Module 2: Knowledge Based Systems

Contents:

- **Types of Knowledge Based Systems**
- **Expert Systems**
- **Intelligent Agents**
- **Data Mining**
- **Case Based Reasoning**
- **Genetic Algorithms**
- **Neural Networks**

Course Objectives & Course Outcomes

- **Course Objective:** To conceptualize the basic ideas underlying the design of Knowledge Based Systems.
- **Course Outcome:** Design and develop basic knowledge based systems, artificial neural networks, genetic algorithms and data mining techniques, and identify their components, risks, and impacts. LO 2.1, LO 2.2, LO 2.3)

Learning Outcomes

A learner will be able to

- LO 2.1: Design expert system and intelligent agents and identify their components along with a brief assessment of their social and economic impact. (PI 3.2.2) (PI 6.3.2)
- LO 2.2 Explore the design mechanism of data mining and case based reasoning techniques and explore advancements in the respective fields ensuring ongoing competence in evolving applications. (PI 3.2.1), (PI 11.2.2)
- LO 2.3 Develop the ability to explore and investigate genetic algorithms and neural networks and identify the risk, impact on various processes, professions and domains along with the trends and advancements in the field. (PI 6.3.1), (PI 11.2.1)

Intelligence in Nature (Activity)

- Do the plant and animals in the example above exhibit evidence of intelligence?
- If you were a knowledge engineer, what particular behavioural skills or knowledge, in generic terms, would you expect to find in the objects listed below?




- Think specifically of the movement (or lack of) for each object, as well as the communication skills that could be expected.

Intelligence in Nature (Activity Cont...)

	Progressive Level of Intelligence
Plants	
Fish	
Chimpanzees:	
Humans	

Intelligence in Nature (Activity)

	Progressive Level of Intelligence
Plants	**Adapt in time and evolve—an individual plant has no skills but as a species they do.**
Fish	
Chimpanzees:	
Humans	

Intelligence in Nature (Activity)

	Progressive Level of Intelligence
Plants	**Adapt in time and evolve—an individual plant has no skills but as a species they do.**
Fish	**Navigation, Visual recognition, Avoid danger.**
Chimpanzees:	
Humans	

Intelligence in Nature (Activity)

	Progressive Level of Intelligence
Plants	Adapt in time and evolve—an individual plant has no skills but as a species they do.
Fish	Navigation, Visual recognition, Avoid danger.
Chimpanzees:	Language/communication about concrete concepts, Use of basic tools, Simple problem solving, Mimic humans, Build mental models.
Humans	

Intelligence in Nature (Activity)

	Progressive Level of Intelligence
Plants	Adapt in time and evolve—an individual plant has no skills but as a species they do.
Fish	Navigation, Visual recognition, Avoid danger.
Chimpanzees:	Language/communication about concrete concepts, Use of basic tools, Simple problem solving, Mimic humans, Build mental models.
Humans	Language/communication about complex concepts, Learn from being told, Learn from the past experience, Identify cause and effect relationships, Teach, Solve complex problems , Design, plan and schedule, Create complex abstract models, Show initiative.

Summary

- one common feature that fish, chimpanzees and humans share is that we are all unique individuals. Within the scope of our mental capacity we have individual choice and make our own decisions. This again is evidence of intelligence.

- The application of artificial intelligence has tried to emulate all of these characteristics within computer systems. Knowledge engineers have the difficult job of attempting to build these characteristics into a computer program.

Recall of KBS

❑Definition: Knowledge Engineering (KE)Knowledge

Engineering is the process of designing systems that mimic expert-level human decision-making. It involves acquiring, organizing, representing, and applying knowledge so that machines (especially in AI) can perform tasks that normally require human intelligence.

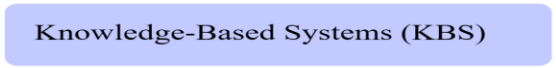
❑ What is a Knowledge-Based System (KBS)?

A Knowledge-Based System is a type of computer program that

- ✓Stores knowledge about a domain (like a doctor's medical knowledge)
- ✓Applies logic or rules to that knowledge
- ✓Makes inferences or suggestions like a human expert would
- ✓Example: A medical expert system that suggests diagnoses based on symptoms.

