Data Science & Business Analytics

An Industrial Training Report submitted to

COMPUTER SCIENCE AND ENGINEERING

in partial fulfilment of the requirements

for the award of the Degree of

BACHELORS OF TECHNOLOGY

Submitted by

Kartikeya (18134503008)

Under the guidance of

Mr. Vijay Bijlwan (Faculty Guide)
&
Mr. Pranav Dubey(Industrial Guide)

HEMVATI NANDAN BAHUGUNA GARHWAL UNIVERSITY

(A CENTRAL UNIVERSITY)

SCHOOL OF ENGINEERING AND TECHNOLOGY

Hemvati Nandan Bahuguna Garhwal University

Srinagar, Uttarakhand-246174, India

December,2021.

SCHOOL OF ENGINEERING AND TECHNOLOGY
HEMVATI NANDAN BAHUGUNA GARHWAL
UNIVERSITY SRINAGAR, UTTARAKHAND-
246174, INDIA

Certificate

This is to certify that the Industrial Training and Report entitled "**Data Science & Business Analytics**",in partial fulfilment of the requirements for the award of the Degree of **B.Tech** is a record of original training undergone by **Kartikeya (18134503008)** during the year 2022 of his study in the **Department of Computer Science and Engineering, School of Engineering and Technology** under my supervision and the report has not formed the basis for the award of any Degree/Fellowship or other similar title to any candidate of any University.

Mr. Vijay Bijlwan

Dept. of Computer Science & Engg.

HNB Garhwal Central University.

Company Certificate

DECLARATION

I, Kartikeya , hereby declare that the Industrial Training Report, entitled " **Data Science & Business Analytics**", is a training completed by me at **Online** for a durationof 4 weeks is submitted to the School of Engineering and Technology in partial fulfilment of the requirements for the award of the Degree of B.Tech is a record of original training undergone by me during the period **Janurary-Februrary, 2022** under the supervision of Mr. Vijay Bijlwan , Department of Computer Science and Engineering, School of Engineering and Technology

Kartikeya

Place: Srinagar, Uttarakhand.

Date: 30-December-2021.

ACKNOWLEDGEMENT

The training opportunity I had with The Sparks Foundation was a great chance for learning and professional development. Therefore, I consider myself as a very lucky individual to be a part of it.

I wish to express my profound gratitude to my Industrial supervisor, **Mr. Pranav Dubey** for his advices and patiently guiding me through while I was a trainee. I very much appreciate for their entire kindness for helping and teaching me.

I wish to extend my sincere gratitude to **Mr. Vijay Bijlwan**, faculty of CSE department, for their guidance, encouragement and valuable suggestion which proved extremely useful and helpful in completion of this industrial training.

I find no words to acknowledgement the sacrifice, help and inspiration rendered by my parents to take up this study.

With thanks to all,

Kartikeya

B. Tech (7^{th} Sem)

(18134503008)

CONTENTS

LIST OF TABLES

LIST OF FIGURES / CHART

LIST OF ABBREVIATIONS

ABBRIEVIATED FORM	EXPANDED FORM
AI	Artificial Intelligence
ML	Machine Learning
KNN	K-Nearest Neighbor
NLP	Natural Programming Language
SVM	Support Vector Machine
PCA	Principal Component Analysis

Chapter – 1

Introduction

Data science is an interdisciplinary field that uses scientific methods, processes, algorithms and systems to extract knowledge and insights from noisy, structured and unstructured data, and apply knowledge and actionable insights from data across a broad range of application domains. Data science is related to data mining, machine learning and big data.

Data science is a "concept to unify statistics, data analysis, informatics, and their related methods" in order to "understand and analyze actual phenomena" with data. It uses techniques and theories drawn from many fields within the context of mathematics, statistics, computer science, information science, and domain knowledge. However, data science is different from computer science and information science. Turing Award winner Jim Gray imagined data science as a "fourth paradigm" of science (empirical, theoretical, computational, and now data-driven) and asserted that "everything about science is changing because of the impact of information technology" and the data deluge.

History of Data Science

In 1962, John Tukey described a field he called "data analysis", which resembles modern data science.In 1985, in a lecture given to the Chinese Academy of Sciences in Beijing, C.F. Jeff Wu used the term Data Science for the first time as an alternative name for statistics. Later, attendees at a 1992 statistics symposium at the University of Montpellier II acknowledged the emergence of a new discipline focused on data of various origins and forms, combining established concepts and principles of statistics and data analysis with computing.

The term "data science" has been traced back to 1974, when Peter Naur proposed it as an alternative name for computer science.In 1996, the International Federation of Classification Societies became the first conference to specifically feature data science as a topic. However, the definition was still in flux. After the 1985 lecture in the Chinese Academy of Sciences in Beijing, in 1997 C.F. Jeff Wu again suggested that statistics should be renamed data science. He reasoned that a new name would help statistics shed inaccurate stereotypes, such as being synonymous with accounting, or limited to describing data. In 1998, Hayashi Chikio argued for data science as a new, interdisciplinary concept, with three aspects: data design, collection, and analysis.

During the 1990s, popular terms for the process of finding patterns in datasets (which were increasingly

large) included "knowledge discovery" and "data mining".

Importance:

There is a veil of mystery surrounding Data Science. While the buzzword of Data Science has been circulating for a while, very few people know about the real purpose of being a Data Scientist.

We will go through the various responsibilities that a Data Scientist must fulfill and understand as to what industries seek from employing Data Scientists. After this, we will look at various types of industries which employ Data Scientists to make better decisions.

Big data is becoming a tool for businesses and companies of all sizes.The availability and interpretation of big data has altered the business models of old industries and enabled the creation of new ones. Data scientists are responsible for breaking down big data into usable information and creating software and algorithms that help companies and organizations determine optimal operations.

Objectives:

The objective of the data scientist is to explore, sort and analyze megadata from various sources in order to take advantage of them and reach conclusions to optimize business processes or for decision support.

Main objectives of training were to learn:

- How to determine and measure program complexity,
- Python Programming
- Jupter Notebook, PyCharm
- ML Library Scikit, Matplotlib, Pandas, Numpy, Theano, TensorFlow
- Statistical Math for the Algorithms.
- Learning to solve statistics and mathematical concepts.
- Supervised and Unsupervised Learning
- Classification and Regression
- Data Science Algorithms

Future Scope:

The scope of Data Science is growing with every passing year. From 2008 to 2020, people across the globe have stepped on the digitalization age. The massive growth of data provides a glimpse of the future scope of Data Science in India.

1.Health care sector

There is a huge requirement of data scientists in the healthcare sector because they create a lot of data on a daily basis. Tackling a massive amount of data is not possible by any unprofessional candidate.

Hospitals need to keep a record of patients' medical history, bills, staff personal history, and much other information. Data scientists are getting hired in the medical sector to enhance the quality and safety of the data.

2.Transport Sector

The transport sector requires a data scientist to analyze the data collected through passenger counting systems, asset management, location system, fare collecting, and ticketing.

3.E-commerce

The e-commerce industry is booming just because of data scientists who analyze the data and create customized recommendation lists for providing great results to end-users

The Training in "Data Science & Bussiness Analytics " Started in 1st week of Jan, 2022 till Feb,2022. It was an online platform training where pre-recorded videos were available as the resource. The trainees day-to-day act was to take help from videos and for any doubts the teachers were present where trainees had to mail their doubts to get sol. within an hrs.

Methodologies:

There were several facilitation techniques used by the trainer which included question and answer, brainstorming, group discussions, case study discussions and practical implementation of some of the topics by trainees on flip charts and paper sheets. The multitude of training methodologies was utilized in order to make sure all the participants get the whole concepts and they practice what they learn, because only listening to the trainers can be forgotten, but what the trainees do by themselves they will never forget. After the post-tests were administered and the final course evaluation forms were filled in by the participants, the trainer

expressed his closing remarks and reiterated the importance of the training for the trainees in their daily activities and their readiness for applying the learnt concepts in their assigned tasks. Certificates of completion were distributed among the participants at the end.

Chapter 2

Company Profile

About The Spark Foundation

To inspire students, help them innovate and let them integrate to build the next generation humankind.

- **Inspire**
  - To inspire, motivate and encourage students to learn, create and help build a better society.
- **Innovate**
  - To teach new ways of thinking, to innovate and solve the problems on their own.
- **Integrate**
  - To let the students integrate, and help each other, learn from each other and do well together.

What can you find on this company/website?

