feature Selection is the process where you automatically or manually select those features which contribute most to your prediction variable or output in which you are interested in. Having irrelevant features in your data can decrease the accuracy of the models and make your model learn based on irrelevant features. How to select features and what are Benefits of performing feature selection before modeling your data? Reduces Overfitting: Less redundant data means less opportunity to make decisions based on noise. Less misleading data means modeling accuracy improves.

Module 4 : Model Training

Models have what are known as "hyper parameters." These are the parameters that govern the model, they define how the model is created. Altering these can give us better (or perhaps worse) results. Examples include: number of neurons in each hidden layer, the number of hidden layers, the activation function, etc. Our goal here is to "tune" these hyperparameters to achieve a lower error tolerance than was possible with our first model. The simplest way to do this, in my opinion, is to increase the number of neurons in the hidden layers. I am by no means a leading source of knowledge on this topic, but I will venture far enough to say that increasing the number of neurons and/or

the number of hidden layers increases the level of abstraction with which the model can represent.

CHAPTER 2

LITERATURE SURVEY

2.1 Literature Survey

Title 1: Machine Learning Approach for Coconut Sugar Quality Assessment and Prediction

Author: Lea Monica B. Alonzo, Francheska B. Chioson, Homer S. Co, Nilo T. Bugtai

This study presents a machine learning approach to accurately assess the quality of coconut sugar using RGB values. Python and scikit-learn were used to run the following machine learning algorithms: artificial neural network (ANN), stochastic gradient descent (SGD), k-nearest neighbors (k-NN) algorithm, support vector machine (SVM), decision tree (DT) and random forest (RF). Comparisons were made between the aforementioned machine learning algorithms by evaluating the accuracy and the average running time of each training model. Results of the study show that the SGD is superior in terms of accuracy but falls short to k-NN and SVC in terms of running time. In this fashion, a plot between the accuracy and the running time was made and it was observed that algorithms with higher accuracies correspondingly have also higher running times. By this very nature, experimental results show that the SGD holds merit in accurately assessing the coconut sugar quality, despite its expense in running time.

Title 2: Hybrid approach towards the assessment of a drill condition using deep learning and the Support Vector Machine

Author: Jaroslaw Kurek

This method describes an application of a novel method which relies on applying the fusion of two sets of features obtained from deep learning and hand crafted features. Moreover, in terms of deep learning, transfer learning in deep learning has been applied. A hybrid method has been compared to classical deep learning approach, and let us state that the new approach can obtain more accuracy than regular methods. Transfer learning in deep learning should be used in case of lack of data to train the use of deep learning algorithms. We have used this method for the classification of drill wear state on the basis of drilled hole images. The specialists divided the whole data set into three classes: red, yellow, and green, which correspond to 3 stages of drill

wear. The red class corresponds to a drill which is worn out and should be replaced; the yellow class should

send a warning message to an operator for a manual check of the state of a drill; and the last one, the green one, is connected with the state of a drill which should be further used in production (it is still sharp enough). The important advantage of this approach is that it collects features from two sources: one set is automatically extracted by means of deep learning and the other is manually extracted by a researcher. The next advantage of this approach is training classification models only on the basis of a small portion of data, which in case of popular deep learning methods is too small to achieve reasonable accuracy. Hence, in order to get around this issue connected with a small portion of training data, transfer learning in deep learning has been applied. The achieved results confirm the fact that this approach can be applied in this situation.

Title 3: Machine Learning for Efficient Assessment and Prediction of Human Performance in Collaborative Learning Environments

Author: Pravin Chopade, Saad M Khan, David Edwards, and Alina von Davier

The objective of this work is to propose a machine learning-based methodology system architecture and algorithms to find patterns of learning, interaction, and relationship and effective assessment for a complex system involving massive data that could be obtained from a proposed collaborative learning environment (CLE). Collaborative learning may take place between dyads or larger team members to find solutions for realtime events or problems, and to discuss concepts or interactions during situational judgment tasks (SJT). Modeling a collaborative, networked system that involves multimodal data presents many challenges. This paper focuses on proposing a Machine Learning - (ML)-based system architecture to promote understanding of the behaviors, group dynamics, and interactions in the CLE. Our framework integrates techniques from computational psychometrics (CP) and deep learning models that include the utilization of convolutional neural networks (CNNs) for feature extraction, skill identification, and pattern recognition. Our framework also identifies the behavioral components at a micro level, and can help us model behaviors of a group involved in learning.

Title 4: Teaching Key Machine Learning Principles Using Anti-learning Datasets

Author: Chris Roadknight, Prapa Rattadilok, Uwe Aickelin

Much of the teaching of machine learning focuses on iterative hill-climbing approaches and the use of local knowledge to gain information leading to local or global maxima. In this paper we advocate the teaching of alternative methods of generalising to the best

possible solution, including a method called anti-learning. By using simple teaching methods, students can achieve a deeper understanding of the importance of validation on data excluded from the training process and that each problem requires its own methods to solve. We also exemplify the requirement to train a model using sufficient data by showing that different granularities of cross-validation can yield very different results.

Title 5: A Comprehension Based Intelligent Assessment Architecture

Author: Sadhu Prasad Kar 1, Jyotsna Kumar Mandal 2, Rajeev Chatterjee

In teaching and learning system assessment is measured to be an essential activity. In most of the automated learning systems, assessment is normally realized using automatic techniques where Multiple Choice Questions (MCQ) with respective answers and distractors are used. Conversely, in order to gauge the comprehension ability of learners, responses based on simple sentences, group of sentences and / or paragraph are necessary to be assessed by the system using automated tools / framework. This research activity is a footstep in this route, in which authors have suggested an automated assessment technique for brief answer type questions where multiple sentences are involved.

Title 6: Security and Privacy Preservation Scheme of Face Identification and Resolution Framework Using Fog Computing in Internet of Things

Author: P. Hu, H. Ning, Y. Wang, X.

Face identification and resolution technology is crucial to ensure the identity consistency of humans in physical space and cyber space. In current Internet of Things (IoT) and big data situation, the increase of applications based on face identification and resolution raises the demands of computation, communication and storage capabilities. Therefore, we have proposed the fog computing based face identification and resolution framework to improve processing capacity and save the bandwidth. However, there are some security and privacy issues brought by the properties of fog computing based framework. In this paper, we propose a security and privacy preservation scheme to solve above issues. We give an outline of the fog computing based face identification and resolution framework, and summarize the security and privacy issues. Then the authentication and session key agreement scheme, data encryption scheme, and data integrity checking scheme are proposed to solve the

issues of confidentiality, integrity, and availability in the processes of face identification and face resolution. Finally, we implement a

prototype system to evaluate the influence of security scheme on system performance. Meanwhile, we also evaluate and analyze the security properties of proposed scheme from the viewpoint of logical formal proof and the CIA (confidentiality, integrity, availability) properties of information security. The results indicate that the proposed scheme can effectively meet the requirements for security and privacy preservation.

Title 7: Survey on Privacy-Preserving Methods for Storage in Cloud Computing

Author: Neethu Mariam Joseph, Esther Daniel, N. A. Vasanthi, Ph.D

At present the mankind are progressively relying more on a number of online storage stores to back up our data or for using it in real time which gives an anywhere, anytime access. All these services bring with it, concerns of security and privacy weaknesses for all the services provided by them since the user's data are stored and maintained out of user's premises. This paper portrays the various issues associated to privacy while storing the user's data on third party service providers, which is more commonly termed as cloud service. Cloud computing refers to the fundamental infrastructure for an up-coming model of service provision that has the benefit of dropping cost by sharing computing and storage resources, united with an on-demand provisioning mechanism depending on a pay-per-use business model. Without appropriate security and privacy solutions designed for clouds this computing paradigm could become a huge failure. There is a lot of research being made to spot out the issues with these cloud service providers and cloud security in general. This paper is on regard of one of the key issue

-privacy that occur in the context of cloud computing and analyse the various works being done to solve the issues in privacy and thus to ensure privacy to outsourced data on cloud storage.