Knowledge-Based Systems (KBS)

- What is a KBS?
- Knowledge
- Objectives of KBS
- KBS vs. Traditional Computer-Based Systems (CBIS)
- General Structure of a KBS
- Types of KBS
- CommonKADS (Methodology for KBS)
- Knowledge Engineering
- Knowledge Acquisition
- Knowledge Representation
- PyKE (Python Knowledge Engine)
- Uncertainty Management
- Knowledge Validation
- Truth Maintenance Systems (TMS)



- What is a KBS?
  - Program capturing and using knowledge
  - Assists in solving complex problems
  - Supports human decision-making, learning
  - Subcategory of Artificial Intelligence (AI)
  - Comprises Knowledge Base and Inference Engine
  - Learning improves system over time
- Knowledge
- Objectives of KBS
- KBS vs. Traditional Computer-Based Systems (CBIS)
- General Structure of a KBS
- Types of KBS
- CommonKADS (Methodology for KBS)
- Knowledge Engineering
- Knowledge Acquisition
- Knowledge Representation
- PyKE (Python Knowledge Engine)
- Uncertainty Management
- Knowledge Validation
- Truth Maintenance Systems (TMS)

- Knowledge-Based Systems (KBS)
  - What is a KBS?
    - Program capturing and using knowledge
    - Assists in solving complex problems
    - Supports human decision-making, learning
    - Subcategory of Artificial Intelligence (AI)
    - Comprises Knowledge Base and Inference Engine
    - Learning improves system over time
  - Knowledge
    - Valuable entity in itself
    - Information used to solve problems
    - Explicit functional associations between information/data (Debenham, 1988)
    - Includes
    - May be
    - Types of Knowledge
  - Objectives of KBS
  - KBS vs. Traditional Computer-Based Systems (CBIS)
  - General Structure of a KBS
  - Types of KBS
  - CommonKADS (Methodology for KBS)
  - Knowledge Engineering
  - Knowledge Acquisition
  - Knowledge Representation
  - PyKE (Python Knowledge Engine)
  - Uncertainty Management
  - Knowledge Validation
  - Truth Maintenance Systems (TMS)

- Knowledge-Based Systems (KBS)
  - What is a KBS?
    - Program capturing and using knowledge
    - Assists in solving complex problems
    - Supports human decision-making, learning
    - Subcategory of Artificial Intelligence (AI)
    - Comprises Knowledge Base and Inference Engine
    - Learning improves system over time
  - Knowledge
    - Valuable entity in itself
    - Information used to solve problems
    - Explicit functional associations between information/data (Debenham, 1988)
    - Includes
      - Facts
      - Concepts
      - Procedures
      - Models
      - Heuristics
      - Examples
    - May be
    - Types of Knowledge
  - Objectives of KBS
  - KBS vs. Traditional Computer-Based Systems (CBIS)
  - General Structure of a KBS
  - Types of KBS
  - CommonKADS (Methodology for KBS)
  - Knowledge Engineering
  - Knowledge Acquisition
  - Knowledge Representation
  - PyKE (Python Knowledge Engine)
  - Uncertainty Management
  - Knowledge Validation
  - Truth Maintenance Systems (TMS)

- Knowledge-Based Systems (KBS)
  - What is a KBS?
    - Program capturing and using knowledge
    - Assists in solving complex problems
    - Supports human decision-making, learning
    - Subcategory of Artificial Intelligence (AI)
    - Comprises Knowledge Base and Inference Engine
    - Learning improves system over time
  - Knowledge
    - Valuable entity in itself
    - Information used to solve problems
    - Explicit functional associations between information/data (Debenham, 1988)
    - Includes
      - Facts
      - Concepts
      - Procedures
      - Models
      - Heuristics
      - Examples
    - May be
      - Specific or general
      - Exact or fuzzy
    - Types of Knowledge
  - Objectives of KBS
  - KBS vs. Traditional Computer-Based Systems (CBIS)
  - General Structure of a KBS
  - Types of KBS
  - CommonKADS (Methodology for KBS)
  - Knowledge Engineering
  - Knowledge Acquisition
  - Knowledge Representation
  - PyKE (Python Knowledge Engine)
  - Uncertainty Management
  - Knowledge Validation
  - Truth Maintenance Systems (TMS)

