Unit – 1: Introduction

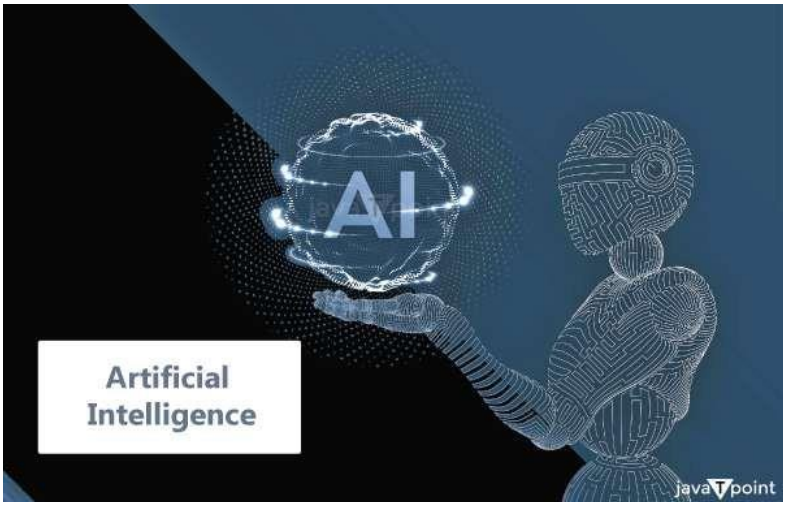
1. **What is artificial intelligence?**
2. **History and applications of AI**
3. **AI Problems**
4. **AI Techniques**
5. **Defining the Problem as a State Space Search,**
6. **Problem Characteristics**
7. **Production Systems**

What is Artificial Intelligence (AI)?

In today's world, technology is growing very fast, and we are getting in touch with different new technologies day by day.

Here, one of the booming technologies of computer science is Artificial Intelligence which is ready to create a new revolution in the world by making intelligent machines. The Artificial Intelligence is now all around us. It is currently working with a variety of subfields, ranging from general to specific, such as self-driving cars, playing chess, proving theorems, playing music, Painting, etc.

AI is one of the fascinating and universal fields of Computer science which has a great scope in future. AI holds a tendency to cause a machine to work as a human.



Artificial Intelligence is composed of two words **Artificial** and **Intelligence**, where Artificial defines *"man-made,"* and intelligence defines *"thinking power"*, hence AI means *"a man-made thinking power."*

So, we can define AI as:

"It is a branch of computer science by which we can create intelligent machines which can behave like a human, think like humans, and able to make decisions."

Artificial Intelligence exists when a machine can have human based skills such as learning, reasoning, and solving problems

With Artificial Intelligence you do not need to preprogram a machine to do some work, despite that you can create a machine with programmed algorithms which can work with own intelligence, and that is the awesomeness of AI.

Why Artificial Intelligence?

Before Learning about Artificial Intelligence, we should know that what is the importance of AI and why should we learn it. Following are some main reasons to learn about AI:

- With the help of AI, you can create such software or devices which can solve real-world problems very easily and with accuracy such as health issues, marketing, traffic issues, etc.
- With the help of AI, you can create your personal virtual Assistant, such as Cortana, Google Assistant, Siri, etc.
- With the help of AI, you can build such Robots which can work in an environment where survival of humans can be at risk.
- AI opens a path for other new technologies, new devices, and new Opportunities.

History of AI

- Throughout history, people have been intrigued by the idea of making non-living things smart. In ancient times, Greek stories mentioned gods creating clever machines, and in Egypt, engineers made statues move. Thinkers like **Aristotle and Ramon Llull** laid the groundwork for AI by describing how human thinking works using symbols.
- In the late 1800s and early 1900s, modern computing started to take shape. Charles Babbage and Ada Lovelace designed machines that could be programmed in the 1830s. In the 1940s, John Von Neumann came up with the idea of storing computer programs. At the same time, Warren McCulloch and Walter Pitts started building the basics of neural networks.
- The **1950s** brought us modern computers, letting scientists dig into machine intelligence. **Alan Turing's** Turing test became a big deal in computer smarts. The term "**artificial intelligence" was first used in a 1956 Dartmouth College meeting,** where they introduced the first AI program, the Logic Theorist.
- The following years had good times and bad times for AI, called "**AI Winters**." In the 1970s and 1980s, we hit limits with computer power and complexity. But in the late 1990s, things got exciting again. Computers were faster, and there was more data. **IBM's Deep Blue** beating chess champion Garry Kasparov in 1997 was a big moment.
- The 2000s started a new era with machine learning, language processing, and computer vision. This led to cool new products and services. The 2010s saw AI take off with things

like voice assistants and self-driving cars. Generative AI, which makes creative stuff, also started getting big.
- In the 2020s, generative AI like **ChatGPT-3** and **Google's Bard** grabbed everyone's attention. These models can create all sorts of new things when you give them a prompt, like essays or art. But remember, this tech is still new, and there are things to fix, like making sure it doesn't make things up.

Advantages of Artificial Intelligence

Following are some main advantages of Artificial Intelligence:

- **High Accuracy with less errors:** AI machines or systems are prone to less errors and high accuracy as it takes decisions as per pre-experience or information.
- **High-Speed:** AI systems can be of very high-speed and fast-decision making, because of that AI systems can beat a chess champion in the Chess game.
- **High reliability:** AI machines are highly reliable and can perform the same action multiple times with high accuracy.
- **Useful for risky areas:** AI machines can be helpful in situations such as defusing a bomb, exploring the ocean floor, where to employ a human can be risky.
- **Digital Assistant:** AI can be very useful to provide digital assistant to the users such as AI technology is currently used by various E-commerce websites to show the products as per customer requirement.
- **Useful as a public utility:** AI can be very useful for public utilities such as a self-driving car which can make our journey safer and hassle-free, facial recognition for security purpose, Natural language processing to communicate with the human in human-language, etc.
- **Enhanced Security:** AI can be very helpful in enhancing security, as It can detect and respond to cyber threats in real time, helping companies protect their data and systems.
- **Aid in Research:** AI is very helpful in the research field as it assists researchers by processing and analyzing large datasets, accelerating discoveries in fields such as astronomy, genomics, and materials science.

Disadvantages of Artificial Intelligence

Every technology has some disadvantages, and the same goes for Artificial intelligence. Being so advantageous technology still, it has some disadvantages which we need to keep in our mind while creating an AI system. Following are the disadvantages of AI:

- **High Cost:** The hardware and software requirement of AI is very costly as it requires lots of maintenance to meet current world requirements.
- **Can't think out of the box:** Even we are making smarter machines with AI, but still they cannot work out of the box, as the robot will only do that work for which they are trained, or programmed.

- **No feelings and emotions:** AI machines can be an outstanding performer, but still it does not have the feeling so it cannot make any kind of emotional attachment with human, and may sometime be harmful for users if the proper care is not taken.
- **Increase dependency on machines:** With the increment of technology, people are getting more dependent on devices and hence they are losing their mental capabilities.
- **No Original Creativity:** As humans are so creative and can imagine some new ideas but still AI machines cannot beat this power of human intelligence and cannot be creative and imaginative.
- **Complexity:** Making and keeping AI systems can be very complicated and need a lot of knowledge. This can make it hard for some groups or people to use them.
- **Job Concerns:** As AI gets better, it might take away not just basic jobs but also some skilled ones. This worries people about losing jobs in different fields.

Challenges of AI

Artificial Intelligence offers incredible advantages, but it also presents some challenges that need to be addressed:

- **Doing the Right Thing:** AI should make the right choices, but sometimes it doesn't. It can make mistakes or do things that aren't fair. We need to teach AI to be better at making good choices.
- **Government and AI:** Sometimes, governments use AI to keep an eye on people. This can be a problem for our freedom. We need to make sure they use AI in a good way.
- **Bias in AI:** AI can sometimes be a bit unfair, especially when it comes to recognizing people's faces. This can cause problems, especially for people who aren't like the majority.
- **AI and Social Media:** What you see on social media is often decided by AI. But sometimes, AI shows things that aren't true or are kind of mean. We need to make sure AI shows the right stuff.
- **Legal and Regulatory Challenges:** The rapid evolution of AI has outpaced the development of comprehensive laws and regulations, leading to uncertainty about issues like liability and responsibility.

Applications of AI

Artificial Intelligence has various applications in today's society. It is becoming essential for today's time because it can solve complex problems with an efficient way in multiple industries, such as Healthcare, entertainment, finance, education, etc. AI is making our daily life more comfortable and fast.Following are some sectors which have the application of Artificial Intelligence:

1. AI in Astronomy

- **Automated Celestial Object Identification:** AI systems can automatically identify and classify celestial objects in astronomical images, aiding in discovering new stars, galaxies, and other cosmic phenomena. In simple words, AI can spot and sort out things in space by looking at pictures. It's like having a cosmic detective that finds new stars, galaxies, and other mysterious objects without human help.
- **Exoplanet Hunting:** AI helps astronomers find planets outside our solar system by looking at lots of data. It can notice tiny changes in the light from stars, which tell us there might be planets around them, such as those caused by exoplanet transits.
- **Analyzing Space Information:** AI plays a crucial role in the study of space. It assists scientists by carefully examining vast amounts of complex data gathered from space observations. This helps astronomers uncover sophisticated patterns, unusual phenomena, and connections that might be otherwise very difficult to notice. Essentially, AI acts as a dedicated assistant, sifting through the cosmic data haystack to find the valuable needles of knowledge.
- **Watching Space Events in Real-time:** AI-powered tools can keep a constant eye on the night sky, looking out for sudden happenings like exploding stars (supernovae) or bursts of powerful gamma rays. This allows scientists to quickly study these events in more detail when they occur.
- **Making Telescopes Smarter:** AI is like a brain for telescopes. It helps them work better by changing their settings on the fly. For example, if the weather gets cloudy or if scientists want to study something specific in space, AI can adjust the telescope to get the best results. It's like having a telescope that can think and adapt to the situation.

2. AI in Healthcare

- **Helping Doctors See Inside the Body Better:** AI is like a super helper for doctors when they look at pictures of the inside of a patient's body, like X-rays or MRIs. It uses smart algorithms to find things like problems, tumors, or broken bones very accurately. This means doctors can figure out what's going on faster and more accurately, which is great for patients and for better diagnosis.
- **Detecting Health Problems Early:** AI acts as a health detective. It looks at your health information to find out if you might get certain diseases in the future. When it sees a high risk, doctors can step in early to help you stay healthy. This is really important for conditions like diabetes and heart problems because catching them at this time means better treatment and less trouble for the patient.
- **Developing Medications Quickly and Cost-Effectively:** AI acts like a super scientist in the lab. It uses certain algorithms to predict how different chemicals can fight diseases. This helps us make new medicines much quicker and at a low cost. So people can get the treatments they need sooner, and it doesn't cost as much money to manufacture them.
- **Personalized Treatment Plans:** AI looks at your health information, like your genes, what happened to you before, and how you've responded to treatments. Then, it makes a special plan just for you. This means your treatment works better and doesn't give you as many problems. It's like having a personalized health coach, which helps in avoiding complications caused by improperly prescribed medicine.
- **Managing Hospital Functions and Resources:** AI acts like a manager for hospitals. It helps with things like when patients come in, where to put resources like doctors and supplies, and how to make sure everything runs well. It can even guess how many patients might come in ahead of time, so hospitals use their staff and resources in the best way possible.

3. AI in Gaming

- **Smart Game Characters:** AI is like the brains behind game characters that aren't controlled by players. They make these characters, called NPCs, act more like real people or clever enemies. They can learn from what players do and change their behavior, which makes games more exciting and lifelike. Imagine playing a game where the bad guys learn and adapt to your moves - that's what AI does.
- **Creating Game Worlds with AI:** AI can make parts of video games all on its own. It can create levels, maps, and places to explore without people having to make them by hand. This means games can have bigger and more interesting worlds because AI does a lot of the work, kind of like a game world builder. It helps game developers, too.
- **Making Games Look and Feel Real:** AI helps to make games look and act more like the real world. They create graphics that look just like the things we see, and they make how things move in games feel realistic, like in real life. They even guess what players might do next so the game looks smooth and natural.

4. AI in Finance

- **Identifying and Prevention of Fraud:** AI keeps an eye on bank transactions all the time. They act like super detectives who can spot strange things happening with money, like someone using a credit card in a weird way. When they see something fishy, they raise the alarm and help the bank stop bad people from stealing money. This happens really fast, without needing people to check every transaction.
- **Automated Trading:** AI helps a skilled trader who works automatically. It uses various algorithms to swiftly buy and sell stocks while analyzing all the market information. This boosts trading strategies, making investments more efficient and profitable.
- **Risk Control:** AI helps in examining lots of data to check how risky something is, like giving out loans or making investments. It looks at things like whether someone can pay back a loan or how safe an investment is. This helps banks and investment firms make smarter choices so they don't lose money and can help others save and grow their money.

5. AI in Data Security

- **Anamoly Detection:** AI works as a digital detective. It looks at big piles of data and watches for anything strange or out of the ordinary, like someone sneaking into a digital vault or trying to steal secrets. When it sees something fishy, it raises the alarm, helping to keep important data safe from cyber-attacks.
- **Predicting Threats:** AI looks at past troubles and keeps an eye on new dangers that are popping up. By doing this, it can predict what bad things might happen in the future, like a security breach or a cyberattack. This way, companies can get ready in advance to protect their important data, sort of like putting up a strong fortress before any attack happens.
- **Automated Safety Response:** AI acts like a digital guardian that can respond when there's trouble. If it sees something bad happening, like a cyberattack, it can automatically take action. It might isolate the part that's under attack. This way, it keeps your important stuff safe in the digital world.

6. AI in Social Media

- **Smart Suggestions:** AI helps as a guide on social media. It watches what you like and what you do, and then it suggests things you might enjoy, like posts, videos, or ads. It acts as someone who knows your tastes and shows you stuff you're really into, making your social media experience more enjoyable and personalized.
- **Virtual Assistants and Chatbots:** AI chatbots and virtual assistants act as digital helpers on social media. They're quick to respond and can talk to you just like a real person. They answer your questions, share information, and even help with problems. It's like having an assistant available 24/7, making your social media experience smoother and more helpful.
- **Sentiment Analysis:** AI can figure out how people feel on social media. It looks at what they say in comments and posts and decides if it's a happy, sad, or neutral kind of message. This helps companies understand what people think so they can react in the right way. It's like having a mood gauge for the internet so businesses can make their customers happier.
- **Trend Analysis:** AI keeps track of all the chats and what's popular right now. This helps companies and regular folks understand what everyone's thinking and talking about. It acts as a social media news reporter that keeps customers in the loop about what's hot and what people are buzzing about.

7. AI in Travel & Transport

- **Optimization of Route:** AI plays a crucial role in optimizing travel routes, be it for parcel deliveries, public transportation, or personal trips. It efficiently calculates the swiftest and most economical paths from one point to another point, resulting in reduced travel time, minimized fuel consumption, and cost savings. Essentially, it serves as a pocket-sized travel advisor, enhancing the speed and budget-friendliness of your journeys.
- **Smart Security Screening:** AI helps in keeping traveling safely. It uses special skills to scan bags and people quickly. It can spot things that might be dangerous and make

security checks faster and smoother. This means you can fly knowing that the airport is working hard to keep you safe without making your travel a hassle.

