BSCS Thesis Proposal

An interaction index framework for predictive and preventive measures



**Submitted By**

**Bilal Khaliq**F-51069 UET-16F-BSCS-APCOMS-63

**Amir Saleem**F- UET-16F-BSCS-APCOMS-

**Muneeb Ahmad**F- UET-16F-BSCS-APCOMS-

**Supervisor**

**Nomica Imran**  
**Computer Science & Software Engineering Department**

**DEPARTMENT OF COMPUTER SCIENCE**  
**ARMY PUBLIC COLLEGE OF MANAGEMENT AND SCIENCES AFFILIATE WITH UET-TAXILA**  
-- July 2020

# UNDERTAKING

This is to declare that the project entitled “An Interaction index framework for predictive and preventive measures” is an original work done by undersigned, in partial fulfillment of the requirements for the degree “Bachelor of Science in Computer Science/software engineering” at Computer Science /software Engineering Department, Army Public College of Management & Sciences, affiliated with UET Taxila, Pakistan.

All the analysis, design and system development have been accomplished by the undersigned. Moreover, this project has not been submitted to any other college or university.

**Date: 29th July 2020**

Bilal Khaliq

Amir Saleem

Muneeb Ahmad

BSCS-26B

# CERTIFICATE

This is to certify that the project titled “**An** **Interaction index framework for predictive and preventive measures**” is the bona fide work carried out by **Bilal Khaliq , Amir Saleem, Muneeb Ahmad** students of Bachelor of Science in Computer Science (BSCS), of Army Public College of Management and Sciences, affiliated to University of Engineering and Technology, Taxila (Pakistan) during the academic year **2019-2020**, in partial fulfillment of the requirements for the award of the degree of Bachelor of Science in Computer Science (BSCS), and that the project has not formed the basis for the award previously of any other degree, diploma, fellowship or any other similar title.

**Signature of the Supervisor**

**Date: 29th July 2020**

# Abstract

Large amount of data is available online from sources like in surveillance a lot of data is available because of cameras and other devices available for securing a system. When there is a lot of data there is a need of data analysis tool that data scientists can use perform their predictive analysis on the basis of available data which help them analyzing data without doing any coding.

# 

# **ACKNOWLEDGEMENT**

I would consider it a proud a privilege to express the cordial gratitude and the deep sense of obligation to my reverend supervisor **Miss Nomica Imran (**Literature at Software Engineering and Computer science Department APCOMS Affiliated from UET Taxila) for his dexterous guidance, inspiring attitude, untiring help and consolatory behavior throughout the project and presentation of this manuscript.

**Table of Contents**

[**UNDERTAKING 2**](#_Toc19403527)

[**CERTIFICATE 3**](#_Toc19403528)

[**ABSTRACT 4**](#_Toc19403529)

[**ACKNOWLEDGEMENT 5**](#_Toc19403530)

[**1. Introduction (Chapter 1) 9**](#_Toc19403531)

[**1.1 Background Information: 9**](#_Toc19403532)

[**1.2 Problem Statement: 11**](#_Toc19403533)

[**1.3 Scope of Project: 11**](#_Toc19403534)

[**1.4 Research Hypothesis: 12**](#_Toc19403535)

[**1.5 Research Question: 12**](#_Toc19403536)

[**1.6 Research Objectives: 13**](#_Toc19403537)

[**1.7 Research Methodology: 13**](#_Toc19403538)

[**1.8 Contribution: 14**](#_Toc19403539)

[**1.9 Summary: 15**](#_Toc19403540)

[**2. Literature Review (Chapter 2) 16**](#_Toc19403541)

[**2.1 Problem Domain: 16**](#_Toc19403542)

[**2.2 Existing Work: 16**](#_Toc19403543)

[**2.3 State of Art Methods and Approaches: 19**](#_Toc19403544)

[**2.4 Comparison and Explanation: 21**](#_Toc19403545)

[**2.5 Problem Solution: 21**](#_Toc19403546)

[**2.6 Summary: 21**](#_Toc19403547)

[**3. Methodology (Chapter 3) 22**](#_Toc19403548)

[**3.1 System Requirements: 22**](#_Toc19403549)

[**3.2 System Overview: 23**](#_Toc19403550)

[**3.3 Inputs to System: 24**](#_Toc19403551)

[**3.4 Method/ Technique: 24**](#_Toc19403552)

[**3.5 Design Structure: 30**](#_Toc19403553)

[**3.6 Scenarios of Processing: 31**](#_Toc19403554)

[**3.7 Expected Output: 36**](#_Toc19403555)

[**3.8 Summary: 36**](#_Toc19403556)

[**4. Implementation (Chapter 4) 37**](#_Toc19403557)

[**4.1 Development Tool: 37**](#_Toc19403558)

[**4.2 Implementation Issues: 37**](#_Toc19403559)

[**4.3 Configuration Management: 37**](#_Toc19403560)

[**4.4 Framework Section for development: 38**](#_Toc19403561)

[**4.5 Deployment Factors: 40**](#_Toc19403562)

[**4.6 Summary: 40**](#_Toc19403563)

[**5. Chapter 5 41**](#_Toc19403564)

[**5.1 Introduction: 41**](#_Toc19403565)

[**5.2 Output: 41**](#_Toc19403566)

[**46**](#_Toc19403567)

[**5.3 Testing: 47**](#_Toc19403568)

[**48**](#_Toc19403569)

[**5.4 Statistical and Graphical Analysis: 48**](#_Toc19403570)

[**5.5 Complete User Interface: 50**](#_Toc19403571)

[**5.6 Summary: 53**](#_Toc19403572)

[**References: 54**](#_Toc19403573)

**Table of Figures**

[Figure 1 : Comparison Graph 14](file:///C:\Users\User\Desktop\FYP%20thesis\FYP%20Thesis.docx#_Toc19403494)

[Figure 2: Framework Flowchart 16](#_Toc19403495)

[Figure 3: Framework Overview 17](file:///C:\Users\User\Desktop\FYP%20thesis\FYP%20Thesis.docx#_Toc19403496)

[Figure 4: Framework Steps 18](file:///C:\Users\User\Desktop\FYP%20thesis\FYP%20Thesis.docx#_Toc19403497)