This aim to provide as many resources as possible for learning analytics. These resources include:

1. **Training and tutorials**: Stuff to get you going and make you better in analytics and data science
2. **Tips and tricks** related to Data Science, Machine Learning, Business Analytics and Business Intelligence tools
3. **Hackathons** – Real life industry problems being released in form of contests
4. **Case studies:** Case studies of problems and their analytical solutions
5. **Interviews** of Business Analytics & Business Intelligence leaders

The Spark Foundation's Team

We are a group of passionate data science professionals, who decided to dedicate their careers in building Analytics and Data Science community. We feel passionately about creating next generation data science community and ecosystem and will leave no stone unturned to do the same. *(www.thesparksfoundationsingapore.org/)*

Chapter – 3

Data Science Work

Theory

A core objective of a learner is to generalize from its experience. The computational analysis of machine learning algorithms and their performance is a branch of theoretical computer science known as computational learning theory. Because training sets are finite and the future is uncertain, learning theory usually does not yield guarantees of the performance of algorithms. Instead, probabilistic bounds on the performance are quite common. The bias–variance decomposition is one way to quantify generalization error. For the best performance in the context of generalization, the complexity of the hypothesis should match the complexity of the function underlying the data. If the hypothesis is less complex than the function, then the model has underfit the data. If the complexity of the model is increased in response, then the training error decreases. But if the hypothesis is too complex, then the model is subject to overfitting and generalization will be poorer.

In addition to performance bounds, learning theorists study the time complexity and feasibility of learning in computational learning theory, a computation is considered feasible if it can be done in polynomial time. There are two kinds of time complexity results. Positive results show that a certain class of functions can be learned in polynomial time. Negative results show that certain classes cannot be learned in polynomial time.

The Challenges Facing Data Science

Fig 3.1 (Data Science Challenge)

- **Finding the data**

The first step of any data science project is unsurprisingly to find the data assets needed to start working. The surprising part is that the availability of the "right" data is still the most common challenge of data scientists, directly impacting their ability to build strong models. But why is data so

hard to find?

The first issue is that most companies collect tremendous volumes of data without determining first whether it is really going to be consumed, and by whom. This is driven by a fear of missing out on key insights that could be derived from it, and the availability of cheap storage. The dark side of this data-collection frenzy is that organizations end up gathering useless data, taking the focus away from actionability. This makes it harder for data users to find the truly relevant data assets for the business strategy. Businesses need to ensure they collect relevant data that is going to be utilized. For that, it is key to understand exactly what needs to be measured in order to drive decision-making, and this varies according to the various organizations.

Secondly, data is scattered in multiple sources, making it difficult for data scientists to find the right asset. Part of the solution is to consolidate the information in a single place. That's why so many companies use a data warehouse, in which they store the data from all their various sources.

However, having a single source of truth for your data assets is not enough without data documentation. What use can you make of a huge data repository if you don't know what's in it? The key for data scientists to find the tables relevant to their work is to maintain a neatly organized inventory of data assets. That is, each table should be enriched with context about what it contains, who imported it in the company, which dashboard and KPI it is related to, and any other information that can help data scientists locate it. This inventory can be maintained manually, in an excel spreadsheet shared with your company's employees. If that's what you need at the moment, we've got a template in store here, and we explain how to use it effectively. If your organization is too large for manual documentation, the alternative solution is to use a data cataloging tool to bring visibility to your data assets. If you prefer this option, make sure you choose a tool that suits your company's needs. We've listed the various options here.

- **Getting access to the data**

Once data scientists locate the right table, the next challenge is accessing the latter. Security and compliance issues are making it harder for data scientists to access datasets. As organizations transition into cloud data management, cyberattacks have become quite common. This has led to two major issues:

Confidential data is becoming vulnerable to these attacks

The response to cyberattacks has been to tighten regulatory requirements for businesses. As a result, data scientists are struggling to get consent to use the data, which drastically slows down their work.

Worse, when they are refused access to a dataset.

Organizations thus face the challenge of keeping data secure and ensure strict adherence to data protection norms such as GDPR, while allowing the relevant parties to access the data they need. Failing at one of these two objectives will either lead to expensive fines and time-consuming audits, or to the impossibility of leveraging data efficiently.

Again, the solution lies in cataloging tools. Data catalogs make regulatory compliance a flawless process while making sure the right people can access the data they need. This is mainly achieved through features of access management, whereby you can grant/restrict access in one click to tables based on employees' statuses. This way, data scientists will seamlessly to the datasets they need. You will find further information here about how data catalogs can be used as regulatory compliance tools.

- **Understanding the data**

You would think that once data scientists find and obtain access to a specific table, they can finally work their magic and build powerful predictive models. sadly, still not. They usually sit scratching their head for ridiculous amounts of time with questions of the type:

What does the column name 'FRPT33' even mean?

Who can I ask this to?

Why are there so many missing values?

Although these questions are simple, getting an answer isn't. There is no ownership over datasets in organizations, and finding the person that knows the meaning of the column name you are enquiring about is like trying to find a needle in a haystack.

The solution to prevent data scientists in your organization from spending too much time on these basic questions is again to ... document data assets. As simple as that. If you can have a written definition for every column in every table of your data warehouse, you will see the productivity of your data scientists skyrocket. Does that seem tedious? We assure you, it takes less time than letting undocumented assets roam around your business with unproductive data scientists spending 80% of their time trying to figure them out. Also, modern data documentation solutions have automation features, meaning that when you define a single column in a table, this definition is propagated to all other columns bearing a similar name in other tables.

- **Data cleaning**

Unfortunately, real-life data is nothing like hackathon data or Kaggle data. It is much messier. The result? Data scientists spend most of their time pre-processing data to make it consistent before

analyzing it, instead of building meaningful models. This tedious task involves cleaning the data, removing outliers, encoding variables, and so on. Although data pre-processing is often considered the worst part of a data scientist's job, it is crucial that models are built on clean, high-quality data. Otherwise, machine learning models learn the wrong patterns, ultimately leading to wrong predictions. How then can data scientists spend less time pre-processing data while ensuring only high quality data is used for training machine learning models?

One solution lies in using augmented analytics. It is the use of technologies such as machine learning and AI to assist with data preparation to augment how data scientists pre-process data. This allows for the possibility of automating certain aspects of data cleansing which can save data scientists significant amounts of time while keeping the same productivity levels.

- **Communicating the results to non-technical stakeholders.**

Data scientists' work is meant to be perfectly aligned with business strategy, as the ultimate goal of data science is to guide and improve decision-making in organizations. Hence, one of their biggest challenges is to communicate their results to business executives. In fact, managers and other stakeholders are ignorant of the tools and the works behind models. They have to base their decisions on data scientists' explanations. If the latter can't explain how their model will affect the performance of the organization, their solution is unlikely to be executed. There are two things making this communication to non-technical stakeholders a challenge:

First, data scientists often have a technical background, making it difficult for them to translate their data findings into clear business insights. But this is something that can be practiced. They can adopt concepts such as "data storytelling" to provide a powerful narrative to their analysis and visualizations.

Second, business terms and KPI's are poorly defined in most companies. For example, everyone knows roughly what the ROI is made of in a company, but there is rarely a common understanding across all departments of how it is computed exactly. There ends up being as many ROI definitions as they are employees calculating it. And it's the same story for other KPIs and business terms. This makes it even harder for data scientists to understand and explain the impact of their work related to specific KPIs. How on earth are they then expected to convince business executives to implement their solutions? The solution is simple. Define your KPI's and make sure everyone has a common understanding of each metric. Proper business KPI's will allow you to measure exactly the business impact generated by data scientists' analyses. A good way of building a single source of truth for your KPIs and business terms is to use a data catalog. This solution ensures everyone is aligned regarding key definitions for your business.

Part of Data Science

The types of data science differ in their approach, the type of data they input and output, and the type of task or problem that they are intended to solve. Broadly Data Science can be categorized into three categories.

I. Artificial Intelligence
II. Machine Learning
III. Deep Learning

Artificial Intelligence-

Artificial Intelligence (AI), the ability of a digital computer or computer-controlled robot to perform tasks commonly associated with intelligent beings. The term is frequently applied to the project of developing systems endowed with the intellectual processes characteristic of humans, such as the ability to reason, discover meaning, generalize, or learn from past experience. Since the development of the digital computer in the 1940s, it has been demonstrated that computers can be programmed to carry out very complex tasks—as, for example, discovering proofs for mathematical theorems or playing chess—with great proficiency. Still, despite continuing advances in computer processing speed and memory capacity, there are as yet no programs that can match human flexibility over wider domains or in tasks requiring much everyday knowledge. On the other hand, some programs have attained the performance levels of human experts and professionals in performing certain specific tasks, so that artificial intelligence in this limited sense is found in applications as diverse as medical diagnosis, computer search engines, and voice or handwriting recognition.

What is intelligence?

All but the simplest human behaviour is ascribed to intelligence, while even the most complicated insect behaviour is never taken as an indication of intelligence. What is the difference? Consider the behaviour of the digger wasp, Sphex ichneumoneus. When the female wasp returns to her burrow with food, she first deposits it on the threshold, checks for intruders inside her burrow, and only then, if the coast is clear, carries her food inside. The real nature of the wasp's instinctual behaviour is revealed if the food is moved a few inches away from the entrance to her burrow while she is inside: on emerging, she will repeat the whole procedure as often as the food is displaced. Intelligence—conspicuously absent in the case of Sphex—must include the ability to adapt to new

circumstances.

Psychologists generally do not characterize human intelligence by just one trait but by the combination of many diverse abilities. Research in AI has focused chiefly on the following components of intelligence: learning, reasoning, problem solving, perception, and using language.