- **Chatbots for Travel Support:** AI chatbots are like digital travel helpers. These chatbots are capable of aiding you in various tasks such as reserving tickets, suggesting interesting destinations to explore, and providing responses to your inquiries, much like an affable travel consultant. This elevates the convenience and pleasure of your travel adventures, as you can access assistance whenever it's required, even during late-night hours.
- **AI Prevents Breakdowns:** AI works like a fortune teller for machines like cars, planes, and roads. It predicts when they might get sick and need fixing. This way, we can fix them before they break down and cause problems. It keeps everything running smoothly, making travel safer and saving a lot of time and money.

8. AI in Automotive Industry

- **Self-Driving Cars:** AI is like the brain of self-driving cars. It looks at what's happening around the car using various sensors and decides what the car should do, like turning or stopping. It's like having a super-smart driver that doesn't need a person. This makes cars drive on their own, making travel more convenient and safer because there's no need for a human to steer.
- **Advanced Driver Assistance Systems (ADAS):** AI adds extra smarts to your car to keep you safe. It possesses the capability to autonomously adjust your vehicle's speed while on the highway, assist in maintaining your lane, and swiftly engage the brakes when detecting potential hazards. These intelligent functionalities function akin to a co-pilot, ensuring your safety by preventing accidents and ensuring your safe arrival at your intended destination.
- **Streamlining Production Processes:** AI watches over machines, checks if they're healthy, and makes sure they don't break. It also helps with ordering materials and makes sure everything is made just right. This makes things faster, cheaper, and better quality, like having a super factory manager.
- **Voice Recognition:** AI-driven voice recognition systems allow drivers to control various functions in their vehicles, such as navigation, music, and communication, using natural language.

9. AI in Robotics:

- **Self-Moving Robots:** AI makes robots really smart at moving around on their own. It's like giving them a built-in GPS and a clever brain. They can figure out where to go and how to get there without bumping into things or needing a person to show them the way. This helps them do tasks like delivering packages or exploring places on their own, making them super independent.
- **Object Recognition and Manipulation:** AI gives robots sharp eyes and clever hands. It helps them see objects clearly and then pick them up and move them just right. This is super useful, especially in places like warehouses, where they can do things like sorting and packing items accurately.
- **Collaboration of Humans and Robots:** AI makes it possible for robots to be great team players with people. They can work alongside humans, helping out and learning from them. If a person does something, the robot can understand and follow their lead. This makes workplaces safer and more efficient, like having a trusty robot colleague who understands and supports you.

10. AI in Entertainment

- **Recommendation of Content:** AI looks at what customers have liked before, such as movies or music, and suggests new things that they might enjoy. It's like having a personal entertainment guide, making their experience more enjoyable by offering just what they like.
- **AI as a Creative Assistant:** AI acts as a creative sidekick for artists and creators. It can make music, art, and videos or help improve what they create. It's like having a helper that speeds up the creative process, making it easier to bring new ideas to life. This way, artists can focus more on their vision, and AI handles the technical bits.
- **Live Event and Performance Enhancements:** AI makes live events and performances even cooler. It can translate what people are saying in real time, add cool effects that blend with what's happening, and even predict what the audience will like. This makes

shows and events more exciting and enjoyable for everyone there. It's like having a magic touch that brings performances to life in new and amazing ways.

11. AI in Agriculture

- **Crop Observation and Control:** AI, with the help of various sensors, acts as a guardian for crops on the farm. It keeps an eye on them, making sure they're healthy and growing well. It tells farmers when it's the best time to plant, water, and harvest the most crops. It's like having a farm expert who ensures the fields are super productive so farmers can get the most out of their hard work.
- **Smart Farming for Efficiency:** AI makes farming super efficient. It helps farmers use just the right amount of things like fertilizer and pesticides, not too much and not too little. This means there's less waste, and the crops grow better. It's like having a precise chef in the field, making sure everything is just perfect for the plants to thrive and produce lots of food.
- **Automated Farming:** AI controls a number of machines like tractors and drones. These machines can plant seeds, remove weeds, and spray stuff on crops all by themselves. They do it super well and exactly as needed, like having expert farmers who never get tired and work perfectly, making farming easier and more efficient.
- **Monitoring Livestock:** AI uses special sensors and smart data analysis to make sure they're healthy and happy. If anything is wrong, it alerts the farmer. This way, the animals are well taken care of, and the farm can run smoothly. It's like having a watchful friend for the animals, making sure they're okay and the farm works better.

12. AI in E-commerce

- **Personalized Product Suggestions:** AI looks at what you've looked at and bought before and suggests things you might really like. It's like having a personal shopper who knows your style, making your online shopping more fun and helping you discover new things you might want to buy. Plus, it's great for the store because it helps them sell more, and as a customer, it saves your time.

- **Managing Inventory:** AI takes care of a store's shelves. It predicts how much of each product people will buy and automatically orders more when needed. In this manner, there exists an optimal balance of products, preventing excessive stock that ties up funds while also ensuring an adequate supply to prevent customers from leaving without making a purchase.
- **Dynamic Pricing:** Artificial intelligence dynamically adjusts pricing according to demand, market competition, and inventory levels, ensuring customers receive optimal value while enhancing the store's profitability.

13. AI in education:

- **Education Content Creation:** AI acts as a teaching assistant for educators. It helps them make things like quizzes, lesson plans, and study materials. This makes teaching easier and better because educators have more time for students, and the materials are top-notch. It's like having a super-efficient helper who does the paperwork, leaving teachers more time to inspire students.
- **Virtual Learning Assistants:** AI is there to answer questions, explain things, and offer help whenever students need it, day or night. This makes learning easier and more fun because students have someone to turn to whenever they're stuck. It also takes some pressure off teachers because AI can handle common questions, leaving more time for personalized teaching.
- **Automated Assessment and Instant Feedback:** AI acts like a super-speedy homework checker. It looks at your assignments and tests and gives you grades and feedback right away. This aids in gauging your progress and pinpointing areas for potential enhancement. Furthermore, it alleviates some of your teacher's grading responsibilities, allowing them to dedicate more time to teaching rather than paper evaluation.
- **Customized Learning Routes:** AI figures out what you're good at and where you might need extra help. Then, it gives you the right stuff to learn and the best way to learn it. This makes learning easier and more fun

AI Problems

We assure you that you will not find any difficulty while learning our AI tutorial. But if there any mistake, kindly post the problem in the contact form.

- **High Cost:** The hardware and software requirement of AI is very costly as it requires lots of maintenance to meet current world requirements.
- **Can't think out of the box:** Even we are making smarter machines with AI, but still they cannot work out of the box, as the robot will only do that work for which they are trained, or programmed.
- **No feelings and emotions:** AI machines can be an outstanding performer, but still it does not have the feeling so it cannot make any kind of emotional attachment with human, and may sometime be harmful for users if the proper care is not taken.
- **Increase dependency on machines:** With the increment of technology, people are getting more dependent on devices and hence they are losing their mental capabilities.
- **No Original Creativity:** As humans are so creative and can imagine some new ideas but still AI machines cannot beat this power of human intelligence and cannot be creative and imaginative.
- **Complexity:** Making and keeping AI systems can be very complicated and need a lot of knowledge. This can make it hard for some groups or people to use them.
- **Job Concerns:** As AI gets better, it might take away not just basic jobs but also some skilled ones. This worries people about losing jobs in different fields.
- **Doing the Right Thing:** AI should make the right choices, but sometimes it doesn't. It can make mistakes or do things that aren't fair. We need to teach AI to be better at making good choices.
- **Government and AI:** Sometimes, governments use AI to keep an eye on people. This can be a problem for our freedom. We need to make sure they use AI in a good way.
- **Bias in AI:** AI can sometimes be a bit unfair, especially when it comes to recognizing people's faces. This can cause problems, especially for people who aren't like the majority.
- **AI and Social Media:** What you see on social media is often decided by AI. But sometimes, AI shows things that aren't true or are kind of mean. We need to make sure AI shows the right stuff.
- **Legal and Regulatory Challenges:** The rapid evolution of AI has outpaced the development of comprehensive laws and regulations, leading to uncertainty about issues like liability and responsibility.
- **Transformers:** Google found a better way to train AI using lots of regular computers with special chips called GPUs. This discovery made transformers possible. Transformers help AI learn from data that doesn't have labels, like teaching a computer to understand language.
- **Hardware Improvements:** Companies like Nvidia improved the inner workings of these GPUs. They made them really good at handling the math AI needs to do. This teamwork between better hardware, smarter AI software, and computer data centers made AI a million times better! Nvidia is also working with companies that offer cloud computing to make it easier for others to use this powerful AI.

- **GPTs:** Before, if a company wanted to use AI, they had to start from scratch, which was expensive and time-consuming. Now, companies like OpenAI, Nvidia, Microsoft, and Google offer pre-trained AI models. These models can be fine-tuned for specific tasks at a lower cost and with less effort. It's like buying a ready-made cake and adding your own frosting instead of baking the whole cake from scratch. This helps companies use AI faster and with less risk.
- **AI in the Cloud:** Using AI can be tricky because it needs lots of data work. Big cloud companies like **AWS, Google, Microsoft, IBM**, and **Oracle** are making it easier. They're offering AI services that help with the hard parts, like getting data ready, building AI models, and putting them into apps.
- **Advanced AI for Everyone:** Some groups are making really smart AI models and sharing them. **OpenAI**, for example, has models that are good at chatting, understanding language, making images, and writing code. Nvidia is another, and they're not tied to one cloud company. Many others are making special AI models for different jobs and industries. It's like having a library of powerful tools for lots of different tasks.
- Any computer language such as C, C++, Java, Python, etc.(knowledge of Python will be an advantage)
- Knowledge of essential Mathematics such as derivatives, probability theory, etc.

Top Artificial Intelligence Techniques

Artificial intelligence has grown from something that was conceptualized in the future to one that is now revolutionizing several industries. Artificial intelligence is revolutionizing how organizations are run, how people interact with technology, and even how we view the world around us. This could be attributed to the fact that it is able to process and analyze massive volumes of data at previously unseen speeds. Sophisticated AI technologies form the backbone of this revolution, enabling robots to carry out jobs classically requiring human intelligence: learning, reasoning, solving problems, perceiving, and understanding language. The further AI makes its way into our lives, the more centrally important it becomes to understand the key technology propelling this innovation. These are the top AI technologies redefining the world and taking machine capabilities to their limits.

01. Natural Language Processing(NLP)

One of the most important and fast-developing approaches in the sphere of artificial intelligence is Natural Language Processing, focused on how computers and human language interact. NLP closes the gap between digital data processing and human communication by giving machines the ability to understand, interpret, and generate human languages in a meaningful and practical way. Probably one of the most exciting and significant fields of AI research and application is a cross-disciplinary discipline combining linguistics, computer science, and artificial intelligence in an attempt to solve the intricacies of human language.

Basically, NLP deals with some major activities such as language translation, production, and understanding. It draws support from many subfields like phonetics, concerned with the sounds; pragmatics, concerned with the meaning in context; semantics, concerned with meaning; and syntax, concerned with the structure of the language. NLP combines these elements to construct systems that can recognize voices, translate machines, conduct sentiment analysis, and chatbots in a comprehensive manner.

NLP encompasses a range of techniques for interpreting, understanding, and generating human language that are:

- **Language Understanding:** This has to do with how well robots can understand the structure and semantics of human language. Methods like dependency parsing, named entity recognition, and part-of-speech tagging aid in dissecting and examining sentences to comprehend their grammatical structure and word connections. For example, in sentiment analysis, natural language processing algorithms may discern, from word and context analysis, whether a text represents positive, negative, or neutral feelings.

- **Language Generation:**This alludes to robots' capacity to generate prose that resembles that of humans. Text generation-where the system can compose articles, tales, or even code depending on inputs-and text summarization-where the system generates succinct summaries of lengthy documents-are two techniques in this field. Recent developments in this area, such as the creation of massive language models like GPT-3, have greatly enhanced the coherence and quality of machine-generated text, making it more challenging to tell it apart from human writing.
- **Language Translation:** Translating text or voice across languages is the aim of machine translation, which is one of the most useful uses of natural language processing. Large dictionaries and rule-based systems played a major role in early machine translation techniques. However, by learning from enormous volumes of bilingual text data, neural

machine translation systems like Google Translate have grown significantly more accurate and fluent with the development of neural networks and deep learning.

02. Computer Vision(CV)

Within the field of artificial intelligence, computer vision is a potent and quickly developing technique that focuses on enabling machines to comprehend and make judgments based on visual data. This multidisciplinary area combines concepts from electrical engineering, neurology, and computer science to imitate the human visual system and enable computers to comprehend digital pictures and movies to a high degree. Applications for CV are many and range from basic tasks like handwritten digit recognition to sophisticated systems like autonomous driving.

Some of the various methods in computer vision are:

- **Classification of Image:** Sorting a picture into one of many specified classes is the goal of image classification, which is a basic problem in computer vision. In this procedure, a picture is fed into a machine learning model that has been trained on a sizable dataset of labeled images, such as a Convolutional Neural Network (CNN). In order to assign a label, the model takes characteristics out of the picture, such as edges, textures, and forms. Image classification is used in face identification, medical diagnosis, and product sorting in warehouses.
- **Identification of Object:** Object detection goes beyond simple categorization to include both the identification of things in a picture and the specific position of those objects. In object detection, methods like Single Shot MultiBox Detector (SSD), You Only Look Once (YOLO), and Region-based CNN (R-CNN) are frequently employed. These models are appropriate for real-time video analysis, driverless cars, and surveillance since they can process pictures in real time.
- **Segmenting Image:** Compared to object identification, image segmentation is a more detailed operation since it entails dividing a picture into several segments or areas, each of which represents a distinct object or component of an item. Semantic segmentation, which assigns a category to every pixel, and instance segmentation, which distinguishes between several objects belonging to the same class, are the two primary categories of picture segmentation. Image segmentation is essential for autonomous driving since it aids in comprehending road situations and is used in medical imaging to detect and evaluate anatomical features.

03. Machine Learning(ML)

A fundamental component of artificial intelligence, machine learning enables computers to learn from data, spot patterns, and make choices with the least amount of human input. By using statistical techniques, machine learning algorithms allow robots to gain experience and become more proficient at activities. This method is different from traditional programming, which codes specific instructions for each and every eventuality that might arise. Rather, machine learning models are highly effective in dynamic and complicated contexts because they learn from examples and adapt to new circumstances.

ML is broadly categorized into supervised learning, unsupervised learning, and reinforcement learning.