[Figure 5 : HMM Example 19](file:///C:\Users\User\Desktop\FYP%20thesis\FYP%20Thesis.docx#_Toc19403498)

[Figure 6: Biological Neuron network 22](file:///C:\Users\User\Desktop\FYP%20thesis\FYP%20Thesis.docx#_Toc19403499)

[Figure 7: Artificial Neural Network model 23](file:///C:\Users\User\Desktop\FYP%20thesis\FYP%20Thesis.docx#_Toc19403500)

[Figure 8: KNN output 29](#_Toc19403501)

[Figure 9: KNN output 2 30](#_Toc19403502)

[Figure 10: Black Box testing 31](file:///C:\Users\User\Desktop\FYP%20thesis\FYP%20Thesis.docx#_Toc19403503)

[Figure 11: Algorithms output analysis 32](file:///C:\Users\User\Desktop\FYP%20thesis\FYP%20Thesis.docx#_Toc19403504)

[Figure 12: Framework main page 33](file:///C:\Users\User\Desktop\FYP%20thesis\FYP%20Thesis.docx#_Toc19403505)

[Figure 13: Class Label interface 33](file:///C:\Users\User\Desktop\FYP%20thesis\FYP%20Thesis.docx#_Toc19403506)

[Figure 14: Loading interface 34](#_Toc19403507)

[Figure 15: Model Selection interface 34](#_Toc19403508)

# **Introduction (Chapter 1)**

## **1.1 Background Information:**

Proposing an interaction index framework for predictive analysis. Our framework will help the data scientists to analyze data. This framework will be based on existing machine learning techniques and libraries to analyze data on multiple techniques with less technical skills.

**How can it help in predictive and preventive measures?**

When the data is processed by framework, the application will display an eye-friendly interface in which data scientists can instinctively see the patterns and trends of the data to take notes for any predictions and prior to any disaster, the respective preventive measures can be taken.

**How can this Interaction index framework help us?**

An Interaction index framework will calculate the dataset feature importance vector and display them in graphical form for data scientists to calculate the impact of feature change on the data.

**Existing System**

* Excel
  + Excel has a variety of features that helps data scientists such as form creation, Pivot Table and VBS.
  + Excel’s analytical system is very large and most commonly used in industry. People can use it according to their needs.
  + Many data scientists use this a small dataset level and for few preprocessing operations
  + With a variety of powerful features such as form creation, PivotTable, VBA, etc., Excel’s system is so large that no analytics tool can surpass it, ensuring that people can analyze data according to their needs.
* BI Tools
  + BI is a tool for business intelligence and specialized in data analysis.
  + BI tools such as Power BI, Fine Report, and Tableau for example. You will find that they are designed according to the data analysis process. First, data processing, data cleaning, and then data modeling, finally data visualization that uses presentation of charts to identify problems and influence decision-making.
  + The repetitive and low value-added work of cleaning data can be simplified with BI tools.

## **1.2 Problem Statement:**

Proposing a Framework that would be built on the top of Artificial Intelligence to reduce the time and effort needed by data scientists to analyze the data and to get intuition of the patterns and trends.

## **1.3 Scope of Project:**

The scope of this project is to create a system that would make use of existing techniques to enhance the impact of decision without too much time loss for data scientists, detect potential disaster and spikes in data. This system will process the data with existing algorithms to convert from dataset to visual plot so analysis can be done faster and more efficiently.

Feature of the system:

* Fast Analysis
* Commonly used algorithms implemented
* Generic and tunable parameters

## **1.8 Contribution:**

Our proposed solution is a Design and Implementation of such framework that will be using current artificial intelligence techniques to create user interface based algorithms and tunable parameters to save valuable time for data scientists.

This Interaction Index Framework does the following:

1. Give user friendly interface
2. Allow data scientists to choose from multiple algorithms to process the dataset.
3. Allow selecting of the parameters from data scientists.
4. Simple data preprocessing.
5. Perform quick analysis on the data.
6. Visualize data for the user.

## **1.9 Summary:**

We trying to implement a framework that enable users to make use of machine learning algorithms with on single click without writing a line of code. The framework will save time and cost because it is very easy to use for data scientist. The currently used systems required lot of programming to do a simple task. Moreover, interaction index framework is extendable and can be updated in future.

# **Literature Review (Chapter 2)**

## **2.1 Problem Domain:**

Huge amount of data is generating from multiple sources like in surveillance a lot of data is available because of cameras and other devices available for securing a system. Along

In this modern age of technology because of the advancement in the computer technology almost every person in this world have access to computer and because of the access of internet as well which mean huge amount of data is out there to be process and get information out of it. Alongside a lot of work is already done on the preprocessing of data and now there is a need of a framework to take this preprocessed data and be able to handle data from multiple use cases. One framework to handle already processed data and work on multiple case scenarios. This is required by the data scientist to analyze data and make predictions accordingly.

## **2.2 Existing Work:**

Frameworks for neural network is important for the implementation of different models. Although, as long as new models based on deep learning techniques, come to usage than current and previous models become less useful because of less efficiency. Existing frameworks are less efficient. Because the training time a model takes to get trained is low and accuracy of new models are high as compare to previous models so old models cannot cope up with new one with respect to efficiency. The training time for new frameworks is low and they yield a high accuracy. Because these new models are created by having previous models problem and issues in mind so different fixes and improvements come as new models in results.

Most existing frameworks use domain specific programing language for representing deep learning models, along with an interpreter to translate them into a data structure stored in memory. Because of use of domain specific language portability problems take place. Therefore, developers using these frameworks cannot use standard programming language debuggers a significant problem as debugging is a major part of developing and tuning deep learning models. The scope of contribution decrease because of not using standard programming language debuggers. Because only people having skills in particular language can only contribute. Although the Define-and-Run paradigm works well for implementing CNN models, when it is used for implementing other types of NN models many major problems become evident.

This framework has simple implementation of complex algorithms because it uses python language as this language is popular so having large community ahead to support, use and improve this framework. Libraries used by chainer such as CuPy is an open source matrix library supported by NIVIDIA this framework also support popular optimization techniques SGD, AdaGrad etc. which attract community to get advantage from it. According to this we think we can use every technique separately and can make changes accordingly to make contribution for this framework. Chainer make use of declarative programming means focuses on design and element rather than working on workflow.

