**1. INTRODUCTION**

Brain tumor is an abnormal growth of cells inside the skull. Normally the tumor will grow from the cells of the brain, blood vessels, nerves that emerge from the brain. There are two types of tumor which are benign (non-cancerous) and malignant (cancerous) tumors. The former is described as slow growing tumors that will exert potentially damaging pressure but it will not spread into surrounding brain tissue. However, the latter is described as rapid growing tumor and it is able to spread into surrounding brain. Tumors can damage the normal brain cells by producing inflammation, exerting pressure on parts of brain and increasing pressure within the skull. it shows the presence of tumor in the Brain. Radiologists examine the patient physically by using Computed Tomography (CT scan) and Magnetic Resonance Imaging (MRI) MRI images showed the brain structures, tumor’s size and location. From the MRI images the information such as tumors location provided radiologists, an easy way to diagnose the tumor plan the surgical approach for its removal.

Machine learning algorithms have the potential to be invested deeply in all fields of medicine, from drug discovery to clinical decision making, significantly altering the way medicine is practiced. The success of machine learning algorithms at computer vision tasks in recent years comes at an opportune time when medical records are increasingly digitalized. The use of electronic health records (EHR) quadrupled from 11.8% to 39.6% amongst office-based physicians in the US from 2007 to 2012. Medical images are an integral part of a patient’s EHR and are currently analyzed by human radiologists, who are limited by speed, fatigue, and experience. It takes years and great financial cost to train a qualified radiologist, and some health-care systems outsource radiology reporting to lower-cost countries such as India via tele-radiology. A delayed or erroneous diagnosis cause’s harm to the patient. Therefore, it is ideal for medical image analysis to be carried out by an automated, accurate and efficient machine learning algorithm.

Medical image analysis is an active field of research for machine learning, partly because the data is relatively structured and labeled, and it is likely that this will be the area where patients first interact with functioning, practical artificial intelligence systems. This is significant for two reasons. Firstly, in terms of actual patient metrics, medical image analysis is a litmus test as to whether artificial intelligence systems will actually improve patient outcomes and survival. Secondly, it provides a tested for human-AI interaction, of how receptive patients will be towards health altering choices being made, or assisted by a non-human actor.

**1.1 BREIF INFORMATION**

Gone are the days, when health-care data was small. Due to the tremendous advancement in image acquisition devices, the data is quite large (moving to big data), that makes it challenging and interesting for image analysis. This rapid growth in medical images and modalities requires extensive and termidious aborts by medical expert that is subjective, prone to human error and may have large variations across deferent expert. Alternative solution is using machine learning techniques to automate diagnosis process however, traditional machine learning methods are not sufficient to deal with complex problem. The machine learning promise the capacity to deal big medical image data for accurate and ancient diagnosis. Deep learning will not only help to select and extract features but also construct new ones, furthermore, it does not only diagnose the disease but also measure predictive target and provides actionable prediction models to help physician efficiently.

Machine Learning (ML) and Artificial Intelligence (AI) have progressed rapidly in recent years. Techniques of ML and AI have played important role in medical field like medical image processing, computer-aided diagnosis, image interpretation, image fusion, image registration, image segmentation, image-guided therapy, image retrieval and analysis Techniques of ML extract information from the images and represents information effectively and efficiently. The ML and AI facilitate and assist doctors that they can diagnose and predict accurate and faster the risk of diseases and prevent them in time. These techniques enhance the abilities of doctors and researchers to understand that how to analyze the generic variations which will lead to disease. These techniques composed of conventional algorithms without learning like Support Vector Machine (SVM), Neural Network (NN), KNN etc. and deep learning algorithms such as Convolutional Neural Network (CNN),Artificial neural network (ANN), Recurrent neural Network (RNN), Long Short term Memory (LSTM), Extreme Learning Model (ELM), Generative Adversarial Networks (GANs) etc. Former algorithms are limited in processing the natural images in their raw form, time consuming, based on expert knowledge and requires a lot time for tuning the features. The later algorithms are fed with raw data, automatic features learner and fast. These algorithms try to learn multiple levels of abstraction, representation and information automatically from large set of images that exhibit the desired behavior of data. Although automated detection of diseases based on conventional methods in medical imaging has been shown significant accuracies around for decades, but new advances in machine learning techniques have ignited a boom in the deep learning. Deep learning based algorithms showed promising performance as well speed in different domains like speech recognition, text recognition, lips reading, computer-aided diagnosis, face recognition, drug discovery.

**1.2 MOTIAVTION**

The motivation is to provide the comprehensive review of deep learning based algorithms in medical image analysis problems in terms of current work and future direction. It provides the fundamental knowledge and the state of the art approaches about deep learning in the domain of medical image processing and analysis.

**1.3 OBJECTIVE**

In medical image analysis, the lack of data is two-fold and more accurate there is general lack of publicly available data, and high quality labeled data is even more scarce. Most of the datasets presented in this review involve fewer than 100 patients. Yet the situation may not be as dire as it seems, as despite the small training datasets, the report relatively satisfactory performance in the various tasks.

* 1. **PROBLEM STATEMENT**

In this system when using machine learning techniques not achieved the 100% accuracy results and Medical human radiologist has support limited speed, fatigue and experience. User has to get lot of delay to the diagnosis causes reports.

**2. SYSTEM ANALYSIS**

* 1. **EXISTING SYSTEM**

In machine learning, you need fewer data to train the algorithm than deep learning. Deep learning requires an extensive and diverse set of data to identify the underlying structure. Besides, machine learning provides a faster-trained model. Most advanced deep learning architecture can take days to a week to train. The advantage of deep learning over machine learning is it is highly accurate. You do not need to understand what features is the best representation of the data, the neural network learned how to select critical features. In machine learning, you need to choose for yourself what features to include in the model.

Accurate diagnoses of disease depend upon image acquisition and image interpretation. image acquisition devices has improved substantially over the recent few years i.e. currently we are getting radiological images ((X-RAY, CT and MRI Scans etc.) with much higher resolution. however, we just started to get benefits for automated image interpretation. one of the best machine learning application is computer vision, though traditional machine learning algorithms for image interpretation rely heavily on expert crafted features i.e. Brain tumor detection requires structure features to be extracted. due to the extensive variation from patient to patient data, traditional learning methods are not reliable. machine learning has evolved over the last few years by its ability to shift through complex and big data.

Now deep learning has got great interest in each and every field and especially in medical image analysis and it is expected that it will hold $300 million medical imaging market by 2021. Thus, by 2021, it alone will get more investment for medical imaging than the entire analysis industry spent in 2016. It is the most effective and supervised machine learning approach. This approach use models of deep neural network which is variation of neural network but with large approximation to human brain using advance mechanism as compare to simple neural network. The term deep learning implies the use of a deep neural network model. The basic computational unit in a neural network is the neuron, a concept inspired by the study of the human brain, which takes multiple signals as inputs, combines them linearly using weights, and then passes the combined signals through nonlinear operations to generate output signals.

**Disadvantages of Existing System**

* Deep learning is gaining more importance than machine learning. Deep learning is proving to be one of the best techniques in state-of-art performance.
* Machine learning works with large amounts of data. It is useful for small amounts of data too. Deep learning on the other hand works efficiently if the amount of data increases rapidly.
* Deep learning is generating a major impact in computer vision and medical imaging.

**2.2 PROPOSED SYSTEM**

AI algorithms moved from heuristics-based techniques to manual, handcrafted feature extraction techniques and then to supervised learning techniques. Unsupervised machine learning methods are also being researched, but the majority of the algorithms from 2015-2017 in the published literature have employed supervised learning methods, namely Convolutional Neural Networks (CNN). Aside from the availability of large labeled data sets being available, hardware advancements in Graphical Processing Units (GPUs) have also led to improvements in CNN performance, and their widespread use in medical image analysis.

Deep learning is the growing trend to develop automated applications and has been termed in 10 breakthrough technologies of 2013. Today, several deep learning based computer vision applications are performing even better than human i.e. identifying indicators for cancer in blood and tumors in MRI scans. It is improvement of artificial neural network that consist of more hidden layer that permits higher level of abstraction and improved image analysis. It becomes extensively applied method due to its recent unparalleled result for several applications i.e. object detection, speech recognition, face recognition and medical imaging.

Artificial neural networks structurally and conceptually inspired by human biological nervous system. Preceptron is one of the earliest neural networks that were based on human brain system. It consists of input layer that is directly connect to output layer and was good to classify linearly separable patterns. To solve more complex pattern, neural network was introduced that has a layered architecture i.e., input layer, output layer and one or more hidden layers. Neural network consist of interconnected neurons that takes input and perform some processing on the input data, and finally forward the current layer output to the coming layer. Each neuron in the network sums up the input data and applies the activation function to the summed data and finally provides the output that might be propagated to the next layer. Thus adding more hidden layer allows dealing with complex as hidden layer capture nonlinear relationship. These neural networks are knows as Deep Neural network.

Both the 2-dimensional and 3-dimensional structures of an organ being studied are crucial in order to identify what is normal versus abnormal. By maintaining these local spatial relationships, CNNs are well-suited to perform image recognition tasks. CNNs have been put to work in many ways, including image classiﬁcation, localization, detection, segmentation and registration. CNNs are the most popular machine learning algorithm in image recognition and visual learning tasks, due to its unique characteristic of preserving local image relations, while performing dimensionality reduction. This captures important feature relationships in an image (such as how pixels on an edge join to form a line), and reduces the number of parameters the algorithm has to compute, increasing computational efﬁciency.CNN separable to take as inputs and process both 2-dimensional images, as well as 3-dimensional images with minor medications. This is a useful advantage in designing a system for hospital use, as some modalities like X-rays are 2-dimensional while others like CT or MRI scans are 3-dimensional volumes. CNNs and Recurrent Neural Networks (RNNs) are examples of supervised machine learning algorithms, which require signiﬁcant amounts of training data. Unsupervised learning algorithms have also been studied for use in medical image analysis. These include Auto encoders, Restricted Boltzmann Machines (RBMs), Deep Belief Networks (DBNs), and Generative Adversarial Networks (GANs).

**Advantages of Proposed System**

* Machine learning works with large amounts of data. It is useful for small amounts of data too. Deep learning on the other hand works efficiently if the amount of data increases rapidly.
* Deep learning algorithms are designed to heavily depend on high-end machines unlike the traditional machine learning algorithms. Deep learning algorithms perform a number of matrix multiplication operations, which require a large amount of hardware support.
* Feature engineering is the process of putting domain knowledge into specified features to reduce the complexity of data and make patterns that are visible to learning algorithms it works.
* The traditional machine learning algorithms follow a standard procedure to solve the problem. It breaks the problem into parts, solve each one of them and combine them to get the required result. Deep learning focusses in solving the problem from end to end instead of breaking them into divisions.
* Interpretability is the major factor for comparison of machine learning and deep learning algorithms. The main reason is that deep learning is still given a second thought before its usage in industry.

**2.3 FEASIBILITY STUDY**

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential. Three key considerations involved in the feasibility analysis are

**2.3.1 Economical Feasibility**

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

### **2.3.2 Technical Feasibility**

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

**2.3.3 Social Feasibility**

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

**2.4 SYSTEM REQUIREMENT SPECIFICATION**

**2.4.1 Functional Requirements**

A functional requirement defines a function of a system or its components. Functional requirements may be calculations , technical details, data manipulation and processing and other specific functionality that defines what a system is supposed to accomplish the functional requirement specification documents the operation and activities that a system able to perform. Functional requirements include functions performed by specific screens, outlines of work flows performed by the system, and other business compliance requirements the system must meet. This project has four modules. There are Convolutional Layers, Rectified Linear Unit (RELU) Layers, Pooling Layers and Fully Connected layer.

**2.4.2 Non Functional Requirements**

Non-functional requirements define the overall qualities or attributes of the resulting System Non-functional requirements place restrictions on the product being developed, the development process, and specify external constraints that the product must meet. Examples of NFR include safety, security, usability, reliability and performance Requirements. Project management issues (costs, time, and schedule) are often considered as non-functional requirements.

**Performance requirements**

Requirements about resources required, response time, transaction rates, throughput, benchmark specifications or anything else having to do with performance. In this project, Data publisher (or data holder, who collects data from record owner ex. Alice and bob) and data miner or the public, called the data recipient and record owners like patients and doctors.

**Modifiability**

Requirements about the effort required to make changes in the software. Often, the measurement is personnel effort (person- months).

**Portability**

The effort required to move the software to a different target platform. The measurement is most commonly person-months or % of modules that need changing.

**Reliability**

Requirements about how often the software fails. The measurement is often expressed in MTBF (mean time between failures). The definition of a failure must be clear. Also, don't confuse reliability with availability which is quite a different kind of requirement.  Be sure to specify the consequences of software failure, how to protect from failure, a strategy for error detection, and a strategy for correction.

**Security**

One or more requirements about protection of your system and its data. The measurement can be expressed in a variety of ways (effort, skill level, time) to break into the system.  Do not discuss solutions (e.g. passwords) in a requirements document.

**Usability**

Requirements about how difficult it will be to learn and operate the system. The requirements are often expressed in learning time or similar metrics.

**Legal**

There may be legal issues involving privacy of information, intellectual property rights, export of restricted technologies, etc.

**2.5 HARDWARE REQUIREMENTS**

* Processor : Intel i3 and above
* RAM : 4 GB(min)
* Hard Disk : 100 GB

**2.6 SOFTWARE REQUIREMENTS**

* Operating system : Windows7 (Min).
* Coding Language : Python
* Front-End : Python with Anaconda
* Dataset : **Feature Dataset**

**3. SYSTEM DESIGN**

**3.1 SYSTEM ARCHITECTURE**

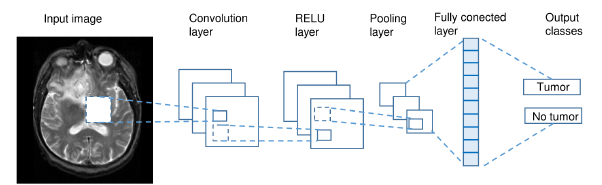


Fig. 3.1 System Architecture

**3.2 MODULES**

The Following Modules are:

* Convolution Layer
* Rectified Linear Unit (RELU) Layer
* Pooling Layer
* Fully Connected layer

**Module Description**

**Convolution Layer**

A convolution is defined as an operation on two functions. In image analysis, one function consists of input values (e.g. pixel values) at a position in the image, and the second function is a filter (or kernel) each can be represented as array of numbers. Computing the dot product between the two functions gives an output. The filter is then shifted to the next position in the image as defined by the stride length. The computation is repeated until the entire image is covered, producing a feature (or activation) map. This is a map of where the filter is strongly activated and ‘sees’ a feature such as a straight line, a dot, or a curved edge. If a photograph of a face was fed into a CNN, initially low-level features such as lines and edges are discovered by the filters. These build up to progressively higher features in subsequent layers, such as a nose, eye or ear, as the feature maps become inputs for the next layer in the CNN architecture.