- **Supervised Learning:** Supervised learning involves training models using labeled data, where each input sample yields the proper output. This method is comparable to classroom instruction. Classification methods like Support Vector Machines (SVM) and Decision Trees, which divide inputs into discrete groups, and Linear Regression, which predicts a continuous output, are common algorithms. Applications include sentiment analysis, spam identification, and medical diagnosis frequently making use of supervised learning.
- **Unsupervised Learning:** In unsupervised learning, models are trained on unlabeled data, enabling the system to find underlying structures or hidden patterns. While Dimensionality Reduction techniques like Principal Component Analysis (PCA) reduce the number of variables in a dataset, clustering algorithms like K-means and hierarchical clustering group comparable data points together. Applications include anomaly detection, exploratory data analysis, and consumer segmentation.
- **Reinforcement Learning(RL):** Reinforcement learning (RL), which takes its cues from behavioral psychology, teaches agents to make decisions by rewarding good behavior and penalizing bad behavior. As it engages with its surroundings, the agent gains knowledge on how to optimize repeated gains. RL makes use of methods such as Policy Gradient Methods and Q-learning. Notable uses for it include robotics, autonomous driving, and gameplay (like AlphaGo), where systems need to learn the best strategies over time.

04. Deep Learning(DL)

A key method in artificial intelligence, deep learning is a subset of machine learning (ML) that focuses on modeling and solving complicated problems with artificial neural networks. Deep learning models can learn from vast amounts of data and make judgments by mimicking the functions of the human brain. These models have transformed artificial intelligence and made major strides possible in a number of areas, including speech recognition, computer vision, natural language processing, and more. Core concepts of deep learning are:

- **ANN:** The artificial neural network serves as the fundamental building block of deep learning models. An artificial neural network (ANN) is made up of layers of networked nodes, or neurons, where each node is a mathematical process. An ANN's main layer types are input layers, which are responsible for accepting data, hidden layers, which analyze data, and output layers, which generate the final prediction. To reduce prediction error, a weight is assigned to each link between nodes during training.
- **DNN:** Deep neural networks (ANNs) are ANNs with more than one hidden layer. DNNs are able to learn hierarchical representations of data, capturing complex patterns and relationships, thanks to the network's depth, or the number of layers. This capacity is

necessary to solve increasingly complicated problems that are beyond the scope of simpler models.

- **CNN:** CNNs are designed specifically to handle structured grid data, including picture data. They make use of fully linked, pooling, and convolutional layers. In order to extract features from the input data, convolutional layers apply filters; pooling layers then lower the dimensionality of these features; and fully connected layers then interpret the features for tasks involving classification or regression. The fields of object identification, image synthesis, and image recognition have all advanced greatly because of CNNs.
- **RNN:** RNNs are intended for time-series analysis and sequential data. Their architecture has loops that enable information to remain constant over time steps. The vanishing gradient issue is resolved by variations like Gated Recurrent Units (GRU) and Long Short-Term Memory (LSTM), which allow the model to capture long-term dependence. Speech recognition, time-series forecasting, and natural language processing (NLP) are three common applications for RNNs.

State Space Search in AI

•

An essential method in artificial intelligence is state space search, which looks for potential states and their transitions to solve issues. According to this method, the problem is modeled as a state space, with each state representing a possible configuration and transitions denoting actions or operations that change the state of the problem. Finding a route that meets predetermined requirements from an initial state to a goal state is the aim.

This article provides an in-depth exploration of state space search in artificial intelligence, detailing its principles, strategies, and applications, with a practical implementation using Breadth-First Search (BFS) to solve the 8-puzzle problem.

Understanding State Space Search

To locate a solution, ***state space search*** entails methodically going through every potential state for an issue. This approach can be used to solve a variety of AI issues, including pathfinding, solving puzzles, playing games, and more. The fundamental concept is to visualize the issue as a graph with nodes standing in for states and edges for transitions.

Important ideas consist of:

- **State:** A specific configuration of the problem.
- **Initial State:** The starting point of the search.
- **Goal State:** The desired end configuration.
- **Transition:** An action that changes one state to another.
- **Path:** A sequence of states connected by transitions.
- **Search Strategy:** The method used to explore the state space.

Principles and Features of State Space Search

The efficiency and effectiveness of state space search are heavily dependent on several principles and characteristics. Understanding these elements is crucial for selecting the right search strategy and optimizing the search process.

1. **Expansiveness**: The number of successors that each state can generate. This impacts how many new states are explored from a given state.
2. **Branching Factor**: The average number of successors in each state. It influences the width of the search tree and the overall complexity of the search.
3. **Depth**: The length from the initial state to the goal state in the search tree. Deeper search trees can increase the time required to find a solution.
4. **Completeness**: A search strategy is complete if it guarantees finding a solution, assuming one exists.
5. **Optimality**: A search strategy is optimal if it guarantees finding the best solution according to a specified criterion.
6. **Time Complexity**: The duration of the state space exploration. It is influenced by the branching factor and the depth of the search tree.
7. **Space Complexity**: The amount of memory required to carry out the search. This depends on the number of states that need to be stored in memory simultaneously.

Steps in State Space Search

The following steps are often involved in the state space search process:

Step 1: Define the State Space

Determine the collection of all potential states and their interchanging states. To do this, the problem must be modelled in a fashion that encompasses all pertinent configurations and actions.

Step 2: Pick a Search Strategy

Decide how to comb over the state space. Typical tactics consist of:

- Before going on to nodes at the following depth level, the **Breadth-First Search (BFS)** method investigates every node at the current depth level. Full and ideal for graphs without weights.
- **Depth-First Search (DFS) i**nvestigates a branch as far as it can go before turning around. less memory-intensive, although completeness and optimality are not assured.
- The best method for locating the lowest-cost solution is **Uniform Cost Search (UCS)**, which expands the least expensive node first.
- **Greedy Best-First Search e**xpands the node that seems to be closest to the objective using a heuristic.
- **A* Search Algorithm** assures completeness and optimality with an admissible heuristic by combining the cost to reach the node with a heuristic calculating the cost to the target.

Step 3: Start the Search

Add the initial state to the frontier (the collection of states to be investigated) by starting there.

Step 4: Extend the Nodes

Using the selected search technique, iteratively expands nodes from the frontier, producing successor states and appending them to the frontier. After each node has been expanded, determine whether it now corresponds to the desired state. If so, retrace your route to the objective and call off the hunt.

Step 5: Address State Repetition

Put in place safeguards to prevent revisiting the same state, including keeping track of the states you've been to.

Step 6: End the Search

The search comes to an end when the desired state is discovered or, in the event that no viable solution is identified, when every state has been investigated.

AI systems are able to tackle complicated issues in an organized and methodical manner by employing these methods to systematically explore the state space.

Problem characteristics:

Introduction

Artificial intelligence stands as a continuously advancing field at the forefront of technological progress. At its core, AI entails crafting algorithms and systems capable of emulating human intelligence, tackling intricate problems, and making informed decisions. In this piece, we will delve into the essence of problem characteristics within artificial intelligence, exploring their significant properties and the systematic approach to effectively address them.

Problems in Artificial Intelligence (AI) manifest in diverse forms, each presenting its own set of challenges and potential for innovation. From image recognition to natural language processing, AI problems exhibit distinct attributes that influence the methodologies and techniques employed to tackle them. In this article, we delve into the fundamental characteristics of AI problems, shedding light on what renders them captivating and substantial.

Key Issues in AI Characteristics

1. The Crux of AI Difficulties

artificial intelligence issues are particular animals. They habitually incorporate a level of complexity and unusual nature not ordinarily present in traditional programming. It is fundamental to grasp these characteristics to make AI arrangements that work.

2. intricacy

Artificial intelligence, taking everything into account, challenges are more troublesome than traditional PC occupations. The huge information volumes that artificial intelligence frameworks should deal with and the confounded calculations they use are the wellsprings of this intricacy.

3. Uncertainty

AI instead of dated computations, occasionally handles unclear and deficient data. man-made brainpower structures ought to use probabilistic reasoning to pursue assumptions and choices because of this weakness.

4. Flexibility

Artificial intelligence frameworks need to conform to new data and evolving environmental elements. Developers are tested by this powerful nature to plan versatile calculations that might develop and change over the long haul.

5. Objective-centered

Artificial intelligence calculations are made to achieve specific targets. These targets may be pretty much as fundamental as arranging information or as modern as facial acknowledgement or language interpretation.

Steps for the Problem Characteristics

The intricacy of AI issues necessitates a methodical approach. Here is a methodical approach to comprehending and resolving these issues:

1. Identifying the Issue

Clearly defining the issue is the first step. Just what problem are you attempting to solve? This might entail analysing big databases, identifying trends, or formulating forecasts. Solving an issue with clarity makes it easier.

2. Gathering & Preparing Data

AI is data-driven. Gather pertinent information and get it ready for analysis. This includes sanitising the data, dealing with null values, and maybe converting it into a format that AI systems can understand.

3. Selecting the Proper Algorithm

Different AI techniques are needed for different challenges. Neural networks, for instance, may perform better for image recognition tasks than decision trees for categorization tasks. Choosing the right algorithm is essential.

4. Getting the Model Ready

To do this, data must be fed into the algorithm so that it may learn from it. Iterative training necessitates continuous modification and improvement.

5. Assessment and Enhancement

Analyse the model's performance after training. Utilise measures like as recall, accuracy, and precision to assess the effectiveness of your AI. Optimise the model to perform better based on these evaluations.

6. Implementation and Tracking

The AI solution is implemented in a real-world setting after it has been optimised. It must be continuously observed to make sure it adjusts to new information and circumstances.

Artificial intelligence (AI) is primarily concerned with the search process, therefore selecting the optimal answer requires a technique.

Before choosing a suitable approach for a given problem, we must classify the problem according to the following attributes.

- Is it possible to break the problem down into manageable, easily solved subproblems?
- Are stages in a solution reversible or ignorable?
- Is the problem's universe predictable?
- Is a sound answer to the issue universal or specific?
- Is there a path or a state where the problem can be solved?
- How does knowledge fit into the artificial intelligence problem-solving process?
- Does a problem-solving task include interaction with people?

1. Can the problem be broken down into manageable, easily solved subproblems?

Separating an issue into more modest, more reasonable sub-issues can frequently make it simpler to tackle. By taking apart the main pressing concern into more modest parts, you can handle each part independently, possibly working on the general errand. This approach is especially helpful for complex issues that could appear to be overpowering when drawn nearer in general. Notwithstanding, whether an issue is decomposable into simple to-tackle sub-issues relies upon the idea of the actual issue methods to address them. This gradual methodology frequently prompts more clear bits of knowledge and more powerful arrangements.

(picture)

2. Can solution steps be ignored or undone?

A verified lemma in the Theorem Proving issue can be disregarded for the remainder of the procedure.

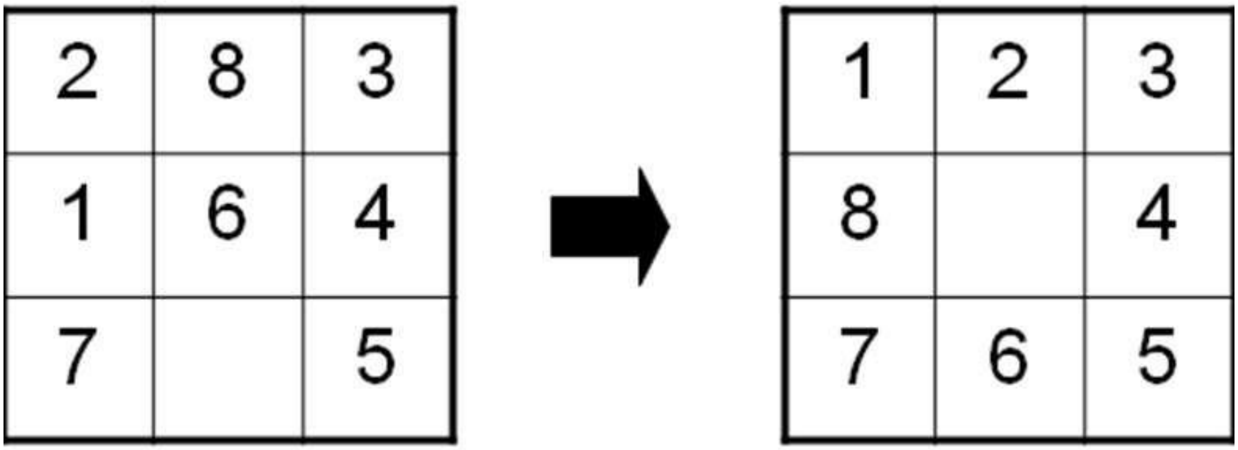
We refer to these issues as ignorable issues.

Moves in the 8-Puzzle can be reversed and retraced.

We refer to these issues as recoverable issues.

Justification:

The 8-Puzzle is viewed as a recoverable issue because any move made to change the riddle's setup can be scattered by switching that move. For example, if a tile is slid into an unfilled space, a similar tile can be moved back to its unique position. This property permits one to backtrack through the grouping of moves to any past state, empowering recuperation and re-investigation in various ways in the arrangement cycle. In this way, the riddle upholds reversible activities and complete recoverability.



Retracted is a move in chess play.

We refer to these issues as irreversible issues.

Justification:

Playing chess is viewed as an irrecoverable issue because, albeit individual moves can be withdrawn in relaxed play, each move on a very basic level changes the game's state in a manner that can't be completely scattered. For example, a withdrawn move doesn't delete the gathered upper hand or information acquired by one or the other player. Not at all like riddles where states can be definitively returned, has chess included developing procedures and places that make genuine recuperation to a past state deficient and unfeasible.

A straightforward, never-backtracking control structure can be used to handle insignificant issues. Backtracking is an effective way to tackle recoverable difficulties. Planning allows recoverable style solutions to address irrecoverable difficulties.

3. Is the problem's universe predictable?

We are unable to predict the movements of other players throughout their turns or the precise locations of all the cards when playing bridge. Planning may be used to create a series of operators for ***certain outcome issues*** that will inevitably result in a solution.

Example:

Think about a robot that is assigned the duty of cleaning a room. The following would be part of the planning process:

- Determining the Objective: A tidy room.

- Defined Operators: such as "move to a location," "pick up object," and "vacuum floor."
- Formulating a Scheme: a series of steps, like this: Go to the first corner.
- The floor is vacuumed.
- Proceed to the following location.
- Clear the clutter.
- Continue until the space is tidy.

A series of produced operators can only have a fair chance of leading to a solution for ***uncertain outcome issues.*** As the plan is implemented and the required input is received, revisions are made.

Example:

- Think of an autonomous vehicle navigating a metropolis to get to its destination, for instance:
- Determining the Objective: Get to a given address.
- Defined Operators: Activities such as "drive straight," "turn left," "turn right," "stop at a traffic light," "yield to pedestrians."
- Formulating a Scheme: a series of steps, as driving straight for two blocks.
- At the junction, make a left.
- Proceed straight ahead for three blocks.
- Make a right turn.

4. Is a good answer for the issue outright or relative?

Regarding the travelling Sales rep Issue, we need to attempt all ways to see it as the most limited one.

Any path problem can be solved using heuristics that suggest good paths to explore. For best-path problems, a much more exhaustive search will be performed.