Tensor flow is a user friendly tool for the implementation and testing of machine learning algorithms. Algorithms processed on TensorFlow can be run on multiple and wide variety of different systems it can be phone, tablets and can be large-scale distributed systems having hundreds of computers and thousands of device such as GPU cards. This machine learning framework is extensible and able to process large number of different algorithms, which include training, testing and inference algorithms for the models based on deep neural networks. It has been used for research purpose and for the installation of machine learning systems into production over multiple. Tensor flow is a user friendly tool for the implementation and testing of machine learning algorithms. Algorithms processed on Tensor Flow can be run on multiple and wide variety of different systems it can be phone, tablets and can be large-scale distributed systems having hundreds of computers and thousands of device such as GPU cards. This machine learning framework is extensible and able to process large number of different algorithms, which include training, testing and inference algorithms for the models based on deep neural networks. It has been used for research purpose and for the installation of machine learning systems into production. I thing because of the advancement in the machine learning technology and with the help of different deep learning techniques the scope of applications where these techniques can be used as an end product increase day by day. So these devices like mobile phones having AI camera feature. So these frameworks can easily deploy the model in such device.

## **2.3 State of Art Methods and Approaches:**

Techniques based on neural networks are the reason for the tremendous increase in different application area and because of increase in demand of these methods often usage result in the development of new frameworks to facilitate the researcher to test their cases using these frameworks. In this papers they are doing comparative study on for of deep learning frameworks widely used for research and development purpose so these frameworks required extensibility, high performance hardware, and speed so that why are making comparison between them on the basis of these factors.

They tested different neural network architectures on a single computer for both case CPU (Multi-threaded) and GPU. Two factors used for speed performance testing

* Gradient Computation (Important at the training phase of network)
* Forward Time (Important for deployment perspective)

How these frameworks support different architecture of convolutional neural networks? Theano and Torch more extensible. Theano outperforms on GPU training an LSTM networks deployment. Caffe is simple and easy to handle for the performance of standard neural network architectures

I think because of the increase in the amount of data deep learning techniques are becoming more popular because of their accuracy rate and learning complex features from large amount of data.

Because of the popularity and these specialties these techniques are widely use in different fields i.e. speech recognition, natural language processing. As these techniques also give more performance as compare to state-of-the-art methods.

Increase in the development of the techniques based on deep learning in different areas, Big tech companies and researcher’s teams created unique and independent tools for the advancement of this technology. So in this papers they introduce 18 similar deep learning frameworks and libraries.

Along with the detail description and display of benchmarks. They collectively explain 18 different common deep learning frameworks and libraries. And as their contribution they also provided the overall score of the existing main deep learning frameworks from six different aspects.

1. Model design ability
2. Interface property
3. Deployment ability
4. Performance
5. Framework design
6. Prospects for development

## **2.4 Comparison and Explanation:**

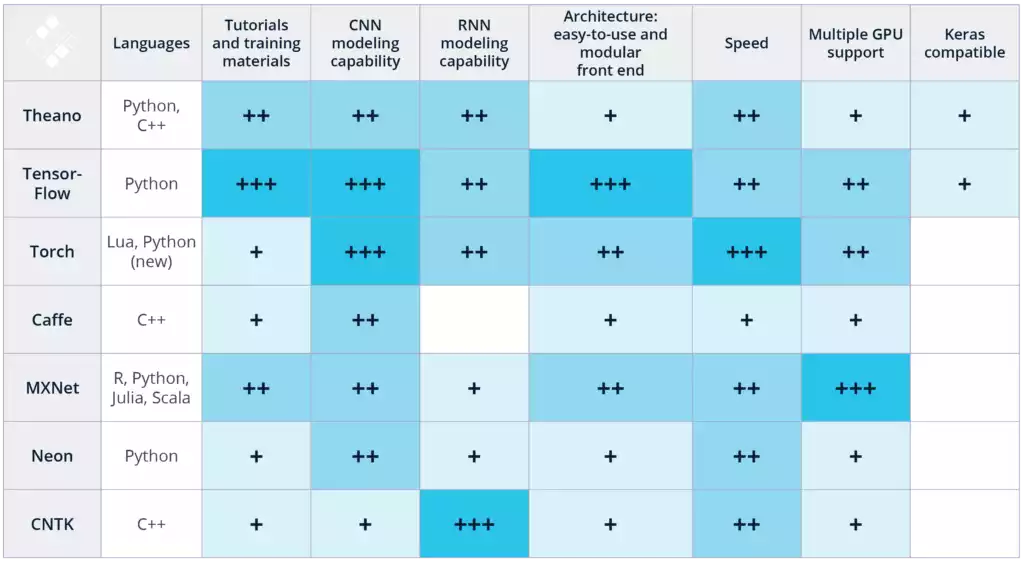


Figure 1 Comparison Graph

## **2.5 Problem Solution:**

As we have discussed earlier that when a person need to implement an algorithm and for testing this algorithm he can se any of these frameworks discussed in the existing work but the issue is they have to create environment manually we want to automate this with command just manually by selecting and clicking on the few options.

## **2.6 Summary:**

We are trying to implement a framework in a complete structured form which include scalability we can add multiple algorithms in this framework and get results with one click. As currently now we have used sate of the art methods we can extend this framework so user can customize all aspects of their algorithm and get results accordingly.

# **Methodology (Chapter 3)**

## **3.1 System Requirements:**

In this part we will briefly describe the basic requirement for the system. Following are the requirements for the system.

Basically we can implement multiple techniques

1. Hidden Markov Model
2. Queue Learning
3. Artificial Neural Networks

**Dataset:**

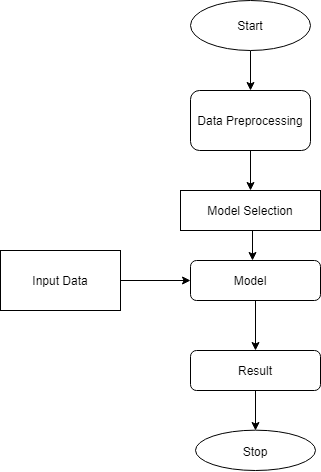
Depends on the techniques we will use to complete this task. Suppose if we have hidden markov model so data needed for it should have these characteristics.