Convolution exploits three ideas intrinsic to perform computationally efficient machine learning: sparse connections, parameter sharing (or weights sharing) and equivariant (or invariant) representation. Unlike some neural networks where every input neuron is connected to every output neuron in the subsequent layer, CNN neurons have sparse connections, meaning that only some inputs are connected to the next layer. By having a small, local receptive ﬁeld (i.e., the area covered by the ﬁlter per stride), meaningful features can be gradually learnt, and the number of weights to be calculated can be drastically reduced, increasing the algorithm’s efficiency. In using each ﬁlter with its ﬁxed weights across different positions of the entire image, CNNs reduce memory storage requirements. Parameter sharing results in the quality of equivariant representation to arise. This means that input translations result in a corresponding feature map translation. The convolution operation is deﬁned by the ∗ symbol. An output (or feature map) s(t) is deﬁned below when input I(t) is convolved with a ﬁlter or kernel K(a).

s(t)=(I ∗K)(t). (1)

If canonry takes integer values, the discretized convolution is given by:

s(t)=X a I(a)·K (t −a). (2)

The above assumes a one-dimensional convolutional operation. A two dimension convolution operation within put I (m, n) and a kernel K (a, b) is deﬁned as:

s(t)=X a X b I (a,b)·K (m−a,n−b). (3)

By the commutative law, the kernel is ﬂipped and the above is equivalent to:

s(t)=X a X b I (m−a,n−b)·K (a,b). (4)

Neural networks implement the cross-correlation function, which is the same as convolution but without ﬂipping the kernel. s(t)=X a X b I (m+a,n+b)·K (a,b). (5)

**Rectified Linear Unit (RELU) Layer**

The RELU layer is an activation function that sets negative input values to zero. This simpliﬁes and accelerates calculations and training, and helps to avoid the vanishing gradient problem. Mathematically it is deﬁned as:

f(x)=max(0,x). where x is the input to the neuron. Other activation functions include the sigmoid, tanh, leaky RELUs, Randomized RELUs and parametric RELUs.

**Pooling Layer**

The Pooling layer is inserted between the Convolution and RELU layers to reduce the number of parameters to be calculated, as well as the size of the image (width and height, but not depth). Max-pooling is most commonly used; other pooling layers include Average pooling and L2-normalization pooling. Max-pooling simply takes the largest input value within a filter and discards the other values; effectively it summarizes the strongest activations over a neighborhood. The rationale is that the relative location of a strongly activated feature to another is more important than its exact location

**Fully Connected Layer**

The final layer in a CNN is the Fully Connected Layer, meaning that every neuron in the preceding layer is connected to every neuron in the Fully Connected Layer. Like the convolution, RELU and pooling layers, there can be 1 or more fully connected layers depending on the level of feature abstraction desired. This layer takes the output from the preceding layer (Convolutional, RELU or Pooling) as its input, and computes a probability score for classification into the different available classes. In essence, this layer looks at the combination of the most strongly activated features that would indicate the image belongs to a particular class. For example, on histology glass slides, cancer cells have a high DNA to cytoplasm ratio compared to normal cells. If features of DNA were strongly detected from the preceding layer, the CNN would be more likely to predict the presence of cancer cells.

**3.3 UML Diagrams**

The underlying premise of UML is that no one diagram can capture the different elements of a System in its entirety. Hence, UML is made up of nine diagrams that can be used to model a System at different points of time in the software life cycle of a system.

A software system can be said to have two distinct characteristics: a structural, "static" part and a behavioral, "dynamic" part. In addition to these two characteristics, an additional characteristic that a software system possesses is related to implementation. Before we categorize UML diagrams into each of these three characteristics, let us take a quick look at exactly what these characteristics are.

* Use case diagram
* Class diagram
* Object diagram
* State diagram
* Activity diagram
* Sequence diagram
* Collaboration diagram
* Component diagram
* Deployment diagram

**3.3.1 Usecase Diagram**

The use case diagram is used to identify the primary elements and processes that form the System. The primary elements are termed as "actors" and the processes are called "use cases." The use case diagram shows which actors interact with each use case.



Fig. 3.2 Usecase Diagram

**3.3.2 Class Diagram**

The class diagram is used to refine the use case diagram and define a detailed design of the System. The class diagram classifies the actors defined in the use case diagram into a set of interrelated classes. The relationship or association between the classes can be either an "is-a" Or "has-a" relationship.

****

Fig. 3.3 Class Diagram

A class consists of three compartments like class name, attributes and operations and in the above class diagram we are having the classes like dataset and user

**3.3.3 Sequence Diagram**

A sequence diagram represents the interaction between different objects in the system. The Important aspect of a sequence diagram is that it is time-ordered. Different objects In the sequence diagram interact with each other by passing "messages".

user

web server

load train dataset

load test dataset

deep learning

convolutional layer

RELU layer

pooling layer

fully connected layer

sigmoid layer

Fig. 3.4 Sequence Diagram

**3.3.4 Activity Diagram**

The process flows in the system are captured in the activity diagram. Similar to a state Diagram, an activity diagram also consists of activities, actions, transitions, initial and final States, and guard conditions.

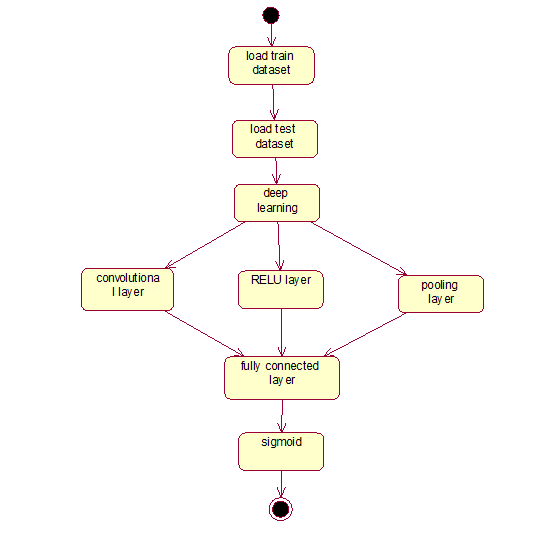


Fig. 3.5 Activity Diagram

**3.3.5 Component Diagram**

The process of this diagram shows the organizations and dependencies among a set of components. It represents the static implementation view of a system.

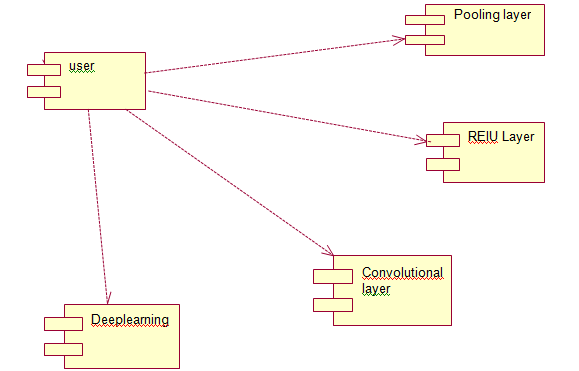


Fig. 3.6 Component Diagram

**3.3.6 Deployment Diagram**

The deployment diagram captures the configuration of the runtime elements of the Application. This diagram is by far most useful when a system is built and ready to be Deployed. The name Deployment itself describes the purpose of the diagram. Deployment diagrams are used for describing the hardware components where software components are deployed. Component diagrams and deployment diagrams are closely related.

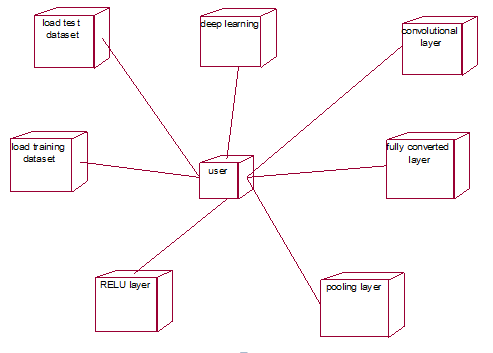


Fig. 3.7 Deployment Diagram

**3.4 DATABASE DESIGN**

The data pertaining to proposed system is voluminous that a careful design of the database must proceed before storing the data in the database. A database management system Provides flexibility in the storage and retrieval of data bad production of information. The DBMS is a bridge between the application programs, which determines what data are needed and how they are processed, and the operating system of the computer, which is Responsible for placing data on the magnetic storage devices.

**3.4.1 Normalization**

Normalization theory is built around the concept of normal forms. A relation is said to be in particular normal form if it satisfies a certain specified set of constraints.

**First Normal form**

A relation R is in first normal form if and only if all underlying domains contained atomic values only

**Second Normal form**

A relation R is said to be in second normal form if and only if it is in first normal form and every non-key attribute is fully dependent on the primary key.

**Third Normal form**

A relation R is said to be in third normal form if and only if it is in second normal form and every non key attribute is non transitively depend on the primary key.

### **Boyce and Codd Normal Form (BCNF)**

Boyce and Codd Normal Form is a higher version of the Third Normal form. This form deals with certain type of anomaly that is not handled by 3NF. A 3NF table which does not have multiple overlapping candidate keys is said to be in BCNF. For a table to be in BCNF, following conditions must be satisfied:

* R must be in 3rd Normal Form
* For each functional dependency ( X → Y ), X should be a super Key.

**Fourth Normal Form (4NF)**

Fourth Normal Form comes into picture when Multi-valued Dependency occurs in any relation.

For a table to satisfy the Fourth Normal Form, it should satisfy the following two conditions:

* It should be in the Boyce-Codd Normal Form.
* The table should not have any Multi-valued Dependency.

**4. SYSTEM IMPLEMENTATION**

**4.1 FRONT END IMPLEMENTATION**

**Python Introduction**

**Python** is a general purpose, dynamic, high level, and interpreted programming language. It supports Object Oriented programming approach to develop applications. It is simple and easy to learn and provides lots of high-level data structures.

Python is easy to learn yet powerful and versatile scripting language, which makes it attractive for Application Development. Python's syntax and dynamic typing with its interpreted nature make it an ideal language for scripting and rapid application development. It supports multiple programming pattern, including object-oriented, imperative, and functional or procedural programming styles. Python is not intended to work in a particular area, such as web programming. That is why it is known as multipurpose programming language because it can be used with web, enterprise, 3D CAD, etc. We don't need to use data types to declare variable because it is dynamically typed so we can write a=10 to assign an integer value in an integer variable. It makes the development and debugging fast because there is no compilation step included in Python development, and edit-test-debug cycle is very fast.

# Python Applications

Python is known for its general purpose nature that makes it applicable in almost each domain of software development. Python as a whole can be used in any sphere of development. Here, we are specifying applications areas where python can be applied.

* **Web Applications:** We can use Python to develop web applications. It provides libraries to handle internet protocols such as HTML and XML, JSON, Email processing, request, beautifulSoup, Feedparser etc. It also provides Frameworks such as Django, Pyramid, Flask etc to design and delelop web based applications. Some important developments are: PythonWikiEngines, Pocoo, PythonBlogSoftware etc. The useful library and package are SciPy, Pandas, IPython etc. SciPy is group of packages of engineering.
* **Desktop GUI Applications:** Python provides Tk GUI library to develop user interface in python based application. Some other useful toolkits wxWidgets, Kivy, pyqt that are useable on several platforms. The Kivy is popular for writing multitouch applications.
* **Software Development:** Python is helpful for software development process. It works as a support language and can be used for build control and management, testing etc.
* **Scientific and Numeric:** Python is popular and widely used in scientific and numeric computing. Some useful library and package are SciPy, Pandas, IPython etc. SciPy is group of packages of engineering, science and mathematics.
* **Business Application:** Python is used to build Bussiness applications like ERP and e-commerce systems. Tryton is a high level application platform.
* **Console Based Application:** We can use Python to develop console based applications. For example: IPython.
* **Audio or Video based Applications:** Python is awesome to perform multiple tasks and can be used to develop multimedia applications. Some of real applications are: TimPlayer, cplay etc.\
* **3D CAD Applications:** To create CAD application Fandango is a real application which provides full features of CAD.
* **Enterprise Applications:** Python can be used to create applications which can be used within an Enterprise or an Organization. Some real time applications are: OpenErp, Tryton, Picalo etc.
* **Applications for Images:** Using Python several application can be developed for image. Applications developed are: VPython, Gogh, imgSeek etc. Python's syntax and dynamic typing with its interpreted nature make it an ideal language for scripting and rapid application development. It supports multiple programming pattern, including object-oriented, imperative, and functional or procedural programming styles.

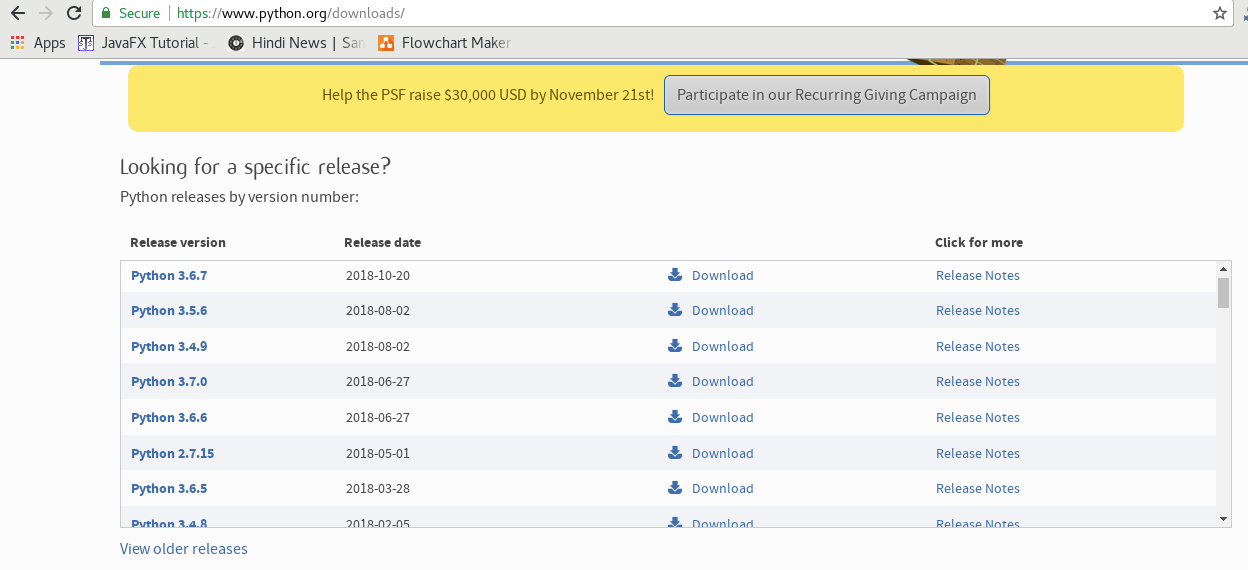
There are several such applications which can be developed using Python

# How to Install Python (Environment Set-up)

In this section of the tutorial, we will discuss the installation of python on various operating systems.