5. Is the solution to the problem a state or a path?

In artificial intelligence, the answer for an issue can be either a state or a way, contingent upon the issue type. The fact that satisfies the objective circumstances makes for state-based issues, the arrangement a last expression. For way-based issues, the arrangement is a succession of states (way) from the underlying state to the objective state.

6. How does knowledge fit into the artificial intelligence problem-solving process?

Playing Chess:

Chess Playing Considering the chess problem once more, let's say you have infinite processing power. What knowledge, if any, would a perfect programme need? The answer is very little, just the rules governing legal moves and a basic control mechanism that carries out a suitable search procedure. Of course, a little more knowledge about things like sound strategy and tactics could greatly aid in limiting the search and expediting the program's execution.

Reading Newspaper:

Now think about the issue of sifting through the daily newspapers to determine which ones are conservative and which are liberal in an impending election. Once more, supposing infinite computational capacity, what level of expertise would a machine seeking to resolve this issue require? The response is substantial this time.

7. Does the task of solving a problem require human interaction?

Sometimes it is useful to program computers to solve problems in ways that the majority of people would not be able to understand. This is fine if the level of the interaction between the computer and its human users is problem-in solution-out.

But increasingly we are building programs that require intermediate interaction with people, both to provide additional input to the program and to provide additional reassurance to the user. The solitary problem, in which there is no intermediate communication and no demand for an explanation of the reasoning process.

The conversational problem, in which intermediate communication is to provide either additional assistance to the computer or additional information to the user.

Examples of AI Applications and Challenges Across Domains

1. Robotics

Problem: A delivery robot navigating a busy warehouse to locate and retrieve a specific item.

Characteristics:

- **Complexity:** Industrial storage is networked, in the middle of things, with obstacles, and other robots and people moving unpredictably. This robot must process the visual scene, plan the route effectively, and detect and avoid possible collisions.
- **Dynamism:** A combination of outside factors leads to change, which is a constant inside the warehouse. Unpredictable system failures or spontaneous tasks can make the robot change its means and decision-making at the moment of need.
- **Uncertainty:** Sensor data (such as images obtained from a camera) might be noisy, incomplete, and unstable. The robot could be handling decisions based on fragmented or formless pieces of information.

2. Natural Language Processing (NLP)

Problem: A sentiment analysis system in NLP classifying customer reviews as positive, negative, or neutral.

Characteristics:

- Subjectivity: Human language is nuanced. Sarcasm, irony, and figurative expressions can be difficult for machines to accurately interpret.

- Need for Context: Understanding sentiment may depend on cultural references, product-specific knowledge, or even the reviewer's prior interactions with the company.
- Ambiguity: A single word or phrase could have multiple meanings, affecting the overall sentiment of the text.

3. Computer Vision

Problem: A medical image recognition system in Computer Vision designed to detect tumours in X-rays or MRI scans.

Characteristics:

- Complexity: Medical images are highly detailed and can exhibit subtle variations. The system needs to distinguish between healthy tissue and potential abnormalities.
- Uncertainty: Images may contain noise or artefacts. The presence of a tumour might not be immediately obvious, requiring the system to handle ambiguity.
- Ethical Considerations: False positives or false negatives have serious consequences for patient health. Accuracy, transparency, and minimizing bias are crucial.

Production System in AI

In artificial intelligence (AI), a production system refers to a type of rule-based system that is designed to provide a structured approach to problem solving and decision-making. This framework is particularly influential in the realm of expert systems, where it simulates human decision-making processes using a set of predefined rules and facts.

Let's consider an example of **Expert System for Medical Diagnosis**.
Scenario: A patient comes to a healthcare facility with the following symptoms: fever, severe headache, sensitivity to light, and stiff neck.
Mediacal diagnosis operates in the following manner:

1. **Input**: A healthcare professional inputs the symptoms into MediDiagnose.
2. **Processing**:
   - **MediDiagnose** reviews its knowledge base for rules that match the given symptoms.
   - It identifies several potential conditions but recognizes a strong match for meningitis based on the combination of symptoms.
3. **Output**:
   - The system suggests that meningitis could be a possible diagnosis and recommends further tests to confirm, such as a lumbar puncture.
   - It also provides a list of other less likely conditions based on the symptoms for comprehensive differential diagnosis.

MediDiagnose uses its rule-based system to quickly filter through vast amounts of medical data to provide preliminary diagnoses. This assists doctors in focusing their investigative efforts more efficiently and potentially speeds up the process of reaching an accurate diagnosis.

Key Components of a Production System in AI

The key components of production system includes:

1. **Knowledge Base**: This is the core repository where all the rules and facts are stored. In AI, the knowledge base is critical as it contains the domain-specific information and the if-then rules that dictate how decisions are made or actions are taken.
2. **Inference Engine**: The inference engine is the mechanism that applies the rules to the known facts to derive new facts or to make decisions. It scans the rules and decides which ones are applicable based on the current facts in the working memory. It can operate in two modes:
   - **Forward Chaining (Data-driven)**: This method starts with the available data and uses the inference rules to extract more data until a goal is reached.
   - **Backward Chaining (Goal-driven)**: This approach starts with a list of goals and works backwards to determine what data is required to achieve those goals.
3. **Working Memory**: Sometimes referred to as the fact list, working memory holds the dynamic information that changes as the system operates. It represents the current state of knowledge, including facts that are initially known and those that are deduced throughout the operation of the system.
4. **Control Mechanism**: This governs the order in which rules are applied by the inference engine and manages the flow of the process. It ensures that the system responds appropriately to changes in the working memory and applies rules effectively to reach conclusions or solutions.

Types of Production Systems

Production systems in AI can be categorized based on how they handle and process knowledge. This categorization includes Rule-Based Systems, Procedural Systems, and Declarative Systems, each possessing unique characteristics and applications.

1. Rule-Based Systems

1. **Explanation of Rule-Based Reasoning**
   - Rule-based systems operate by applying a set of pre-defined rules to the given data to deduce new information or make decisions. These rules are generally in the form of conditional statements (if-then statements) that link conditions with actions or outcomes.
2. **Examples of Rule-Based Systems in AI**
   - **Diagnostic Systems**: Like medical diagnosis systems that infer diseases from symptoms.
   - **Fraud Detection Systems**: Used in banking and insurance, these systems analyze transaction patterns to identify potentially fraudulent activities.

2. Procedural Systems

1. **Description of Procedural Knowledge**
   - Procedural systems utilize knowledge that describes how to perform specific tasks. This knowledge is procedural in nature, meaning it focuses on the steps or procedures required to achieve certain goals or results.
2. **Applications of Procedural Systems**
   - **Manufacturing Control Systems**: Automate production processes by detailing step-by-step procedures to assemble parts or manage supply chains.
   - **Interactive Voice Response (IVR) Systems**: Guide users through a series of steps to resolve issues or provide information, commonly used in customer service.

3. Declarative Systems

1. **Understanding Declarative Knowledge**

- Declarative systems are based on facts and information about what something is, rather than how to do something. These systems store knowledge that can be queried to make decisions or solve problems.

2. **Instances of Declarative Systems in AI**
   - **Knowledge Bases in AI Assistants**: Power virtual assistants like Siri or Alexa, which retrieve information based on user queries.
   - **Configuration Systems**: Used in product customization, where the system decides on product specifications based on user preferences and declarative rules about product options.

Each type of production system offers different strengths and is suitable for various applications, from straightforward rule-based decision-making to complex systems requiring intricate procedural or declarative reasoning.

How Production Systems Function?

The operation of a production system in AI follows a cyclic pattern:

- **Match**: The inference engine checks which rules are triggered based on the current facts in the working memory.
- **Select**: From the triggered rules, the system (often through the control mechanism) selects one based on a set of criteria, such as specificity, recency, or priority.
- **Execute**: The selected rule is executed, which typically modifies the facts in the working memory, either by adding new facts, changing existing ones, or removing some.

Applications of Production Systems in AI

Production systems are used across various domains where decision-making can be encapsulated into clear, logical rules:

- **Expert Systems**: For diagnosing medical conditions, offering financial advice, or making environmental assessments.
- **Automated Planning**: Used in logistics to optimize routes and schedules based on current data and objectives.
- **Game AI**: Manages non-player character behavior and decision-making in complex game environments.

Unit 2. Search Techniques

- **Un-Informed Search,**
- **Best-First Search,**
- **DFS;**
- **Heuristic Search Techniques:**
- **Generate-And-Test, Hill Climbing**
- **Best-First Search,**
- **A*Algorithm**

1. Uninformed Search Techniques (Blind Search)

Breadth-First Search (BFS)

BFS is an uninformed search that explores all the nodes at the current depth before moving to the nodes at the next depth level.

- **Example**: Suppose you have a graph representing a maze, where each node represents a position in the maze and edges represent the possible paths.
  - Start from the initial position, explore all the possible positions at distance 1, then move to distance 2, and so on until you find the goal.
- **Properties**:
  - **Completeness**: BFS will always find the goal if it exists.
  - **Optimality**: BFS finds the shallowest goal node (the shortest path in an unweighted graph).
  - **Time complexity**: O(bd)O(b^d) where bb is the branching factor (average number of child nodes) and dd is the depth of the shallowest solution.
  - **Space complexity**: O(bd)O(b^d), since BFS stores all nodes at each depth level.

Depth-First Search (DFS)

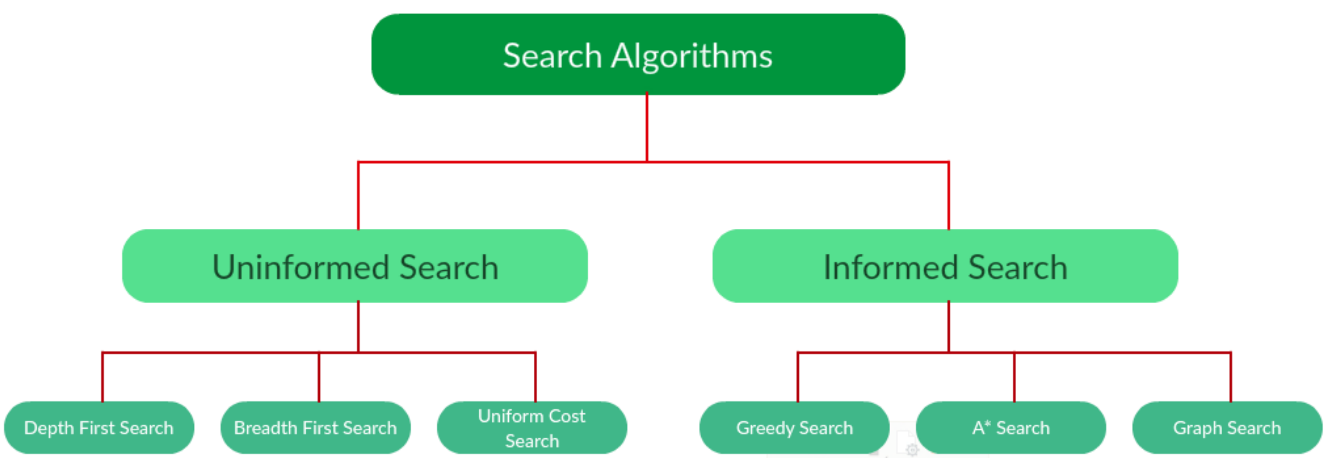
DFS explores as far as possible along a branch before backtracking.

- **Example**: In a maze-solving problem, DFS will follow one path as far as possible until it hits a dead end or finds the goal. If it reaches a dead end, it backtracks to explore other branches.
- **Properties**:
  - **Completeness**: DFS may fail to find the goal if there are infinite paths or loops.
  - **Optimality**: DFS does not guarantee finding the shortest path.
  - **Time complexity**: O(bm)O(b^m), where mm is the maximum depth of the search tree.
  - **Space complexity**: O(bm)O(bm), because DFS only needs to store the nodes along the current path.

Uniform Cost Search (UCS)

UCS expands the node with the least cost (where cost is typically the distance or resource consumption from the start node to the current node).

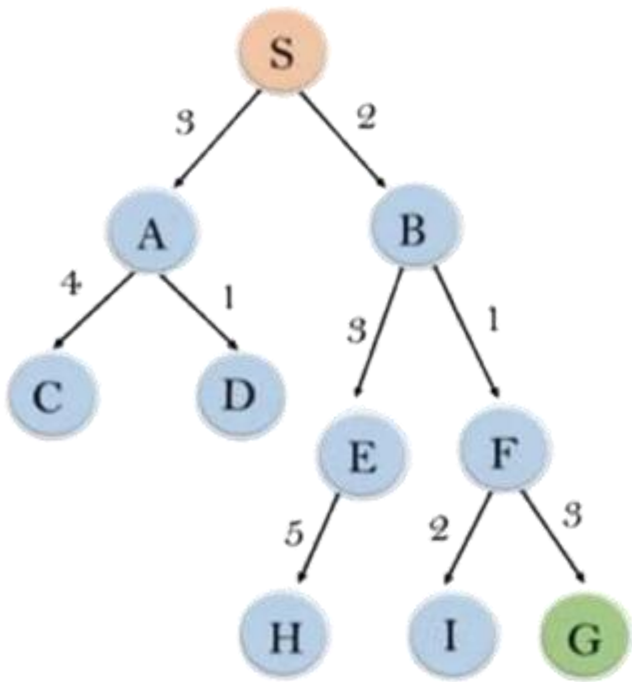
- **Example**: In a weighted graph, where nodes represent cities and edges represent roads with distances as weights, UCS will expand the closest city and continue until the goal city is reached.
- **Properties**:
  - **Completeness**: UCS is complete if the cost of every action is non-negative.
  - **Optimality**: UCS guarantees finding the least-cost solution.
  - **Time and Space complexity**: O(bC∗/ϵ)O(b^{C^*/\epsilon}), where C∗C^* is the cost of the optimal solution and ϵ\epsilon is the minimum edge cost.



2. **Best-First Search (Greedy Search)**

Best-First Search chooses the node that seems closest to the goal, according to a heuristic function. It uses a priority queue to always expand the node that has the lowest estimated cost.

- **Example**: In a navigation problem, the heuristic could be the straight-line distance between the current position and the goal. The algorithm will prioritize exploring nodes that are physically closer to the goal based on the heuristic.
- **Properties**:
  - **Completeness**: May not be complete, as it can get stuck in loops.
  - **Optimality**: Not optimal, as it may not consider the overall cost.
  - **Time complexity**: Depends on the heuristic.
  - **Space complexity**: Requires storing all generated nodes.



node	H (n)
A	12
B	4
C	7
D	3
E	8
F	2
H	4
I	9
S	13
G	0

3. **Heuristic Search Techniques**

Heuristic search techniques improve upon blind searches by using domain-specific knowledge to make more informed decisions about which paths to explore.

Generate-and-Test

This is a trial-and-error approach, where potential solutions are generated and tested to check if they satisfy the goal criteria.

- **Example**: Solving a puzzle like Sudoku. You generate a possible arrangement of numbers and then check if it satisfies the Sudoku rules. If it doesn't, you generate a new arrangement and repeat.
- **Properties**:
    - **Completeness**: Complete if the generation process can eventually generate all possible solutions.
    - **Optimality**: Not guaranteed to find the best solution.
    - **Time and space complexity**: Can be very inefficient depending on the problem size.