TITLE 8: Survey on Secure Storage in Cloud Computing

AUTHOR: A. Rajathi^{1*} and N. Saravanan²

Cloud Computing is an environment for providing information and resources that are delivered as a service to end-users over the Internet on demand. Thus cloud enables users to access their data from any geographical locations at any time and also has brought benefits in the form of online storage services. Cloud storage service avoids the cost expensive on software, personnel maintenance and provides better performance, less storage cost and scalability. But the maintenance of stored data in a

secure manner is not an easy task in cloud environment and especially that stored data may not be

completely trustworthy. Cloud delivers services through internet which increases their exposure to storage security vulnerabilities. However security is one of the major drawbacks that preventing several large organizations to enter into cloud computing environment. This work surveyed on several existing cloud storage frameworks, techniques and their advantages, drawbacks and also discusses the challenges that are required to implement secure cloud data storage. This survey results help to identify the future research areas and methods for improving the existing drawbacks.

TITLE 9: Security and Privacy Preservation Scheme of Face Identification and Resolution Framework Using Fog Computing in Internet of Things

AUTHOR : Yanna Wang, and Xuanxia Yao

Face identification and resolution technology is crucial to ensure the identity consistency of humans in physical space and cyber space. In current Internet of Things (IoT) and big data situation, the increase of applications based on face identification and resolution raises the demands of computation, communication and storage capabilities. Therefore, we have proposed the fog computing-based face identification and resolution framework to improve processing capacity and save the bandwidth. However, there are some security and privacy issues brought by the properties of fog computing-based framework. In this paper, we propose a security and privacy preservation scheme to solve above issues. We give an outline of the fog computing-based face identification and resolution framework, and summarize the security and privacy issues. Then the authentication and session key agreement scheme, data encryption scheme, and data integrity checking scheme are proposed to solve the issues of confidentiality, integrity, and availability in the processes of face identification and face resolution. Finally, we implement a prototype system to evaluate the influence of security scheme on system performance. Meanwhile, we also evaluate and analyze the security properties of proposed scheme from the viewpoint of logical formal proof and the CIA (confidentiality, integrity, availability) properties of information security. The results indicate that the proposed scheme can effectively meet the requirements for security and privacy preservation.

TITLE 10: A Secure Cloud-assisted Urban Data Sharing Framework for Ubiquitous-cities

AUTHOR: Jian Shena^{a,b,c,_,}, Dengzhi Li^{uc}, Jun Shen^c, Qi Liu^{a,c}, Xingming Sun^{a,c}

With the accelerated process of urbanization, more and more people tend to live in cities. In order to deal with the big data that are generated by citizens and public city departments, new information and communication technologies are utilized to process the urban data, which makes it more easier to manage. Cloud computing is a novel computation technology. After cloud computing was commercialized, there have been lot of cloud-based applications. Since the cloud service is provided by the third party, the cloud is semi-trusted. Due to the features of cloud computing, there are many security issues in cloud computing. Attribute-based encryption (ABE) is a promising cryptography technique which can be used in the cloud to solve many security issues. In this paper, we propose a framework for urban data sharing by exploiting the attribute-based cryptography. In order to fit the real world ubiquitous-cities utilization, we extend our scheme to support dynamic operations. In particular, from the part of performance analysis, it can be concluded that our scheme is secure and can resist possible attacks. Moreover, experimental results and comparisons show that our scheme is more efficient in terms of computation

CHAPTER 3

SCOPE AND OBJECTIVE

SCOPE

A Multi-Layer Privacy Preserving Cloud Storage Scheme Based on Computational Intelligence in Fog Computing. The privacy preservation is our focus, some active attacks are beyond the scope of this work. The multilayer cloud storage stores in to the multi different parts of data parts. If the one data part missing we lost the data information. In this proposed framework using the bucket concept-based algorithms. We are using a BCH code algorithm. It's High flexible.

OBJECTIVE

Cloud storage also causes a series of secure problems. When using cloud storage, users do not actually control the physical storage of their data and it results in the separation of ownership and management of data. In order to solve the problem of privacy protection in cloud storage, we propose a TLS framework based on fog computing model and design a Hash-Solomon algorithm. Through the theoretical safety analysis, the scheme is proved to be feasible. By allocating the ratio of data blocks stored in different servers reasonably, we can ensure the privacy of data in each server. On another hand, cracking the encoding matrix is impossible theoretically. Besides, using hash transformation can protect the fragmentary information. Through the experiment test, this scheme can efficiently complete encoding and decoding without influence of the cloud storage efficiency. The multi-layer cloud storage stores in to the multi different parts of data parts. If the one data part missing, we lost the data information. In this proposed framework using the bucket concept-based algorithms

CHAPTER 4

SYSTEM DESIGN & METHODOLOGY

EXISTING SYSTEM

- The development and evaluation of a data-driven approach that uses machine learning to help VC investors scouting and selecting the best companies to support. As in some previous works, our approach relies on the data provided through Crunchbase.
- The main features of existing approach are to focus on early-stage companies: We will consider that early stage companies are defined as active companies with less than a precise age at the start of a simulation window and that are in an early funding stage , according to the startup fundraising stages.
- Time-aware analysis approach defines a realistic time window, which represents a reasonable investment window for a VC investor. subsequent funding may be a proxy of non-success since it will indicate that a company is burning money and need a new capital injection. In the industry though, this is not the case because VCs urge companies to spend money quickly to grow faster to automatically elicit the ones that can return them the money in a reasonable time frame from the ones that are dead ends.
- Multi-class prediction problem target variable represents the next event that a company will achieve in a defined simulation window.
- The prediction of these events is useful for the VC investor to make an investment decision because they represent the company progress. Namely: a company closes; it obtains investment through a new funding round; the company is acquired by another company; or it goes public via an IPO (allowing the company to raise capital from public markets).

EXISTING SYSTEM DISADVANTAGES:

- Increases the evaluation overhead
- Expense in running time
- Drill enough to leave more damage
- High latency
- If Number of folds increases, then accuracy never goes above 50%

PROPOSED SYSTEM

- The venture capital consists of four phases, capital collection, decision-making of investment, operation, with drawl. During the course of capital collection, capital mainly derives from the government, agencies and individuals.
- The definition of principal-agent efficiency is various; the general definition is the result of comparing principal agent earnings with principal-agent cost.
- If the principal agent earning exceeds the principal-agent cost, then the principal-agent relationship is efficient, otherwise it's inefficient. But the general definition is defective, the principal affects agent and the vice versa, so it's different to research the principal-agent efficiency from the different angles, that is to say, the corresponding research should be divided into two parts, the research on the principal efficiency and the research on the agent efficiency.
- For principals, investment means forming the principal agent relationship. Aimed at maintaining their own interests, the principal is sure to supervise the process of operation and the corresponding operating results.
- The supervision depends on the two factors, one is motivation's strength of supervision, and the other is feasibility of the supervision. The research should begins with the principal's target for researching the motivation's strength of supervision, the reasons why the principal trusts the agents with capital is the expectation to obtain the earnings higher than they does it themselves.