Learning

There are a number of different forms of learning as applied to artificial intelligence. The simplest is learning by trial and error. For example, a simple computer program for solving mate-in-one chess problems might try moves at random until mate is found. The program might then store the solution with the position so that the next time the computer encountered the same position it would recall the solution. This simple memorizing of individual items and procedures—known as rote learning—is relatively easy to implement on a computer. More challenging is the problem of implementing what is called generalization. Generalization involves applying past experience to analogous new situations. For example, a program that learns the past tense of regular English verbs by rote will not be able to produce the past tense of a word such as jump unless it previously had been presented with jumped, whereas a program that is able to generalize can learn the "add ed" rule and so form the past tense of jump based on experience with similar verbs.

Reasoning

To reason is to draw inferences appropriate to the situation. Inferences are classified as either deductive or inductive. An example of the former is, "Fred must be in either the museum or the café. He is not in the café; therefore he is in the museum," and of the latter, "Previous accidents of this sort were caused by instrument failure; therefore this accident was caused by instrument failure." The most significant difference between these forms of reasoning is that in the deductive case the truth of the premises guarantees the truth of the conclusion, whereas in the inductive case the truth of the premise lends support to the conclusion without giving absolute assurance. Inductive reasoning is common in science, where data are collected and tentative models are developed to describe and predict future behaviour—until the appearance of anomalous data forces the model to be revised. Deductive reasoning is common in mathematics and logic, where elaborate structures of irrefutable theorems are built up from a small set of basic axioms and rules.

There has been considerable success in programming computers to draw inferences, especially deductive inferences. However, true reasoning involves more than just drawing inferences; it involves drawing inferences relevant to the solution of the particular task or situation. This is one of the hardest problems confronting AI.

Problem solving

Problem solving, particularly in artificial intelligence, may be characterized as a systematic search through a range of possible actions in order to reach some predefined goal or solution. Problem-solving methods divide into special purpose and general purpose. A special-purpose method is tailor-made for a particular problem and often exploits very specific features of the situation in which the problem is embedded. In contrast, a general-purpose method is applicable to a wide variety of problems. One general-purpose technique used in AI is means-end analysis—a step-by-step, or incremental, reduction of the difference between the current state and the final goal. The program selects actions from a list of means—in the case of a simple robot this might consist of PICKUP, PUTDOWN, MOVEFORWARD, MOVEBACK, MOVELEFT, and MOVERIGHT—until the goal is reached.

Many diverse problems have been solved by artificial intelligence programs. Some examples are finding the winning move (or sequence of moves) in a board game, devising mathematical proofs, and manipulating “virtual objects” in a computer-generated world.

Perception

In perception the environment is scanned by means of various sensory organs, real or artificial, and the scene is decomposed into separate objects in various spatial relationships. Analysis is complicated by the fact that an object may appear different depending on the angle from which it is viewed, the direction and intensity of illumination in the scene, and how much the object contrasts with the surrounding field.

At present, artificial perception is sufficiently well advanced to enable optical sensors to identify individuals, autonomous vehicles to drive at moderate speeds on the open road, and robots to roam through buildings collecting empty soda cans. One of the earliest systems to integrate perception and action was FREDDY, a stationary robot with a moving television eye and a pincer hand, constructed at the University of Edinburgh, Scotland, during the period 1966–73 under the direction of Donald Michie. FREDDY was able to recognize a variety of objects and could be instructed to assemble simple artifacts, such as a toy car, from a random heap of components.

Language

A language is a system of signs having meaning by convention. In this sense, language need not be confined to the spoken word. Traffic signs, for example, form a minilanguage, it being a matter of

convention that ⚠ means “hazard ahead” in some countries. It is distinctive of languages that linguistic units possess meaning by convention, and linguistic meaning is very different from what is called natural meaning, exemplified in statements such as “Those clouds mean rain” and “The fall in pressure means the valve is malfunctioning.”

An important characteristic of full-fledged human languages—in contrast to birdcalls and traffic signs—is their productivity. A productive language can formulate an unlimited variety of sentences.
It is relatively easy to write computer programs that seem able, in severely restricted contexts, to respond fluently in a human language to questions and statements. Although none of these programs actually understands language, they may, in principle, reach the point where their command of a language is indistinguishable from that of a normal human. What, then, is involved in genuine understanding, if even a computer that uses language like a native human speaker is not acknowledged to understand? There is no universally agreed upon answer to this difficult question. According to one theory, whether or not one understands depends not only on one’s behaviour but also on one’s history: in order to be said to understand, one must have learned the language and have been trained to take one’s place in the linguistic community by means of interaction with other language users.

Methods and goals in AI

Symbolic vs. connectionist approaches

AI research follows two distinct, and to some extent competing, methods, the symbolic (or “top-down”) approach, and the connectionist (or “bottom-up”) approach. The top-down approach seeks to replicate intelligence by analyzing cognition independent of the biological structure of the brain, in terms of the processing of symbols—whence the symbolic label. The bottom-up approach, on the other hand, involves creating artificial neural networks in imitation of the brain’s structure—whence the connectionist label.

To illustrate the difference between these approaches, consider the task of building a system, equipped with an optical scanner, that recognizes the letters of the alphabet. A bottom-up approach typically involves training an artificial neural network by presenting letters to it one by one, gradually improving performance by “tuning” the network. (Tuning adjusts the responsiveness of different neural pathways to different stimuli.) In contrast, a top-down approach typically involves writing a computer program that compares each letter with geometric descriptions. Simply put, neural activities are the basis of the bottom-up approach, while symbolic descriptions are the basis of the top-down approach.

In The Fundamentals of Learning (1932), Edward Thorndike, a psychologist at Columbia University, New York City, first suggested that human learning consists of some unknown property of connections between neurons in the brain. In The Organization of Behavior (1949), Donald Hebb, a psychologist at McGill University, Montreal, Canada, suggested that learning specifically involves strengthening certain patterns of neural activity by increasing the probability (weight) of induced neuron firing between the associated connections. The notion of weighted connections is described in a later section, Connectionism.

In 1957 two vigorous advocates of symbolic AI—Allen Newell, a researcher at the RAND Corporation, Santa Monica, California, and Herbert Simon, a psychologist and computer scientist at Carnegie Mellon University, Pittsburgh, Pennsylvania—summed up the top-down approach in what they called the physical symbol system hypothesis. This hypothesis states that processing structures of symbols is sufficient, in principle, to produce artificial intelligence in a digital computer and that, moreover, human intelligence is the result of the same type of symbolic manipulations.

During the 1950s and ’60s the top-down and bottom-up approaches were pursued simultaneously, and both achieved noteworthy, if limited, results. During the 1970s, however, bottom-up AI was neglected, and it was not until the 1980s that this approach again became prominent. Nowadays both approaches are followed, and both are acknowledged as facing difficulties. Symbolic techniques work in simplified realms but typically break down when confronted with the real world; meanwhile, bottom-up researchers have been unable to replicate the nervous systems of even the simplest living things. Caenorhabditis elegans, a much-studied worm, has approximately 300 neurons whose pattern of interconnections is perfectly known. Yet connectionist models have failed to mimic even this worm. Evidently, the neurons of connectionist theory are gross oversimplifications of the real thing.

Strong AI, applied AI, and cognitive simulation

Employing the methods outlined above, AI research attempts to reach one of three goals: strong AI, applied AI, or cognitive simulation. Strong AI aims to build machines that think. (The term strong AI was introduced for this category of research in 1980 by the philosopher John Searle of the University of California at Berkeley.) The ultimate ambition of strong AI is to produce a machine whose overall intellectual ability is indistinguishable from that of a human being. As is described in the section Early milestones in AI, this goal generated great interest in the 1950s and ’60s, but such optimism has given way to an appreciation of the extreme difficulties involved. To date, progress has been meagre. Some critics doubt whether research will produce even a system with the overall intellectual ability of an ant in the foreseeable future. Indeed, some researchers working in AI’s other two branches view strong

AI as not worth pursuing.

Applied AI, also known as advanced information processing, aims to produce commercially viable "smart" systems—for example, "expert" medical diagnosis systems and stock-trading systems. Applied AI has enjoyed considerable success, as described in the section Expert systems.

In cognitive simulation, computers are used to test theories about how the human mind works—for example, theories about how people recognize faces or recall memories. Cognitive simulation is already a powerful tool in both neuroscience and cognitive psychology.

Machine Learning

Machine learning (ML) is a type of artificial intelligence (AI) that allows software applications to become more accurate at predicting outcomes without being explicitly programmed to do so. Machine learning algorithms use historical data as input to predict new output values.

Recommendation engines are a common use case for machine learning. Other popular uses include fraud detection, spam filtering, malware threat detection, business process automation (BPA) and predictive maintenance.

Why is machine learning important?

Machine learning is important because it gives enterprises a view of trends in customer behavior and business operational patterns, as well as supports the development of new products. Many of today's leading companies, such as Facebook, Google and Uber, make machine learning a central part of their operations. Machine learning has become a significant competitive differentiator for many companies.

What are the different types of machine learning?