## Installation on Windows:

Visit the link <https://www.python.org/downloads/> to download the latest release of Python. In this process, we will install Python 3.6.7 on our Windows operating system.



Double-click the executable file which is downloaded; the following window will open. Select Customize installation and proceed.

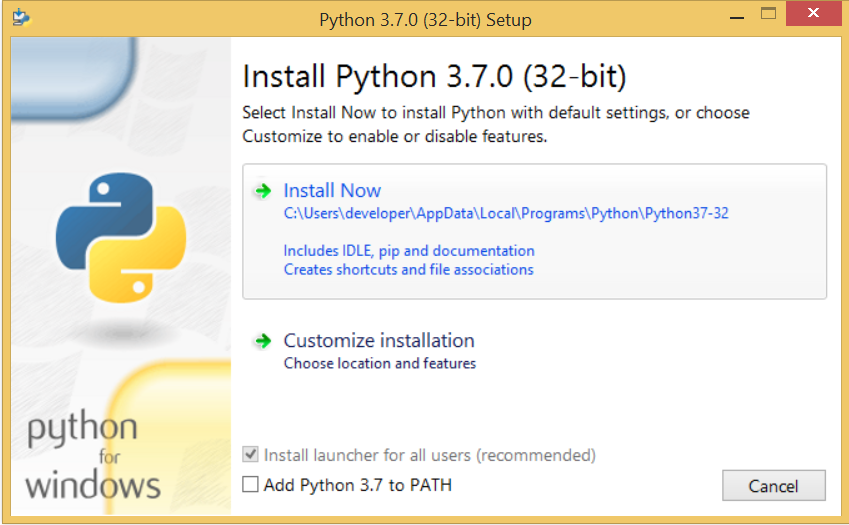


Fig. 4.1 Python Installation

The following window shows all the optional features. All the features need to be installed and are checked by default; we need to click next to continue.

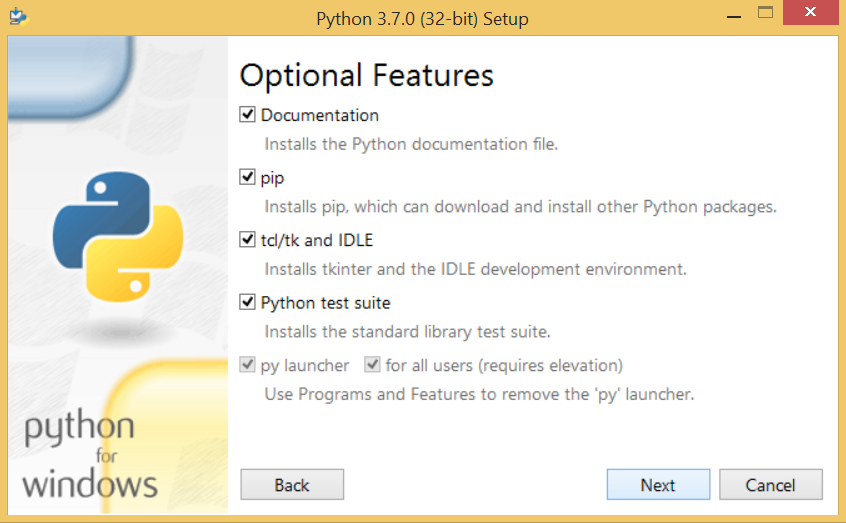
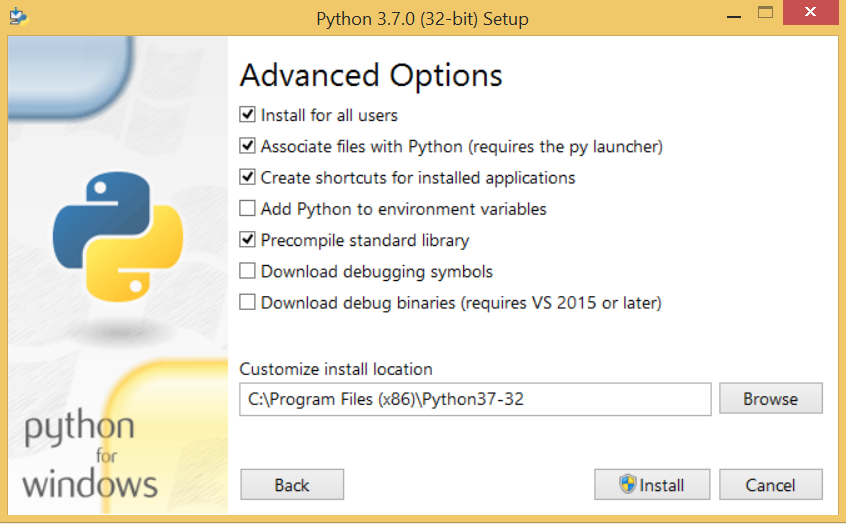


Fig. 4.2 Python Setup

The following window shows a list of advanced options. Check all the options which you want to install and click next. Here, we must notice that the first check-box (install for all users) must be checked.



Now, we are ready to install python-3.6.7. Let's install it.

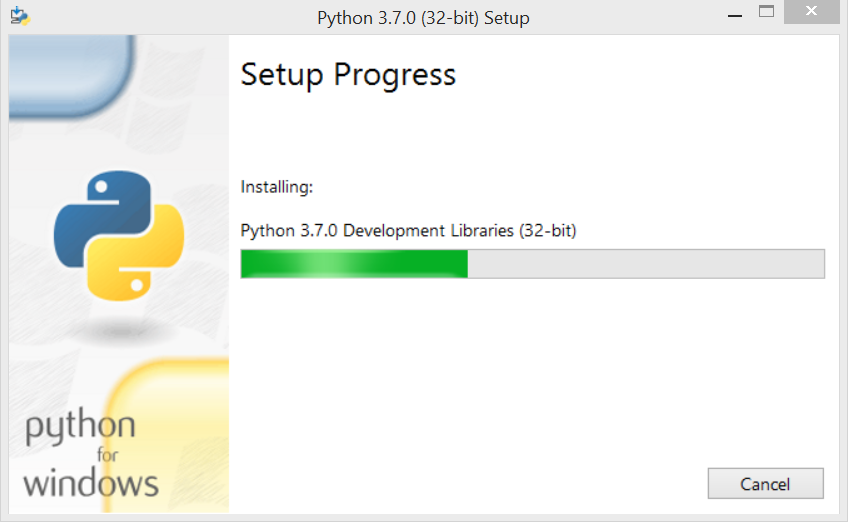
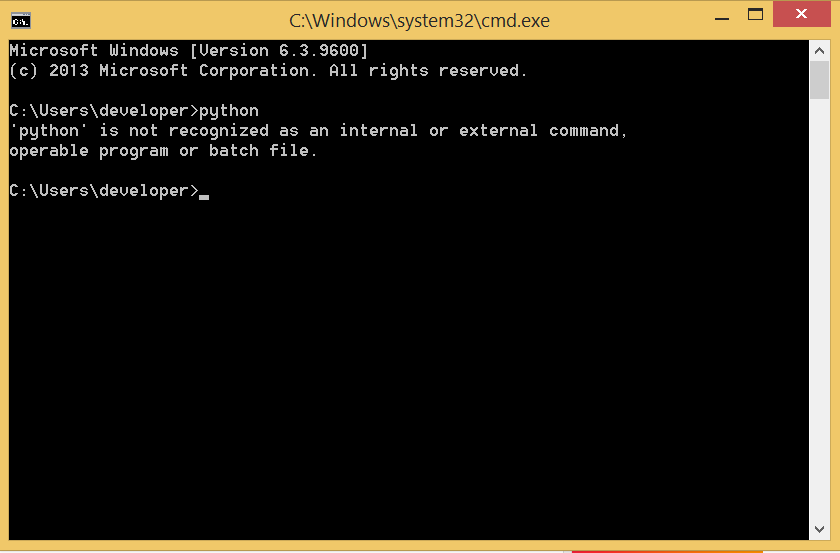
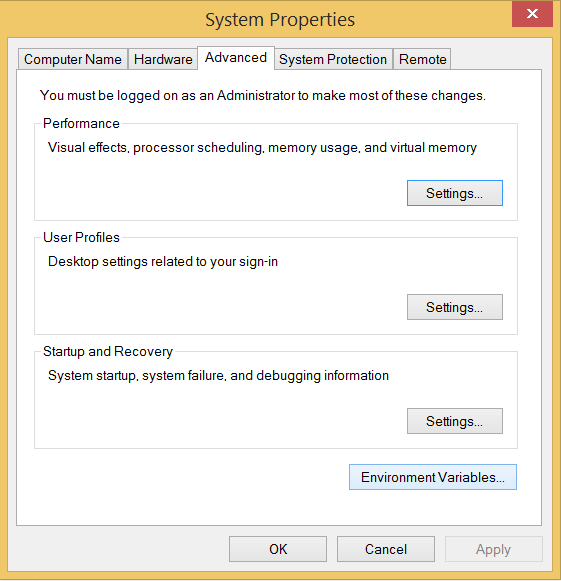


Fig. 4.3 Python Setup Progress

Now, try to run python on the command prompt. Type the command **python** in case of python2 or python3 in case of **python3**. It will show an error as given in the below image. It is because we haven't set the path.



To set the path of python, we need to the right click on "my computer" and go to Properties → Advanced → Environment Variables.

Add the new path variable in the user variable section.

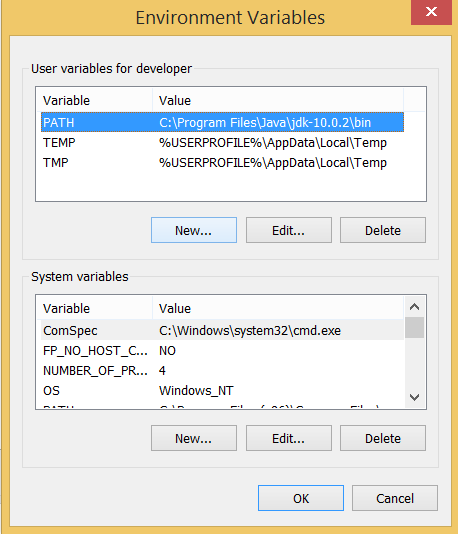
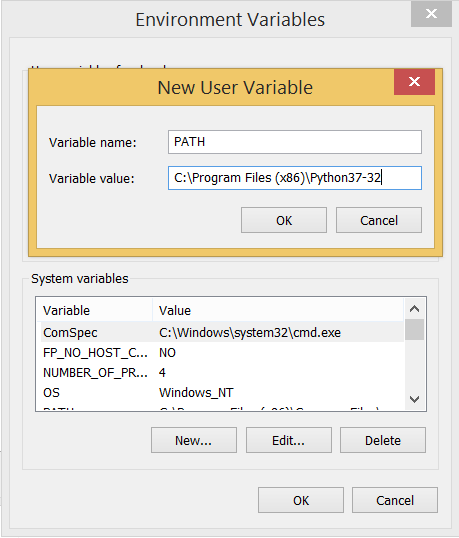


Fig. 4.4 Python Path Setup

Type **PATH** as the variable name and set the path to the installation directory of the python shown in the below image.



Now, the path is set, we are ready to run python on our local system. Restart CMD, and type **python** again. It will open the python interpreter shell where we can execute the python statements.

# First Python Program

In this Section, we will discuss the basic syntax of python by using which, we will run a simple program to print hello world on the console.

Python provides us the two ways to run a program:

* Using Interactive interpreter prompt
* Using a script file

Let's discuss each one of them in detail.

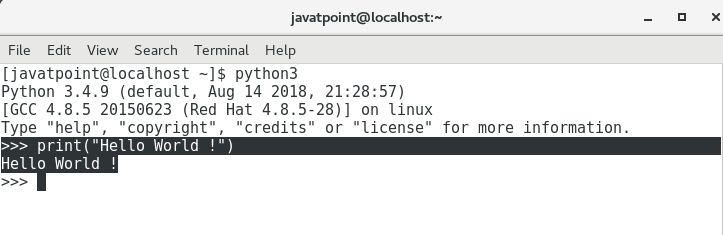
## Interactive interpreter prompt

Python provides us the feature to execute the python statement one by one at the interactive prompt. It is preferable in the case where we are concerned about the output of each line of our python program. To open the interactive mode, open the terminal (or command prompt) and type python (python3 in case if you have python2 and python3 both installed on your system).

It will open the following prompt where we can execute the python statement and check their impact on the console.



Let's run a python statement to print the traditional hello world on the console. Python3 provides print() function to print some message on the console. We can pass the message as a string into this function. Consider the following image.



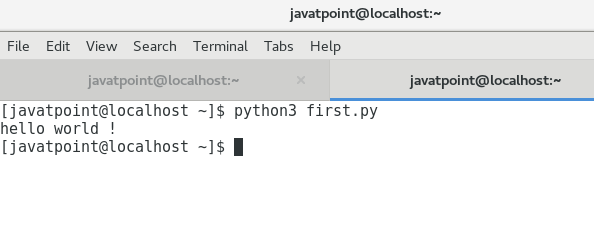
Here, we get the message **"Hello World !"** printed on the console.

## Using a script file

Interpreter prompt is good to run the individual statements of the code. However, we cannot write the code every-time on the terminal. We need to write our code into a file which can be executed later. For this purpose, open an editor like notepad, create a file named first.py (python used .py extension) and write the following code in it.

* Print ("hello world"); #here, we have used print() function to print the message on the console.  To run this file named as first.py, we need to run the following command on the terminal.

**$ python3 first.py**



Hence, we get our output as the message **Hello World !** is printed on the console.

## Get Started with PyCharm

In our first program, we have used gedit on our CentOS as an editor. On Windows, we have an alternative like notepad or notepad++ to edit the code. However, these editors are not used as IDE for python since they are unable to show the syntax related suggestions. JetBrains provides the most popular and a widely used cross-platform IDE **PyCharm** to run the python programs.