- Knowledge-Based Systems (KBS)
  - What is a KBS?
    - Program capturing and using knowledge
    - Assists in solving complex problems
    - Supports human decision-making, learning
    - Subcategory of Artificial Intelligence (AI)
    - Comprises Knowledge Base and Inference Engine
    - Learning improves system over time
  - Knowledge
    - Valuable entity in itself
    - Information used to solve problems
    - Explicit functional associations between information/data (Debenham, 1988)
    - Includes
      - Facts
      - Concepts
      - Procedures
      - Models
      - Heuristics
      - Examples
    - May be
      - Specific or general
      - Exact or fuzzy
    - Types of Knowledge
      - Tacit knowledge
      - Explicit knowledge
      - Factual knowledge
      - Procedural knowledge
      - Commonsense knowledge
      - Domain knowledge
      - Meta knowledge
      - Heuristic knowledge
  - Objectives of KBS
  - KBS vs. Traditional Computer-Based Systems (CBIS)
  - General Structure of a KBS
  - Types of KBS
  - CommonKADS (Methodology for KBS)
  - Knowledge Engineering
  - Knowledge Acquisition
  - Knowledge Representation
  - PyKE (Python Knowledge Engine)
  - Uncertainty Management
  - Knowledge Validation
  - Truth Maintenance Systems (TMS)

Expert Systems

en.akinator.com/theme-selection

Enter passphrase

PBLApp | Homepage

English

Select the next game thematic

- Characters
- Objects
- Animals

11:17 AM

en.akinator.com/game

Enter passphrase

PBLApp | Homepage

akinator

English

1

Does your character play in IPL?

Yes

No

Don't know

Probably

Probably not

11:18 AM

en.akinator.com/game#

Enter passphrase

PBLApp | Homepage

akinator

English

2

Is your character of Indian origins?

Yes

No »

Don't know

Probably

Probably not

en.akinator.com/game#

11:18 AM

en.akinator.com/game#

Enter passphrase

PBLApp | Homepage

akinator

English

3

Is your character a sportsman?

Yes

No

Don't know

Probably

Probably not

11:18 AM

en.akinator.com/game#

Enter passphrase

PBLApp | Homepage

akinator

English

4

Is your character a real person?

Yes

No

Don't know

Probably

Probably not

11:18 AM

en.akinator.com/game#

Enter passphrase

PBLApp | Homepage

akinator

English

5

Does your character live in India?

Yes

No

Don't know

Probably

Probably not

11:19 AM

en.akinator.com/game#

Enter passphrase

PBLApp | Homepage

akinator

English

6

Is your character a female?

Yes

No

Don't know

Probably

Probably not

11:19 AM

en.akinator.com/game#

Enter passphrase

PBLApp | Homepage

akinator

English

7

Is your character American ?

Yes

No

Don't know

Probably

Probably not

11:19 AM

en.akinator.com/game#

Enter passphrase

PBLApp | Homepage

akinator

English

8

Is your character dead ?

Yes

No

Don't know

Probably

Probably not

11:19 AM

en.akinator.com/game#

Enter passphrase

PBLApp | Homepage

akinator

English

9

Does your character live in Korea?

Yes

No

Don't know

Probably

Probably not

11:20 AM

en.akinator.com/game#

Enter passphrase

PBLApp | Homepage

akinator

English

10

Is your character in a group consisting of 7 members?

Yes

No

Don't know

Probably

Probably not

11:20 AM

en.akinator.com/game#

Enter passphrase

PBLApp | Homepage

akinator

English

11

Is your character a member of a boyband?

Yes

No

Don't know

Probably

Probably not

11:20 AM

en.akinator.com/game#

Enter passphrase

PBLApp | Homepage

akinator

English

12

Is your character from Australia?

Yes

No

Don't know

Probably

Probably not

11:21 AM

en.akinator.com/game#

Enter passphrase

PBLApp | Homepage

akinator

English

13

Is your character a member a band of 5 persons?

Yes

No

Don't know

Probably

Probably not

11:21 AM

en.akinator.com/game#

Enter passphrase

PBLApp | Homepage

akinator

English

14

Is your character from 'Record of Youth'?

Yes

No

Don't know

Probably

Probably not

11:21 AM

en.akinator.com/game#

Enter passphrase

PBLApp | Homepage

akinator

English

15

Is your character in a group consisting of 10+ members?

Yes

No

Don't know

Probably

Probably not

11:21 AM

en.akinator.com/game#

Enter passphrase

PBLApp | Homepage

akinator

English

16

Is your character in JYP entertainment ?

Yes

No

Don't know

Probably

Probably not

11:22 AM

en.akinator.com/game#

Enter passphrase

PBLApp | Homepage

akinator

English

17

Did your character's private videos get leaked?

Yes

No

Don't know

Probably

Probably not

11:22 AM

en.akinator.com/game#

Enter passphrase

PBLApp | Homepage

akinator

English

18

Is your character a king?