Hill Climbing

Hill Climbing is a local search algorithm that moves toward the solution based on a heuristic function. It only moves to better (higher-valued) neighbors.

- **Example**: Suppose you're climbing a hill blindfolded, and you can only feel the slope of the hill. You take small steps in the direction that increases your height until you reach the top (local maximum).
- **Properties**:
    - **Completeness**: May not be complete if it gets stuck in local maxima or plateaus.
    - **Optimality**: Not guaranteed; can get stuck at a local maximum, plateau, or ridge.
    - **Time and Space complexity**: O(b)O(b), since it only needs to store one current node at a time.

Best-First Search (with Heuristic)

In a heuristic version of Best-First Search, we use a heuristic function h(n)h(n) to prioritize nodes. It evaluates how close each node is to the goal.

- **Example**: In a maze, the heuristic might be the Euclidean distance from a position to the exit. Best-First Search will expand nodes that seem closest to the exit.
- **Properties**:
    - **Completeness**: May not be complete.
    - **Optimality**: Not guaranteed.
    - **Time and Space complexity**: Depends on the heuristic.

A* Algorithm

A* is a combination of Uniform Cost Search and Best-First Search, using both the actual cost g(n)g(n) from the start and a heuristic h(n)h(n) to the goal. It chooses the path that minimizes f(n)=g(n)+h(n)f(n) = g(n) + h(n).

- **Example**: In pathfinding for a robot, g(n)g(n) could be the distance already traveled, and h(n)h(n) could be the estimated distance to the goal. A* will choose paths that minimize both the traveled and remaining distances.
- **Properties**:
  - **Completeness**: A* is complete if the cost is non-negative.
  - **Optimality**: A* is optimal if the heuristic h(n)h(n) is admissible (never overestimates) and consistent.
  - **Time complexity**: Exponential in the worst case, but can be efficient with a good heuristic.
  - **Space complexity**: A* requires memory proportional to the number of nodes explored, which can grow large.

Summary of Key Properties

Algorithm	**Complete**	**Optimal**	**Time Complexity**	**Space Complexity**
BFS	Yes	Yes	O(bd)O(b^d)	O(bd)O(b^d)
DFS	No	No	O(bm)O(b^m)	O(bm)O(bm)
UCS	Yes	Yes	O(bC∗/ϵ)O(b^{C^*/\epsilon})	O(bC∗/ϵ)O(b^{C^*/\epsilon})
Greedy Best-First	No	No	Depends on heuristic	Depends on heuristic
Generate-and-Test	Yes	No	Inefficient	Inefficient
Hill Climbing	No	No	O(b)O(b)	O(b)O(b)
A* Algorithm	Yes	Yes	Exponential in worst case	Exponential in worst case

These search techniques are widely applied in artificial intelligence, game development, robotics, and optimization problems. Choosing the right algorithm depends on the problem's nature, constraints, and resources.

Unit-3 : Knowledge Representation

1. Types of Knowledge

2. Approach and techniques of knowledge representation

3. Forward Chaining

4. Backward chaining

What is knowledge representation?

Humans are best at understanding, reasoning, and interpreting knowledge. Human knows things, which is knowledge and as per their knowledge they perform various actions in the real world. **But how machines do all these things comes under knowledge representation and reasoning**. Hence we can describe Knowledge representation as following:

- Knowledge representation and reasoning (KR, KRR) is the part of Artificial intelligence which concerned with AI agents thinking and how thinking contributes to intelligent behavior of agents.
- It is responsible for representing information about the real world so that a computer can understand and can utilize this knowledge to solve the complex real world problems such as diagnosis a medical condition or communicating with humans in natural language.
- It is also a way which describes how we can represent knowledge in artificial intelligence. Knowledge representation is not just storing data into some database, but it also enables an intelligent machine to learn from that knowledge and experiences so that it can behave intelligently like a human.

What to Represent:

Following are the kind of knowledge which needs to be represented in AI systems:

- **Object:** All the facts about objects in our world domain. E.g., Guitars contains strings, trumpets are brass instruments.
- **Events:** Events are the actions which occur in our world.
- **Performance:** It describe behavior which involves knowledge about how to do things.
- **Meta-knowledge:** It is knowledge about what we know.
- **Facts:** Facts are the truths about the real world and what we represent.
- **Knowledge-Base:** The central component of the knowledge-based agents is the knowledge base. It is represented as KB. The Knowledgebase is a group of the Sentences (Here, sentences are used as a technical term and not identical with the English language).

Knowledge: Knowledge is awareness or familiarity gained by experiences of facts, data, and situations. Following are the types of knowledge in artificial intelligence:

Types of knowledge

Following are the various types of knowledge:

1. Declarative Knowledge:

- Declarative knowledge is to know about something.
- It includes concepts, facts, and objects.
- It is also called descriptive knowledge and expressed in declarativesentences.
- It is simpler than procedural language.

2. Procedural Knowledge

- It is also known as imperative knowledge.
- Procedural knowledge is a type of knowledge which is responsible for knowing how to do something.
- It can be directly applied to any task.
- It includes rules, strategies, procedures, agendas, etc.
- Procedural knowledge depends on the task on which it can be applied.

3. Meta-knowledge:

- Knowledge about the other types of knowledge is called Meta-knowledge.

4. Heuristic knowledge:

- Heuristic knowledge is representing knowledge of some experts in a filed or subject.

- Heuristic knowledge is rules of thumb based on previous experiences, awareness of approaches, and which are good to work but not guaranteed.

5. Structural knowledge:

- Structural knowledge is basic knowledge to problem-solving.
- It describes relationships between various concepts such as kind of, part of, and grouping of something.
- It describes the relationship that exists between concepts or objects.

Approaches to knowledge representation:

There are mainly four approaches to knowledge representation, which are givenbelow:

1. Simple relational knowledge:

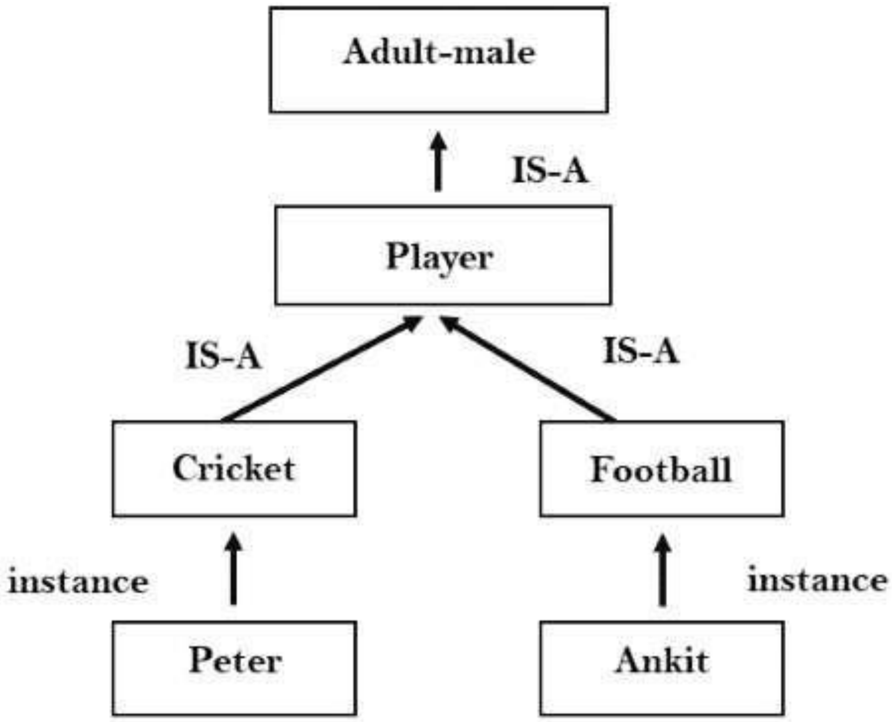
- It is the simplest way of storing facts which uses the relational method, and each fact about a set of the object is set out systematically in columns.
- This approach of knowledge representation is famous in database systems where the relationship between different entities is represented.
- This approach has little opportunity for inference.

Example: The following is the simple relational knowledge representation.

Player1	65	23
Player2	58	18
Player3	75	24

2. Inheritable knowledge:

- In the inheritable knowledge approach, all data must be stored into a hierarchy of classes.
- All classes should be arranged in a generalized form or a hierarchal manner.
- In this approach, we apply inheritance property.
- Elements inherit values from other members of a class.
- This approach contains inheritable knowledge which shows a relation between instance and class, and it is called instance relation.
- Every individual frame can represent the collection of attributes and its value.
- In this approach, objects and values are represented in Boxed nodes.
- We use Arrows which point from objects to their values.
- **Example:**



3. Inferential knowledge:

- Inferential knowledge approach represents knowledge in the form of formal logics.
- This approach can be used to derive more facts.
- It guaranteed correctness.
- **Example:** Let's suppose there are two statements:
    - Marcus is a man
    - All men are mortal Then it can represent as;

      man(Marcus)

      ∀x = man (x) ----------- > mortal (x)s

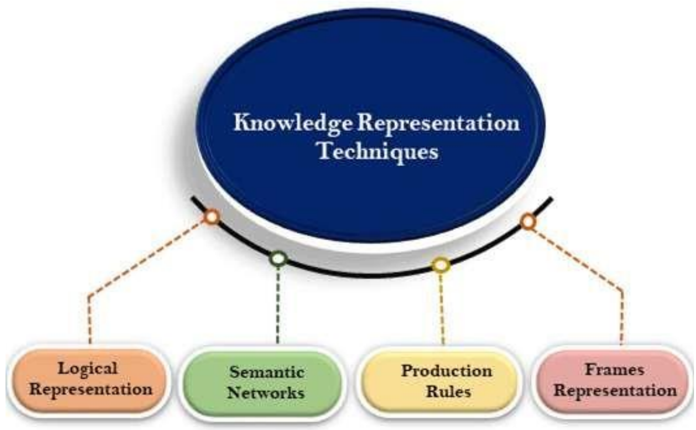
4. Procedural knowledge:

- Procedural knowledge approach uses small programs and codes which describes how to do specific things, and how to proceed.
- In this approach, one important rule is used which is **If-Then rule**.
- In this knowledge,we can use various coding languages such as **LISP language** and **Prolog language**.
- We can easily represent heuristic or domain-specific knowledge using this approach.
- But it is not necessary that we can represent all cases in this approach.

Techniques of knowledge representation

There are mainly four ways of knowledge representation which are given as follows:

1. Logical Representation
2. Semantic Network Representation
3. Frame Representation
4. Production Rules



1. Logical Representation

Logical representation is a language with some concrete rules which deals with propositions and has no ambiguity in representation. Logical representation means drawing a conclusion based on various conditions. This representation lays down some important communication rules. It consists of precisely defined syntax and semantics which supports the sound inference. Each sentence can be translated into logics using syntax and semantics.

Syntax:

- Syntaxes are the rules which decide how we can construct legal sentences in the logic.
- It determines which symbol we can use in knowledge representation.
- How to write those symbols.

Semantics:

- Semantics are the rules by which we can interpret the sentence in the logic.
- Semantic also involves assigning a meaning to each sentence.

Logical representation can be categorised into mainly two logics:

1. Propositional Logics

2. Predicate logics

Note: We will discuss Prepositional Logics and Predicate logics in later chapters.

Advantages of logical representation:

1. Logical representation enables us to do logical reasoning.
2. Logical representation is the basis for the programming languages.

Disadvantages of logical Representation:

1. Logical representations have some restrictions and are challenging to work with.
2. Logical representation technique may not be very natural, and inference may not be so efficient.

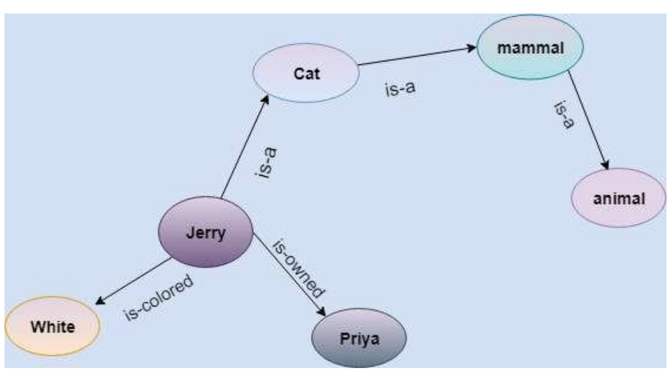
2. Semantic Network Representation

Semantic networks are alternative of predicate logic for knowledge representation. In Semantic networks, we can represent our knowledge in the form of graphical networks. This network consists of nodes representing objects and arcs which describe the relationship between those objects. Semantic networks can categorize the object in different forms and can also link those objects. Semantic networks are easy to understand and can be easily extended.

Example: Following are some statements which we need to represent in the form of nodes and arcs.

Statements:

1. Jerry is a cat.
2. Jerry is a mammal
3. Jerry is owned by Priya.
4. Jerry is brown colored.
5. All Mammals are animal.



In the above diagram, we have represented the different type of knowledge in the form of nodes and arcs. Each object is connected with another object by some relation.

Drawbacks in Semantic representation:

1. Semantic networks take more computational time at runtime as we need to traverse the complete network tree to answer some questions. It might be possible in the worst case scenario that after traversing the entire tree, we find that the solution does not exist in this network.
2. Semantic networks try to model human-like memory (Which has 1015 neurons and links) to store the information, but in practice, it is not possible to build such a vast semantic network.
3. These types of representations are inadequate as they do not have any equivalent quantifier, e.g., for all, for some, none, etc.
4. Semantic networks do not have any standard definition for the link names.
5. These networks are not intelligent and depend on the creator of the system.

Advantages of Semantic network:

1. Semantic networks are a natural representation of knowledge.
2. Semantic networks convey meaning in a transparent manner.
3. These networks are simple and easily understandable.

3. Frame Representation

A frame is a record like structure which consists of a collection of attributes and its values to describe an entity in the world. Frames are the AI data structure which divides knowledge into substructures by representing stereotypes situations. It consists of a collection of slots and slot values. These slots may be of any type and sizes. Slots have names and values which are called facets.

Facets: The various aspects of a slot is known as **Facets**. Facets are features of frames which enable us to put constraints on the frames. Example: IF-NEEDED facts are called when data of any particular slot is needed. A frame may consist of any number of slots, and a slot may include any number of facets and facets may have any number of values. A frame is also known as **slot-filter knowledge representation** in artificial intelligence.

Frames are derived from semantic networks and later evolved into our modern-day classes and objects. A single frame is not much useful. Frames system consist of a collection of frames which are connected. In the frame, knowledge about an object or event can be stored together in the knowledge base. The frame is a type of technology which is widely used in various applications including Natural language processing and machine visions.