## PyCharm installation

As we have already stated, PyCharm is a cross-platform IDE, and hence it can be installed on a variety of the operating systems. In this section of the tutorial, we will cover the installation process of PyCharm on Windows, MacOS, CentOS, and Ubuntu.

### **Windows**

Installing PyCharm on Windows is very simple. To install PyCharm on Windows operating system, visit the link <https://www.jetbrains.com/pycharm/download/download-thanks.html?platform=windows> to download the executable installer. **Double click** the installer (.exe) file and install PyCharm by clicking next at each step.

**Anaconda**

Some tutorials install packages with conda instead of pip. So what is conda?



Anaconda is a distribution of Python (and R). It is free and open-source and makes package management and deployment simpler. Keep reading to see how. It is the standard platform for python data Science and open-source machine learning. Anaconda is used by data scientists, IT professionals and business leaders.

### Benefits of Using Python Anaconda

Why should you use Anaconda for your project? Well, it does have the following benefits:

* It is free and open-source.
* It has more than 1500 Python/R data science packages.
* Anaconda simplifies package management and deployment.
* It has tools to easily collect data from sources using machine learning and AI.
* It creates an environment that is easily manageable for deploying any project
* Anaconda is the industry standard for developing, testing and training on a single machineIt has good community support- you can ask your questions there.

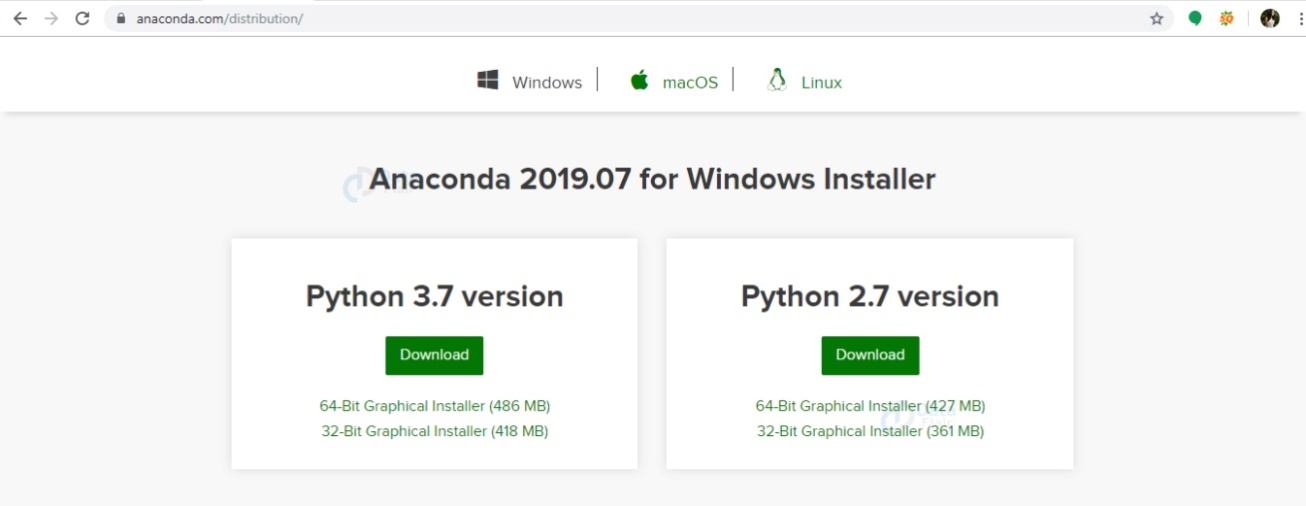
What you get

* Download more than 1500 Python/R data science packages
* Manage libraries, dependencies, and environments with conda
* Build and train ML and deep learning models with scikit-learn, TensorFlow and Theano
* Use Dask, NumPy, Pandas and Numba to analyze data scalably and fast
* Perform visualization with Matplotlib, Bokeh, Datashader, and Holoviews

### **Python Anaconda Installation**

Next in the Python anaconda tutorial is its installation. The latest version of Anaconda at the time of writing is 2019.07. Follow these steps to download and install Anaconda on your machine:

* Go to this link and download Anaconda for Windows, Mac, or Linux: – [Download anaconda](https://www.anaconda.com/distribution/)



You can download the installer for Python 3.7 or for Python 2.7 (at the time of writing). And you can download it for a 32-bit or 64-bit machine.

* Click on the downloaded .exe to open it. This is the Anaconda setup. Click next.

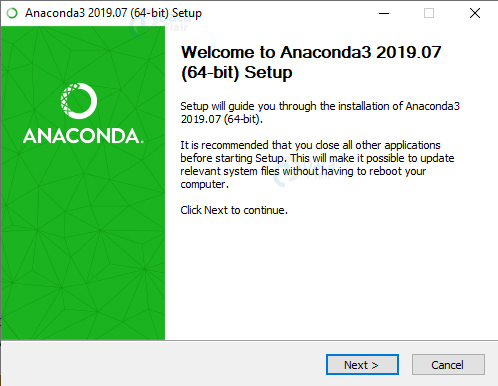
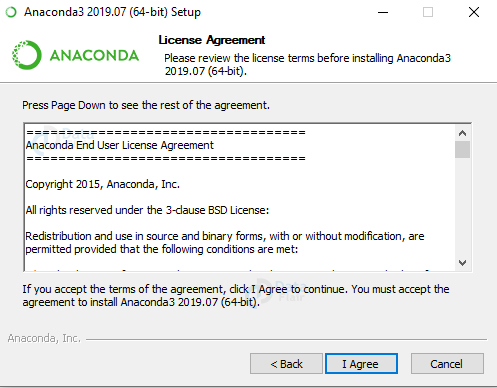
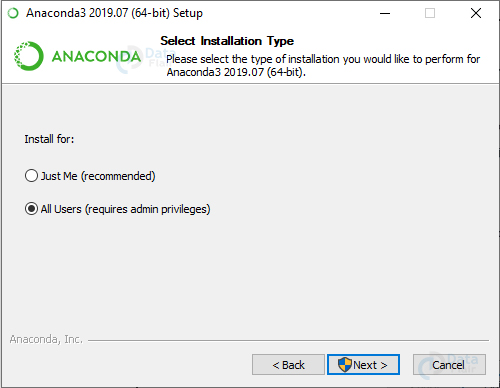


Fig: 4.5 Anaconda Setup

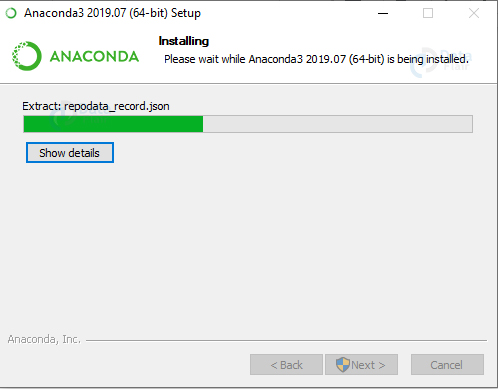
* Now, you’ll see the license agreement. Click on ‘I Agree’.



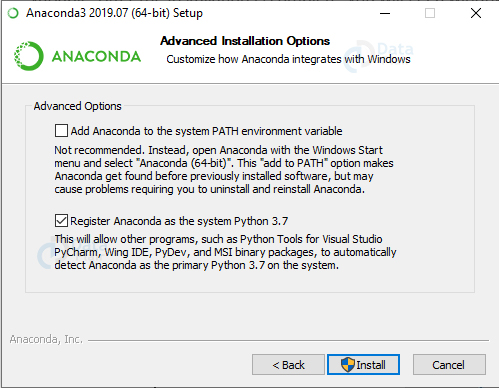
* You can install it for all users or just for yourself. If you want to install it for all users, you need administrator privileges.



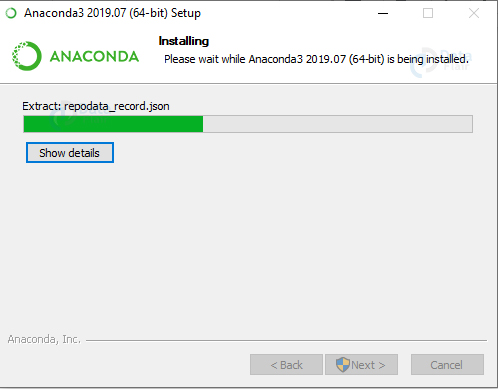
* Choose where you want to install it. Here, you can see the available space and how much you need.



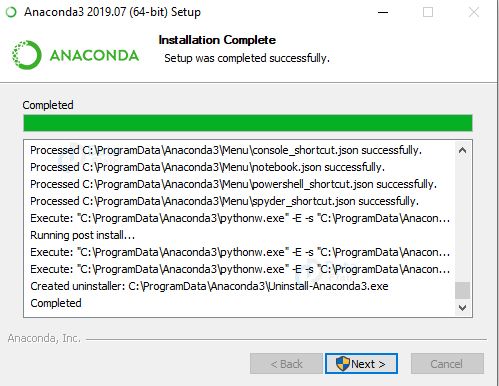
* Now, you’ll get some advanced options. You can add Anaconda to your system’s PATH environment variable, and register it as the primary system Python 3.7. If you add it to PATH, it will be found before any other installation. Click on ‘Install’.



* It will unpack some packages and extract some files on your machine. This will take a few minutes.



* The installation is complete. Click Next.



* This screen will inform you about PyCharm. Click Next.



Fig: 4.6 Pycharm IDE

* The installation is complete. You can choose to get more information about Anaconda cloud and how to get started with Anaconda.



* If you search for Anaconda now, you will see the following options:

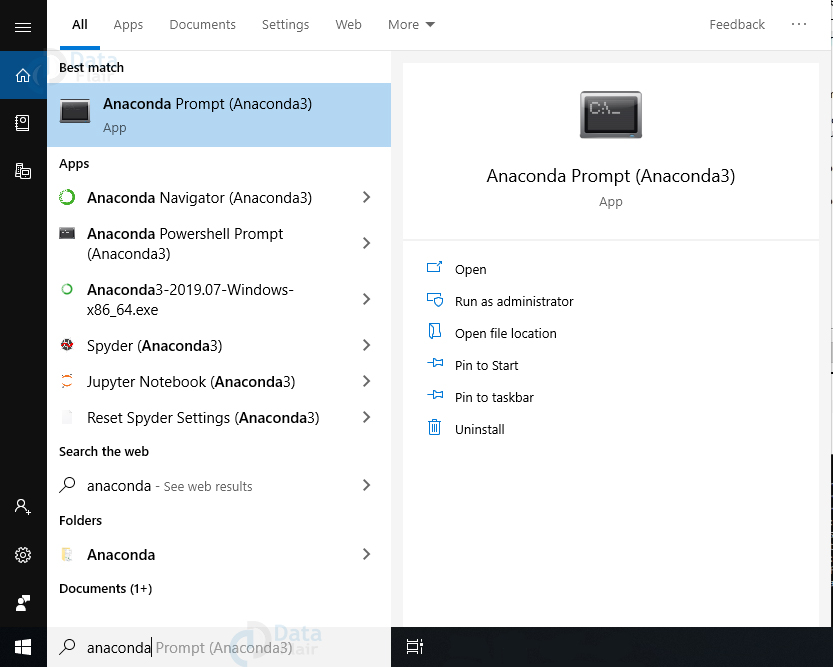
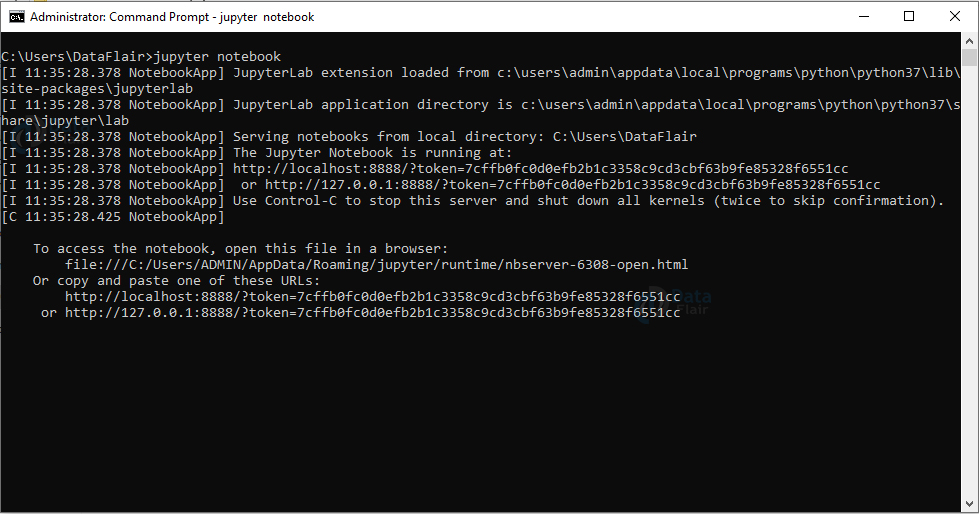


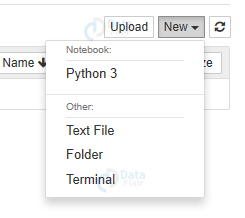
Fig: 4.7 Anaconda Prompt

* The Anaconda prompt
* Anaconda Navigator
* Anaconda Powershell prompt
* Spyder IDE and
* Jupyter Notebook

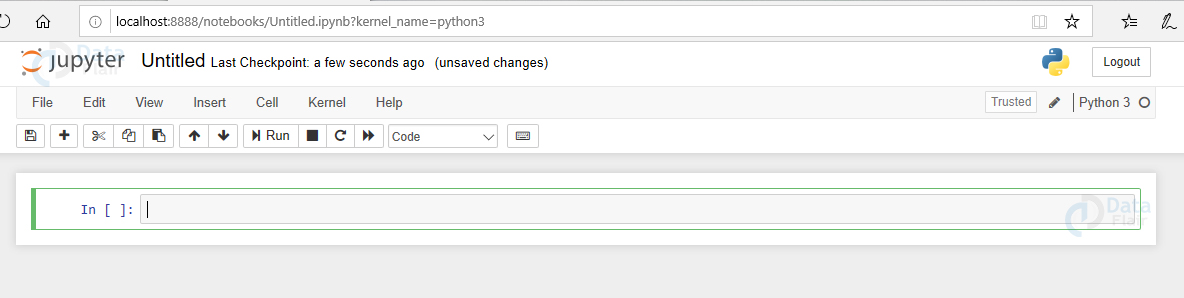
If you go to your command prompt and type ‘jupyter notebook’, it will open the Jupyter dashboard for you.