Specific feature should be selected that can depend on the use case scenario that we will be implementing as in our case it would be that no of time particular whether took place.

In case of neural network, we required a dataset as required for supervised learning such dataset contain already selected features and all scenarios defined in the form of tuples and label class that will be used to train the model and later use that model for prediction.

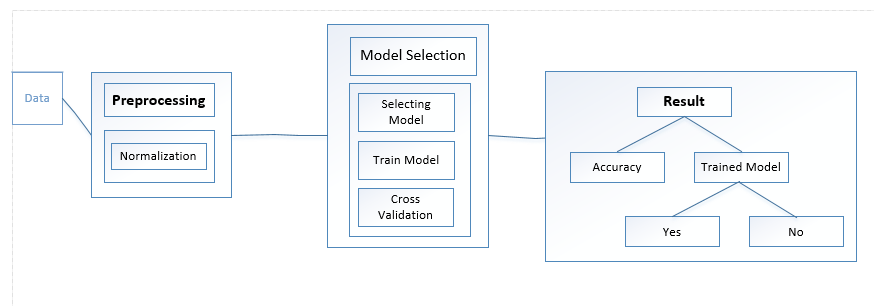
## **3.2 System Overview:**

By using these techniques many research has been already done but they worked on a specific use case but we want to create a framework that can make a use of multiple techniques so multiple use cases dataset can be train and used for disaster management for predictive and preventive measures.

****

***Figure 2 Framework Flowchart***

The proposed technique is expected to work on multiple use cases and provide results with a reasonable degree of accuracy. You can feed multidimensional data unless it should be in preprocessed format our framework will reduce losses in the training phases and increase performance of learning. Our system can be based on multiple artificial intelligence techniques so user can perform cross validation to achieve higher accuracy for their use case.



***Figure 3 Framework Overview***

## **3.3 Inputs to System:**

Input in our system also depend on the techniques we are using.

**Training time inputs:**

In case of neural networks in the phase of training, dataset will be used as input to the system because that dataset has multiple tuples as an input to the input layer of the neural network and also cross validate the output using label attribute in the output layer of neural networks.

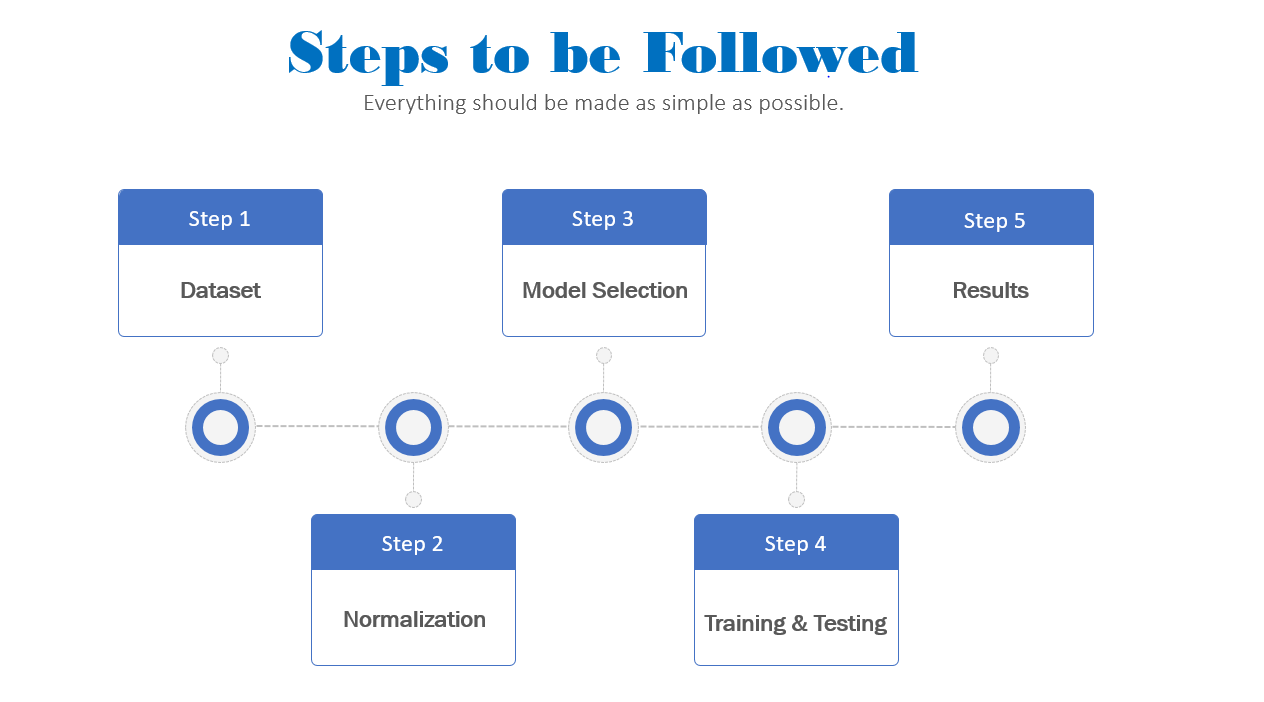
In case of hidden markov model input will be transition probability matrix and starting probability vector matrix. We can also train to optimize the parameters of HMM to achieve more accuracy in the scenarios in knowledge not remain consistent.

**Model Testing Time Inputs**:

User provide that trained model and also the inputs according to their use case on which he/she trained that data.

## **3.4 Method/ Technique:**

The diagram 3.1 below explains the working scheme of the framework in a general view



**Figure 4 Framework Steps**

There are basically three algorithms we will be using. The best technique providing the accurate results will be used. Following are the three algorithms that will be the basic principle of working.