ADVANTAGES OF PROPOSED SYSTEM

- High accuracy has been observed
- Improving the quality assessment system
- Improved and cost effective product quality
- Assessments using advanced computational techniques
- More heterogeneous structure

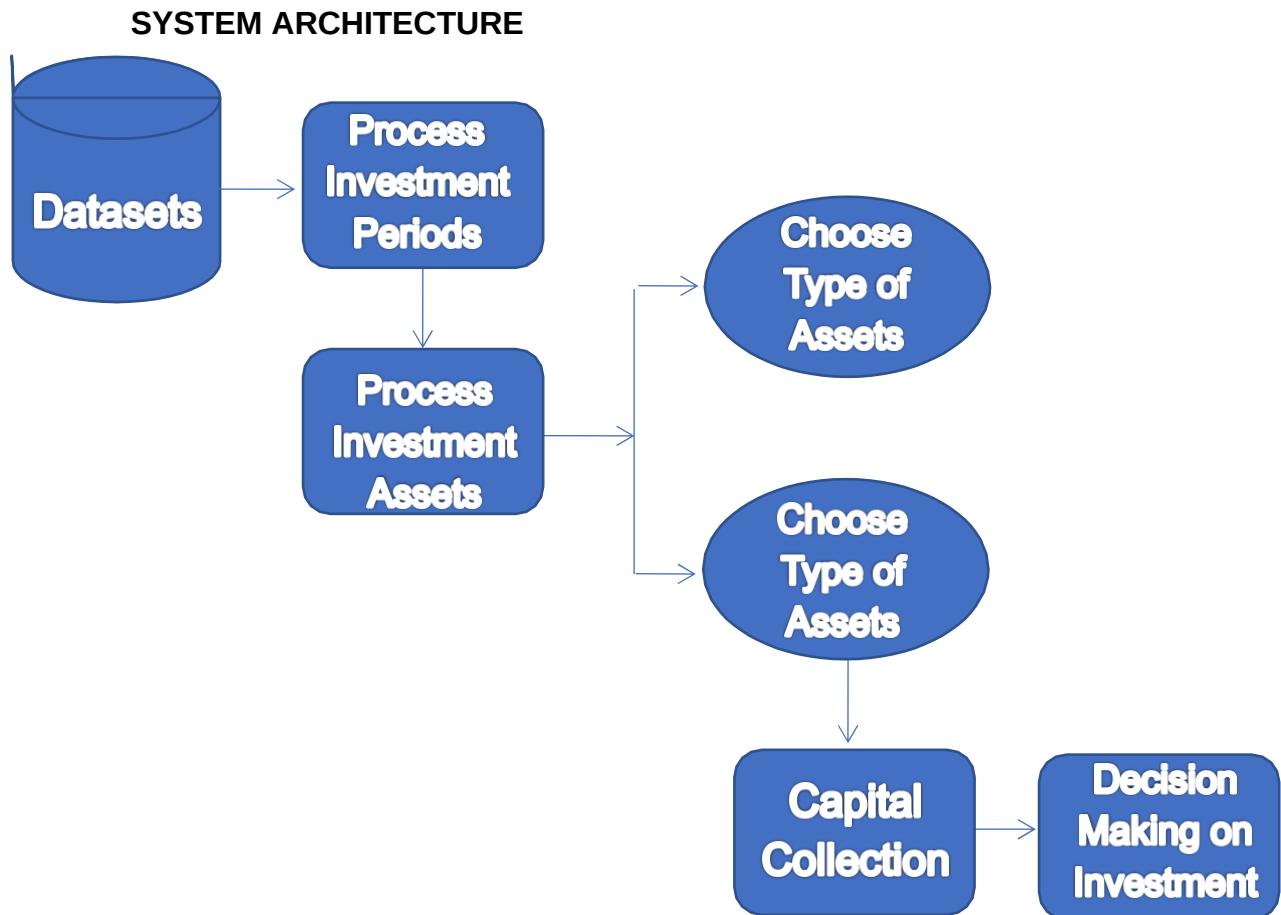


FIG 4.1 SYSTEM ARCHITECTURE

ALGORITHMS

EXISTING ALGORITHM

- Decision trees

PROPOSED ALGORITHM

- Random forest

ADVANTAGES OF PROPOSED ALGORITHM

- The predictive performance can compete with the best supervised learning algorithms
- They provide a reliable feature importance estimate
- They offer efficient estimates of the test error without incurring the cost of repeated model training associated with cross-validation

SYSTEM REQUIREMENTS

HARDWARE REQUIREMENTS:

- System - Pentium-IV
- Speed - 2.4GHZ
- Hard disk - 40GB
- Monitor - 15VGA color
- RAM - 512MB

SOFTWARE REQUIREMENTS:

- Operating System - Windows
- Coding language - Python
- IDE - Anaconda
- Database - SQL Server

TECHNOLOGY USED

BACKEND

TECHNOLOGIES

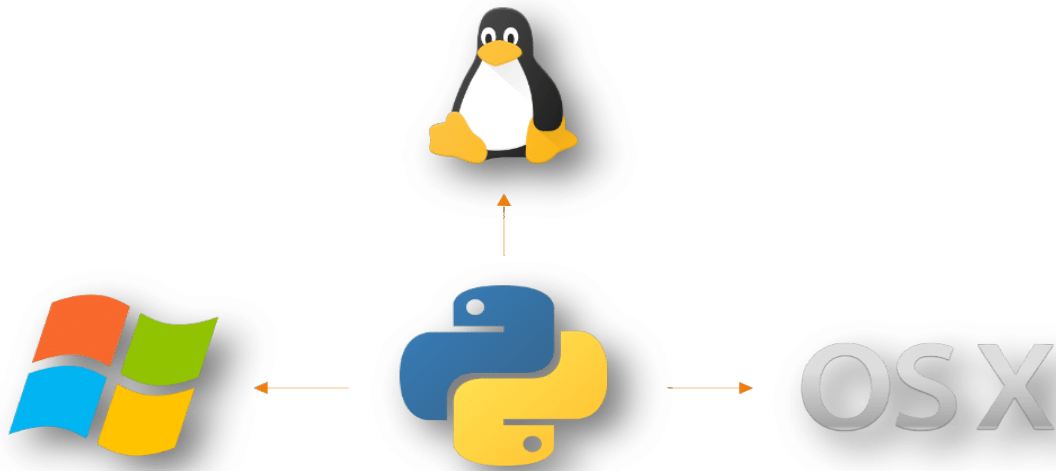
- Python
- NumPy
- Sci-learn
- Eclipse IDE

FRONTEND TECHNOLOGIES

- Web Technologies
- Bootstrap

INTRODUCTION TO PYTHON

Python is a free, open-source programming language. Therefore, all you have to do is install Python once, and you can start working with it. Not to mention that you can contribute your own code to the community. Python is also a cross-platform compatible language. So, what does this mean? Well, you can install and run Python on several operating systems. Whether you have a Windows, Mac or Linux, you can rest assure that Python will work on all these operating systems.



Python is also a great visualization tool. It provides libraries such as Matplotlib, seaborn and bokeh to create stunning visualizations.



In addition, Python is the most popular language for machine learning and deep learning. As a matter of fact, today, all top organizations are investing in Python to implement

machine learning in the back-end.

PYTHON LINE STRUCTURE

Python coding style comprises physical lines as well as logical lines or statements. A physical line in a Python program is a sequence of characters, and the end of the line terminates the line sequence as opposed to some other languages, such as C and C++ where a semi-colon is used to mark the end of the statement. A logical line, on the other hand, is composed of one or more physical lines. The use of a semi-colon is not prohibited in Python, although it's not mandatory. The NEWLINE token denotes the end of the logical line. A logical line that only contains spaces, comments, or tabs are called blank lines and they are ignored by the interpreter. As we saw that in Python, a new line simply means that a new statement has started. Although, Python does provide a way to split a statement into a multiline statement or to join multiple statements into one logical line. This can be helpful to increase the readability of the statement. Following are the two ways to split a line into two or more lines:

EXPLICIT LINE JOINING

In explicit line joining, we use a backward slash to split a statement into a multiline statement.

IMPLICIT LINE JOINING

Statements that reside inside [], {}, or () parentheses can be broken down into two or more physical lines without using a back slash.

MULTIPLE STATEMENTS ON A SINGLE LINE

In Python, it is possible to club multiple statements in the same line using a semi-colon; however, most programmers do not consider this to be a good practice as it reduces the readability of the code.