* You can create a new notebook from the New menu at the top right.

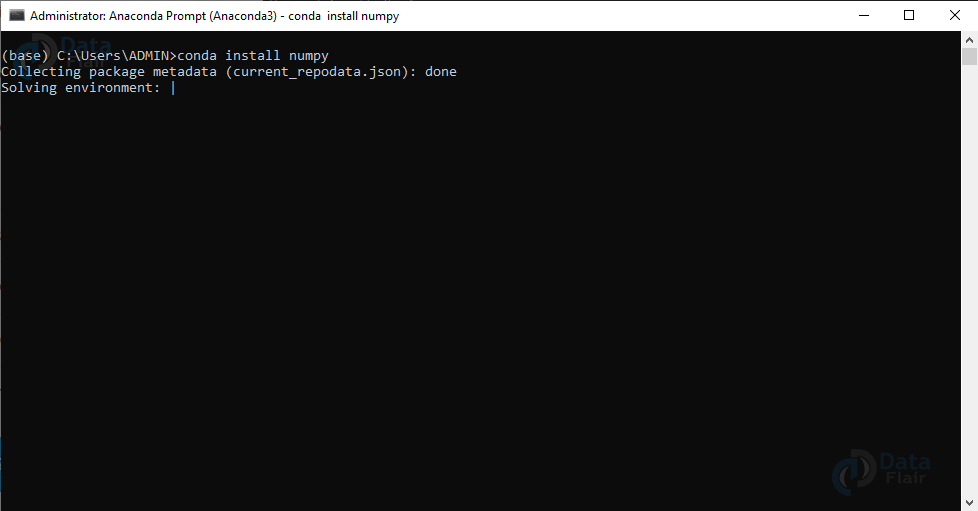


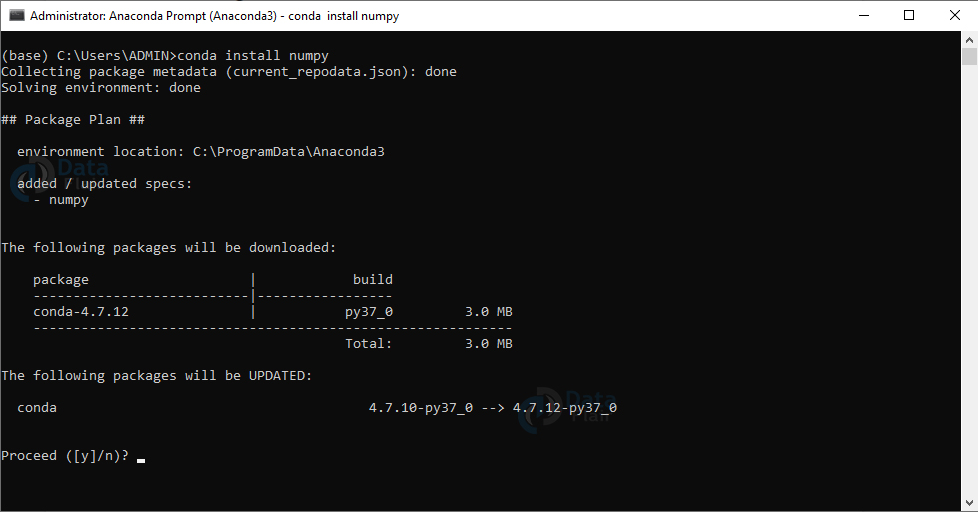
This is your new notebook. You can type in this. To execute a statement or multiple statements at once, press Shift+Enter. Pressing only Enter will only take you to the next line.

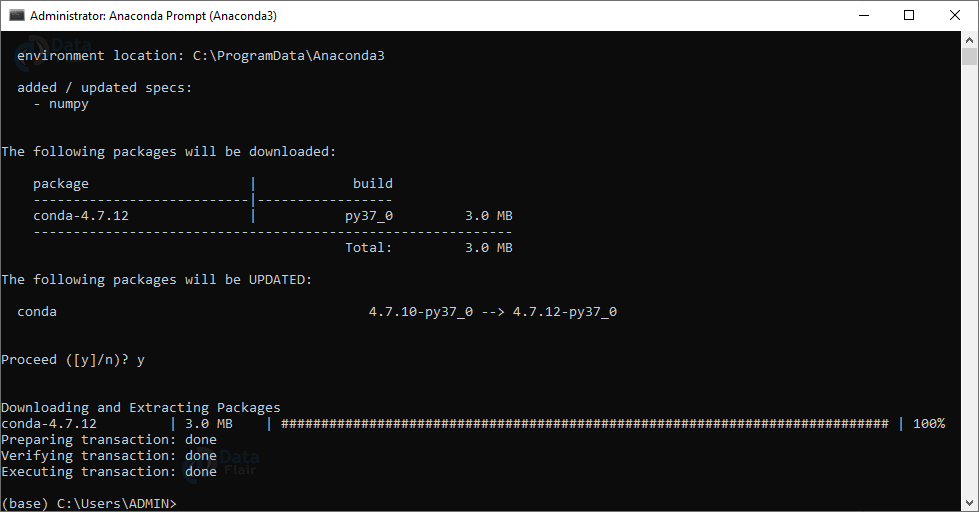


### **Installing Python Anaconda Libraries**

You have successfully installed Anaconda. Now, how will you install libraries or packages with it? Let’s see how. Open the Anaconda Prompt and use the ‘conda install command’ with the package name.

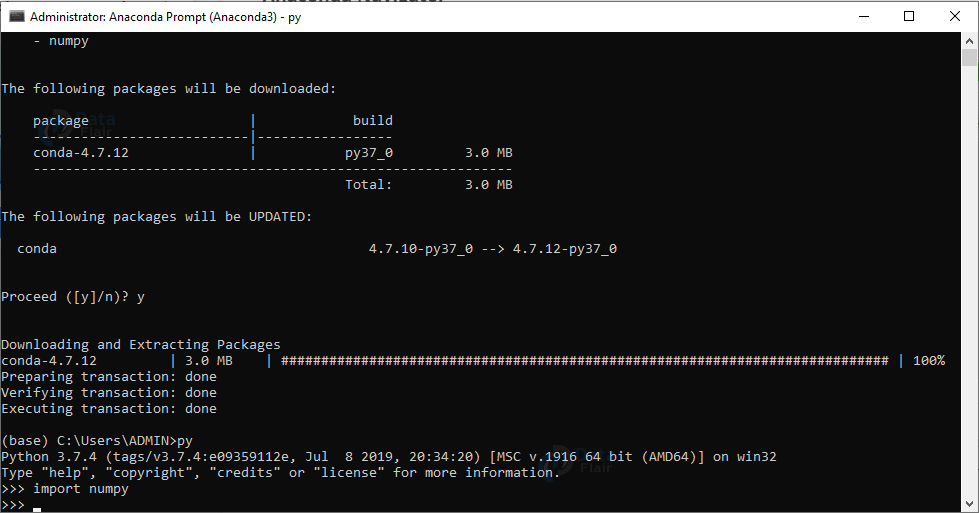






You must check [top python libraries](https://data-flair.training/blogs/python-libraries/)

Now, you can install this package.



## Anaconda Navigator

Anaconda Navigator is a desktop GUI that ships with Anaconda and lets you launch applications and manage conda packages, environments, and channels without having to use a command-line interface. It can search for packages in a local Anaconda repository or on Anaconda Cloud. With Navigator, you don’t need to type commands in a terminal, it lets you work with packages and environments with just a click.

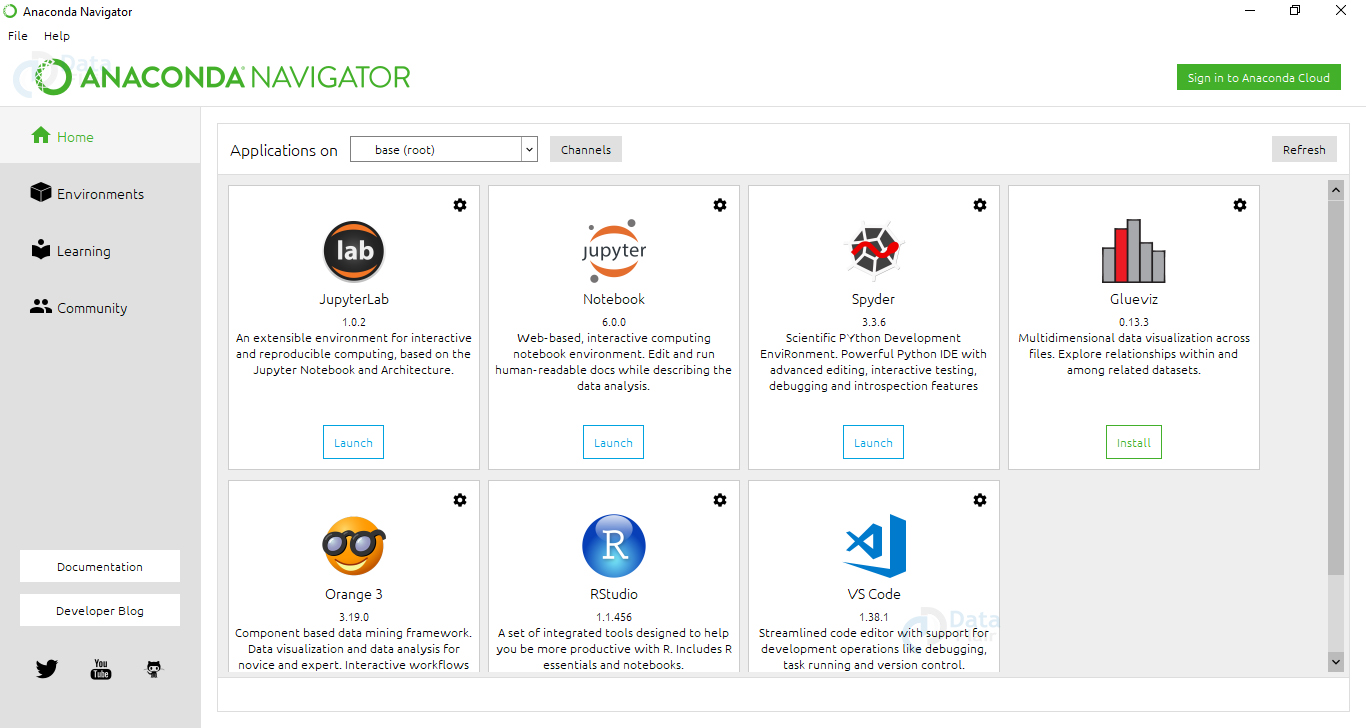


Fig: 4.8 Anaconda Navigator

With Navigator, you can access:

* Jupyter Lab
* Jupyter Notebook
* Spyder

**4.2 Back End Implementation**

Artificial Intelligence is one of the most popular trends of recent times. Machine learning and deep learning constitute artificial intelligence. The Venn diagram shown below explains the relationship of machine learning and deep learning.

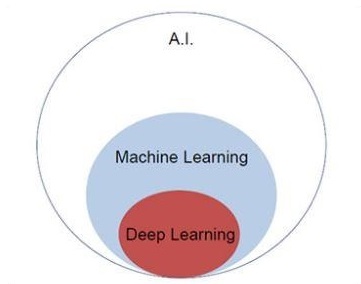


Fig: 4.9 Venn diagram

## Machine Learning

Machine learning is the art of science of getting computers to act as per the algorithms designed and programmed. Many researchers think machine learning is the best way to make progress towards human-level AI. Machine learning includes the following types of patterns.

* Supervised learning pattern
* Unsupervised learning pattern

## Deep Learning

Deep learning is a subfield of machine learning where concerned algorithms are inspired by the structure and function of the brain called artificial neural networks. All the value today of deep learning is through supervised learning or learning from labeled data and algorithms. Each algorithm in deep learning goes through the same process. It includes a hierarchy of nonlinear transformation of input that can be used to generate a statistical model as output.

Consider the following steps that define the Machine Learning process

* Identifies relevant data sets and prepares them for analysis.
* Chooses the type of algorithm to use.
* Builds an analytical model based on the algorithm used.
* Trains the model on test data sets, revising it as needed.
* Runs the model to generate test scores.

The Two Important Types of Deep Neural Networks

* Convolutional Neural Networks
* Recurrent Neural Networks

In this we will focus on the CNN, Convolutional Neural Networks.

Deep learning algorithms are constructed with connected layers.

* The first layer is called the Input Layer
* The last layer is called the Output Layer
* All layers in between are called Hidden Layers. The word deep means the network join neurons in more than two layers.
* Each Hidden layer is composed of neurons. The neurons are connected to each other. The neuron will process and then propagate the input signal it receives the layer above it. The strength of the signal given the neuron in the next layer depends on the weight, bias and activation function.
* The network consumes large amounts of input data and operates them through multiple layers; the network can learn increasingly complex features of the data at each layer.

## Deep learning Process

A deep neural network provides state-of-the-art accuracy in many tasks, from object detection to speech recognition. They can learn automatically, without predefined knowledge explicitly coded by the programmers.

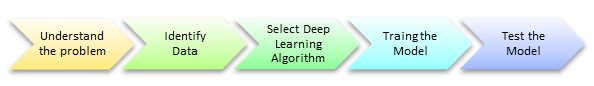


Fig: 4.10 Deep Learning Process

A neural network works quite the same. Each layer represents a deeper level of knowledge, i.e., the hierarchy of knowledge. A neural network with four layers will learn more complex feature than with that with two layers. The learning occurs in two phases.

* Create a statistical model as output.
* The second phase aims at improving the model with a mathematical method known as derivative.

The neural network repeats these two phases hundreds to thousands of time until it has reached a tolerable level of accuracy. The repeat of this two-phase is called an iteration..

## Convolutional Neural Networks

Convolutional Neural networks are designed to process data through multiple layers of arrays. This type of neural networks is used in applications like image Analysis or face recognition. The primary difference between CNN and any other ordinary neural network is that CNN takes input as a two-dimensional array and operates directly on the images rather than focusing on feature extraction which other neural networks focus on.

The dominant approach of CNN includes solutions for problems of recognition. Top companies like Google and Facebook have invested in research and development towards recognition projects to get activities done with greater speed. A convolutional neural network uses three basic ideas : Local respective fields, Convolution, Pooling.

CNN utilizes spatial correlations that exist within the input data. Each concurrent layer of a neural network connects some input neurons. This specific region is called local receptive field. Local receptive field focusses on the hidden neurons. The hidden neurons process the input data inside the mentioned field not realizing the changes outside the specific boundary.