Example: 1

Let's take an example of a frame for a book

Title	Artificial Intelligence
Genre	Computer Science
Author	Peter Norvig
Edition	Third Edition
Year	1996
Page	1152

Advantages of frame representation:

1. The frame knowledge representation makes the programming easier by grouping the related data.
2. The frame representation is comparably flexible and used by many applications in AI.
3. It is very easy to add slots for new attribute and relations.
4. It is easy to include default data and to search for missing values.
5. Frame representation is easy to understand and visualize.

Disadvantages of frame representation:

1. In frame system inference mechanism is not be easily processed.
2. Inference mechanism cannot be smoothly proceeded by frame representation.
3. Frame representation has a much generalized approach.

4. Production Rules

Production rules system consist of (**condition, action**) pairs which mean, "If condition then action". It has mainly three parts:

- The set of production rules
- Working Memory
- The recognize-act-cycle

In production rules agent checks for the condition and if the condition exists then production rule fires and corresponding action is carried out. The condition part of the rule determines which rule may be applied to a problem. And the action part carries out the associated problem-solving steps. This complete process is called a recognize-act cycle.

The working memory contains the description of the current state of problems-solving and rule can write knowledge to the working memory. This knowledge match and may fire other rules.

If there is a new situation (state) generates, then multiple production rules will be fired together, this is called conflict set. In this situation, the agent needs to select a rule from these sets, and it is called a conflict resolution.

Example:

- **IF (at bus stop AND bus arrives) THEN action (get into the bus)**
- **IF (on the bus AND paid AND empty seat) THEN action (sit down).**
- **IF (on bus AND unpaid) THEN action (pay charges).**
- **IF (bus arrives at destination) THEN action (get down from the bus).**

Advantages of Production rule:

1. The production rules are expressed in natural language.
2. The production rules are highly modular, so we can easily remove, add or modify an individual rule.

Disadvantages of Production rule:

1. Production rule system does not exhibit any learning capabilities, as it does not store the result of the problem for the future uses.
2. During the execution of the program, many rules may be active hence rule-based production systems are inefficient.

Forward Chaining and backward chaining in AI

In artificial intelligence, forward and backward chaining is one of the important topics, but before understanding forward and backward chaining lets first understand that from where these two terms came.

Inference engine:

The inference engine is the component of the intelligent system in artificial intelligence, which applies logical rules to the knowledge base to infer new information from known facts. The first inference engine was part of the expert system. Inference engine commonly proceeds in two modes, which are:

1. **Forward chaining**
2. **Backward chaining**

A. Forward Chaining

Forward chaining is also known as a forward deduction or forward reasoning method when using an inference engine. Forward chaining is a form of reasoning which start with atomic sentences in the knowledge base and applies inference rules (Modus Ponens) in the forward direction to extract more data until a goal is reached.

The Forward-chaining algorithm starts from known facts, triggers all rules whose premises are satisfied, and add their conclusion to the known facts. This process repeats until the problem is solved.

Properties of Forward-Chaining:

- It is a down-up approach, as it moves from bottom to top.
- It is a process of making a conclusion based on known facts or data, by starting from the initial state and reaches the goal state.
- Forward-chaining approach is also called as data-driven as we reach to the goal using available data.
- Forward -chaining approach is commonly used in the expert system, such as CLIPS, business, and production rule systems.

Consider the following famous example which we will use in both approaches:

Example:

"As per the law, it is a crime for an American to sell weapons to hostile nations. Country A, an enemy of America, has some missiles, and all the missiles were sold to it by Robert, who is an American citizen."

Prove that **"Robert is criminal."**

To solve the above problem, first, we will convert all the above facts into first-order definite clauses, and then we will use a forward-chaining algorithm to reach the goal.

Facts Conversion into FOL:

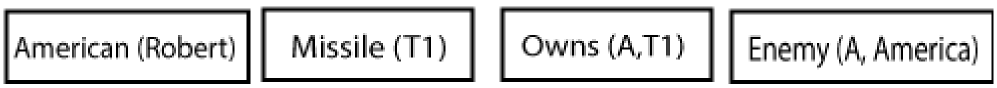
- It is a crime for an American to sell weapons to hostile nations. (Let's say p, q, and r are variables)
  American (p) ∧ weapon(q) ∧ sells (p, q, r) ∧ hostile(r) → Criminal(p) ...(1)
- Country A has some missiles. **?p Owns(A, p) ∧ Missile(p)**. It can be written in two definite clauses by using Existential Instantiation, introducing new Constant T1.
  Owns(A, T1) .... (2)
  Missile(T1)...................... (3)
- All of the missiles were sold to country A by Robert.
  ?p Missiles(p) ∧ Owns (A, p) → Sells (Robert, p, A)...................... (4)
- Missiles are weapons.
  Missile(p) → Weapons (p) .... (5)
- Enemy of America is known as hostile.
  Enemy(p, America) →Hostile(p) ....(6)
- Country A is an enemy of America.
  Enemy (A, America)..................... (7)

- Robert is American **American(Robert).............................(8)**

Forward chaining proof:

Step-1:

In the first step we will start with the known facts and will choose the sentences which do not have implications, such as: **American(Robert), Enemy(A, America), Owns(A, T1), and Missile(T1)**. All these facts will be represented as below.



Step-2:

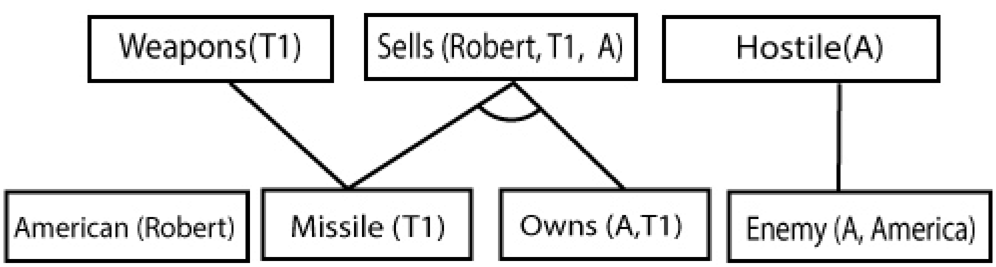
At the second step, we will see those facts which infer from available facts and with satisfied premises.

Rule-(1) does not satisfy premises, so it will not be added in the first iteration. Rule-

(2) and (3) are already added.

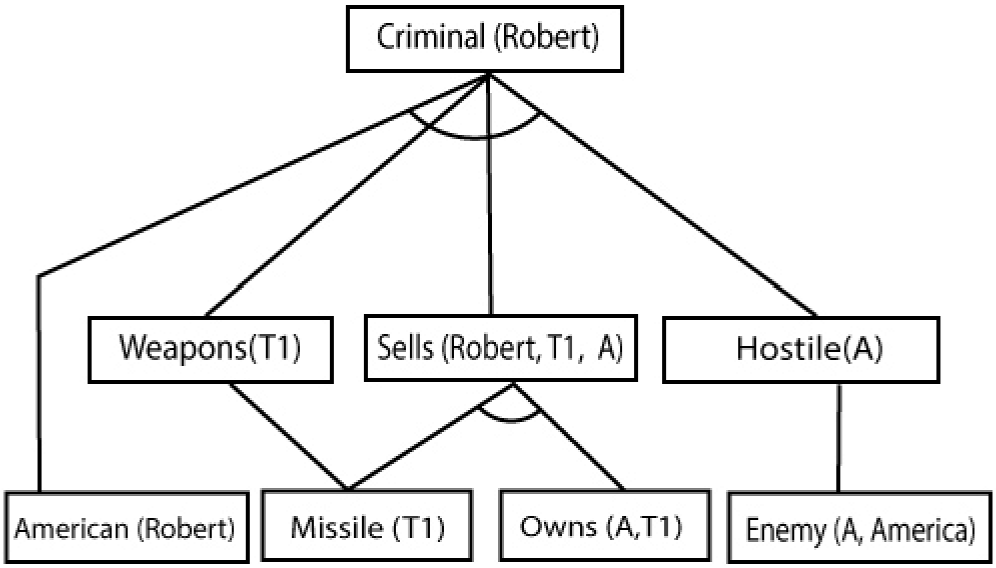
Rule-(4) satisfy with the substitution {p/T1}, **so Sells (Robert, T1, A)** is added, which infers from the conjunction of Rule (2) and (3).

Rule-(6) is satisfied with the substitution(p/A), so Hostile(A) is added and which infers from Rule-(7).



Step-3:

At step-3, as we can check Rule-(1) is satisfied with the substitution **{p/Robert, q/T1, r/A}, so we can add Criminal(Robert)** which infers all the available facts. And hence we reached our goal statement.



Hence it is proved that Robert is Criminal using forward chaining approach.

B. Backward Chaining:

Backward-chaining is also known as a backward deduction or backward reasoning method when using an inference engine. A backward chaining algorithm is a form of reasoning, which starts with the goal and works backward, chaining through rules to find known facts that support the goal.

Properties of backward chaining:

- It is known as a top-down approach.
- Backward-chaining is based on modus ponens inference rule.
- In backward chaining, the goal is broken into sub-goal or sub-goals to prove the facts true.
- It is called a goal-driven approach, as a list of goals decides which rules are selected and used.
- Backward -chaining algorithm is used in game theory, automated theorem proving tools, inference engines, proof assistants, and various AI applications.
- The backward-chaining method mostly used a **depth-first search** strategy for proof.

Example:

In backward-chaining, we will use the same above example, and will rewrite all the rules.

- **American (p) ∧ weapon(q) ∧ sells (p, q, r) ∧ hostile(r) → Criminal(p) ...(1)**
  Owns(A, T1) ....(2)
- **Missile(T1)**
- **?p Missiles(p) ∧ Owns (A, p) → Sells (Robert, p, A)**............................**(4)**

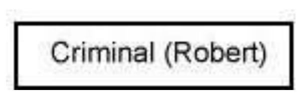
- **Missile(p) → Weapons (p)...............................(5)**
- **Enemy(p, America) →Hostile(p)...............................(6)**
- **Enemy (A, America)...............................(7)**
- **American(Robert)................................ (8)**

Backward-Chaining proof:

In Backward chaining, we will start with our goal predicate, which is **Criminal(Robert)**, and then infer further rules.

Step-1:

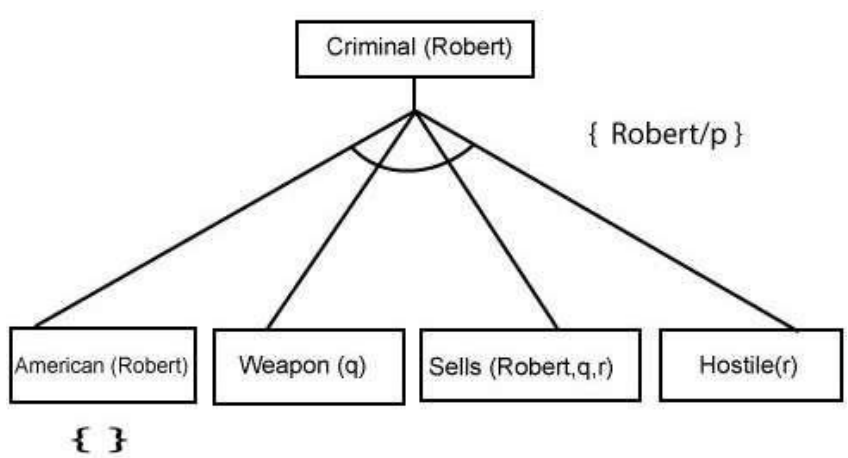
At the first step, we will take the goal fact. And from the goal fact, we will infer other facts, and at last, we will prove those facts true. So our goal fact is "Robert is Criminal," so following is the predicate of it.



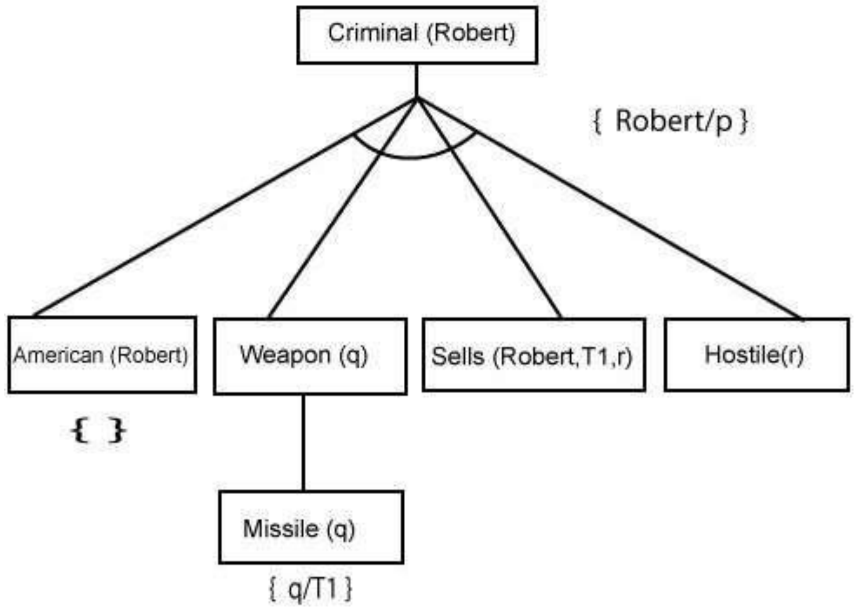
Step-2:

At the second step, we will infer other facts form goal fact which satisfies the rules. So as we can see in Rule-1, the goal predicate Criminal (Robert) is present with substitution {Robert/P}. So we will add all the conjunctive facts below the first level and will replace p with Robert.

Here we can see American (Robert) is a fact, so it is proved here.

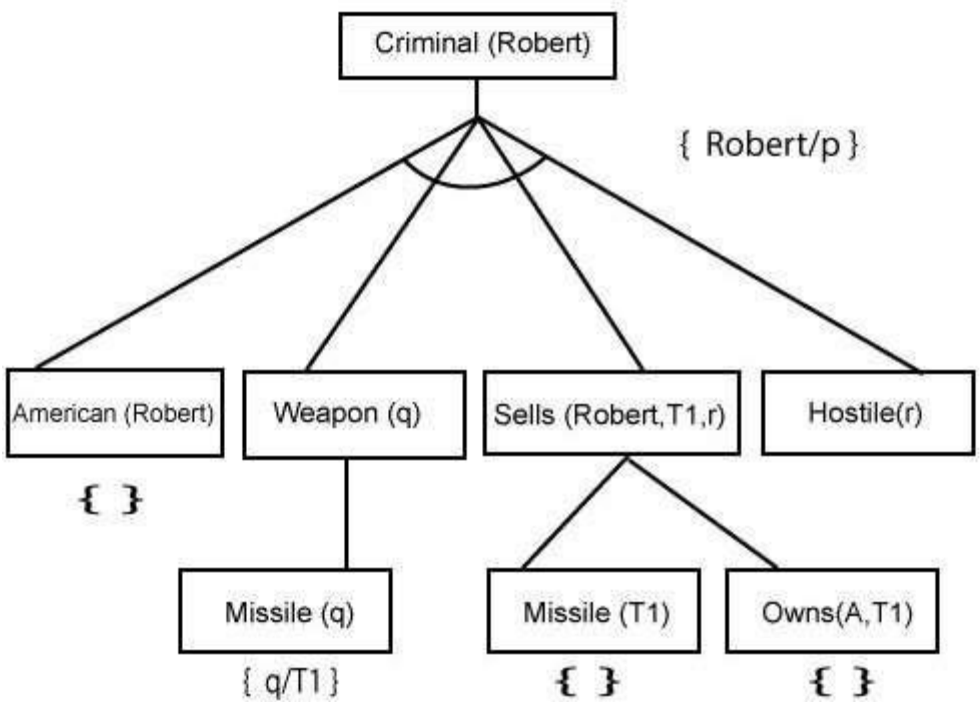


Step-3:t At step-3, we will extract further fact Missile(q) which infer from Weapon(q), as it satisfies Rule-(5). Weapon (q) is also true with the substitution of a constant T1 at q.