WHITESPACES AND INDENTATION

Unlike most of the programming languages, Python uses indentation to mark a block of code. According to Python coding style guideline or PEP8, we should keep an indent

size of four. Most of the programming languages provide indentation for better code formatting and don't enforce to have it. But in Python it is mandatory. This is why indentation is so crucial in Python. Comments in any programming language are used to increase the readability of the code. Similarly, in Python, when the program starts getting complicated, one of the best ways to maintain the readability of the code is to use Python comments. It is considered a good practice to include documentations and notes in the python syntax since it makes the code way more readable and understandable to other programmers as well, which comes in handy when multiple programmers are simultaneously working on the same project. The code can only explain how it does something and not why it does that, but Python comments can do that. With Python comments, we can make documentations for various explanations in our code itself. Comments are nothing but tagged lines of codes which increase the readability of a code and make it self-explanatory. There are different ways of creating comments depending on the type of comment we want to include in our code. Following are different kinds of comments that can be included in our Python program:

1. Single Line Comments
2. Docstring Comments
3. Multiline Comments

Single line Python comments are marked with `#` character. These comments end at the end of the physical line, which means that all characters starting after the `#` character (and lasts till the end of the line) are part of the comment. Python has the documentation strings (or docstrings) feature which is usually the first statement included in functions and modules. Rather than being ignored by the Python Interpreter like regular comments, docstrings can actually be accessed at the run time using the dot operator. It gives programmers an easy way of adding quick notes with every Python module, function, class, and method. To use this feature, we use triple quotes in the beginning of the documentation string or comment and the closing triple quotes at the end of the documentation comment. Docstrings can be one-liners as well as multi-liners. Unlike some programming languages that support multiline comments, such as C, Java, and more, there is no specific feature for multiline comments in Python. But that does not mean that it is totally impossible to make multiline comments in Python. There are two ways we can include comments that can span across multiple lines in our Python code.

PYTHON BLOCK COMMENTS

We can use several single line comments for a whole block. This type of comment is usually created to explain the block of code that follows the Block comment. Python Block comment is the only way of writing a real comment that can span across multiple lines. It is supported and preferred by Python's PEP8 style guide since Block comments are ignored by Python interpreter or parser. However, nothing is stopping programmers from using the second 'non-real' way of writing multiline comments in Python which is explained below

.

USING DOCSTRINGS

Docstrings are largely used as multiline comments in Python by many programmers since it is the closest thing to having a multiline comment feature in Python. While it is not wrong to use docstrings when we need to make multiline comments, it is important to keep in mind that there is a significant difference between docstrings and comments. Comments in Python are totally ignored by the Python Interpreter, while docstrings, when used inside the Python function, can be accessed at the run time.

Python Data Types

One of the most crucial part of learning any programming language is to understand how data is stored and manipulated in that language. Users are often inclined toward Python because of its ease of use and the number of versatile features it provides. One of those features is dynamic typing.

In Python, unlike statically typed languages like C or Java, there is no need to specifically declare the data type of the variable. In dynamically typed languages such as Python, the interpreter itself predicts the data type of the [Python Variable](#) based on the type of value assigned to that variable.

ADVANTAGES OF PYTHON

- Universal Language Construct
- Support both High Level and Low-Level Programming
- Language Interoperability

- Fastest Development life cycle therefore more productive coding environment
- Less memory used because a single container holds multiple data types and each type doesn't require its own function
- Learning Ease and open source development
- Speed and user-friendly data structure
- Extensive and extensible libraries.
- Simple & support IoT

SELECTION OF INTEGRATED DEVELOPMENT ENVIRONMENT

IDE – Integrated Development Environment. It is software suite It contains different instruments for web application software engineers to compose and test applications.

Programming Web application software engineers utilize wide scope of devices habitually all through application source code creation, building and application testing. Programming Development apparatuses frequently incorporate word processors for source code creation, code libraries for reuse, compilers and test stages.

An Integrated Development Environment (IDE) is a product application that gives a programming domain to streamline creating and troubleshooting programming. Instead of playing out every one of the means required to make an executable program as disconnected individual assignments, it brings every one of the devices required into one application and workspace. Every one of the devices has a familiarity with the earth, and they cooperate to display a consistent improvement set for the designer.

Without an IDE, a product web application software engineer must choose, send, incorporate and deal with these instruments independently. An IDE brings huge numbers of those improvement related instruments all together system, application or administration. The incorporated toolset is intended to streamline programming advancement and can distinguish and limit coding missteps and grammatical errors.

- Komodo
- NetBeans
- PyCharm
- Python for Visual Studio.NET
- PyDev
- PyStudio
- Liclipse (Eclipse)

LICLIPSE

LiClipse is at the core for all Python based application development activities. It's been available for many years. LiClipse is a set of plugins to enhance Eclipse. It improves the overall Eclipse experience. It is best IDE tool to develop Python Applications.

Advantages of LiClipse

- Perfect Debugger
- Error checking features
- An enhanced ability to analyze the code for quality and security concerns
- Auto complete feature

Based on the various IDE that are described above, we decided to use LiClipse as IDE (Integrated Development Environment) for developing this e-Voting web based application.

Important Features That Make LiClipse our IDE of Choice

- Rich Reusable Libraries are available
- Model Driven Development
- Enterprise Python Tooling
- Powerful Debugging options

SELECTION OF RELATIONAL DATABASE MANAGEMENT SYSTEM (RDBMS)

There are various open source and commercial database management systems are available in the market. Normally, databases are divided into three categories

- Hierarchical databases
- Network databases
- Relational databases
- Object-oriented databases
- Graph databases
- ER model databases
- Document databases

Out of which Relational database management system is widely used. RDBMS represents social database the board framework. A social model can be spoken to as a table of lines and segments.

RDBMS is a data the board framework which is arranged on an information model. Here

all the data is appropriately put away as tables. It offers data to be spared in the tables. A relational database contains tables which store information that is connected here and there. SQL is the language that permits recovery and control of table information in a social database.

SQL can be used for storing and retrieving information from one spot considered database and utilize that information at whatever point it required in other programming applications.

There are approximately 121 relational database management systems are available in the market.

We have decided to use SQL Server for our e-Voting application.

SQL Server is a relational database technology developed by Microsoft. SQL Server can be used to develop from small scale to large enterprises with complex data requirements, data warehousing and Web-enabled databases.

SQL Server uses Transact-SQL (T-SQL) which is an extension of SQL.

Important Features That Make SQL Server our Database of Choice

- Better Performance Features
- Better Security Features
- Lower Cost Of Ownership
- Excellent Data Restoration and Recovery Mechanism

SELECTION OF OPERATING SYSTEM

There are various open source and commercial Operating systems are available in the market. Computer Operating systems can be broadly classified into two types:

- Normal Operating System
- Real Time Operating System

We have decided to use Windows 10 for our e-Voting application.

Windows 10 is a is the most powerful Microsoft operating system it can be used for desktop/laptop computers as well as IoT devices.

Important Features That Make Windows 10 our Operating System of Choice

- The loading time is very fast compare to other OS
- Supports built-in mobile device executive facilities
- Stability and performance
- Have hologram technology

CHAPTER 5

Results and Discussion

5.1 TESTING

Testing documentation is the documentation of artifacts that are created during or before the testing of a software application. Documentation reflects the importance of processes for the customer, individual and organization. Projects which contain all documents have a high level of maturity. Careful documentation can save the time, efforts and wealth of the organization.

If the testing or development team gets software that is not working correctly and developed by someone else, so to find the error, the team will first need a document. Now, if the documents are available then the team will quickly find out the cause of the error by examining documentation. But, if the documents are not available then the tester need to do black box and white box testing again, which will waste the time and money of the organization. More than that, Lack of documentation becomes a problem for acceptance.

Benefits of using Documentation

- This study Documentation clarifies the quality of methods and objectives.
- It ensures internal coordination when a customer uses software application.
- It ensures clarity about the stability of tasks and performance.
- It provides feedback on preventive tasks.
- It provides feedback for your planning cycle.
- It creates objective evidence for the performance of the quality management system.