1. Hidden Markov Model
2. Artificial Neural Networks
3. Q - Learning

**Hidden Markov Model:**

Below is the hidden markov model algorithm, the code is written in the python under process.

import numpy as np

# The statespace

states = ["Sunny","Rainy","Windy"]

# Possible sequences of events

transitionName = [["SS","SR","SW"],["RS","RR","RW"],["WS","WR","WW"]]

# Probabilities matrix (transition matrix)

transitionMatrix = [[0.2,0.6,0.2],[0.1,0.6,0.3],[0.2,0.7,0.1]]mport random as rm

if sum(transitionMatrix[0])+sum(transitionMatrix[1])+sum(transitionMatrix[1]) != 3:

print("Somewhere, something went wrong. Transition matrix, perhaps?")

else: print("All is gonna be okay, you should move on!! ;)")

#All is gonna be okay, you should move on!! ;)

**Hidden Markov Model**

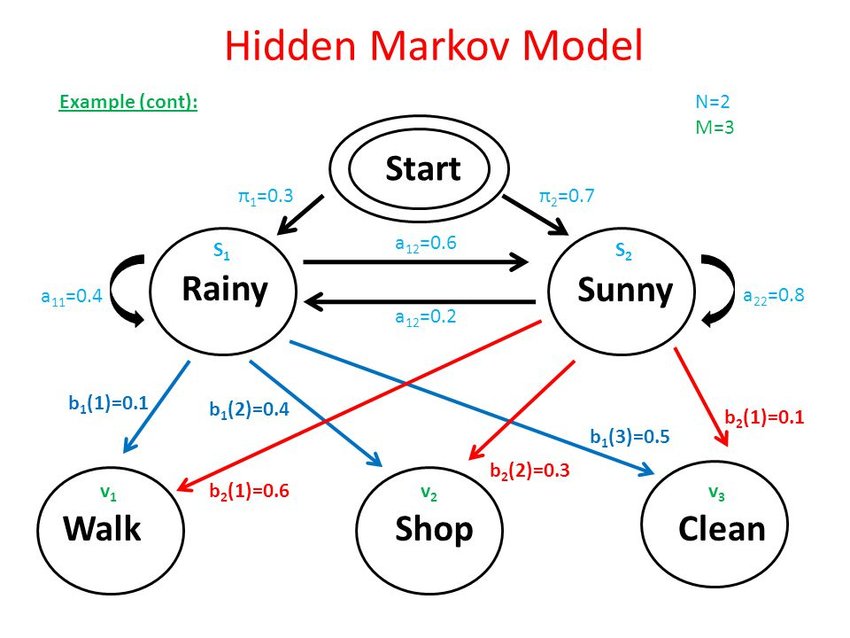


Figure 5 : HMM Example

**Steps Involved in Hidden Markov model**

**Step 1**

Analyzing data and calculate probabilities of transition from one state to another.

**Step 2**

Create a Markov probabilistic matrix (That hold Markova property)

**Step 3**

Create starting probability vector.

**Step 4**

Multiple the vector with the Markov probabilistic matrix to get one step event prediction.

## **3.5 Design Structure:**

**Pseudo Code**

Main Framework Pseudo Code

1 df=read\_csv(“Dataset.csv”)Take dataset as input to the system

2 Normalize dataset (DSn=DS-DS(min)/DS(max)-DS(min))

3 Model selection and dataset as input

5 Splitting data (TrainingData=0.7, TestingData=0.3)

6 Training and Testing (Train(TrainingDataset),Test(TestingDataset))

7 Cross Validation(Checking multiple factors of Trained Model accuracy)

8 If Trainedmodel Accuracy is not greater than previously TrainedModel

9 repeat step 6 and step 7

11 Else

12 Provide Trained Model file

END

## **3.6 Scenarios of Processing:**

**Hidden Markov Model Algorithm:**

The HMM algorithm works in two steps. The HMM is trained in the first step using the training sequences. At the initial state (at time t0), the state transition probabilities and the observable output probabilities are randomly assigned.

However, assigning these probabilities according to prior knowledge of the system, instead of the random assignment, can improve the performance of HMM. At this point, the model is denoted with λ0. Then, applying the Baum-Welch algorithm, the HMM λ0 is adjusted according to the input training sequences and construct the new model λ1 [1].

After every adjustment of λ, the probability difference of the previous model and the adjusted model is calculated. If the difference is below the preset probability difference threshold, the model is known to be the final HMM. Otherwise, further adjustment is required.

In the next step, the unknown sequences are applied to the model and the likelihood of the sequences (i. e., the probability of how much a sequence conforms the HMM) are determined. If the probability is above the predefined acceptable probability, the sequence is concluded as a non-anomalous sequence [6]. Otherwise, it is concluded as an anomalous one.

The HMM algorithm has very accurate prediction of anomaly and has been used for complex sequence analysis. However, the model training time is very high in HMM algorithm.

**Artificial Neural Networks:**

The Artificial Neural Network (ANN) is based on the belief that working of human brain by making the right connections, can be imitated using silicon and wires as living neurons and dendrites.

The human brain is composed of 86 billion nerve cells called neurons [6]. They are connected to other thousand cells by Axons. Stimuli from external environment or inputs from sensory organs are accepted by dendrites. These inputs create electric impulses, which quickly travel through the neural network. A neuron can then send the message to other neuron to handle the issue or does not send it forward. [2]

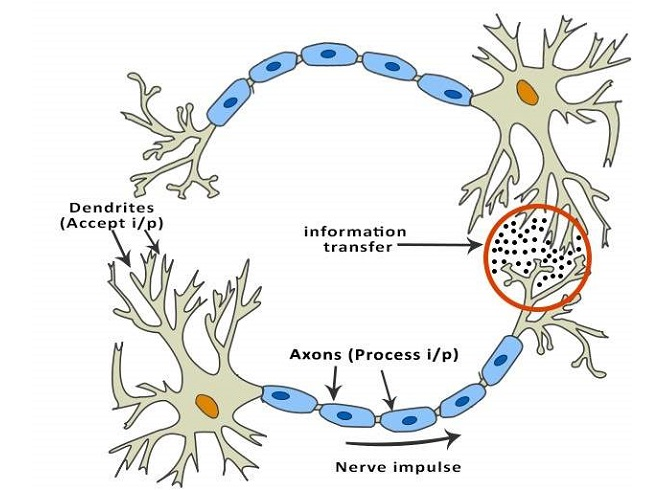


Figure 6: Biological Neuron network

ANNs are composed of multiple nodes, which imitate biological neurons of human brain. The neurons are connected by links and they interact with each other. The nodes can take input data and perform simple operations on the data. The result of these operations is passed to other neurons. The output at each node is called its activation or node value [2].

