UNCERTAINTY in AI

With knowledge representation, we write A→B, which means if A is true then B is true, but consider a situation where we are not sure about whether A is true or not then we cannot express this statement, this situation is called uncertainty.  
Uncertainty arises due to:  
1. Lack of Information: Uncertainty often arises when there is a lack of complete or reliable information, making it difficult to make confident decisions or predictions.

2. Doubtful Information: Uncertainty can also stem from information that is questionable, ambiguous, or subject to doubt, leading to hesitancy in relying on such information for decision-making.

3. Also due to: experimental errors, temperature variation, climate change, equipment fault

Most AI systems have some degree of uncertainty associated with them.

1. Uncertain Inputs:

Uncertain inputs refer to situations where the data or information provided to a system or model is itself uncertain or contains variability.  
when data is missing unreliable ambiguous inconsistent subjective noisy or derived from default it represents and expert’s guess and it is referred to as uncertain data

1. Uncertain Knowledge:

Uncertain knowledge pertains to situations where the information or knowledge used to make decisions or predictions is inherently uncertain or lacks precision.

Uncertain knowledge occurs when the information we have can be linked to multiple different causes, which in turn can lead to various outcomes.

1. Uncertain Outputs:

Uncertain outputs arise when the predictions, decisions, or results generated by a system or model are not definitively certain and come with a degree of probability or confidence.

1. Uncertain Knowledge representation

Uncertain knowledge representation is a way of describing things that acknowledges that we don't have all the information, and it might not capture everything perfectly. It's like creating a simplified model of something in the real world, but knowing that it might not cover all the details accurately.

The representation itself might not be very flexible or expressive, meaning it can't describe everything about the thing you're trying to represent. And the data used for this representation might have gaps or errors, which can cause different kinds of uncertainty or doubt about the accuracy of the information.

Approaches to handling uncertainty:

Probabilistic reasoning:

Probabilistic reasoning is a branch of artificial intelligence and statistics that deals with uncertainty and probability to make informed decisions and predictions.

It's a way to model and reason about uncertainty in a formal and systematic manner.

It combines probability theory with logic to deal with uncertain outcomes.

When we use?uncertain outcomes

* uncertain outcomes
* Possibilities of outcomes is too large
* When an unknown error occurs during the experiment

There are 2 ways to solve problems with uncertain knowledge

1. Bayes rule

Bayes rule is also called as Bayes law or Bayes Theorem.

Bayes' rule is a fundamental theorem in probability theory that allows updating probabilities based on new evidence. It provides a principled way to combine prior knowledge with new data to update the probabilities of different outcomes. Bayes' rule has been widely used in AI for classification, prediction, and decision-making tasks where uncertainty needs to be addressed.

Mathematically, Bayes' Theorem is expressed as:

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

Where:

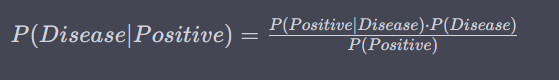
P(A|B) represents the posterior probability, which is the probability of event A occurring given that event B has occurred.

P(B|A) represents the likelihood, which is the probability of observing event B given that event A has occurred.

P(A) represents the prior probability, which is the initial probability of event A occurring before considering any new evidence.

P(B) represents the marginal likelihood, which is the probability of observing event B, regardless of whether event A has occurred.

The beauty of Bayes' Theorem is that it allows you to revise your beliefs as you receive more information. It's often used in scenarios where you want to make informed decisions or predictions based on uncertain or incomplete information.. It is particularly useful in handling uncertainty and making decisions when there is incomplete or ambiguous Information.



1. Bayesian statistics

Bayesian statistics is a branch of statistics that uses probabilistic reasoning to analyze and interpret data. It provides a framework for making statistical inferences and estimating probabilities based on data and prior knowledge.

Bayesian analysis involves two key probabilities: prior and posterior. The prior probability represents initial beliefs before considering evidence, while the posterior probability reflects updated beliefs after incorporating new data.

This process of updating makes Bayesian inference powerful in handling uncertainty.

BAYESIAN NETWORK - techneo

Advantages:

1. \*\*Effective for Uncertain Data:\*\* BBNs are particularly useful when dealing with uncertain or incomplete data. They can model and propagate uncertainty, making them robust in situations where traditional deterministic methods may struggle.

2. \*\*Probabilistic Reasoning:\*\* BBNs provide a principled way to perform probabilistic reasoning. They can estimate the probabilities of events or outcomes, which is essential for decision-making under uncertainty.

3. \*\*Graphical Representation:\*\* The graphical structure of a BBN offers a visual and intuitive way to understand and communicate complex relationships among variables. It can aid in the interpretation of models and facilitate collaboration among experts.

4. \*\*Causal Inference:\*\* BBNs are excellent tools for modeling and exploring cause-and-effect relationships. They allow for causal reasoning and can help identify the root causes of observed events.

5. \*\*Efficient Inference:\*\* BBNs can efficiently compute posterior probabilities and marginal probabilities, making them suitable for real-time or near-real-time applications. This is crucial in fields like medical diagnosis, finance, and decision support systems.

Disadvantages:

1. \*\*Complexity and Expertise:\*\* Constructing a BBN can be complex, especially for large and intricate systems. It often requires expertise in the domain to define the network structure and specify conditional probabilities accurately.

2. \*\*Data Intensive:\*\* BBNs require data for parameter estimation. In cases where data is scarce or expensive to obtain, building an accurate BBN can be challenging.

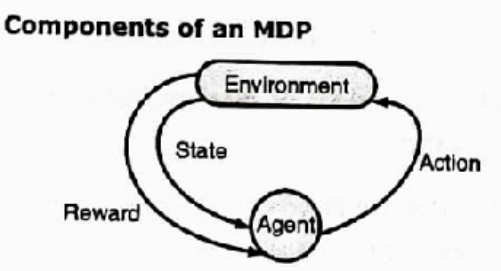
3. \*\*Curse of Dimensionality:\*\* BBNs can become computationally intractable when dealing with a large number of variables. The number of parameters and the computational complexity grow exponentially with the number of variables, leading to the "curse of dimensionality."

4. \*\*Discretization:\*\* BBNs typically require discretizing continuous variables, which can lead to information loss and introduce discretization bias. Selecting appropriate discretization intervals can be challenging.

5. \*\*Sensitivity to Parameter Estimates:\*\* The quality of BBN inference is highly sensitive to the accuracy of the parameter estimates. Small errors in probability values can lead to significant inaccuracies in results.

6. \*\*Limited Handling of Cyclic Graphs:\*\* BBNs typically assume acyclic network structures. When cyclic relationships are essential, other graphical models like dynamic Bayesian networks may be more suitable.

MARKOV DECISION PROCESS:



Think of a Markov decision process (MDP) as a way to make decisions step by step, like in a game. It's like a set of rules that help you figure out what action to take at each moment. These actions can lead to rewards or penalties, like winning points or losing them in a game.Reinforcement learning uses MDPs as a framework to learn and make smart decisions over time.

So, MDPs help us structure problems for this learning process in a systematic way.

Limitations of MDP:

1. Curse of Dimensionality: As the number of states and actions in an MDP increases, the computational complexity of solving it grows exponentially. This can make MDPs impractical for problems with large state or action spaces.
2. Assumption of Stationarity: MDPs assume that the underlying system is stationary, meaning the transition probabilities and rewards remain constant over time. In reality, many systems are non-stationary, and MDPs may not accurately capture their dynamics.
3. Single-Agent Framework: MDPs are primarily designed for single-agent decision-making. They do not directly address problems involving multiple agents or competitive interactions, which are common in real-world scenarios.