Classical machine learning is often categorized by how an algorithm learns to become more accurate in its predictions. There are four basic approaches:supervised learning, unsupervised learning, semi-supervised learning and reinforcement learning. The type of algorithm data scientists choose to use depends on what type of data they want to predict.

Supervised learning:

In this type of machine learning, data scientists supply algorithms with labeled training data and define the variables they want the algorithm to assess for correlations. Both the input and the output of the algorithm is specified.

Unsupervised learning:

This type of machine learning involves algorithms that train on unlabeled data. The algorithm scans through data sets looking for any meaningful connection. The data that algorithms train on as well as the predictions or recommendations they output are predetermined.

Semi-supervised learning:

This approach to machine learning involves a mix of the two preceding types. Data scientists may feed an algorithm mostly labeled training data, but the model is free to explore the data on its own and develop its own understanding of the data set.

Reinforcement learning:

Data scientists typically use reinforcement learning to teach a machine to complete a multi-step process for which there are clearly defined rules. Data scientists program an algorithm to complete a task and give it positive or negative cues as it works out how to complete a task. But for the most part, the algorithm decides on its own what steps to take along the way.

How does supervised machine learning work?

Supervised machine learning requires the data scientist to train the algorithm with both labeled inputs and desired outputs. Supervised learning algorithms are good for the following tasks:

Binary classification: Dividing data into two categories.
Multi-class classification: Choosing between more than two types of answers.
Regression modeling: Predicting continuous values.
Ensembling: Combining the predictions of multiple machine learning models to produce an accurate prediction.

How does unsupervised machine learning work?

Unsupervised machine learning algorithms do not require data to be labeled. They sift through unlabeled data to look for patterns that can be used to group data points into subsets. Most types of deep learning, including neural networks, are unsupervised algorithms. Unsupervised learning algorithms are good for the following tasks:

Clustering: Splitting the dataset into groups based on similarity.

Anomaly detection: Identifying unusual data points in a data set.

Association mining: Identifying sets of items in a data set that frequently occur together.

Dimensionality reduction: Reducing the number of variables in a data set.

How does semi-supervised learning work?

Semi-supervised learning works by data scientists feeding a small amount of labeled training data to an algorithm. From this, the algorithm learns the dimensions of the data set, which it can then apply to new, unlabeled data. The performance of algorithms typically improves when they train on labeled data sets. But labeling data can be time consuming and expensive. Semi-supervised learning strikes a middle ground between the performance of supervised learning and the efficiency of unsupervised learning. Some areas where semi-supervised learning is used include:

Machine translation: Teaching algorithms to translate language based on less than a full dictionary of words.

Fraud detection: Identifying cases of fraud when you only have a few positive examples.

Labelling data: Algorithms trained on small data sets can learn to apply data labels to larger sets automatically.

How does reinforcement learning work?

Reinforcement learning works by programming an algorithm with a distinct goal and a prescribed set of rules for accomplishing that goal. Data scientists also program the algorithm to seek positive rewards -- which it receives when it performs an action that is beneficial toward the ultimate goal -- and avoid punishments -- which it receives when it performs an action that gets it farther away from its ultimate goal. Reinforcement learning is often used in areas such as:

Robotics: Robots can learn to perform tasks the physical world using this technique.

Video gameplay: Reinforcement learning has been used to teach bots to play a number of video games.

Resource management: Given finite resources and a defined goal, reinforcement learning can help enterprises plan out how to allocate resources.

HOW MACHINE LEARNING WORKS

Machine learning is like statistics on steroids.

Who's using machine learning and what's it used for?

Today, machine learning is used in a wide range of applications. Perhaps one of the most well-known examples of machine learning in action is the recommendation engine that powers Facebook's news feed.

Facebook uses machine learning to personalize how each member's feed is delivered. If a member frequently stops to read a particular group's posts, the recommendation engine will start to show more of that group's activity earlier in the feed.

Behind the scenes, the engine is attempting to reinforce known patterns in the member's online behavior. Should the member change patterns and fail to read posts from that group in the coming weeks, the news feed will adjust accordingly.

In addition to recommendation engines, other uses for machine learning include the following:

Customer relationship management. CRM software can use machine learning models to analyze email and prompt sales team members to respond to the most important messages first. More advanced systems can even recommend potentially effective responses.

Business intelligence. BI and analytics vendors use machine learning in their software to identify potentially important data points, patterns of data points and anomalies.

Human resource information systems. HRIS systems can use machine learning models to filter through applications and identify the best candidates for an open position.

Self-driving cars. Machine learning algorithms can even make it possible for a semi-autonomous car to recognize a partially visible object and alert the driver.

Virtual assistants. Smart assistants typically combine supervised and unsupervised machine learning models to interpret natural speech and supply context.

What are the advantages and disadvantages of machine learning?

Machine learning has seen use cases ranging from predicting customer behavior to forming the operating system for self-driving cars.

When it comes to advantages, machine learning can help enterprises understand their customers at a deeper level. By collecting customer data and correlating it with behaviors over time, machine learning algorithms can learn associations and help teams tailor product development and marketing initiatives to customer demand.

Some companies use machine learning as a primary driver in their business models. Uber, for example, uses algorithms to match drivers with riders. Google uses machine learning to surface the ride advertisements in searches.

But machine learning comes with disadvantages. First and foremost, it can be expensive. Machine learning projects are typically driven by data scientists, who command high salaries. These projects also require software infrastructure that can be expensive.

There is also the problem of machine learning bias. Algorithms trained on data sets that exclude certain populations or contain errors can lead to inaccurate models of the world that, at best, fail and, at worst, are discriminatory. When an enterprise bases core business processes on biased models it can run into regulatory and reputational harm.

How to choose the right machine learning model

The process of choosing the right machine learning model to solve a problem can be time consuming if not approached strategically.

Step 1: Align the problem with potential data inputs that should be considered for the solution. This step requires help from data scientists and experts who have a deep understanding of the problem.

Step 2: Collect data, format it and label the data if necessary. This step is typically led by data scientists, with help from data wranglers.

Step 3: Chose which algorithm(s) to use and test to see how well they perform. This step is usually carried out by data scientists.

Step 4: Continue to fine tune outputs until they reach an acceptable level of accuracy. This step is usually carried out by data scientists with feedback from experts who have a deep understanding of the problem.

Importance of human interpretable machine learning

Explaining how a specific ML model works can be challenging when the model is complex. There are some vertical industries where data scientists have to use simple machine learning models because it's important for the business to explain how every decision was made. This is especially true in industries with heavy compliance burdens such as banking and insurance.

Complex models can produce accurate predictions, but explaining to a lay person how an output was determined can be difficult.

What is the future of machine learning?

While machine learning algorithms have been around for decades, they've attained new popularity as artificial intelligence has grown in prominence. Deep learning models, in particular, power today's most advanced AI applications.

Machine learning platforms are among enterprise technology's most competitive realms, with most major vendors, including Amazon, Google, Microsoft, IBM and others, racing to sign customers up for platform services that cover the spectrum of machine learning activities, including data collection, data preparation, data classification, model building, training and application deployment.

As machine learning continues to increase in importance to business operations and AI becomes more practical in enterprise settings, the machine learning platform wars will only intensify.

Continued research into deep learning and AI is increasingly focused on developing more general applications. Today's AI models require extensive training in order to produce an algorithm that is highly optimized to perform one task. But some researchers are exploring ways to make models more flexible and are seeking techniques that allow a machine to apply context learned from one task to future, different tasks.

How deep learning differs from traditional machine learning

Deep learning works in very different ways than traditional machine learning.

How has machine learning evolved?

1642 - Blaise Pascal invents a mechanical machine that can add, subtract, multiply and divide.

1679 - Gottfried Wilhelm Leibniz devises the system of binary code.

1834 - Charles Babbage conceives the idea for a general all-purpose device that could be programmed with punched cards.

1842 - Ada Lovelace describes a sequence of operations for solving mathematical problems using Charles Babbage's theoretical punch-card machine and becomes the first programmer.

1847 - George Boole creates Boolean logic, a form of algebra in which all values can be reduced to the binary values of true or false.

1936 - English logician and cryptanalyst Alan Turing proposes a universal machine that could decipher and execute a set of instructions. His published proof is considered the basis of computer science.

1952 - Arthur Samuel creates a program to help an IBM computer get better at checkers the more it plays.

1959 - MADALINE becomes the first artificial neural network applied to a real-world problem: removing echoes from phone lines.

1985 - Terry Sejnowski's and Charles Rosenberg's artificial neural network taught itself how to correctly pronounce 20,000 words in one week.

1997 - IBM's Deep Blue beat chess grandmaster Garry Kasparov.

1999 - A CAD prototype intelligent workstation reviewed 22,000 mammograms and detected cancer 52% more accurately than radiologists did.

2006 - Computer scientist Geoffrey Hinton invents the term deep learning to describe neural net research.

2012 - An unsupervised neural network created by Google learned to recognize cats in YouTube videos with 74.8% accuracy.

2014 - A chatbot passes the Turing Test by convincing 33% of human judges that it was a Ukrainian teen named Eugene Goostman.

2014 - Google's AlphaGo defeats the human champion in Go, the most difficult board game in the world.