The test scenario is a detailed document of test cases that cover end to end functionality of a software application in liner statements. The liner statement is considered as a scenario. The test scenario is a high-level classification of testable requirements. These requirements are grouped on the basis of the functionality of a module and obtained from the use cases.

In the test scenario, there is a detailed testing process due to many associated test cases. In the test scenario, testers need to put themselves in the place of the user

because they test the software application under the user's point of view. Preparation of scenarios is the most critical part, and it is necessary to seek advice or help from customers, stakeholders or developers to prepare the scenario. As per the IEEE Documentation describing plans for, or results of, the testing of a system or component, Types include test case specification, test incident report, test log, test plan, test procedure, test report. Hence the testing of all the above mentioned documents is known as documentation testing. This is one of the most cost effective approaches to testing. If the documentation is not right: there will be major and costly problems. The documentation can be tested in a number of different ways to many different degrees of complexity. These range from running the documents through a spelling and grammar checking device, to manually reviewing the documentation to remove any ambiguity or inconsistency. Documentation testing can start at the very beginning of the software process and hence save large amounts of money, since the earlier a defect is found the less it will cost to be fixed. the most popular testing documentation files are test reports, plans, and checklists. These documents are used to outline the team's workload and keep track of the process. Let's take a look at the key requirements for these files and see how they contribute to the process. Test strategy. An outline of the full approach to product testing. As the project moves along, developers, designers, product owners can come back to the document and see if the actual performance corresponds to the planned activities. Test data. The data that testers enter into the software to verify certain features and their outputs. Examples of such data can be fake user profiles, statistics, media content, similar to files that would be uploaded by an end-user in a ready solution. Test plans. A file that describes the strategy, resources, environment, limitations, and schedule of the testing process. It's the fullest testing document, essential for informed planning. Such a document is distributed between team members and shared with all stakeholders. Test scenarios. In scenarios, testers break down the product's functionality and interface by modules and provide real-time status updates at all testing stages. A module can be described by a single statement, or require hundreds of statuses, depending on its size and scope. Test cases. If the test scenario describes the object of testing (what), a scenario describes a procedure (how). These files cover step-by-step guidance, detailed conditions, and current inputs of a testing task. Test cases have their own kinds that depend on the type of testing — functional, UI, physical, logical cases, etc. Test cases compare available resources and current conditions with desired outcomes and determine if the functionality can be released or not. Traceability

Matrix. This software testing documentation maps test cases and their requirements. All entries have their custom IDs — team members and stakeholders can track the progress of any tasks by simply entering its ID to the search. External documentation collects information from inner documentation but also emphasize on providing a visual data representation — graphs, diagrams, etc. External reports — these documents collect information on test results and can describe an entire project or a particular piece of functionality. Test summary report — the file with final test results and findings, presented to stakeholders. Bug reports — such files keep track of newly encountered bugs and their fixes. We prefer to keep our bug documentation numbered, so it's easier to mention them in further documentation. Reports are concise and focus on offering tangible solutions. Sometimes, bug reports can only include issue description, if the team hasn't yet found the best approach to fixing the problem.

In simple words, Unit Testing means – writing a piece of code (unit test) to verify the code (unit) written for implementing requirements.



Testing Process

Unit Testing is used to design robust software components that help maintain code and eliminate the issues in code units. We all know the importance of finding and fixing defects in the early stage of the software development cycle. Unit Testing serves the same purpose. Unit Testing is an integral part of the agile software development process. When a nightly build run unit test suite should run and report should be generated. If any of the unit tests have failed then the QA team should not accept that build for verification. If we set this as a standard process, many defects would be caught

in the early development cycle, saving much testing time. I know many developers hate to write unit tests. They either ignore or write bad unit test cases due to tight scheduled or lack of seriousness (yea they write empty unit tests, so 100% of them pass successfully ;-)). It's important to write good unit tests or don't write them at all. It's even more important to provide sufficient time and a supportive environment for real benefits.

- Testing can be done in the early phases of the software development lifecycle when other modules may not be available for integration
- Fixing an issue in Unit Testing can fix many other issues occurring in later development and testing stages
- Cost of fixing a defect found in Unit Testing is very less than the one found in the system or acceptance testing
- Code coverage can be measured
- Fewer bugs in the System and Acceptance testing
- Code completeness can be demonstrated using unit tests. This is more useful in the agile process. Testers don't get the functional builds to test until integration is completed. Code completion cannot be justified by showing that you have written and checked in the code. But running Unit tests can demonstrate code completeness.
- Expect robust design and development as developers write test cases by understanding the specifications first.
- Easily identify who broke the build
- Saves development time: Code completion may take more time but due to decreased defect count overall development time can be saved.

unit Testing frameworks are mostly used to help write unit tests quickly and easily. Most of the programming languages do not support unit testing with the inbuilt compiler. Third- party open source and commercial tools can be used to make unit testing even more fun.

List of popular Unit Testing tools for different programming languages:

- ▶ Java framework – JUnit
- ▶ PHP framework – PHPUnit
- ▶ C++ frameworks – UnitTest++ and Google C++

-

Functional Testing is a type of black box testing whereby each part of the system is tested against functional specification/requirements. For instance, seek answers to the following questions, Are you able to login to a system after entering correct credentials? Does your payment gateway prompt an error message when you enter incorrect card number? Does your “add a customer” screen adds a customer to your records successfully? Well, the above questions are mere samples to perform full-fledged functional testing of a system. Test Driven Development, or TDD, is a code design technique where the programmer writes a test before any production code, and then writes the code that will make that test pass. The idea is that with a tiny bit of assurance from that initial test, the programmer can feel free to refactor and refactor some more to get the cleanest code they know how to write. The idea is simple, but like most simple things, the execution is hard. TDD requires a completely different mind set from what most people are used to and the tenacity to deal with a learning curve that may slow you down at first.

1. Unit Testing
2. Integration Testing
3. System Testing

4. Sanity Testing
5. Smoke Testing
6. Interface Testing
7. Regression Testing
8. Beta/Acceptance Testing
9. Non-functional Testing types include:
 - Performance Testing
 - Load Testing
 - Stress Testing
 - Volume Testing
 - Security Testing
 - Compatibility Testing
 - Install Testing
 - Recovery Testing
 - Reliability Testing
 - Usability Testing
 - Compliance Testing
 - Localization Testing

Unit Testing

UNIT TESTING is a level of software testing where individual units/ components of a software are tested. The purpose is to validate that each unit of the software performs as designed. A unit is the smallest testable part of any software. It usually has one or a few inputs and usually a single output. In procedural programming, a unit may be an individual program, function, procedure, etc. In object-oriented programming, the smallest unit is a method, which may belong to a base/ super class, abstract class or derived/ child class. (Some treat a module of an application as a unit. This is to be

discouraged as there will probably be many individual units within that module.) Unit testing frameworks, drivers, stubs, and mock/ fake objects are used to assist in unit testing. A unit can be almost anything you want it to be -- a line of code, a method, or a class. Generally, though, smaller is better. Smaller tests give you a much more granular view of how your code is performing. There is also the practical aspect that when you test very small units, your tests can be run fast; like a thousand tests in a second fast. Black Box testers don't care about Unit Testing. Their main goal is to validate the application against the requirements without going into the implementation details.

Unit Testing is not a new concept. It's been there since the early days of programming. Usually, developers and sometimes White box testers write Unit tests to improve code quality by verifying each and every unit of the code used to implement functional requirements (aka test drove development TDD or test-first development). Most of us might know the classic definition of Unit Testing – “Unit Testing is the method of verifying the smallest piece of testable code against its purpose.” If the purpose or requirement failed then the unit test has failed. In simple words, Unit Testing means – writing a piece of code (unit test) to verify the code (unit) written for implementing requirements.