Yes

No

Don't know

Probably

Probably not

11:22 AM

en.akinator.com/game#

Enter passphrase

PBLApp | Homepage

akinator

English

19

Does your character own a cat?

Yes

No

Don't know

Probably

Probably not

11:22 AM

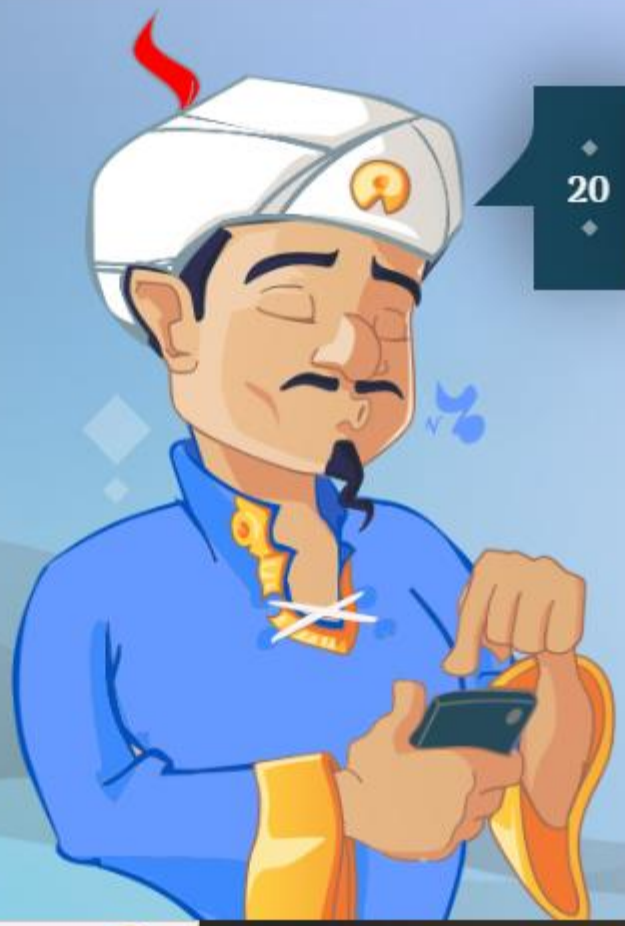
en.akinator.com/game#

Enter passphrase

PBLApp | Homepage

akinator

English



20

Does your character love cats very much?

Yes

No

Don't know

Probably

Probably not

11:23 AM

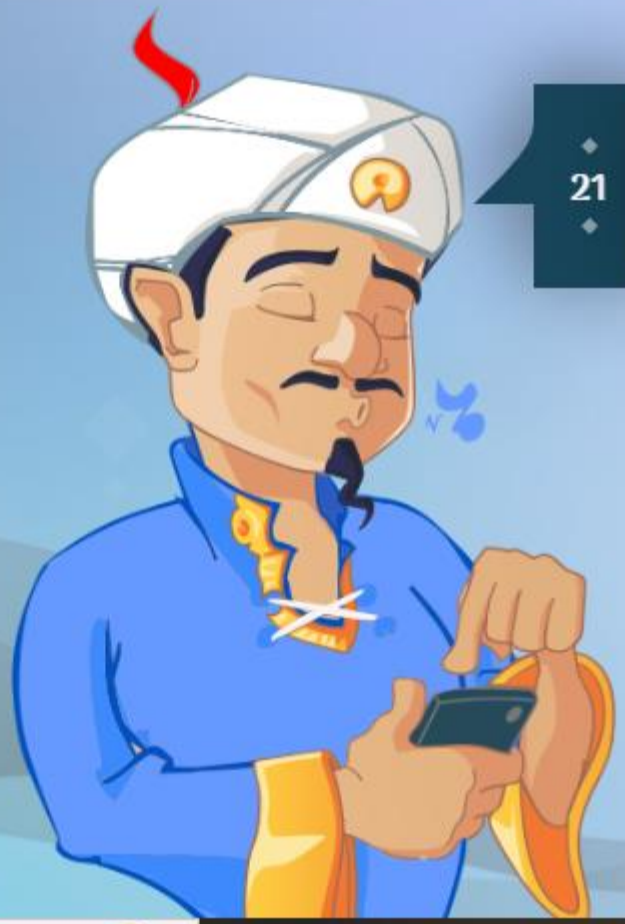
en.akinator.com/game#

Enter passphrase

PBLApp | Homepage

akinator

English



21

Is your character a professional dancer?