Each link is associated with weight. ANNs are capable of learning, which takes place by altering weight values.

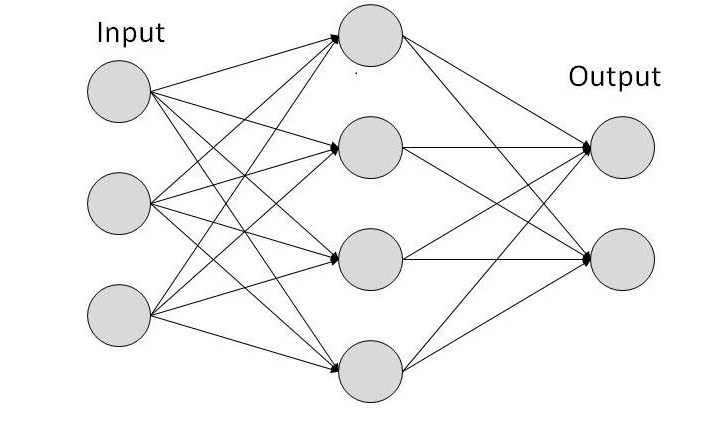


Figure 7 : Artificial Neural Network model

Q – Learning:

**Reinforcement Learning** briefly is a paradigm of Learning Process in which a learning agent learns, overtime, to behave optimally in a certain environment by interacting continuously in the environment. The agent during its course of learning experience various different situations in the environment it is in. These are called states.

**Q-Learning** is a basic form of Reinforcement Learning which uses Q-values (also called action values) to iteratively improve the behavior of the learning agent [4]

1. Q-Values or Action-Values: Q-values are defined for states and actions.  Q (S, A) is an estimation of how good is it to take the action A at the state S[3].
2. Rewards and Episodes: An agent over the course of its lifetime starts from a start state, makes a number of transitions from its current state to a next state based on its choice of action and also the environment the agent is interacting in. At every step of transition, the agent from a state takes an action, observes a reward from the environment, and then transits to another state. If at any point of time the agent ends up in one of the terminating states that means, there are no further transition possible. This is said to be the completion of an episode [3].
3. This estimation of  Q(S, A) will be iteratively computed using the **TD- Update rule**https://media.geeksforgeeks.org/wp-content/uploads/TD_Update-300x30.jpg
4. The update rule to estimate the value of Q is applied at every time step of the agent’s interaction with the environment.
5. **Choosing the Action to take using e-greedy policy:**
6. E-greedy policy of is a very simple policy of choosing actions using the current Q-value estimations. With probability (1- e) choose the action which has the highest Q-value.  
   With probability (e) choose any action at random.

The terms used in Q – Learning are:

* S: Current State of the agent.
* A: Current Action Picked according to some policy.
* S’: Next State where the agent ends up.
* A’: Next best action to be picked using current Q-value estimation, i.e. pick the action with the maximum Q-value in the next state.
* R: Current Reward observed from the environment in Response of current action.
* Y: (>0 and <=1): Discounting Factor for Future Rewards. Future rewards are less valuable than current rewards so they must be discounted. Since Q-value is an estimation of expected rewards from a state, discounting rule applies here as well.
* a: Step length taken to update the estimation of Q (S, A).

## **3.7 Expected Output:**

Expected output will be the accuracy and the time taken training of models. We will test the model with custom inputs from outside of the dataset and check what will be the prediction of the system. We assume that implemented techniques will taking less time to train and with reasonably high accuracy. It may performance of system and better helpful in implemented use case scenario.

## **3.8 Summary:**

In this chapter we have introduced three techniques basically which will help us achieve our goal of making the more reliable and convenient framework for taking predictive and preventive measures. Neural network and hidden markov model are the techniques on which we are working and about to complete and will be implemented in next phase. One of the algorithm will be used for the training and help in predicting accordingly. The streams of data captured by using any of one technique will help us to improve it for future use by making the results more accurate.

# **4. Implementation (Chapter 4)**

## **4.1 Development Tool:**

PyCharm IDE is used as it provides us with multiple tools to integrate existing libraries and frameworks like NumPy, Scipy and Scikit-learn.

The design of our tool is made via Chart.js and standard web technologies for Django version of the tool. We also used a Django real-time server to take care of processing at the backend of Web Interface, our desktop version is designed via PyQt5 framework.

**Documentation Tools:**

* For writing the thoughts and discussions with the supervisor we have used OneNote tool. It was best for organizing all the information.
* For writing a thesis, we have used Word.
* JabRef is used to save all the papers related to our thesis.
* For making diagrams we have used Visio.

**Technology:**

* Python
* PyQt5
* Django

**Training and Validation of Dataset:**

* Input will be given in the form of Dataset .csv file.

**Other tools used in development:**

* Anaconda Navigator
* Jupiter Notebook
* PyCharm
* Spiders

## **4.2 Implementation Issues:**

Following were the implementation issues faced in beginning during the implementation phase:

* Find the correct datasets with similar properties for testing and training of Algorithms. Data pre-processing issues.
* Transforming data into predefined templates. High computation power required.
* Measure the performance on different performance Metrics.

## **4.3 Configuration Management:**

In our research work, we have focused on following:

Check and verify the training and testing of algorithms.

Performance Metrics – Measuring Performance for different scenarios by using different metrics.

Teamwork – Checking that team is provided with aid related to the project.

Environment Managing -Managing the libraries that are necessary for the framework. Process Management – Ensuring to follow the organization's buildup process.

Examining – Making sure that project consists of all parts.

Quality – During the training and testing process all information must be stored. Control – Implementing a controlled change process.

**Example:**

During the phase of training and testing of algorithms all the results which are extracted must be store correctly to identify the best algorithm for given dataset.

## **4.4 Framework Section for development:**

We have selected a well renowned framework called “NumPy” for data manipulation and “Scikit-learn” for statistical analysis on data. The use of Scikit learn in our framework allows us to deliver a free machine learning library for Python. Our tool features various algorithms like support vector machines, random forests, and k-neighbors, and it also supports Python numerical and scientific libraries like NumPy for manipulation of data at runtime and Scipy. The used framework will allow us to integrate and make use of many algorithms that will need by data scientists for various data analysis. We use PyCharm IDE which allow us to work on multiple development frameworks such as PyQt for desktop application development and Django for web application development.