2016 - LipNet, DeepMind's artificial intelligence system, identifies lip-read words in video with an accuracy of 93.4%.

2019 - Amazon controls 70% of the market share for virtual assistants in the U.S.

Deep Learning-

Deep learning is a type of machine learning and artificial intelligence (AI) that imitates the way humans gain certain types of knowledge. Deep learning is an important element of data science, which includes statistics and predictive modeling. It is extremely beneficial to data scientists who are tasked with collecting, analyzing and interpreting large amounts of data; deep learning makes this process faster and easier.

At its simplest, deep learning can be thought of as a way to automate predictive analytics. While traditional machine learning algorithms are linear, deep learning algorithms are stacked in a hierarchy of increasing complexity and abstraction.

To understand deep learning, imagine a toddler whose first word is dog. The toddler learns what a dog is -- and is not -- by pointing to objects and saying the word dog. The parent says, "Yes, that is a dog," or, "No, that is not a dog." As the toddler continues to point to objects, he becomes more aware of the features that all dogs possess. What the toddler does, without knowing it, is clarify a complex abstraction -- the concept of dog -- by building a hierarchy in which each level of abstraction is created with knowledge that was gained from the preceding layer of the hierarchy.

How deep learning works

Computer programs that use deep learning go through much the same process as the toddler

learning to identify the dog. Each algorithm in the hierarchy applies a nonlinear transformation to its input and uses what it learns to create a statistical model as output. Iterations continue until the output has reached an acceptable level of accuracy. The number of processing layers through which data must pass is what inspired the label deep.

In traditional machine learning, the learning process is supervised, and the programmer has to be extremely specific when telling the computer what types of things it should be looking for to decide if an image contains a dog or does not contain a dog. This is a laborious process called feature extraction, and the computer's success rate depends entirely upon the programmer's ability to accurately define a feature set for dog. The advantage of deep learning is the program builds the feature set by itself without supervision. Unsupervised learning is not only faster, but it is usually more accurate.

Initially, the computer program might be provided with training data -- a set of images for which a human has labeled each image dog or not dog with metatags. The program uses the information it receives from the training data to create a feature set for dog and build a predictive model. In this case, the model the computer first creates might predict that anything in an image that has four legs and a tail should be labeled dog. Of course, the program is not aware of the labels four legs or tail. It will simply look for patterns of pixels in the digital data. With each iteration, the predictive model becomes more complex and more accurate.

Unlike the toddler, who will take weeks or even months to understand the concept of dog, a computer program that uses deep learning algorithms can be shown a training set and sort through millions of images, accurately identifying which images have dogs in them within a few minutes.

To achieve an acceptable level of accuracy, deep learning programs require access to immense amounts of training data and processing power, neither of which were easily available to programmers until the era of big data and cloud computing. Because deep learning programming can create complex statistical models directly from its own iterative output, it is able to create accurate predictive models from large quantities of unlabeled, unstructured data. This is important as the internet of things (IoT) continues to become more pervasive because most of the data humans and machines create is unstructured and is not labeled.

Deep learning methods

Various methods can be used to create strong deep learning models. These techniques include learning rate decay, transfer learning, training from scratch and dropout.

Learning rate decay. The learning rate is a hyperparameter -- a factor that defines the system or set

conditions for its operation prior to the learning process -- that controls how much change the model experiences in response to the estimated error every time the model weights are altered. Learning rates that are too high may result in unstable training processes or the learning of a suboptimal set of weights. Learning rates that are too small may produce a lengthy training process that has the potential to get stuck.

The learning rate decay method -- also called learning rate annealing or adaptive learning rates -- is the process of adapting the learning rate to increase performance and reduce training time. The easiest and most common adaptations of learning rate during training include techniques to reduce the learning rate over time.

Transfer learning. This process involves perfecting a previously trained model; it requires an interface to the internals of a preexisting network. First, users feed the existing network new data containing previously unknown classifications. Once adjustments are made to the network, new tasks can be performed with more specific categorizing abilities. This method has the advantage of requiring much less data than others, thus reducing computation time to minutes or hours.

Training from scratch. This method requires a developer to collect a large labeled data set and configure a network architecture that can learn the features and model. This technique is especially useful for new applications, as well as applications with a large number of output categories. However, overall, it is a less common approach, as it requires inordinate amounts of data, causing training to take days or weeks.

Dropout. This method attempts to solve the problem of overfitting in networks with large amounts of parameters by randomly dropping units and their connections from the neural network during training. It has been proven that the dropout method can improve the performance of neural networks on supervised learning tasks in areas such as speech recognition, document classification and computational biology.

Deep learning neural networks?

A type of advanced machine learning algorithm, known as an artificial neural network, underpins most deep learning models. As a result, deep learning may sometimes be referred to as deep neural learning or deep neural networking.

Neural networks come in several different forms, including recurrent neural networks, convolutional neural networks, artificial neural networks and feedforward neural networks, and each has benefits for specific use cases. However, they all function in somewhat similar ways -- by feeding data in and letting the model figure out for itself whether it has made the right interpretation or decision about a given data element.

Neural networks involve a trial-and-error process, so they need massive amounts of data on which to train. It's no coincidence neural networks became popular only after most enterprises embraced big data analytics and accumulated large stores of data. Because the model's first few iterations involve somewhat educated guesses on the contents of an image or parts of speech, the data used during the training stage must be labeled so the model can see if its guess was accurate. This means, though many enterprises that use big data have large amounts of data, unstructured data is less helpful. Unstructured data can only be analyzed by a deep learning model once it has been trained and reaches an acceptable level of accuracy, but deep learning models can't train on unstructured data.

Deep learning examples

Because deep learning models process information in ways similar to the human brain, they can be applied to many tasks people do. Deep learning is currently used in most common image recognition tools, natural language processing (NLP) and speech recognition software. These tools are starting to appear in applications as diverse as self-driving cars and language translation services.

Use cases today for deep learning include all types of big data analytics applications, especially those focused on NLP, language translation, medical diagnosis, stock market trading signals, network security and image recognition.

Specific fields in which deep learning is currently being used include the following:

- **Customer experience (CX).** Deep learning models are already being used for chatbots. And, as it continues to mature, deep learning is expected to be implemented in various businesses to improve CX and increase customer satisfaction.
- **Text generation.** Machines are being taught the grammar and style of a piece of text and are then using this model to automatically create a completely new text matching the proper spelling, grammar and style of the original text.
- **Aerospace and military.** Deep learning is being used to detect objects from satellites that identify areas of interest, as well as safe or unsafe zones for troops.

  Industrial automation. Deep learning is improving worker safety in environments like factories and

warehouses by providing services that automatically detect when a worker or object is getting too close to a machine.

- **Adding color.** Color can be added to black-and-white photos and videos using deep learning models. In the past, this was an extremely time-consuming, manual process.
  Medical research. Cancer researchers have started implementing deep learning into their practice as a way to automatically detect cancer cells.
- **Computer vision.** Deep learning has greatly enhanced computer vision, providing computers with extreme accuracy for object detection and image classification, restoration and segmentation.

Limitations and challenges

The biggest limitation of deep learning models is they learn through observations. This means they only know what was in the data on which they trained. If a user has a small amount of data or it comes from one specific source that is not necessarily representative of the broader functional area, the models will not learn in a way that is generalizable.

The issue of biases is also a major problem for deep learning models. If a model trains on data that contains biases, the model will reproduce those biases in its predictions. This has been a vexing problem for deep learning programmers because models learn to differentiate based on subtle variations in data elements. Often, the factors it determines are important are not made explicitly clear to the programmer. This means, for example, a facial recognition model might make determinations about people's characteristics based on things like race or gender without the programmer being aware.

The learning rate can also become a major challenge to deep learning models. If the rate is too high, then the model will converge too quickly, producing a less-than-optimal solution. If the rate is too low, then the process may get stuck, and it will be even harder to reach a solution.

The hardware requirements for deep learning models can also create limitations. Multicore high-performing graphics processing units (GPUs) and other similar processing units are required to ensure improved efficiency and decreased time consumption. However, these units are expensive and use large amounts of energy. Other hardware requirements include random access memory and a hard disk drive (HDD) or RAM-based solid-state drive (SSD).

Other limitations and challenges include the following:

Deep learning requires large amounts of data. Furthermore, the more powerful and accurate models will need more parameters, which, in turn, require more data.

Once trained, deep learning models become inflexible and cannot handle multitasking. They can

deliver efficient and accurate solutions but only to one specific problem. Even solving a similar problem would require retraining the system.

Any application that requires reasoning -- such as programming or applying the scientific method -- long-term planning and algorithmlike data manipulation is completely beyond what current deep learning techniques can do, even with large data.

Deep learning vs. machine learning

Deep learning is a subset of machine learning that differentiates itself through the way it solves problems. Machine learning requires a domain expert to identify most applied features. On the other hand, deep learning understands features incrementally, thus eliminating the need for domain expertise. This makes deep learning algorithms take much longer to train than machine learning algorithms, which only need a few seconds to a few hours. However, the reverse is true during testing. Deep learning algorithms take much less time to run tests than machine learning algorithms, whose test time increases along with the size of the data.