Black box Testing

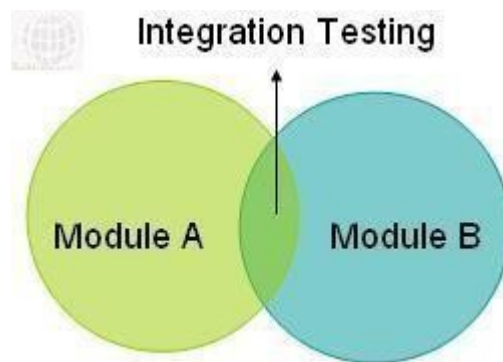
During functional testing, testers verify the app features against the user specifications. This is completely different from testing done by developers which is unit testing. It checks whether the code works as expected. Because unit testing focuses on the internal structure of the code, it is called the white box testing. On the other hand, functional testing checks app's functionalities without looking at the internal structure of the code, hence it is called black box testing. Despite how flawless the various individual code components may be, it is essential to check that the app is functioning as expected, when all components are combined. Here you can find a detailed comparison between functional testing vs unit testing.

Integration Testing

INTEGRATION TESTING is a level of software testing where individual units are combined and tested as a group. The purpose of this level of testing is to expose faults

in the interaction between integrated units. Test drivers and test stubs are used to assist

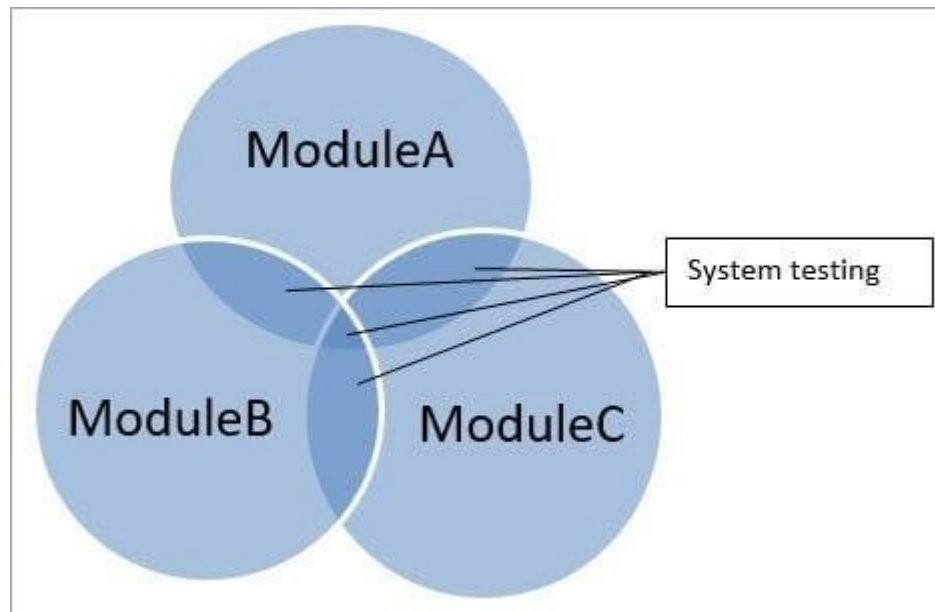
in Integration Testing. integration testing: Testing performed to expose defects in the interfaces and in the interactions between integrated components or systems. See also component integration testing, system integration testing. component integration testing: Testing performed to expose defects in the interfaces and interaction between integrated components. System integration testing: Testing the integration of systems and packages; testing interfaces to external organizations (e.g. Electronic Data Interchange, Internet). Integration tests determine if independently developed units of software work correctly when they are connected to each other. The term has become blurred even by the diffuse standards of the software industry, so I've been wary of using it in my writing. In particular, many people assume integration tests are necessarily broad in scope, while they can be more effectively done with a narrower scope. As often with these things, it's best to start with a bit of history. When I first learned about integration testing, it was in the 1980's and the waterfall was the dominant influence of software development thinking. In a larger project, we would have a design phase that would specify the interface and behavior of the various modules in the system. Modules would then be assigned to developers to program. It was not unusual for one programmer to be responsible for a single module, but this would be big enough that it could take months to build it. All this work was done in isolation, and when the programmer believed it was finished they would hand it over to QA for testing. Integration testing tests integration or interfaces between components, interactions to different parts of the system such as an operating system, file system and hardware or interfaces between systems. Integration testing is a key aspect of software testing.



System Testing

System Testing is a level of software testing where a complete and integrated software is tested. The purpose of this test is to evaluate the system's compliance with the specified requirements. System Testing means testing the system as a whole. All the

modules/components are integrated in order to verify if the system works as expected or not. System Testing is done after Integration Testing. This plays an important role in delivering a high-quality product.



System testing is a method of monitoring and assessing the behaviour of the complete and fully-integrated software product or system, on the basis of pre-decided specifications and functional requirements. It is a solution to the question "whether the complete system functions in accordance to its pre-defined requirements?" It's comes under black box testing i.e. only external working features of the software are evaluated during this testing. It does not require any internal knowledge of the coding, programming, design, etc., and is completely based on users-perspective. A black box testing type, system testing is the first testing technique that carries out the task of testing a software product as a whole. This System testing tests the integrated system and validates whether it meets the specified requirements of the client. System testing is a process of testing the entire system that is fully functional, in order to ensure the system is bound to all the requirements provided by the client in the form of the functional specification or system specification documentation. In most cases, it is done next to the Integration testing, as this testing should be covering the end-to-end system's actual routine. This type of testing requires a dedicated Test Plan and other test documentation derived from the system specification document that should cover both software and hardware requirements. By this test, we uncover the errors. It

ensures that all the system works as expected. We check System performance and functionality to get a quality

product. System testing is nothing but testing the system as a whole. This testing checks complete end-to-end scenario as per the customer's point of view. Functional and Non- Functional tests also done by System testing. All things are done to maintain trust within the development that the system is defect-free and bug-free. System testing is also intended to test hardware/software requirements specifications. System testing is more of a limited type of testing; it seeks to detect both defects within the "inter-assemblages".

Sanity Testing

Sanity Testing is done when as a QA we do not have sufficient time to run all the test cases, be it Functional Testing, UI, OS or Browser Testing. Sanity testing is a subset of regression testing. After receiving the software build, sanity testing is performed to ensure that the code changes introduced are working as expected. This testing is a checkpoint to determine if testing for the build can proceed or not. The main purpose of this testing is to determine that the changes or the proposed functionality are working as expected. If the sanity test fails, the build is rejected by the testing team to save time and money. It is performed only after the build has cleared the smoke test and been accepted by the Quality Assurance team for further testing. The focus of the team during this testing process is to validate the functionality of the application and not detailed testing. Let's assume that in an e-commerce project, there are mainly five modules- like login page, home page, user profile page, new user creation, etc. There is a defect in the login page when the password field accepts less than four alpha numeric characters and the requirement mentions that this password field should not be below eight characters. So, the bug is reported by the testing team to the development team to resolve it. When the development team fixes the defect and passes it to the testing team. Then the testing team checks if the changes done are working fine. It also checks if it does not have an impact on other related functionalities. Now there is a functionality to update the password in the user profile page. As part of the sanity check, we would need to validate the login page as well as the profile page to ensure that the checks are working fine at both the places. Sanity tests are generally performed on build where the production deployment is required immediately - like a critical bug fix. Smoke Testing is done to make sure if the build we received from the development team is testable or not. It is also called as "Day 0" check. It is done at the

“build level”. It helps not to waste the testing time to simply testing the whole application when the key features don’t work or

the key bugs have not been fixed yet. Here our focus will be on primary and core application work flow. To conduct smoke testing, we don't write test cases. We just pick the necessary test cases from already written test cases. Do we really write test cases for all testing types? Here in this article, we have given clear idea on choosing testing types to write test cases. As mentioned earlier, here in Smoke Testing, our main focus will be on core application work flow. So we pick the test cases from our test suite which cover major functionality of the application. In general, we pick minimal number of test cases that won't take more than half an hour to execute. After receiving a Software build with the minor issues fixes in code or functionality, Sanity testing is carry out to check whether the bugs reported in previous build are fixed & there is regression introduced due to these fixes i.e. not breaking any previously working functionality. The main aim of Sanity testing to check the planned functionality is working as expected. Instead of doing whole regression testing the Sanity testing is perform. Sanity tests helps to avoid wasting time and cost involved in testing if the build is failed. Tester should reject the build upon build failure. After completion of regression testing the Sanity testing is started to check the defect fixes & changes done in the software application is not breaking the core functionality of the software. Typically this is done nearing end of SDLC i.e. while releasing the software. You can say that sanity testing is a subset of acceptance testing. We can also say Tester Acceptance Testing for Sanity testing.