## **4.5 Deployment Factors:**

Applications of this system are

* Rapid testing and analysis of data
* Manipulation of data
* Testing of multiple algorithms over data to feed it into other systems

The tool is highly flexible with multiple datasets as it cares for availability of multiple algorithms.

## **4.6 Summary:**

In the development of this framework for statistical analysis of data to prevent or predict events (disasters, market crashes), the main focus is to target those data scientists that perform statistical analysis on daily use. This tool will provide time efficiency and flexibility to data scientists because all the algorithms used will be general purpose.

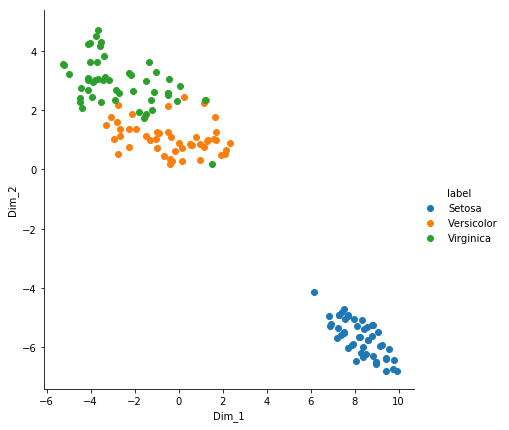
# **5. Chapter 5**

## **5.1 Introduction:**

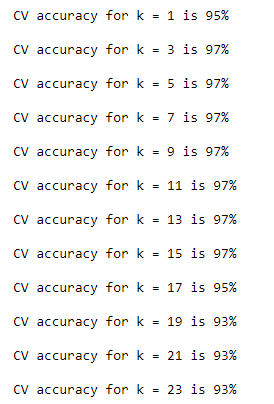
Proposing an interaction index framework for predictive analysis. Our framework will help the data scientists to analyze data. This framework will be based on existing machine learning techniques and libraries to analyze data on multiple techniques with less technical skills.

## **5.2 Output:**

Expected output will be the accuracy and the time taken training of models. We will test the model with custom inputs from outside of the dataset and check what will be the prediction of the system. We assume that implemented techniques will taking less time to train and with reasonably high accuracy. It may performance of system and better helpful in implemented use case scenario.



***Figure 8 KNN output***



***Figure 9 KNN output***

## **5.3 Testing:**

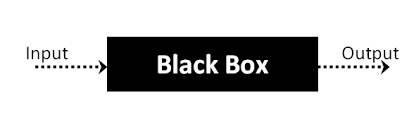
Testing is done in order to check if the actual output matches the expected results. It is done to check that a software is working properly or is there any defect. The execution time is observed as well as to check bugs or any missing requirements. It can be done using a tool or either achieved by manual process.

In other words, it is narrated as black box testing or white box testing.

**Black Box Testing:**

In this type of testing the external test or functional test of the system is done.

**White Box Testing:**

The type of testing in which the internal structure of the system is verified. The internal working of the code, its functionality.

The internal structure of our code has been tested and verified as it works without any faults after the final stages. The code was being tested in all the phases of development.

***Figure10 Black Box testing***

## **5.4 Statistical and Graphical Analysis:**

The statistical and graphical analysis is done to check if the system is working properly in a given time with the help of a graph. Below given graph help in understanding the process.

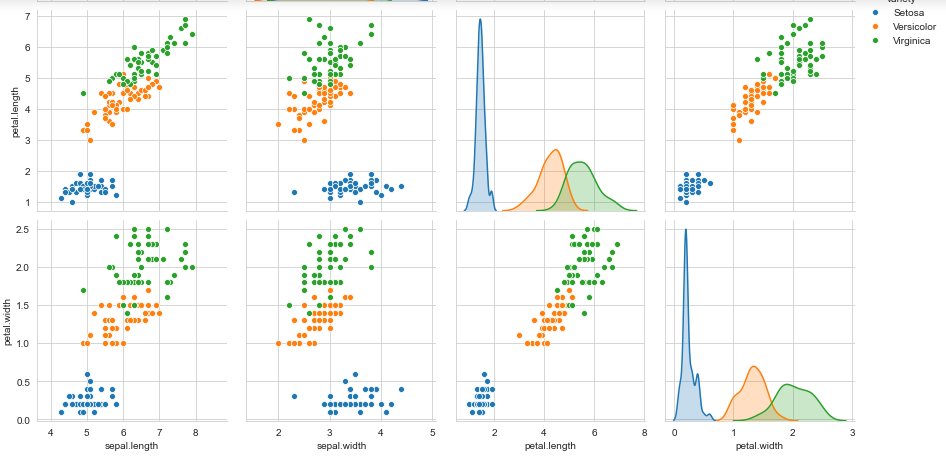


Figure 11: Algorithms output analysis

## **5.5 Complete User Interface:**

The code is written in python.

The tool used is PyCharm, Anaconda, Visual Studio Code.

Users Dataset file also provide label class name system normalize the data and ask for multiple algorithm implementation option to user for predictive analysis.