Furthermore, machine learning does not require the same costly, high-end machines and high-performing GPUs that deep learning does.

In the end, many data scientists choose traditional machine learning over deep learning due to its superior interpretability, or the ability to make sense of the solutions. Machine learning algorithms are also preferred when the data is small.

Instances where deep learning becomes preferable include situations where there is a large amount of data, a lack of domain understanding for feature introspection, or complex problems, such as speech recognition and NLP.

Chapter - 4

TECHNOLOGY IMPLEMENTED

Python – The New Generation Language

Python is a widely used general-purpose, high level programming language. It was initially designed by Guido van Rossum in 1991 and developed by Python Software Foundation. It was mainly developed for an emphasis on code readability, and its syntax allows programmers to express concepts in fewer lines of code Python is dynamically typed and garbage-collected. It supports multiple programming paradigms, including procedural, object-oriented, and functional programming. Python is often described as a "batteries included" language due to its comprehensive standard library.

Features

• Interpreted

In Python there is no separate compilation and execution steps like C/C++. It directly run the program from the source code. Internally, Python converts the source code into an intermediate form called bytecodes which is then translated into native language of specific computer to run it.

• Platform

Independent Python programs can be developed and executed on the multiple operating system platform. Python can be used on Linux, Windows, Macintosh, Solaris and many more.

• Multi-Paradigm

Python is a multi-paradigm programming language. Object-oriented programming and structured programming are fully supported, and many of its features support functional programming and aspect-oriented programming.

• Simple

Python is a very simple language. It is a very easy to learn as it is closer to English language. In python more emphasis is on the solution to the problem rather than the syntax.

• Rich Library Support

Python standard library is very vast. It can help to do various things involving regular expressions, documentation generation, unit testing, threading, databases, web browsers, CGI, email, XML, HTML, WAV files, cryptography, GUI and many more.

• Free and Open Source Firstly

Python is freely available. Secondly, it is open-source. This means that its source code is available to the public. We can download it, change it, use it, and distribute it. This is called FLOSS(Free/Libre and Open Source Software). As the Python community, we're all headed toward one goal-an ever-bettering Python.

Why Python is perfect language for Data Science?

1. **A great library ecosystem –**

A great choice of libraries is one of the main reasons Python is the most popular programming language used for AI. A library is a module or a group of modules published by different sources which include a pre-written piece of code that allows users to reach some functionality or perform different actions. Python libraries provide base level items so developers don't have to code them from the very beginning every time. ML requires continuous data processing, and Python's libraries let us access, handle and transform data. These are some of the most widespread libraries you can use for ML and AI:

a. Scikit-learn for handling basic ML algorithms like clustering, linear and logistic regressions, regression, classification, and others.
b. Pandas for high-level data structures and analysis. It allows merging and filtering of data, as well as gathering it from other external sources like Excel, for instance.
c. Keras for deep learning. It allows fast calculations and prototyping, as it uses the GPU in addition to the CPU of the computer.
d. Tensor Flow for working with deep learning by setting up, training, and utilizing artificial neural networks with massive datasets.
e. Matplotlib for creating 2D plots, histograms, charts, and other forms of visualization.

f. NLTK for working with computational linguistics, natural language recognition, and processing.
g. Scikit-image for image processing.
h. Py-Brain for neural networks, unsupervised and reinforcement learning.
i. Caffe for deep learning that allows switching between the CPU and the GPU and processing60+ mln images a day using a single NVIDIA K40 GPU.
j. Stats Models for statistical algorithms and data exploration

In the PyPI repository, we can discover and compare more python libraries.

2. A low entry barrier –

Working in the ML and AI industry means dealing with a bunch of data that we need to process in the most convenient and effective way. The low entry barrier allows more data scientists to quickly pick up Python and start using it for AI development without wasting too much effort into learning the language in addition to this, there's a lot of documentation available, and Python's community is always there to help out and give advice.

3. Flexibility-

Python for data science is a great choice, as this language is very flexible:

▪It offers an option to choose either to use OOPs or scripting.

▪There's also no need to recompile the source code, developers can implement any changes and quickly see the results.

▪Programmers can combine Python and other languages to reach their goals.

4. Good Visualization Options-

For AI developers, it's important to highlight that in artificial intelligence, deep learning, and machine learning, it's vital to be able to represent data in a human-readable format. Libraries like Matplotlib allow data scientists to build charts, histograms, and plots for better data comprehension effective presentation, and visualization. Different application programming interfaces also simplify the visualization process and make it easier to create clear reports.

5. Community Support-

It’s always very helpful when there’s strong community support built around the programming language. Python is an open-source language which means that there’s a bunch of resources open for programmers starting from beginners and ending with pros. A lot of Python documentation is available online as well as in Python communities and forums, where programmers and machine learning developers discuss errors, solve problems, and help each other out. Python programming language is absolutely free as is the variety of useful libraries and tools.6. Growing Popularity-As a result of the advantages discussed above, Python is becoming more and more popular among data scientists. According to Stack Overflow, the popularity of Python is predicted to grow until 2020 atleast. This means it’s easier to search for developers and replace team players if required. Also, the cost of their work maybe not as high as when using a less popular programming language.

Data Preprocessing, Analysis & Visualization

Data Science algorithms don't work so well with processing raw data. Before we can feed such data to an Data Science algorithm, we must preprocess it. We must apply some transformations on it. With data preprocessing, we convert raw data into a clean data set. To perform data this, there are 7 techniques-

1. Rescaling Data –

   For data with attributes of varying scales, we can rescale attributes to possess the same scale. We rescale attributes into the range 0 to 1 and call it normalization. We use the Min Max Scaler class from scikit-learn. This gives us values between 0 and 1.

2. Standardizing Data –

   With standardizing, we can take attributes with a Gaussian distribution and different means and standard deviations and transform them into a standard Gaussian distribution with a mean of 0 and a standard deviation of 1.

3. Normalizing Data –

   In this task, we rescale each observation to a length of 1 (a unit norm). For this, we use the Normalizer class.

4. Binarizing Data –

   Using a binary threshold, it is possible to transform our data by marking the values above it 1 and those equal to or below it, 0. For this purpose, we use the Binarizer class.

5. Mean Removal-

   We can remove the mean from each feature to center it on zero.

6. One Hot Encoding –

   When dealing with few and scattered numerical values, we may not need to store these. Then, we can perform One Hot Encoding. For k distinct

values, we can transform the feature into a k-dimensional vector with one value of 1 and 0 as the rest values.

7. Label Encoding –

   Some labels can be words or numbers. Usually, training data is labelled with words to make it readable Label encoding converts word labels into numbers to let algorithms work on them.

Data Science Algorithm

The implementation of Data Science to any problem requires a set of skills. Machine Learning is an integral part of this skill set.

For doing Data Science, you must know the various Machine Learning algorithms used for solving different types of problems, as a single algorithm cannot be the best for all types of use cases. These algorithms find an application in various tasks like prediction, classification, clustering, etc. from the dataset under consideration.

The most popular Machine Learning algorithms used by the Data Scientists are:

1.Linear Regression-

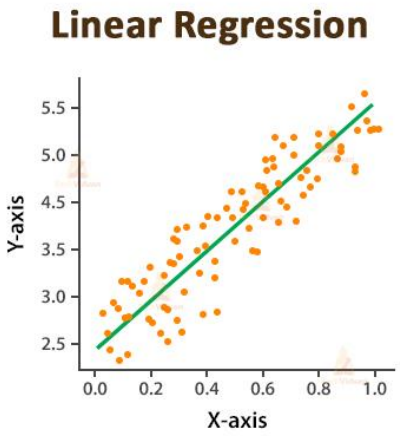

Fig 4.1(Linear Regression)

Linear regression method is used for predicting the value of the dependent variable by using the values of the independent variable.

The linear regression model is suitable for predicting the value of a continuous quantity.

OR

The linear regression model represents the relationship between the input variables (x) and the output variable (y) of a dataset in terms of a line given by the equation,

$y = b0 + b1x$

Where,

y is the dependent variable whose value we want to predict.

x is the independent variable whose values are used for predicting the dependent variable.

b0 and b1 are constants in which b0 is the Y-intercept and b1 is the slope.

The main aim of this method is to find the value of b0 and b1 to find the best fit line that will be covering or will be nearest to most of the data points.

2. Logistic Regression

Linear Regression is always used for representing the relationship between some continuous values. However, contrary to this Logistic Regression works on discrete values.

Logistic regression finds the most common application in solving binary classification problems, that is, when there are only two possibilities of an event, either the event will occur or it will not occur (0 or 1).

Thus, in Logistic Regression, we convert the predicted values into such values that lie in the range of 0 to 1 by using a non-linear transform function which is called a logistic function.