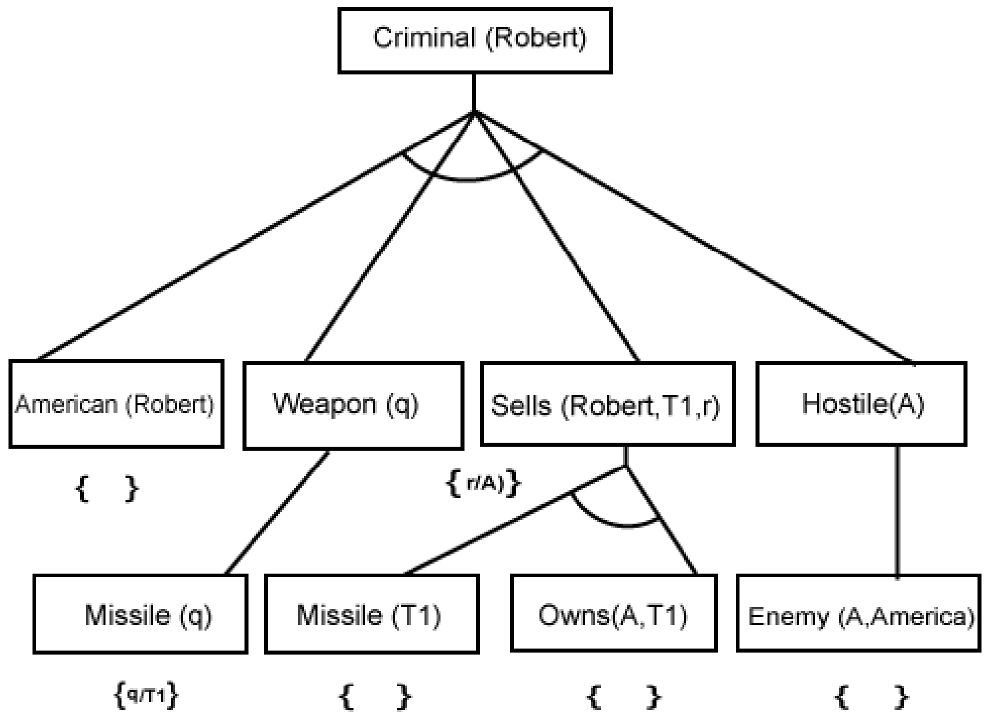
Step-4:

At step-4, we can infer facts Missile(T1) and Owns(A, T1) form Sells(Robert, T1, r) which satisfies the **Rule- 4**, with the substitution of A in place of r. So these two statements are proved here.



Step-5:

At step-5, we can infer the fact **Enemy(A, America)** from **Hostile(A)** which satisfies Rule-6. And hence all the statements are proved true using backward chaining.



Unit 4. Introduction to Expert System

An **Expert System** is an intelligent computer program that uses knowledge and reasoning to solve problems in specific domains, usually those requiring human expertise. It is one of the most important applications of Artificial Intelligence (AI). Expert systems are designed to simulate the reasoning processes of human experts and can make decisions, diagnose problems, and recommend solutions.

Key Components of Expert Systems:

- **Knowledge Base**: Contains domain-specific knowledge, facts, and rules about the problem. Knowledge can be gathered from human experts, books, or databases.
- **Inference Engine**: The core component that applies logical rules to the knowledge base to deduce new information or to make decisions.
- **User Interface**: The medium through which users interact with the expert system. It collects input from users and provides results based on reasoning.
- **Explanation System**: Provides reasoning for the decisions made by the expert system (i.e., explaining how conclusions were derived).
- **Knowledge Acquisition System**: Helps to gather and update the knowledge base from experts or external sources.

2. Block Diagram of Expert System

- **Knowledge Base**: Contains the expert knowledge and rules about the domain (e.g., in a medical expert system, it will include medical knowledge, symptoms, diseases, treatments).
- **Inference Engine**: Processes input from users and uses rules to draw conclusions from the knowledge base. It typically follows deductive reasoning to apply rules such as IF-THEN statements.
- **Explanation Facility**: Explains to the user how the system reached a particular decision or solution.
- **User Interface**: This is the front-end of the system, where users input their problem or data and receive a solution or recommendation.
- **Knowledge Acquisition System**: This component helps update the knowledge base by adding new knowledge (e.g., adding new diseases or treatments in a medical expert system).

3. Characteristics of Expert Systems

1. **High Performance**: Expert systems can process complex rules quickly and provide expert-level solutions in real-time.
2. **Domain-Specific Knowledge**: They are focused on a single domain of expertise (e.g., medicine, engineering) and are tailored to solve problems within that domain.
3. **Consistency**: Unlike human experts who can be influenced by emotions, fatigue, or inconsistency, expert systems provide the same solution for the same problem every time.

4. **Explainability**: Many expert systems can explain the reasoning behind their conclusions. This is crucial for gaining the user's trust and understanding the decision-making process.
5. **Modularity**: Knowledge in the system is often structured into modules or chunks, making it easier to update or extend the system.
6. **Inference and Reasoning Ability**: The system can apply reasoning and derive new knowledge or conclusions from the known facts in the knowledge base.
7. **Flexibility**: Expert systems can adapt to new knowledge by updating their knowledge base without significant reprogramming.

---

4. Types of Expert Systems

1. Rule-Based Expert Systems

Rule-based systems use a set of "if-then" rules to represent knowledge. Each rule consists of an antecedent (if part) and a consequent (then part).

- **Example**:
  - In a medical expert system: "IF the patient has a fever AND a sore throat, THEN suggest the possibility of the flu."
  - A classic example is **MYCIN**, which diagnoses bacterial infections based on symptoms provided by a user.
- **Advantages**: Easy to create, explain, and maintain.
- **Disadvantages**: Difficulty in managing large numbers of rules and handling uncertainties.

2. Frame-Based Expert Systems

Frame-based systems use data structures called "frames" to represent stereotypical situations or objects. Frames contain slots (attributes) that store information about an object.

- **Example**:
  - A frame might represent a vehicle with slots for type, color, speed, and capacity. Based on the values of these slots, the system can reason about the type of vehicle.

3. Fuzzy Logic Expert Systems

Fuzzy logic allows the system to handle uncertain or imprecise information. Instead of binary true/false logic, fuzzy logic works with degrees of truth.

- **Example**:
  - In an expert system for controlling air conditioning, fuzzy logic can be used to handle vague inputs like "hot," "cold," or "slightly warm" temperatures and adjust the cooling accordingly.

4. Neural Network-Based Expert Systems

Neural networks are used to make systems that can learn from examples and improve their decision-making over time. These systems are adaptive and can handle noisy and incomplete data.

- **Example**:
  - Neural networks are used in expert systems for stock market prediction, where the system learns from past data to make future predictions.

5. Hybrid Expert Systems

These combine several AI techniques, like rule-based systems with fuzzy logic, neural networks, or probabilistic reasoning, to improve their decision-making.

5. Design of Expert System

The design of an expert system involves several steps:

1. **Problem Identification**: Understand and define the problem that the expert system will solve. The problem must be narrow and well-defined to ensure that the system can simulate expertise in that domain.
2. **Knowledge Acquisition**: Extract knowledge from human experts through interviews, documentation, and case studies. This process can be time-consuming and is often referred to as the "knowledge acquisition bottleneck."
3. **Knowledge Representation**: Organize the knowledge into a structured format such as rules, frames, or semantic networks. This step ensures the system can process the knowledge efficiently.
4. **Choice of Inference Engine**: Design or choose the inference engine that will interpret and process the knowledge. The engine uses logical rules (forward chaining, backward chaining) to derive new conclusions.
   - **Forward Chaining**: Start with the known facts and apply rules to generate new facts until the goal is achieved.
   - **Backward Chaining**: Start with the goal and work backward, applying rules to determine which facts are needed to achieve the goal.
5. **User Interface Development**: Create a user-friendly interface that allows users to input data and get results from the system easily.
6. **Explanation Facility**: Provide an explanation module that allows the system to explain its reasoning process to the user.
7. **Testing and Validation**: Test the expert system on real-world cases to ensure accuracy and reliability. It is important to validate the system with expert feedback.

Example:

A **medical expert system** can be designed to diagnose respiratory diseases. The knowledge base includes rules about symptoms and diseases, such as:

- IF the patient has a cough AND fever, THEN suggest bronchitis. The inference engine will process the patient's symptoms and provide a diagnosis, which can be further explained to the user.

6. Advantages and Limitations of Expert Systems

Advantages:

1. **Consistency**: Provides consistent, accurate responses as it always follows the same reasoning process.
2. **Availability**: Available 24/7 and does not suffer from fatigue like human experts.
3. **Cost-Effective**: Reduces the need for hiring expensive experts, making expert-level knowledge available at lower costs.
4. **Fast Responses**: Can process large amounts of data and solve problems much faster than humans.
5. **Training and Learning**: Can act as a training tool for new or less experienced individuals, providing them with expert guidance.
6. **Handle Complex Problems**: Capable of solving very complex problems that may be difficult for non-experts.

Limitations:

1. **Knowledge Acquisition Bottleneck**: The process of gathering and encoding expert knowledge into the system is difficult and time-consuming.
2. **Lack of Common Sense**: Expert systems cannot handle problems that require general world knowledge or common sense, unlike humans.
3. **Limited to Specific Domains**: Expert systems are designed for very specific problems, and extending them to other domains requires substantial reprogramming.
4. **Maintenance**: Expert systems need to be updated frequently to ensure their knowledge base is current, which can be a challenge.
5. **No Self-Learning**: Most traditional expert systems do not learn from new experiences or data unless explicitly programmed with machine learning algorithms.

7. Applications of Expert Systems

Expert systems are used across various industries and sectors to solve domain-specific problems:

1. **Medical Diagnosis**:
   - **Example**: MYCIN, an early medical expert system, diagnoses bacterial infections and recommends antibiotics based on patient symptoms.
   - **Benefits**: Provides fast, accurate, and consistent diagnoses, reducing the workload on human doctors.
2. **Financial Services**:
   - **Example**: Expert systems are used to provide financial advice, manage portfolios, or perform risk analysis.
   - **Benefits**: Helps in making investment decisions, predicting stock trends, and detecting fraud.
3. **Manufacturing**:

- **Example**: Expert systems are used in manufacturing to manage production schedules, quality control, and fault diagnosis in machinery.
- **Benefits**: Improves efficiency

, reduces downtime, and enhances product quality.

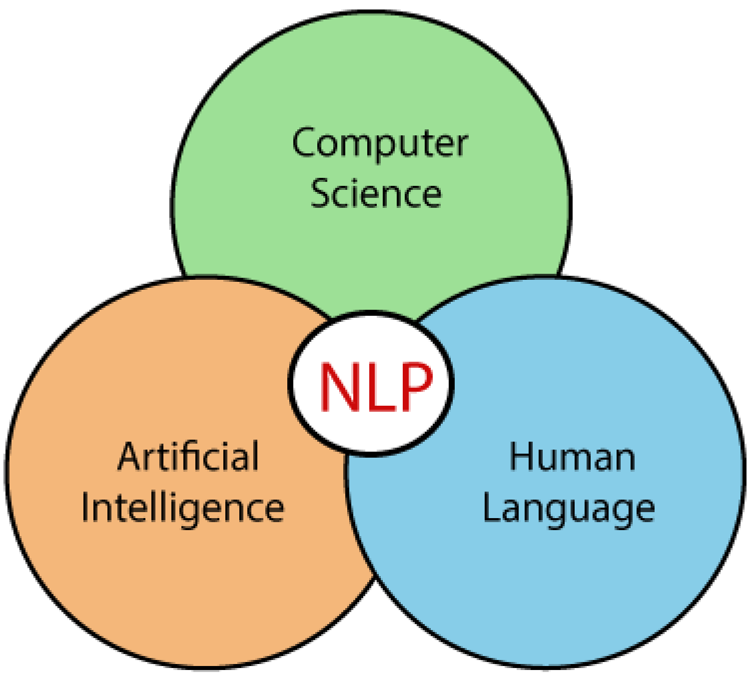
4. **Agriculture**:
   - **Example**: Expert systems can help farmers decide what crops to plant based on soil type, weather forecasts, and past yield data.
   - **Benefits**: Helps optimize crop production, pest control, and resource management.
5. **Education**:
   - **Example**: Intelligent Tutoring Systems (ITS) provide personalized learning experiences to students by adapting lessons based on their progress.
   - **Benefits**: Offers customized education and helps students learn at their own pace.
6. **Legal Decision Making**:
   - **Example**: Expert systems assist lawyers and judges in analyzing legal cases and suggesting possible outcomes based on past precedents.
   - **Benefits**: Speeds up the legal process and helps in making well-informed decisions.
7. **Weather Forecasting**:
   - **Example**: Expert systems analyze weather patterns and predict future weather conditions.
   - **Benefits**: Helps in disaster management, agriculture, and planning.

Unit-5 : Natural Language Processing

1. **Introduction**
2. **Advantages of NLP**
3. **Components of NLP**
4. **Application of NLP**
5. **Phases of NLP**

What is NLP?

NLP stands for **Natural Language Processing**, which is a part of **Computer Science, Human language,** and **Artificial Intelligence**. It is the technology that is used by machines to understand, analyse, manipulate, and interpret human's languages. It helps developers to organize knowledge for performing tasks such as **translation, automatic summarization, Named Entity Recognition (NER), speech recognition, relationship extraction,** and **topic segmentation**.



History of NLP

(1940-1960) - Focused on Machine Translation (MT)

The Natural Languages Processing started in the year 1940s.

1948 - In the Year 1948, the first recognisable NLP application was introduced in Birkbeck College, London.

1950s - In the Year 1950s, there was a conflicting view between linguistics and computer science. Now, Chomsky developed his first book syntactic structures and claimed that language is generative in nature.

In 1957, Chomsky also introduced the idea of Generative Grammar, which is rule based descriptions of syntactic structures.

(1960-1980) - Flavored with Artificial Intelligence (AI)

In the year 1960 to 1980, the key developments were:

Augmented Transition Networks (ATN) : Augmented Transition Networks is a finite state machine that is capable of recognizing regular languages.

Case Grammar : Case Grammar was developed by **Linguist Charles J. Fillmore** in the year 1968. Case Grammar uses languages such as English to express the relationship between nouns and verbs by using the preposition.