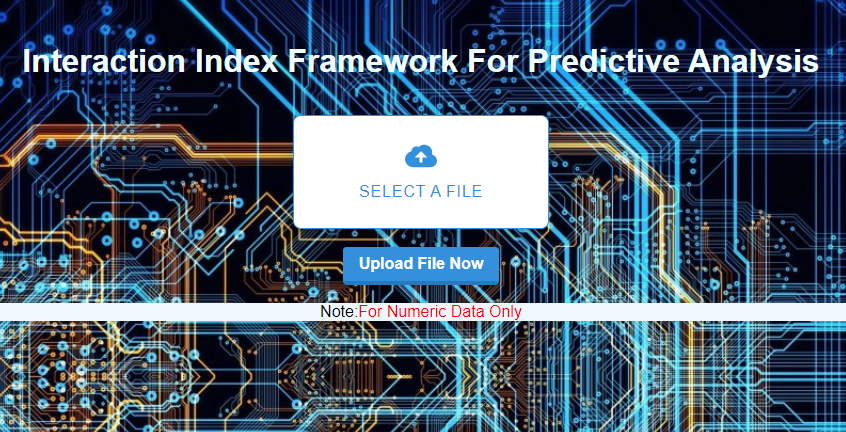
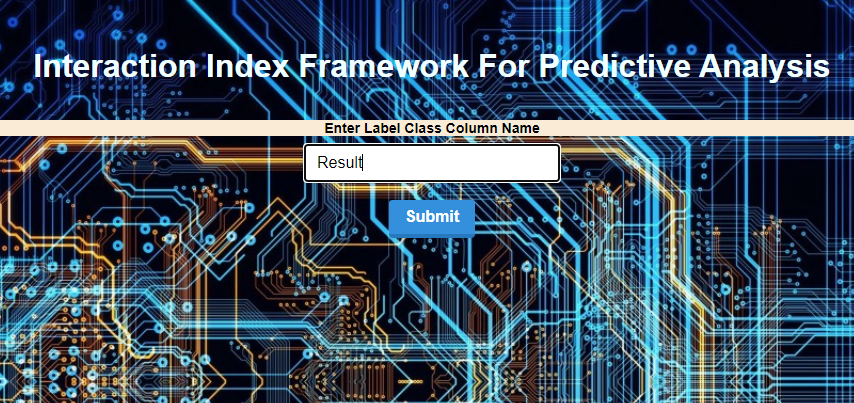
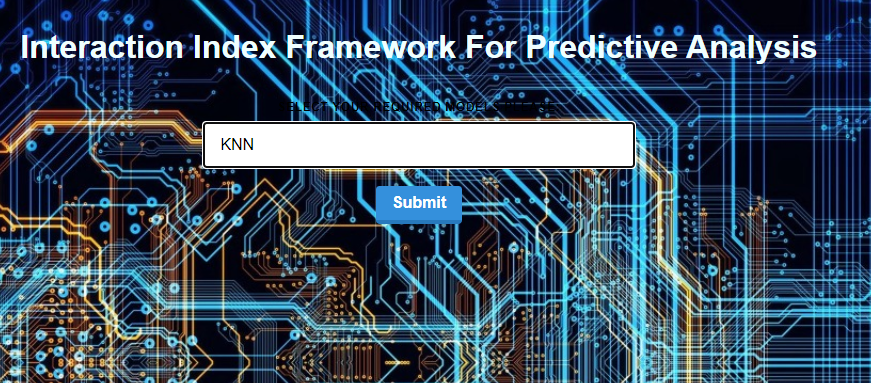


Figure 12 Framework main pageFigure 13 Class Label interface

## Figure 14 Loading interface

Figure 15 Model Selection interface

## **5.6 Summary:**

We have developed this framework in a complete structured form which include expendability we can add multiple algorithms in this framework and get results with one click. As currently now we have used sate of the art methods we can extend this framework so user can customize all aspects of their algorithm and get results accordingly.

# **References:**

[1] Leonard E. Baum, Ted Petrie, George Soules and Norman Weiss, “A Maximization Technique Occurring in the Statistical Analysis of Probabilistic Functions of Markov Chains”, The Annals of Mathematical Statistics, vol. 41(1)  
<https://sci-hub.tw/10.1145/3158233.3159306>

[2] Q-learning Watkins, Christopher J. C. H volume 8 machine learning journal.<https://sci-hub.tw/10.1145/3078240>

[3] Barto, A.G., Bradtke, S.J. & Singh, S.P. Real-time learning and control using asynchronous dynamic programming. (COINS technical report 91-57). Amherst: University of Massachusetts.  
<https://sci-hub.tw/10.1145/3005719>

[4] Mahadevan & Connell .Automatic programming of behavior-based robots using reinforcement learning.Proceedings of the National Conference on AI (pp. 768–773)<https://sci-hub.tw/10.1145/2989250.2989272>

[5] Sato, M., Abe, K. & Takeda, H. (1988). Learning control of finite Markov chains with explicit trade-off between estimation and control.IEEE Transactions on Systems, Man and Cybernetics, 18, pp. 677–684.   
<https://sci-hub.tw/10.1109/ECTI-NCON.2018.8378271>

[6] Sutton, R.S. (1984).Temporal credit assignment in reinforcement learning. PhD Thesis, University of Massachusetts, Amherst, MA  
<https://sci-hub.tw/10.1109/ICETETS.2016.7603012>

[7] Werbos, P.J. (1977). Advanced forecasting methods for global crisis warning and models of intelligence.General Systems Yearbook, 22, pp. 25–38.  
<https://sci-hub.tw/10.1145/3239283.3239306>

[8] Big data driven mobile traffic understanding and forecasting: A time series approach

F Xu, Y Lin, J Huang, D Wu, H Shi, J Song, Y Li IEEE transactions on services computing 9 (5), 796-805<https://sci-hub.tw/10.1145/3067695.3075976>

[9] Sharma, V., Cali, U., Hagenmeyer, V., Mikut, R., &Ordiano, J. Á. G. (2018). Numerical Weather Prediction Data Free Solar Power Forecasting with Neural Networks.   
<https://sci-hub.tw/10.1145/3208903.3210279>

[10] Wang, M., Zhou, W., Tian, Q., Pu, J., & Li, H. (2017). Deep Supervised Quantization by Self-Organizing Map.   
<https://sci-hub.tw/10.1145/3123266.3123415>

[11] Braga, P. H. M., &Bassani, H. F. (2018). A Semi-Supervised Self-Organizing Map for Clustering and Classification.  
<https://sci-hub.tw/10.1109/IJCNN.2018.8489675>

[12] Kobayashi, M. (2018). Decomposition of Rotor Hopfield Neural Networks Using Complex Numbers.  
<https://sci-hub.tw/10.1109/TNNLS.2017.2657781>

[13] Azayite, F. Z., &Achchab, S. (2018). Topology design of bankruptcy prediction neural networks using Particle swarm optimization and backpropagation.  
<https://sci-hub.tw/10.1145/3230905.3230951>

[14] Li, L.-H., Lin, C.-T., & Chen, S.-F. (2017). Micro-lending Default Awareness Using Artificial Neural Network.  
<https://sci-hub.tw/10.1145/3145511.3145515>