If we observe the above representation, each connection learns a weight of the hidden neuron with an associated connection with movement from one layer to another. Here, individual neurons perform a shift from time to time. This process is called “convolution”.

The mapping of connections from the input layer to the hidden feature map is defined as “shared weights” and bias included is called “shared bias”.

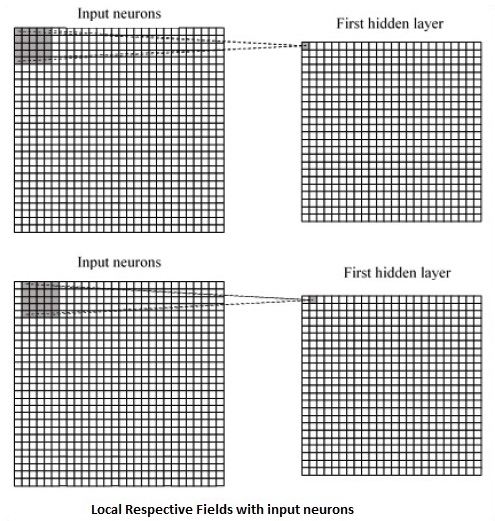


Fig: 4.11 Local Respective fields with input neurons

CNN or convolutional neural networks use pooling layers, which are the layers, positioned immediately after CNN declaration. It takes the input from the user as a feature map that comes out of convolutional networks and prepares a condensed feature map. Pooling layers helps in creating layers with neurons of previous layers.

**Tensor flow**

Tensor Flow is a software library or framework, designed by the Google team to implement machine learning and deep learning concepts in the easiest manner. It combines the computational algebra of optimization techniques for easy calculation of many mathematical expressions. Let us now consider the following important features of Tensor Flow −

* It includes a feature of that defines, optimizes and calculates mathematical expressions easily with the help of multi-dimensional arrays called tensors.
* It includes a programming support of deep neural networks and machine learning techniques.
* It includes a high scalable feature of computation with various data sets.
* Tensor Flow uses GPU computing, automating management. It also includes a unique feature of optimization of same memory and the data used.
* Tensor Flow is well-documented and includes plenty of machine learning libraries. It offers a few important functionalities and methods for the same.
* Tensor Flow is also called a “Google” product. It includes a variety of machine learning and deep learning algorithms. Tensor Flow can train and run deep neural networks for handwritten digit classification, image recognition, word embedding and creation of various sequence models.

**Tensor Flow – Keras**

One of the most powerful and easy-to-use Python libraries for developing and evaluating deep learning models is Keras; It wraps the efficient numerical computation libraries Thaana and Tensor Flow. The advantage of this is mainly that you can get started with neural networks in an easy and fun way.Keras is compact, easy to learn, high-level Python library run on top of TensorFlow framework. It is made with focus of understanding deep learning techniques, such as creating layers for neural networks maintaining the concepts of shapes and mathematical details. The creation of framework can be of the following two types

* Sequential API
* Functional API

Consider the following eight steps to create deep learning model in Keras –

* Loading the data
* Preprocess the loaded data
* Definition of model
* Compiling the model
* Fit the specified model
* Evaluate it
* Make the required predictions
* Save the model

We will use the Jupyter Notebook for execution and display of output.

**Step 1** − Loading the data and preprocessing the loaded data is implemented first to execute the deep learning model.

This step can be defined as “Import libraries and Modules” which means all the libraries and modules are imported as an initial step.

**Step 2** − In this step, we will define the model architecture

**Step 3** − Let us now compile the specified model

**Step 4** − We will now fit the model using training data

The output of iterations created.

**4.3 SOURCE CODE**

import tensorflow as tf

import keras

from keras.models import Sequential

from keras.layers import Conv2D

from keras.layers import MaxPooling2D

from keras.layers import Flatten

from keras.layers import Dense

classifier = Sequential()

# Input layer

classifier.add(Conv2D(32, (3, 3), input\_shape = (64, 64, 3), activation = 'relu'))

classifier.add(MaxPooling2D(pool\_size = (2, 2)))

# Hidden layer 1

classifier.add(Conv2D(32, (3, 3), activation = 'relu'))

classifier.add(MaxPooling2D(pool\_size = (2, 2)))

# Hidden layer 2

classifier.add(Conv2D(32, (3, 3), activation = 'relu'))

classifier.add(MaxPooling2D(pool\_size = (2, 2)))

# Hidden layer 3

classifier.add(Flatten())

# Hidden layer 4

classifier.add(Dense(activation = 'relu',units=128))

classifier.add(Dense(activation = 'sigmoid',units=1))

classifier.compile(optimizer ='adam', loss = 'binary\_crossentropy', metrics = ['accuracy'])

classifier.summary()

from keras.preprocessing.image import ImageDataGenerator

train\_datagen = ImageDataGenerator(rescale = 1./255,

                                   shear\_range = 0.2,

                                   zoom\_range = 0.2,

                                   horizontal\_flip = True

test\_datagen = ImageDataGenerator(rescale = 1./255)

import os

os.getcwd()

os.chdir(‘C:/Users/nares/Desktop/move/VBEC/proj5/DEEP LEARNING APPLICATIONS IN MEDICAL IMAGE ANALYSIS-BRAIN TUMOR/Brain\_tumor’)

print(os.getcwd())

training\_set = train\_datagen.flow\_from\_directory('C:/Users/nares/Desktop/move/VBEC/proj5/DEEP LEARNING APPLICATIONS IN MEDICAL IMAGE ANALYSIS-BRAIN TUMOR/Brain\_tumor/train/',

                                                 target\_size = (64, 64),

                                                 batch\_size = 32,

                                                 class\_mode = 'binary')

test\_set = test\_datagen.flow\_from\_directory('C:/Users/nares/Desktop/move/VBEC/proj5/DEEP LEARNING APPLICATIONS IN MEDICAL IMAGE ANALYSIS-BRAIN TUMOR/Brain\_tumor/test/',

                                            target\_size = (64, 64),

                                            batch\_size = 32,

                                            class\_mode = 'binary')

classifier.fit\_generator(training\_set, steps\_per\_epoch=None,epochs=100, verbose=1, callbacks=None, validation\_data=test\_set, validation\_steps=None, class\_weight=None, max\_queue\_size=10,workers=1, use\_multiprocessing=False, shuffle=True, initial\_epoch=0)

import numpy as np

from keras.preprocessing import image  test\_image = image.load\_img('C:/Users/nares/Desktop/move/VBEC/proj5/DEEP LEARNING APPLICATIONS IN MEDICAL IMAGE ANALYSIS-BRAIN TUMOR/ TestImages/ images.jpg', target\_size = (64, 64))

test\_image

test\_image = image.img\_to\_array(test\_image)

test\_image = np.expand\_dims(test\_image, axis = 0)

test\_image

result = classifier.predict(test\_image)

Result

training\_set.class\_indices

if result[0][0] == 0:

    prediction = 'Benign'

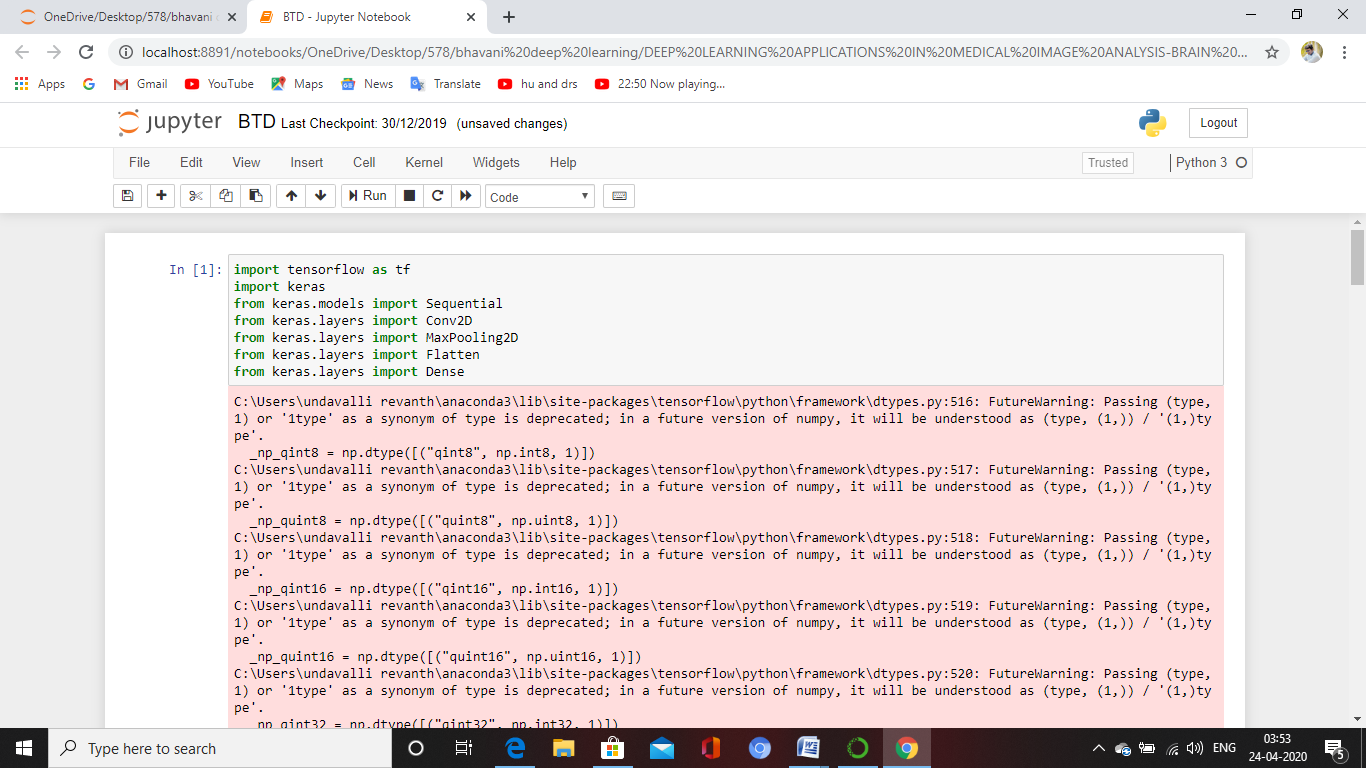
else:

    prediction = 'Malignent'

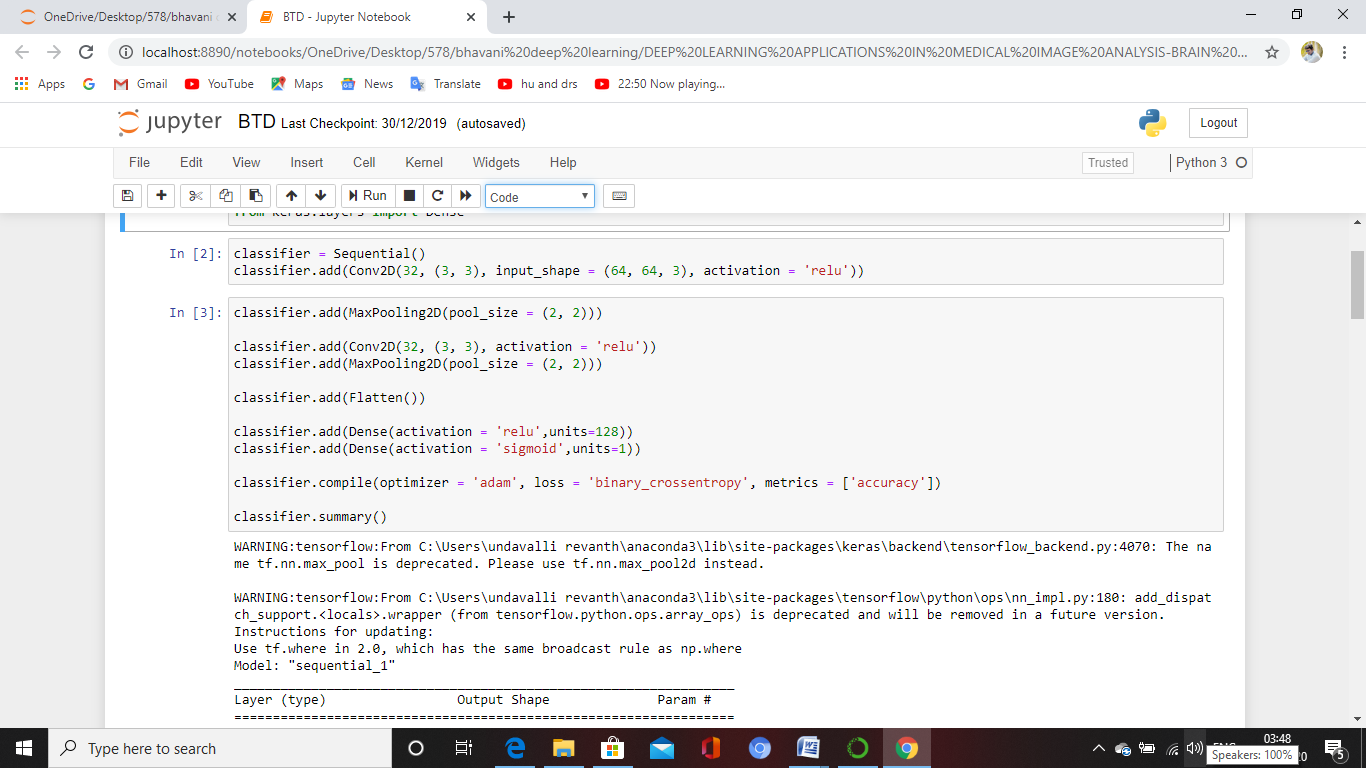
print("Detected tumor type is %s"%prediction)