The logistic function results in an S-shaped curve and is therefore also called a Sigmoid function given by the equation,

?(x) = 1/1+e^-x data science algorithm - logistic regressions

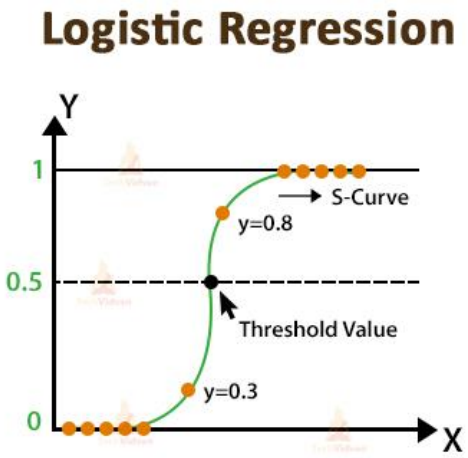

Fig 4.2(Logistic Regression)

The equation of Logistic Regression is,

P(x) = e^(b0+b1x)/1 + e^(b0+b1x)

Where b0 and b1 are coefficients and the goal of Logistic Regression is to find the value of these coefficients.

3. Decision Trees

Decision trees help in solving both classification and prediction problems. It makes it easy to understand the data for better accuracy of the predictions. Each node of the Decision tree represents a feature or an attribute, each link represents a decision and each leaf node holds a class label, that is, the outcome.

The drawback of decision trees is that it suffers from the problem of overfitting.

Basically, these two Data Science algorithms are most commonly used for implementing the Decision trees.

ID3 (Iterative Dichotomiser 3) Algorithm

This algorithm uses entropy and information gain as the decision metric.

Cart (Classification and Regression Tree) Algorithm

This algorithm uses the Gini index as the decision metric. The below image will help you to understand things better.

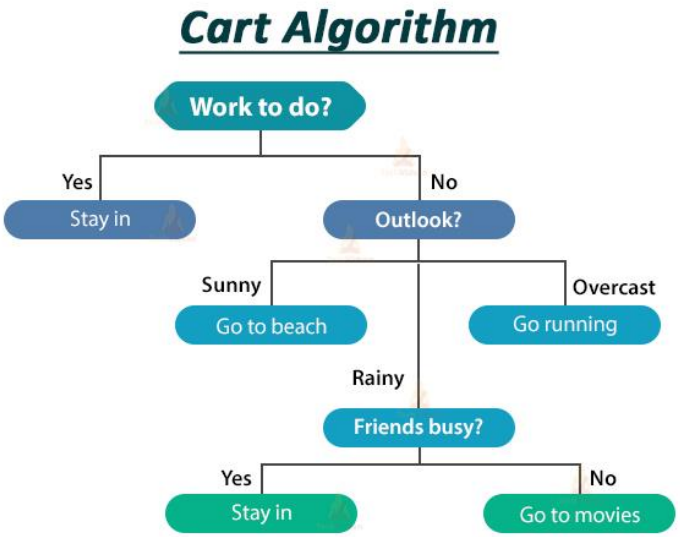

Fig 4.3(Decision Tree)

4. Naive Bayes

The Naive Bayes algorithm helps in building predictive models. We use this Data Science algorithm when we want to calculate the probability of the occurrence of an event in the future. Here, we have prior knowledge that another event has already occurred.

The Naive Bayes algorithm works on the assumption that each feature is independent and has an individual contribution to the final prediction.

The Naive Bayes theorem is represented by:

$P(A|B) = P(B|A)\ P(A) / P(B)$

Where A and B are two events.

P(A|B) is the posterior probability i.e. the probability of A given that B has already occurred.

P(B|A) is the likelihood i.e. the probability of B given that A has already occurred.

P(A) is the class prior to probability.

P(B) is the predictor prior probability.

5. KNN

KNN stands for K-Nearest Neighbors. This Data Science algorithm employs both classification and regression problems.

The KNN algorithm considers the complete dataset as the training dataset. After training the model using the KNN algorithm, we try to predict the outcome of a new data point.

Here, the KNN algorithm searches the entire data set for identifying the k most similar or nearest neighbors of that data point. It then predicts the outcome based on these k instances.

For finding the nearest neighbors of a data instance, we can use various distance measures like Euclidean distance, Hamming distance, etc.

To better understand, let us consider the following example.KNN data science algorithms

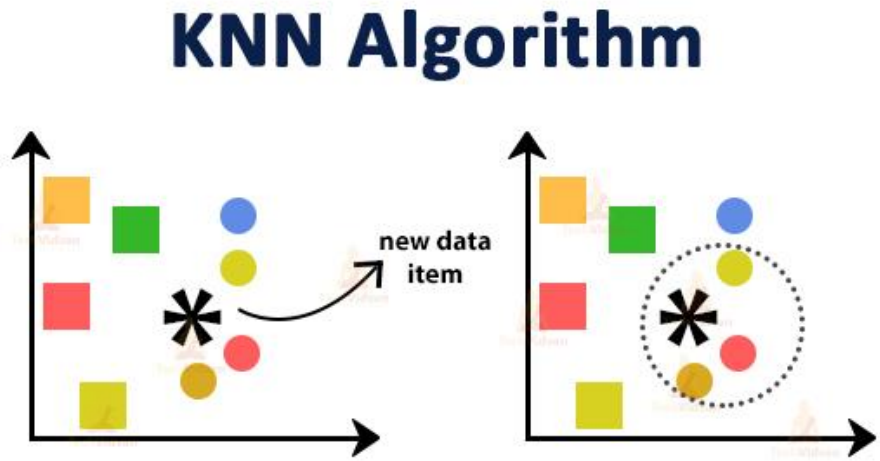

Fig 4.4(KNN Algorithm)

Here we have represented the two classes A and B by the circle and the square respectively.

Let us assume the value of k is 3.

Now we will first find three data points that are closest to the new data item and enclose them in a dotted circle. Here the three closest points of the new data item belong to class A. Thus, we can say that the new data point will also belong to class A.

Now you all might be thinking that how we assumed k=3?

The selection of the value of k is a very critical task. You should take such a value of k that it is neither too small nor too large. Another simpler approach is to take k = √n where n is the number of data points.

6. Support Vector Machine (SVM)-

Support Vector Machine or SVM comes under the category of supervised Machine Learning

algorithms and finds an application in both classification and regression problems. It is most commonly used for classification of problems and classifies the data points by using a hyperplane.

The first step of this Data Science algorithm involves plotting all the data items as individual points in an n-dimensional graph.

Here, n is the number of features and the value of each individual feature is the value of a specific coordinate. Then we find the hyperplane that best separates the two classes for classifying them.

Finding the correct hyperplane plays the most important role in classification. The data points which are closest to the separating hyperplane are the support vectors.svm support vectors

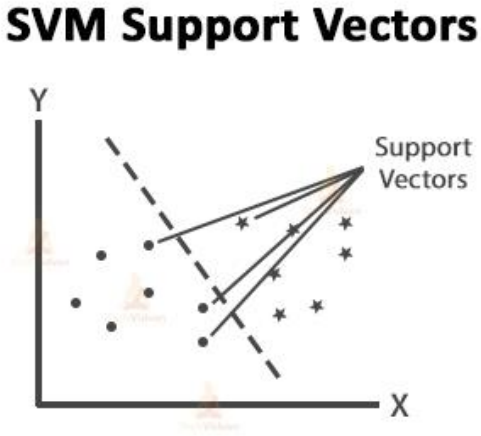

Fig 4.5(SVM Support Vectors)

Let us consider the following example to understand how you can identify the right hyperplane. The basic principle for selecting the best hyperplane is that you have to choose the hyperplane that separates the two classes very well.svm hyperlanes

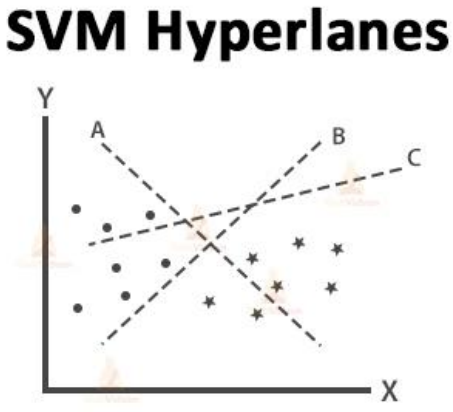

Fig 4.6(SVM Hyperlanes)

In this case, the hyperplane B is classifying the data points very well. Thus, B will be the right

hyperplane.hyperplane classifying data points

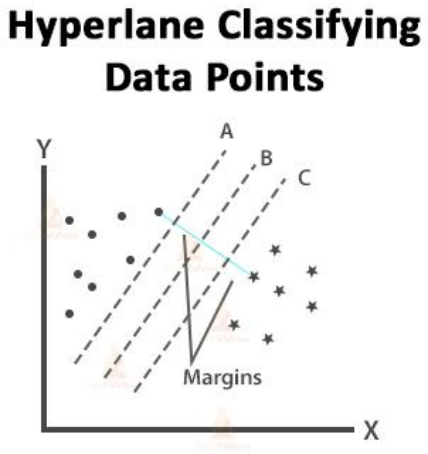

Fig 4.7(Hyperlane Classifying Data Points)

All three hyperplanes are separating the two classes properly. In such cases, we have to select the hyperplane with the maximum margin.