In Case Grammar, case roles can be defined to link certain kinds of verbs and objects.

For example: "Neha broke the mirror with the hammer". In this example case grammar identify Neha as an agent, mirror as a theme, and hammer as an instrument.

In the year 1960 to 1980, key systems were:

SHRDLU : SHRDLU is a program written by **Terry Winograd** in 1968-70. It helps users to communicate with the computer and moving objects. It can handle instructions such as "pick up the green boll" and also answer the questions like "What is inside the black box." The main importance of SHRDLU is that it shows those syntax, semantics, and reasoning about the world that can be combined to produce a system that understands a natural language.

LUNAR : LUNAR is the classic example of a Natural Language database interface system that is used ATNs and Woods' Procedural Semantics. It was capable of translating elaborate natural language expressions into database queries and handle 78% of requests without errors.

1980 - Current

Till the year 1980, natural language processing systems were based on complex sets of hand-written rules. After 1980, NLP introduced machine learning algorithms for language processing.

In the beginning of the year 1990s, NLP started growing faster and achieved good process accuracy, especially in English Grammar. In 1990 also, an electronic text introduced, which provided a good resource for training and examining natural language programs. Other factors may include the availability of computers with fast CPUs and more memory. The major factor behind the advancement of natural language processing was the Internet.

Now, modern NLP consists of various applications, like **speech recognition, machine translation,** and **machine text reading**. When we combine all these applications then it allows the artificial intelligence to gain knowledge of the world. Let's consider the example of AMAZON ALEXA, using this robot you can ask the question to Alexa, and it will reply to you.

Advantages of NLP

A list of advantages of NLP is given below:

- NLP helps users to ask questions about any subject and get a direct response within seconds.
- NLP offers exact answers to the question means it does not offer unnecessary and unwanted information.
- NLP helps computers to communicate with humans in their languages.
- It is very time efficient.
- Most of the companies use NLP to improve the efficiency of documentation processes, accuracy of documentation, and identify the information from large databases.

Disadvantages of NLP

A list of disadvantages of NLP is given below:

- NLP may not show context.
- NLP is unpredictable
- NLP may require more keystrokes.
- NLP is unable to adapt to the new domain, and it has a limited function that's why NLP is built for a single and specific task only.

Components of NLP

There are the following two components of NLP -

1. Natural Language Understanding (NLU)

Natural Language Understanding (NLU) helps the machine to understand and analyse human language by extracting the metadata from content such as concepts, entities, keywords, emotion, relations, and semantic roles.

NLU mainly used in Business applications to understand the customer's problem in both spoken and written language.

NLU involves the following tasks -

- It is used to map the given input into useful representation.
- It is used to analyze different aspects of the language.

2. Natural Language Generation (NLG)

Natural Language Generation (NLG) acts as a translator that converts the computerized data into natural language representation. It mainly involves Text planning, Sentence planning, and Text Realization.

Applications of NLP

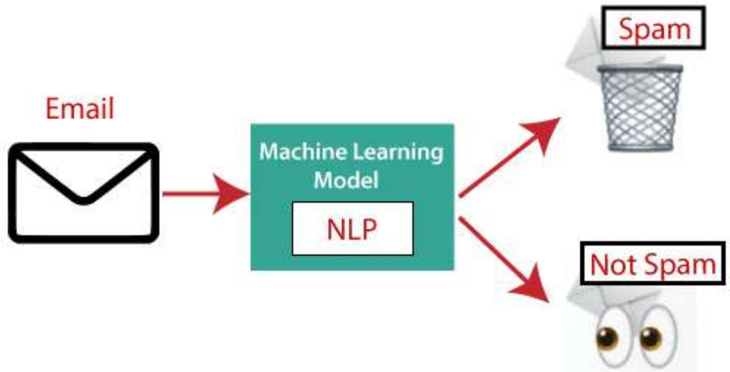
There are the following applications of NLP -

1. Question Answering

Question Answering focuses on building systems that automatically answer the questions asked by humans in a natural language.

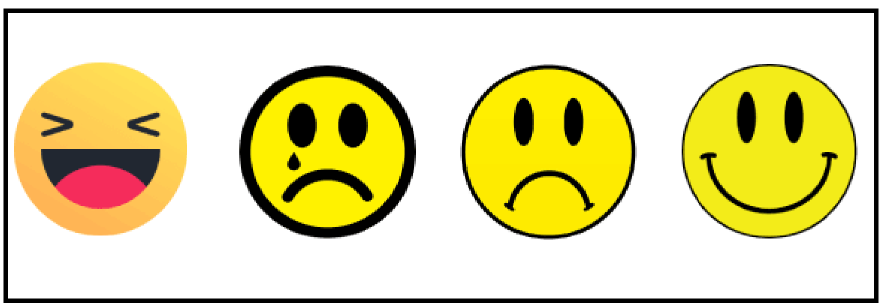
2. Spam Detection

Spam detection is used to detect unwanted e-mails getting to a user's inbox.



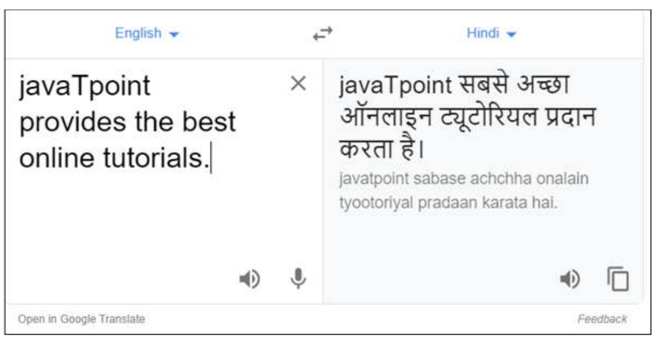
3. Sentiment Analysis

Sentiment Analysis is also known as **opinion mining**. It is used on the web to analyse the attitude, behaviour, and emotional state of the sender. This application is implemented through a combination of NLP (Natural Language Processing) and statistics by assigning the values to the text (positive, negative, or natural), identify the mood of the context (happy, sad, angry, etc.)



4. Machine Translation

Machine translation is used to translate text or speech from one natural language to another natural language.



Example: Google Translator

5. Spelling correction

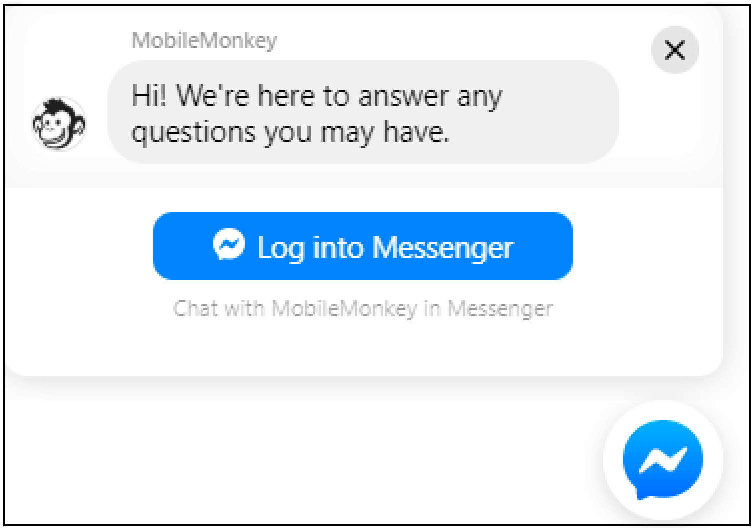
Microsoft Corporation provides word processor software like MS-word, PowerPoint for the spelling correction.

6. Speech Recognition

Speech recognition is used for converting spoken words into text. It is used in applications, such as mobile, home automation, video recovery, dictating to Microsoft Word, voice biometrics, voice user interface, and so on.

7. Chatbot

Implementing the Chatbot is one of the important applications of NLP. It is used by many companies to provide the customer's chat services.



8. Information extraction

Information extraction is one of the most important applications of NLP. It is used for extracting structured information from unstructured or semi-structured machine-readable documents.

9. Natural Language Understanding (NLU)

It converts a large set of text into more formal representations such as first-order logic structures that are easier for the computer programs to manipulate notations of the natural language processing.

How to build an NLP pipeline

There are the following steps to build an NLP pipeline -

Step1: Sentence Segmentation

Sentence Segment is the first step for building the NLP pipeline. It breaks the paragraph into separate sentences.

Example: Consider the following paragraph -

Independence Day is one of the important festivals for every Indian citizen. It is celebrated on the 15th of August each year ever since India got independence from the British rule. The day celebrates independence in the true sense.

Sentence Segment produces the following result:

1. "Independence Day is one of the important festivals for every Indian citizen."
2. "It is celebrated on the 15th of August each year ever since India got independence from the British rule."
3. "This day celebrates independence in the true sense."

Step2: Word Tokenization

Word Tokenizer is used to break the sentence into separate words or tokens.

Example:

JavaTpoint offers Corporate Training, Summer Training, Online Training, and Winter Training.

Word Tokenizer generates the following result:

"JavaTpoint", "offers", "Corporate", "Training", "Summer", "Training", "Online", "Training", "and", "Winter", "Training", "."

Step3: Stemming

Stemming is used to normalize words into its base form or root form. For example, celebrates, celebrated and celebrating, all these words are originated with a single root word "celebrate." The big problem with stemming is that sometimes it produces the root word which may not have any meaning.

For Example, intelligence, intelligent, and intelligently, all these words are originated with a single root word "intelligen." In English, the word "intelligen" do not have any meaning.

Step 4: Lemmatization

Lemmatization is quite similar to the Stamming. It is used to group different inflected forms of the word, called Lemma. The main difference between Stemming and lemmatization is that it produces the root word, which has a meaning.

For example: In lemmatization, the words intelligence, intelligent, and intelligently has a root word intelligent, which has a meaning.

Step 5: Identifying Stop Words

In English, there are a lot of words that appear very frequently like "is", "and", "the", and "a". NLP pipelines will flag these words as stop words. **Stop words** might be filtered out before doing any statistical analysis.

Example: He **is a** good boy.

Note: When you are building a rock band search engine, then you do not ignore the word "The."

Step 6: Dependency Parsing

Dependency Parsing is used to find that how all the words in the sentence are related to each other.

Step 7: POS tags

POS stands for parts of speech, which includes Noun, verb, adverb, and Adjective. It indicates that how a word functions with its meaning as well as grammatically within the sentences. A word has one or more parts of speech based on the context in which it is used.

Example: "Google" something on the Internet.

In the above example, Google is used as a verb, although it is a proper noun.

Step 8: Named Entity Recognition (NER)

Named Entity Recognition (NER) is the process of detecting the named entity such as person name, movie name, organization name, or location.

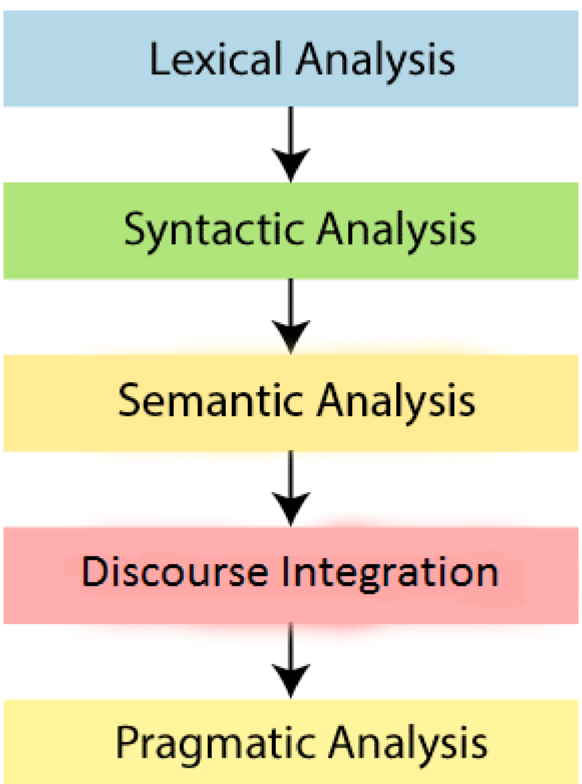
Example: Steve Jobs introduced iPhone at the Macworld Conference in San Francisco, California.

Step 9: Chunking

Chunking is used to collect the individual piece of information and grouping them into bigger pieces of sentences.

Phases of NLP

There are the following five phases of NLP:



1. Lexical Analysis and Morphological

The first phase of NLP is the Lexical Analysis. This phase scans the source code as a stream of characters and converts it into meaningful lexemes. It divides the whole text into paragraphs, sentences, and words.

2. Syntactic Analysis (Parsing)

Syntactic Analysis is used to check grammar, word arrangements, and shows the relationship among the words.

Example: Agra goes to the Poonam

In the real world, Agra goes to the Poonam, does not make any sense, so this sentence is rejected by the Syntactic analyzer.

3. Semantic Analysis

Semantic analysis is concerned with the meaning representation. It mainly focuses on the literal meaning of words, phrases, and sentences.

4. Discourse Integration

Discourse Integration depends upon the sentences that proceeds it and also invokes the meaning of the sentences that follow it.

5. Pragmatic Analysis

Pragmatic is the fifth and last phase of NLP. It helps you to discover the intended effect by applying a set of rules that characterize cooperative dialogues.

For Example: "Open the door" is interpreted as a request instead of an order.

Why NLP is difficult?

NLP is difficult because Ambiguity and Uncertainty exist in the language.

Ambiguity

There are the following three ambiguity -

- **Lexical Ambiguity**

Lexical Ambiguity exists in the presence of two or more possible meanings of the sentence within a single word.

Example:

Manya is looking for a **match**.

In the above example, the word match refers to that either Manya is looking for a partner or Manya is looking for a match. (Cricket or other match)

- **Syntactic Ambiguity**

Syntactic Ambiguity exists in the presence of two or more possible meanings within the sentence.

Example:

I saw the girl with the binocular.

In the above example, did I have the binoculars? Or did the girl have the binoculars?

- **Referential Ambiguity**

Referential Ambiguity exists when you are referring to something using the pronoun.

Example: Kiran went to Sunita. She said, "I am hungry."

In the above sentence, you do not know that who is hungry, either Kiran or Sunita.

NLP Libraries

Scikit-learn: It provides a wide range of algorithms for building machine learning models in Python.

Natural language Toolkit (NLTK): NLTK is a complete toolkit for all NLP techniques.

Pattern: It is a web mining module for NLP and machine learning.

TextBlob: It provides an easy interface to learn basic NLP tasks like sentiment analysis, noun phrase extraction, or pos-tagging.

Quepy: Quepy is used to transform natural language questions into queries in a database query language.

SpaCy: SpaCy is an open-source NLP library which is used for Data Extraction, Data Analysis, Sentiment Analysis, and Text Summarization.

Gensim: Gensim works with large datasets and processes data streams.

Difference between Natural language and Computer Language

Natural language	Computer Language
Natural language has a very large vocabulary.	Computer language has a very limited vocabulary.
Natural language is easily understood by humans.	Computer language is easily understood by the machines.
Natural language is ambiguous in nature.	Computer language is unambiguous.