**4.4 OUTPUT SCREENS**

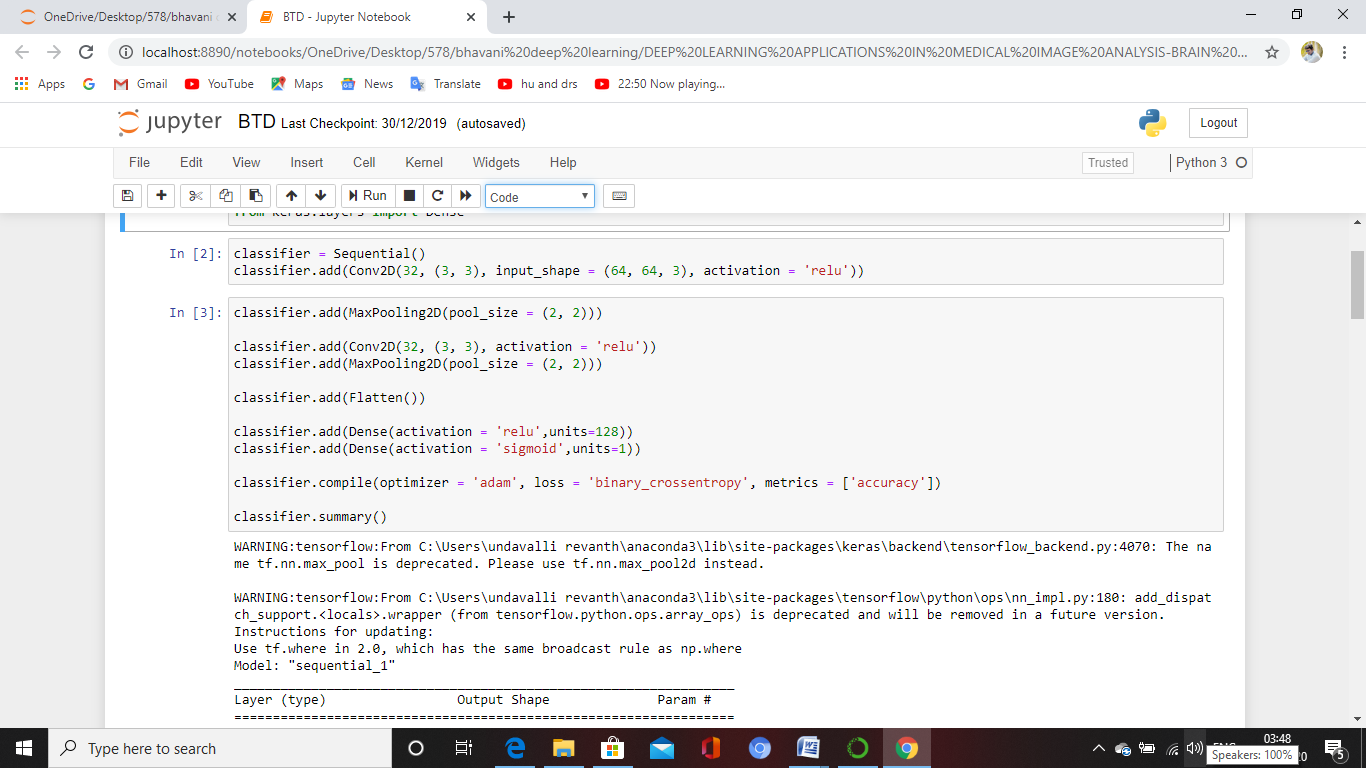
**Import Libraries and Tensorflow**

****

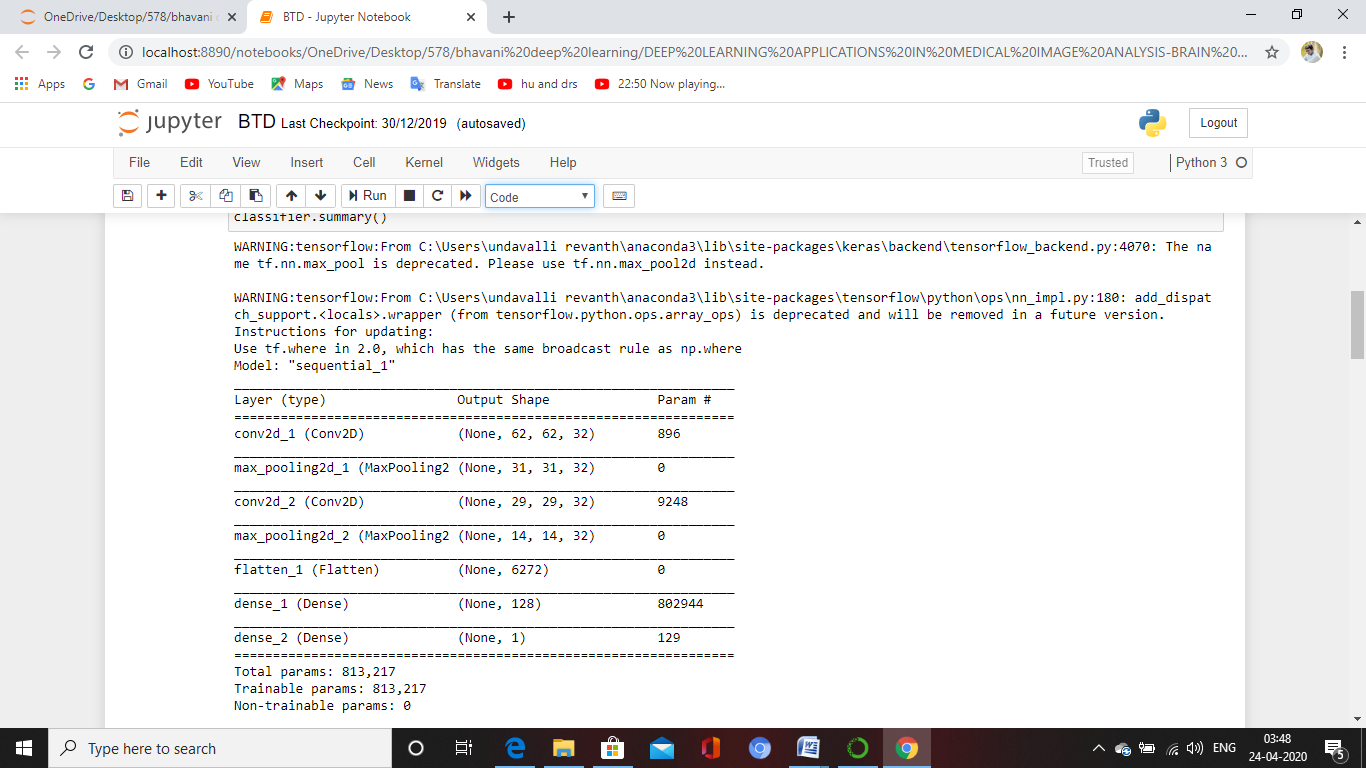
**Import Sequential Model**

****

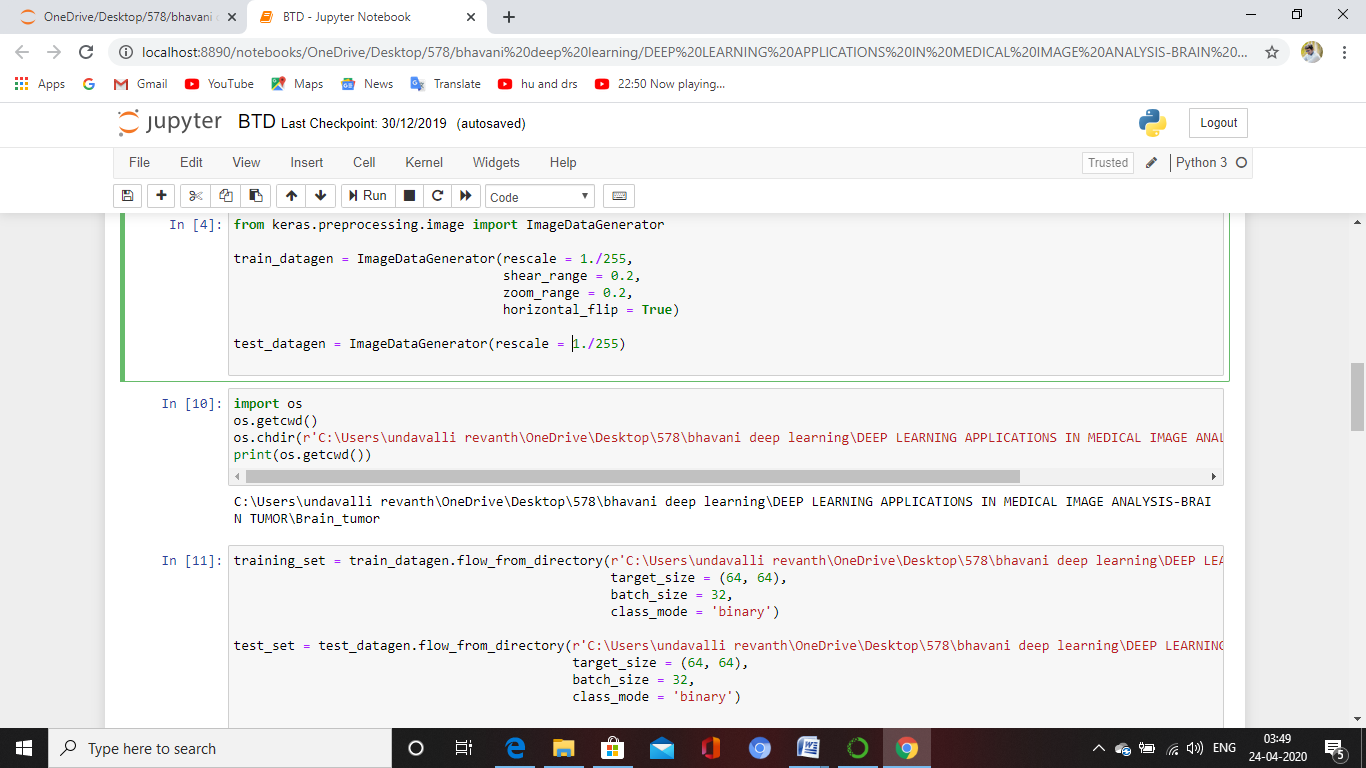
**Import Input values to Hidden Layer**

****

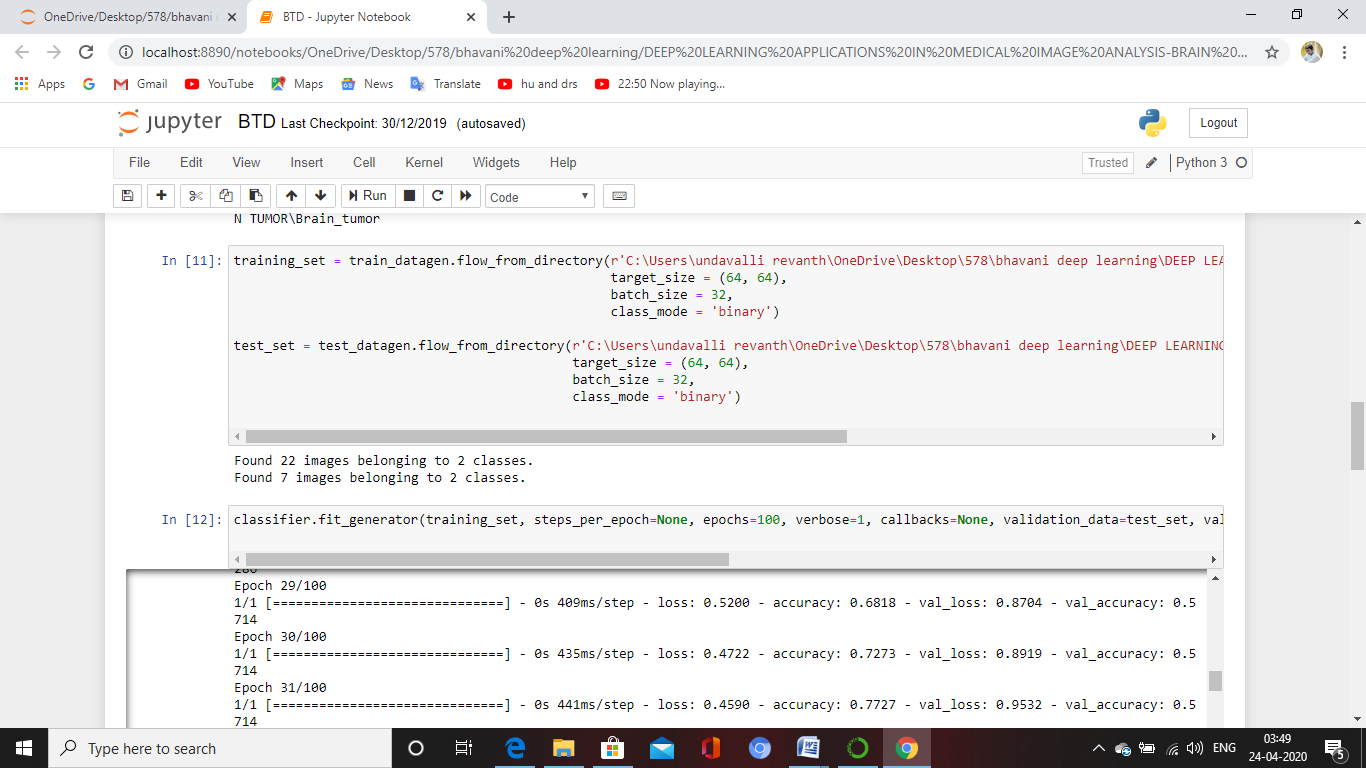
**Preprocessing image data to train and test data**