Regression Testing

Regression Testing is a type of testing that is done to verify that a code change in the software does not impact the existing functionality of the product. This is to make sure the product works fine with new functionality, bug fixes or any change in the existing feature. Previously executed test cases are re-executed in order to verify the impact of change. Regression Testing is a Software Testing type in which test cases are re-executed in order to check whether the previous functionality of the application is working fine and the new changes have not introduced any new bugs. This test can be performed on a new build when there is a significant change in the original functionality that too even in a single bug fix. For regression testing to be effective, it needs to be seen as one part of a comprehensive testing methodology that is cost-effective and efficient while still incorporating enough variety—such as well-designed frontend UI

automated tests alongside targeted unit testing, based on smart risk prioritization—to prevent any

aspects of your software applications from going unchecked. These days, many Agile work environments employing workflow practices such as XP (Extreme Programming), RUP (Rational Unified Process), or Scrum appreciate regression testing as an essential aspect of a dynamic, iterative development and deployment schedule. But no matter what software development and quality-assurance process your organization uses, if you take the time to put in enough careful planning up front, crafting a clear and diverse testing strategy with automated regression testing at its core, you can help prevent projects from going over budget, keep your team on track, and, most importantly, prevent unexpected bugs from damaging your products and your company's bottom line. Performance testing is the practice of evaluating how a system performs in terms of responsiveness and stability under a particular workload. Performance tests are typically executed to examine speed, robustness, reliability, and application size. The process incorporates "performance" indicators such as:

- Browser, page, and network response times
- Server request processing times
- Acceptable concurrent user volumes
- Processor memory consumption; number and type of errors that might be encountered with app

Performance testing gathers all the tests that verify an application's speed, robustness, reliability, and correct sizing. It examines several indicators such as a browser, page and network response times, server query processing time, number of acceptable concurrent users architected, CPU memory consumption, and number/type of errors which may be encountered when using an application. Performance testing is the testing that is performed to ascertain how the components of a system are performing under a certain given situation. Resource usage, scalability, and reliability of the product are also validated under this testing. This testing is the subset of performance engineering, which is focused on addressing performance issues in the design and architecture of a software product.



5.4 Performance Testing

Software Performance testing is type of testing perform to determine the performance of system to major the measure, validate or verify quality attributes of the system like responsiveness, Speed, Scalability, Stability under variety of load conditions. The system is tested under a mixture of load conditions and check the time required responding by the system under varying workloads. Software performance testing involves the testing of application under test to ensure that application is working as expected under variety of load conditions. The goal of performance testing is not only find the bugs in the system but also eliminate the performance bottlenecks from the system. Load Testing is type of performance testing to check system with constantly increasing the load on the system until the time load is reaches to its threshold value. Here Increasing load means increasing number of concurrent users, transactions & check the behavior of application under test. It is normally carried out underneath controlled environment in order to distinguish between two different systems. It is also called as "Endurance testing" and "Volume testing". The main purpose of load testing is to monitor the response time and staying power of application when system is performing well under heavy load. Load testing comes under the Non-Functional Testing & it is designed to test the non-functional requirements of a software application. Load testing is performed to make sure that what amount of load can be

withstand the application under test. The successfully executed load testing is only if the specified test cases are

executed without any error in allocated time.

Simple examples of load testing:

- Testing printer by sending large job.
- Editing a very large document for testing of word processor
- Continuously reading and writing data into hard disk.
- Running multiple applications simultaneously on server.
- Testing of mail server by accessing thousands of mailboxes
- In case of zero-volume testing & system fed with zero load..

CHAPTER 6

CONCLUSION AND FUTURE WORK

The impact of policy support on venture capital investments in general, and on the decision making by venture capitalist in particular, has neither theoretically nor empirically been examined in depth. On the assumption of mean-variance and factor models, the effect of policy support on asset allocation and effort level of venture capitalist and the effect on asset allocation of venture capital provider are studied. The study shows that venture capitalist would enhance his efforts level and venture capital company would increase the weight of investment to VC Projects because of government's policy support. Based on analyzing the relations among venture capital, corporate governance, firm capability, and firm performance, this paper puts forward the frame model of impact of venture capital on firm performance based on firm capability and corporate governance. The model can be conducive to deepening the mechanism research on the impact of venture capital upon firm performance and have some realistic significance to post-investment activities of venture capital companies

REFERENCES

- [1] C. E. Halabí and R. N. Lussier, "A model for predicting small firm performance," *J. Small Bus. Enterprise Develop.*, vol. 21, no. 1, pp. 425, Feb. 2014.
- [2] B. Yankov, P. Ruskov, and K. Haralampiev, "Models and tools for technology start-up companies success analysis," *Econ. Alternatives*, no. 3, pp. 1524, 2014.
- [3] F.R. da Silva Ribeiro Bento, "Predicting start-up success with machine learning," M.S. thesis, Dept. Inf. Manage., Universidade Nova do Lisboa, Lisbon, Portugal, 2018.
- [4] Tahiri, J. S., Bennani, S., and Idrissi, M. K.: An Assessment System adapted to Differentiated Learning within Massive Open Online Courses Using Psychometric Testing, 15th International Conference on Information Technology Based Higher Education and Training (ITHET), Istanbul, Turkey, Sep 08-10, 2016.
- [5] Chatterjee R, Mandal J.K, "Two Dimensional Assessment Technique for CBL", International Conference on Teaching Assessment and Learning in Engineering (TALE) 2016, Bangkok, Thailand, Dec 7-9, 2016.
- [6] Kar S. P., Chatterjee R, Mandal J.K, "A novel automated assessment technique in e-Learning using short answer type questions", 1st International Conference on Computational Intelligence, Communications and Business Analytics (CICBA) 2017, Kolkata, India, Mar 24-25, 2017, in press.
- [7] M. Hardt, B. Recht, and Y. Singer, "Train faster, generalize better: stability of stochastic gradient descent," *arXiv preprint arXiv:1509.01240*, 2015.
- [8] Y. Yan-yan Song, "Decision tree methods: applications for classification and prediction", *PubMed Central (PMC)*, 2018. [Online]. Available: <https://www.ncbi.nlm.nih.gov> [Accessed: 05-Nov-2018].
- [9] T. Stewart, D. Zeng and M. Wu, "Constructing support vector machines with missing data", *Wiley Interdisciplinary Reviews: Computational Statistics*, vol. 10, no. 4, p. e1430, 2018.

[10] M. Kruk, B. Świdorski, S. Osowski, J. Kurek, M. Słowińska and I. Walecka, "Melanoma recognition using extended set of descriptors and classifiers", *Eurasip Journal on Image and Video Processing* 43, 1-10, (2015).

[11] M. Kruk, A. Jegorowa, J. Kurek, S. Osowski, J. Gorski, „ Automatic recognition of drill condition on the basis of images of drilled holes”, *Computational Problems of Electrical Engineering (CPEE)*, 2016 17th International Conference, IEEE, 1-4, 2016/11/10

[12] Z. Fu, F. Huang, K. Ren, J.Weng, and C.Wang, "Privacy-preserving smart semantic search based on conceptual graphs over encrypted outsourced data," *IEEE Trans. Inf. Forensics Security*, vol. 12, no. 8, pp. 1874–1884, Aug. 2017.