As we can see in the above image, hyperplane B has the maximum margin therefore it will be the right hyperplane.svm hyperlane with maximum margins

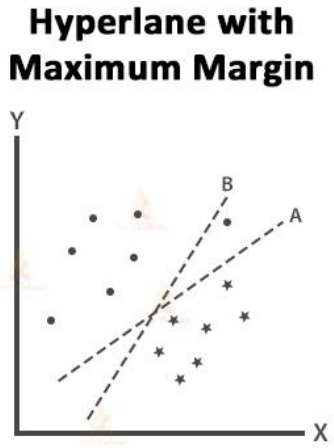

Fig 4.8(Hyperlane with Maximum Margin)

In this case, the hyperplane B has the maximum margin but it is not classifying the two classes accurately. Thus, A will be the right hyperplane.

7. K-Means Clustering

K-means clustering is a type of unsupervised Machine Learning algorithm.

Clustering basically means dividing the data set into groups of similar data items called clusters. K means clustering categorizes the data items into k groups with similar data items.

For measuring this similarity, we use Euclidean distance which is given by,

$D = \sqrt{(x1-x2)^2 + (y1-y2)^2}$

K means clustering is iterative in nature.

The basic steps followed by the algorithm are as follows:

First, we select the value of k which is equal to the number of clusters into which we want to categorize our data.

Then we assign the random center values to each of these k clusters.

Now we start searching for the nearest data points to the cluster centers by using the Euclidean distance formula.

In the next step, we calculate the mean of the data points assigned to each cluster.

Again we search for the nearest data points to the newly created centers and assign them to their closest clusters.

We should keep repeating the above steps until there is no change in the data points assigned to the k clusters.

8. Principal Component Analysis (PCA)

PCA is basically a technique for performing dimensionality reduction of the datasets with the least effect on the variance of the datasets. This means removing the redundant features but keeping the important ones.

To achieve this, PCA transforms the variables of the dataset into a new set of variables. This new set of variables represents the principal components.

The most important features of these principal components are:

All the PCs are orthogonal (i.e. they are at a right angle to each other).

They are created in such a way that with the increasing number of components, the amount of variation that it retains starts decreasing.

This means the 1st principal component retains the variation to the maximum extent as compared to the original variables.

PCA is basically used for summarizing data. While dealing with a dataset there might be some features related to each other. Thus PCA helps you to reduce such features and make predictions with less number of features without compromising with the accuracy.

For example, consider the following diagram in which we have reduced a 3D space to a 2D space.

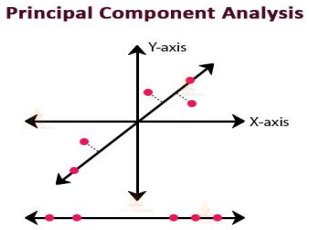

Fig 4.9(Principal Component Analysis)

9. Neural Networks

Neural Networks are also known as Artificial Neural Networks.

Let us understand this by an example.neural networks data science algorithms

Fig 4.10(Neural Networks)

Identifying the digits written in the above image is a very easy task for humans. This is because our brain contains millions of neurons that perform complex calculations for identifying any visual easily in no time.

But for machines, this is a very difficult task to do.

Neural networks solve this problem by training the machine with a large number of examples. By this, the machine automatically learns from the data for recognizing various digits.

Thus we can say that Neural Networks are the Data Science algorithms that work to make the machine identify the various patterns in the same way as a human brain does.

10. Random Forests

Random Forests overcomes the overfitting problem of decision trees and helps in solving both classification and regression problems. It works on the principle of Ensemble learning.

The Ensemble learning methods believe that a large number of weak learners can work together for giving high accuracy predictions.

Random Forests work in a much similar way. It considers the prediction of a large number of

individual decision trees for giving the final outcome. It calculates the number of votes of predictions of different decision trees and the prediction with the largest number of votes becomes the prediction of the model.

Let us understand this by an example.random forests data science algorithms

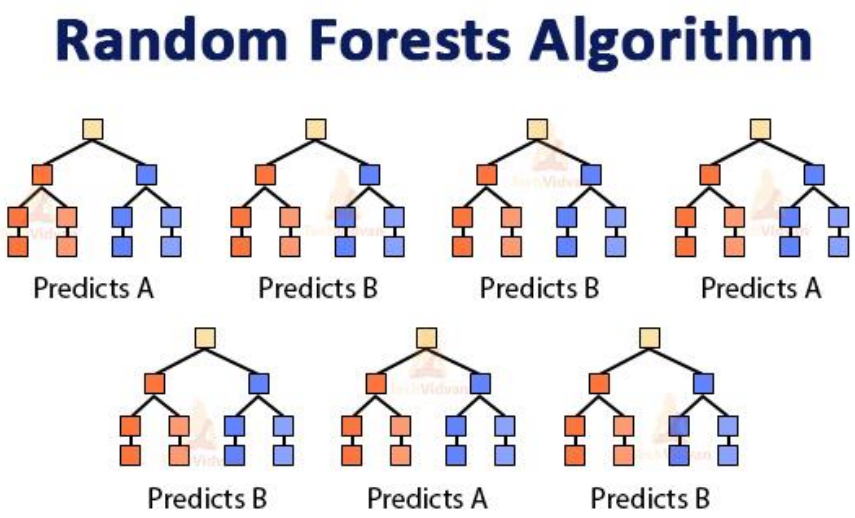

Fig 4.11(Random Forests Algorithm)

In the above image, there are two classes labeled as A and B. In this random forest consisting of 7 decision trees, 3 have voted for class A and 4 voted for class B. As class B has received the maximum votes thus the model's prediction will be class B.

Chapter 5

Conclusion

Data science education is well into its formative stages of development; it is evolving into a self-supporting discipline and producing professionals with distinct and complementary skills relative to professionals in the computer, information, and statistical sciences. However, regardless of its potential eventual disciplinary status, the evidence points to robust growth of data science education that will indelibly shape the undergraduate students of the future. In fact, fueled by growing student interest and industry demand, data science education will likely become a staple of the undergraduate experience. There will be an increase in the number of students majoring, minoring, earning certificates, or just taking courses in data science as the value of data skills becomes even more widely recognized. The adoption of a general education requirement in data science for all undergraduates will endow future generations of students with the basic understanding of data science that they need to become responsible citizens. Continuing education programs such as data science boot camps, career accelerators, summer schools, and incubators will provide another stream of talent. This constitutes the emerging watershed of data science education that feeds multiple streams of generalists and specialists in society; citizens are empowered by their basic skills to examine, interpret, and draw value from data.

Today, the nation is in the formative phase of data science education, where educational organizations are pioneering their own programs, each with different approaches to depth, breadth, and curricular emphasis (e.g., business, computer science, engineering, information science, mathematics, social science, or statistics). It is too early to expect consensus to emerge on certain best practices of data science education. However, it is not too early to envision the possible forms that such practices might take. Nor is it too early to make recommendations that can help the data science education community develop strategic vision and practices. The following is a summary of the findings and recommendations discussed in the preceding four chapters of this report.

Finding 2.1: Data scientists today draw largely from extensions of the "analyst" of years past trained in traditional disciplines. As data science becomes an integral part of many industries and enriches research and development, there will be an increased demand for more holistic and more nuanced data science roles.

Finding 2.2: Data science programs that strive to meet the needs of their students will likely evolve to emphasize certain skills and capabilities. This will result in programs that prepare different types of data scientists.

Recommendation 2.1: Academic institutions should embrace data science as a vital new field that requires specifically tailored instruction delivered through majors and

minors in data science as well as the development of a cadre of faculty equipped to teach in this new field.

Recommendation 2.2: Academic institutions should provide and evolve a range of educational pathways to prepare students for an array of data science roles in the workplace.

Finding 2.3: A critical task in the education of future data scientists is to instill data acumen. This requires exposure to key concepts in data science, real-world data and problems that can reinforce the limitations of tools, and ethical considerations that permeate many applications. Key concepts involved in developing data acumen include the following:

- Mathematical foundations,
- Computational foundations,
- Statistical foundations,
- Data management and curation,
- Data description and visualization,
- Data modeling and assessment,
- Workflow and reproducibility,
- Communication and teamwork,
- Domain-specific considerations, and
- Ethical problem solving.

Recommendation 2.3: To prepare their graduates for this new data-driven era, academic institutions should encourage the development of a basic understanding of data science in all undergraduates.

Recommendation 2.4: Ethics is a topic that, given the nature of data science, students should learn and practice throughout their education. Academic institutions should ensure that ethics is woven into the data science curriculum from the beginning and throughout.

References

- https://www.techtarget.com/searchenterpriseai/definition/deep-learning-deep-neural-network
- https://www.techtarget.com/searchenterpriseai/definition/machine-learning-ML
- https://www.britannica.com/technology/artificial-intelligence/Methods-and-goals-in-AI
- https://intellipaat.com/blog/tutorial/data-science-tutorial/data-science-algorithms/
- https://techvidvan.com/tutorials/data-science-algorithms/
- https://www.castordoc.com/blog/top-5-challenges-of-data-scientists
- https://www.saagie.com/blog/what-is-data-science/
- https://data-flair.training/blogs/purpose-of-data-science/