****

**Import train and test data from directory**

****

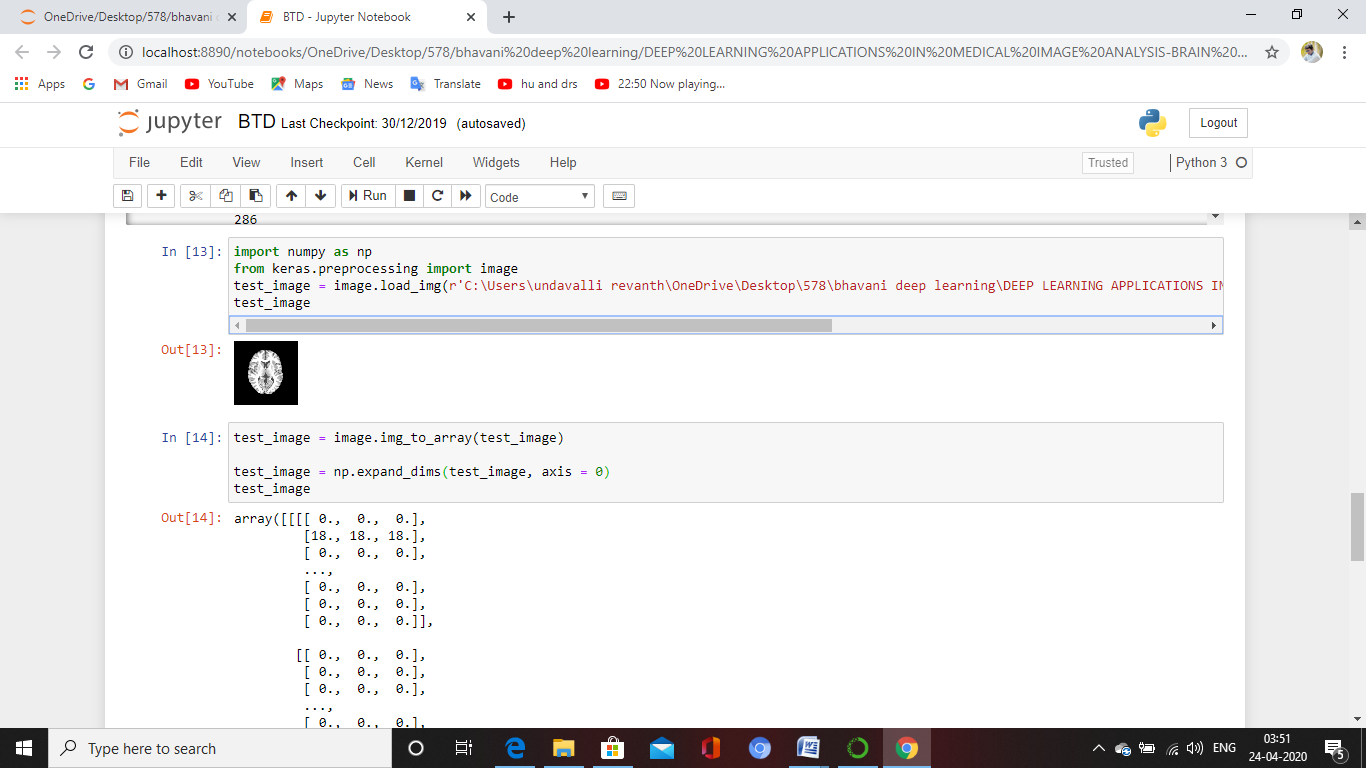
**Import Training set**

****

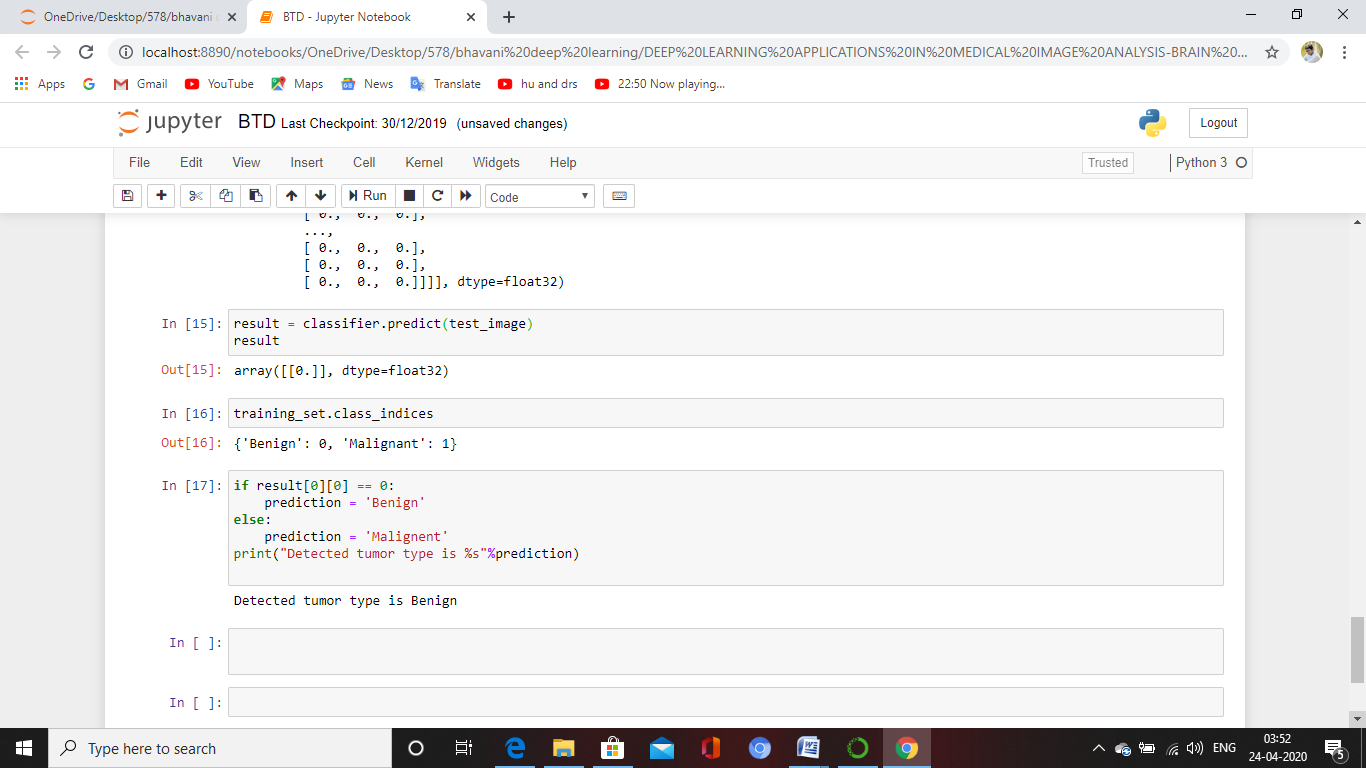
**Classifier**

****

**Import image and preprocessing test dataset from directory**

****

**Predict the result from test data**

****

**5. SYSTEM TESTING**

**5.1 TESTING CONCEPTS**

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

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

**5.2 TESTING STRATEGIES**

**Unit testing**

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

**Integration testing**

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

**Functional test**

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

Valid Input : identified classes of valid input must be accepted.

Invalid Input : identified classes of invalid input must be rejected.

Functions : identified functions must be exercised.

Output : identified classes of application outputs must be exercised.

Systems/Procedures : interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

**System Test**

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

**White Box Testing**

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

**Black Box Testing**

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box .you cannot “see” into it. The test provides inputs and responds to outputs without considering how the software works.

**5.3 TEST CASES**

**Negative Test Cases:**

|  |  |
| --- | --- |
| **Test Case 1**: User Registration | Priority(H.L):High |
| **Test Objective:** to check the user registration wheatear register success or fail | |
| **23454 Test Description:** In this Registration screen, when the user enters his/her details in the registration form and when he clicks the submit button then the database should respond if the user enters any invalid data and responds like please enter valid information, and if the data is correct the admin should store the information in the database and redirect him to the login page, or display an error. | |
| **Requirement Verified:** no | |
| **Test Environment**: System connected with the database. | |
| Actions | Expected Results |
| * when the user enters his/her details in the registration form and when he clicks the submit button, and the details are valid then it shows a message * When he enters a wrong password and then clicks submit button * When he enters a wrong e-mail and phone number and then clicks submit button. | * Registration fail * Enter valid data * Invalid user * Enter Valid email * Enter valid Phone number |
| **Pass**: no **Condition Pass**: No **Fail:** yes | |
| **Problems/Issues**: yes | |
| **Notes:** Registration is fail | |

|  |  |
| --- | --- |
| **Test Case 2**: User Login | Priority (H.L): High |
| **Test Objective:** has to check user login successful or fail | |
| **22353 Test Description:** In this login page user enter his user name and password. After enter the user name password it has to compare with database values and validate the values. Once its match returns fail or success | |
| **Requirement Verified:** no | |
| **Test Environment**: System connected with the database. | |
| Actions | Expected Results |
| * Enter the user name and password * After the enter username and password verify with database | * Login fail * Enter valid data * Invalid user * Enter Valid email * Enter valid Phone number |
| **Pass**: no **Condition Pass**: No **Fail:** yes | |
| **Problems/Issues**: yes | |
| **Notes:** Login fail | |

**Positive Test Cases:**

|  |  |  |
| --- | --- | --- |
| **Test Case 2**: User Login | Priority (H.L): High | |
| **Test Objective:** has to check user login successful or fail | | |
| **22354 Test Description:** In this login page user enter his user name and password. | | |
| **Requirement Verified:** no | | |
| **Test Environment**: System connected with the database. | | |
| Actions | Expected Results | |
| * Enter the user name and password * After the enter username and password verify with database | * Login fail * Invalid user * Enter Valid email * Enter valid Phone number | |
| **Pass**: yes **Condition Pass**: No **Fail:** yes | | |
| **Problems/Issues**: no | | |
| **Notes:** Login Successful | | |
| **Test Case 1**: User Registration | | Priority (H.L): High | |
| **Test Objective:** to check the user registration wheatear register success or fail | | | |
| **23456 Test Description:** In this Registration screen, when the user enters his/her details in the registration form and when he clicks the submit button then the database should respond if the user enters any invalid data and responds like please enter valid information and if the data is correct the admin should store the information in the database and redirect him to the login page, or display an error. | | | |
| **Requirement Verified:** yes | | | |
| **Test Environment**: System connected with the database. | | | |
| Actions | | Expected Results | |
| * when the user enters his/her details in the registration form and when he clicks the submit button, and the details are valid then it shows a message * When he enters a wrong password and then clicks submit button * When he enters a wrong e-mail and phone number and then clicks submit button. | | * Registration successful * Enter valid data * Invalid user * Enter Valid email * Enter valid Phone number | |
| **Pass**: yes **Condition Pass**: No **Fail:** No | | | |
| **Problems/Issues**: no | | | |
| **Notes:** Registration successfully completed | | | |

**6. CONCLUSION AND FUTURE WORK**

During the recent few years, deep learning has gained a central position toward the automation of our daily life and delivered considerable improvements as compared to traditional machine learning algorithms. Based on the tremendous performance, most researchers believe that within next 15 years, deep learning based applications will take over human and most of the daily activities with be performed by autonomous machine. However, Many big research organization are working on deep learning based solution that encourage to use deep learning to apply deep learning on medical images. Looking to the brighter side of machine learning, we are hoping the sooner human will be replaced in most of the medical application especially diagnosis. However, we should not consider it as only solution as there are several challenges that reduce its growth. One of the big barriers is unavailability of annotated dataset. Thus, this question is still answerable, that whether we will be able to get enough training data without eﬀecting the performance of deep learning algorithms. Recent development on other application showed that bigger the data, better the result, however, how big data could be used in healthcare.

So far deep learning based application provided positive feedback, however, but due to the sensitivity of healthcare data and challenges, we should look more sophisticated deep learning methods that can deal complex healthcare data eﬃciently. Lastly we conclude that there are unlimited opportunities to improve healthcare system

### **FUTURE WORK**

Deep learning is gaining more importance than machine learning. Deep learning is proving to be one of the best techniques in state-of-art performance. With the increasing trend of using data science and machine learning in the industry, it will become important for each organization to inculcate machine learning in their businesses. Machine learning and deep learning will prove beneficial in research and academics field.

**7. REFERENCES**

[1] C.-J. Hsiao, E. Hing, and J. Ashman, ‘‘Trends in electronic health record system use among office-based physicians: United states, 2007–2012,’’ Nat. Health Stat. Rep., vol. 75, pp. 1–18, May 2014.

[2] R. Smith-Bindman et al., ‘‘Use of diagnostic imaging studies and associated radiation exposure for patients enrolled in large integrated health care systems, 1996–2010,’’ JAMA, vol. 307, no. 22, pp. 2400–2409, 2012.

[3] E. H. Shortliffe, Computer-Based Medical Consultations: MYCIN, vol. 2. New York, NY, USA: Elsevier, 1976.

[4] G. Litjens et al. (Jun. 2017). ‘‘A survey on deep learning in medical image analysis.’’ [Online]. Available: <https://arxiv.org/abs/1702.05747>.

[5] W. S. McCulloch and W. Pitts, ‘‘A logical calculus of the ideas immanent in nervous activity,’’ Bull. Math. Biol., vol. 5, nos. 4, pp. 115–133, 1943.

[6] F. Rosenblatt, ‘‘The perceptron: A probabilistic model for information storage and organization in the brain,’’ Psychol. Rev., vol. 65, no. 6, pp. 365–386, 1958.

[7] K. Suzuki, ‘‘Overview of deep learning in medical imaging,’’ Radiol. Phys. Technol., vol. 10, no. 3, pp. 257–273, 2017.

[8] H. Greenspan, B. van Ginneken, and R. M. Summers, ‘‘Guest editorial deep learning in medical imaging: Overview and future promise of an exciting new technique,’’ IEEE Trans. Med. Imag., vol. 35, no. 5, pp. 1153–1159, May 2016.

[9] D. Adams. (2017). Publish or Perish 5. Accessed: Oct. 30, 2017. [Online]. Available: https://harzing.com/resources/publish-or-perish/windows

[10] K. Clark et al., ‘‘The cancer imaging archive (TCIA): Maintaining and operating a public information repository,’’ J. Digit. Imag., vol. 26, no. 6, pp. 1045–1057, 2013.

[11] X. Wang, Y. Peng, L. Lu, Z. Lu, M. Bagheri, and R. M. Summers. (Dec. 2017). ‘‘ChestX-ray8: Hospital-scale chest X-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases.’’ [Online]. Available: https://arxiv.org/abs/1705. 02315.

[12] E. Gibson et al. (Oct. 2017). ‘‘NiftyNet: A deep-learning platform for medical imaging.’’ [Online]. Available: https://arxiv.org/abs/1709.03485.

[13] O. Ronneberger, P. Fischer, and T. Brox, ‘‘U-net: Convolutional networks for biomedical image segmentation,’’ in Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent., 2015, pp. 234–241.

[14] F. Milletari, N. Navab, and S.-A. Ahmadi, ‘‘V-net: Fully convolutional neural networks for volumetric medical image segmentation,’’ in Proc. 4th Int. Conf. 3D Vis. (3DV), Oct. 2016, pp. 565–571.

[15] K. Kamnitsas et al., ‘‘Efficient multi-scale 3D CNN with fully connected CRF for accurate brain lesion segmentation,’’ Med. Image Anal., vol. 36, pp. 61–78, Feb. 2017.

[16] I. Goodfellow, Y. Bengio, and A. Courville, Deep Learning. Cambridge, MA, USA: MIT Press, 2016.

[17] O. Russakovsky et al., ‘‘ImageNet large scale visual recognition challenge,’’ Int. J. Comput. Vis., vol. 115, no. 3, pp. 211–252, Dec. 2015, doi: 10.1007/s11263-015-0816-y.

[18] H.-C. Shin et al., ‘‘Deep convolutional neural networks for computeraided detection: CNN architectures, dataset characteristics and transfer learning,’’ IEEE Trans. Med. Imag., vol. 35, no. 5, pp. 1285–1298, May 2016.