[13] J. Hou, C. Piao, and T. Fan, "Privacy preservation cloud storage architecture research," *J. Hebei Acad. Sci.*, vol. 30, no. 2, pp. 45–48, 2013.

Appendices

- A. Sample code
- B. Screen Shots

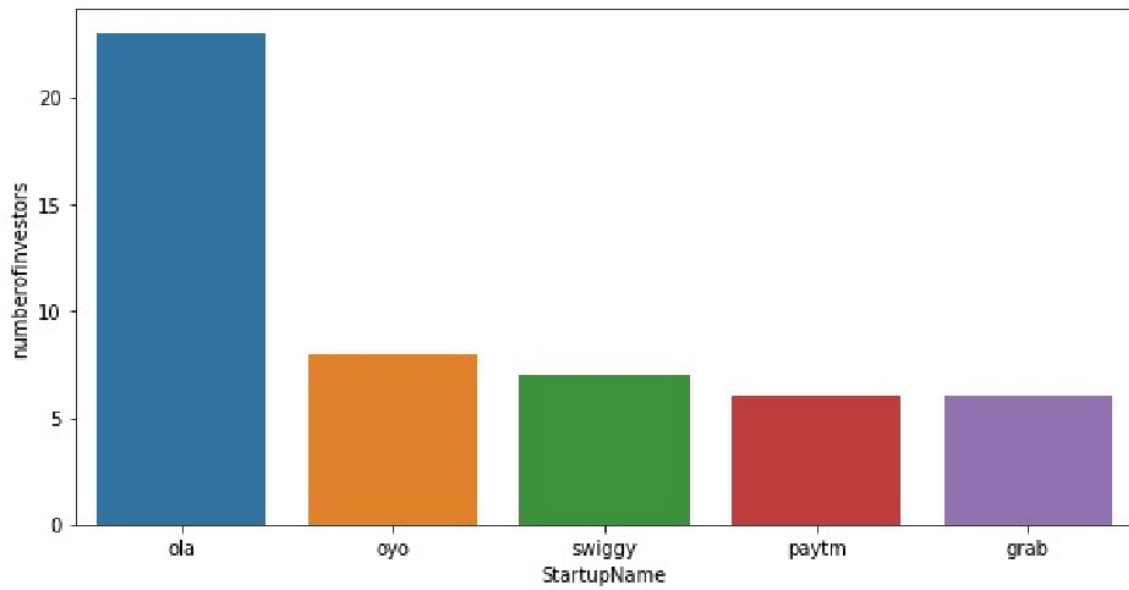
```
In [3]: df.head(4)
```

Out[3]:	SNo	Date	StartupName	IndustryVertical	SubVertical	CityLocation	InvestorsName	InvestmentType	AmountInUSD	Remarks
	0	01/08/2017	TouchKin	Technology	Predictive Care Platform	Bangalore	Kae Capital	Private Equity	1,300,000	NaN
	1	02/08/2017	Ethinos	Technology	Digital Marketing Agency	Mumbai	Triton Investment Advisors	Private Equity	NaN	NaN
	2	02/08/2017	Leverage Edu	Consumer Internet	Online platform for Higher Education Services	New Delhi	Kashyap Deorah, Anand Sankeshwar, Deepak Jain, ...	Seed Funding	NaN	NaN
	3	02/08/2017	Zepo	Consumer Internet	DIY Ecommerce platform	Mumbai	Kunal Shah, LetsVenture, Anupam Mittal, Hetal ...	Seed Funding	500,000	NaN

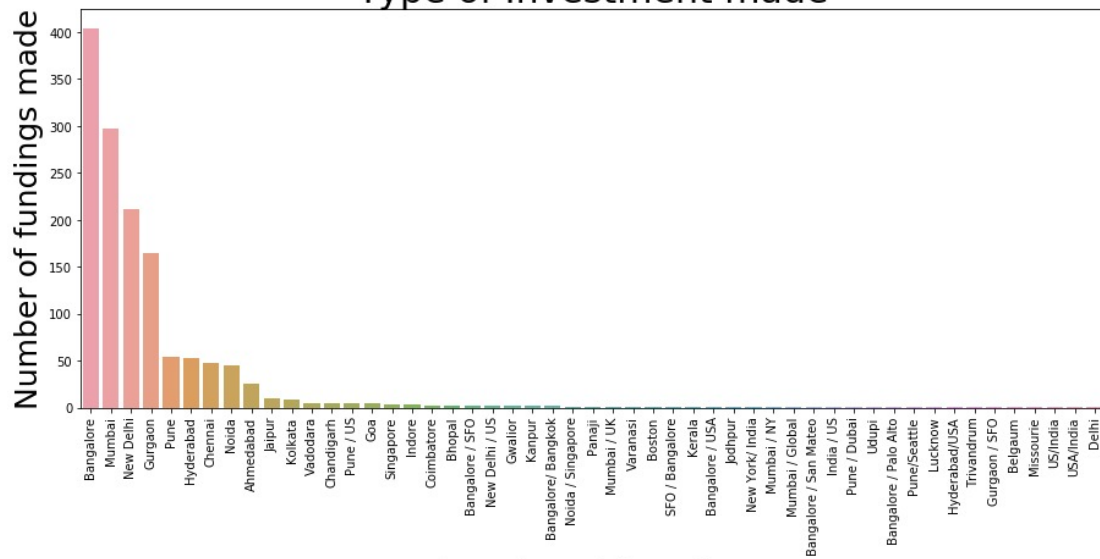
```
In [4]: df.tail(4)
```

Out[4]:	SNo	Date	StartupName	IndustryVertical	SubVertical	CityLocation	InvestorsName	InvestmentType	AmountInUSD	Remarks
2368	2368	29/01/2015	Graphene	NaN	NaN	NaN	KARSEMVEN Fund	Private Equity	825,000	Govt backed VC Fund
2369	2369	30/01/2015	Mad Street Den	NaN	NaN	NaN	Exfinity Fund, GrowX Ventures.	Private Equity	1,500,000	NaN
2370	2370	30/01/2015	Simpletel	NaN	NaN	NaN	MakeMyTrip	Private Equity	NaN	Strategic Funding, Minority stake
2371	2371	31/01/2015	couponmachine.in	NaN	NaN	NaN	UK based Group of Angel Investors	Seed Funding	140,000	NaN

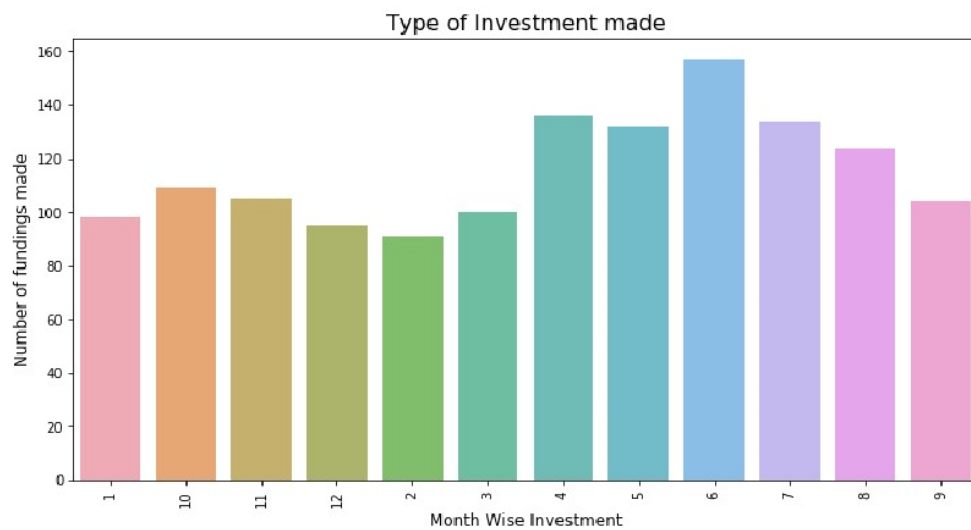




Type of Investment made



Investment Location



Most funded startups