Yes

No

Don't know

Probably

Probably not

11:23 AM

en.akinator.com/game#

Enter passphrase

PBLApp | Homepage

akinator

English

I THINK OF

Lee Know (Lee Minho)

Stray Kids

Yes • No

11:23 AM

en.akinator.com/game#

Enter passphrase

PBLApp | Homepage

I THINK OF

Mahendra Singh Dhoni

Cricketer

Yes • No

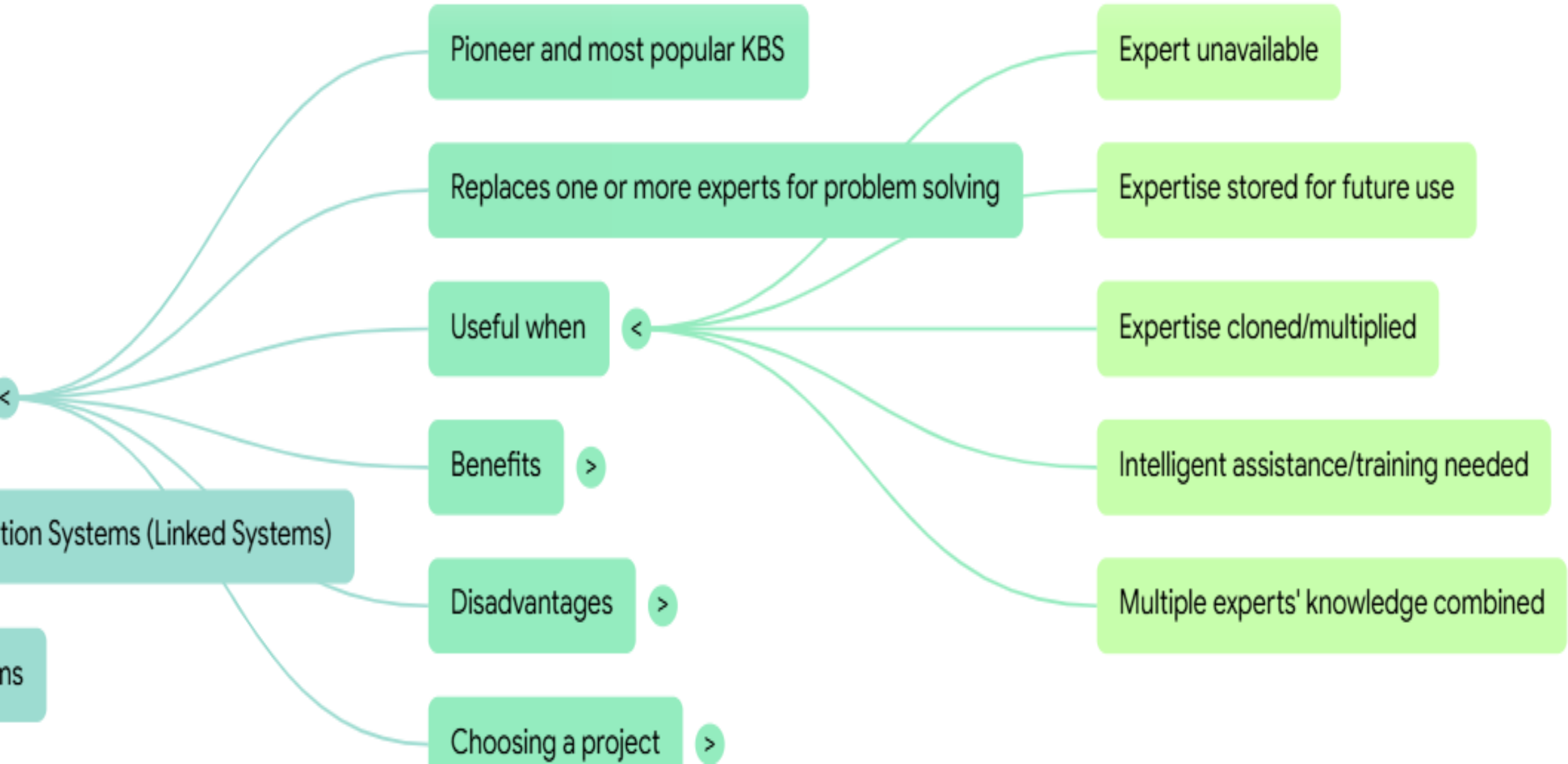
Submitted by Thala

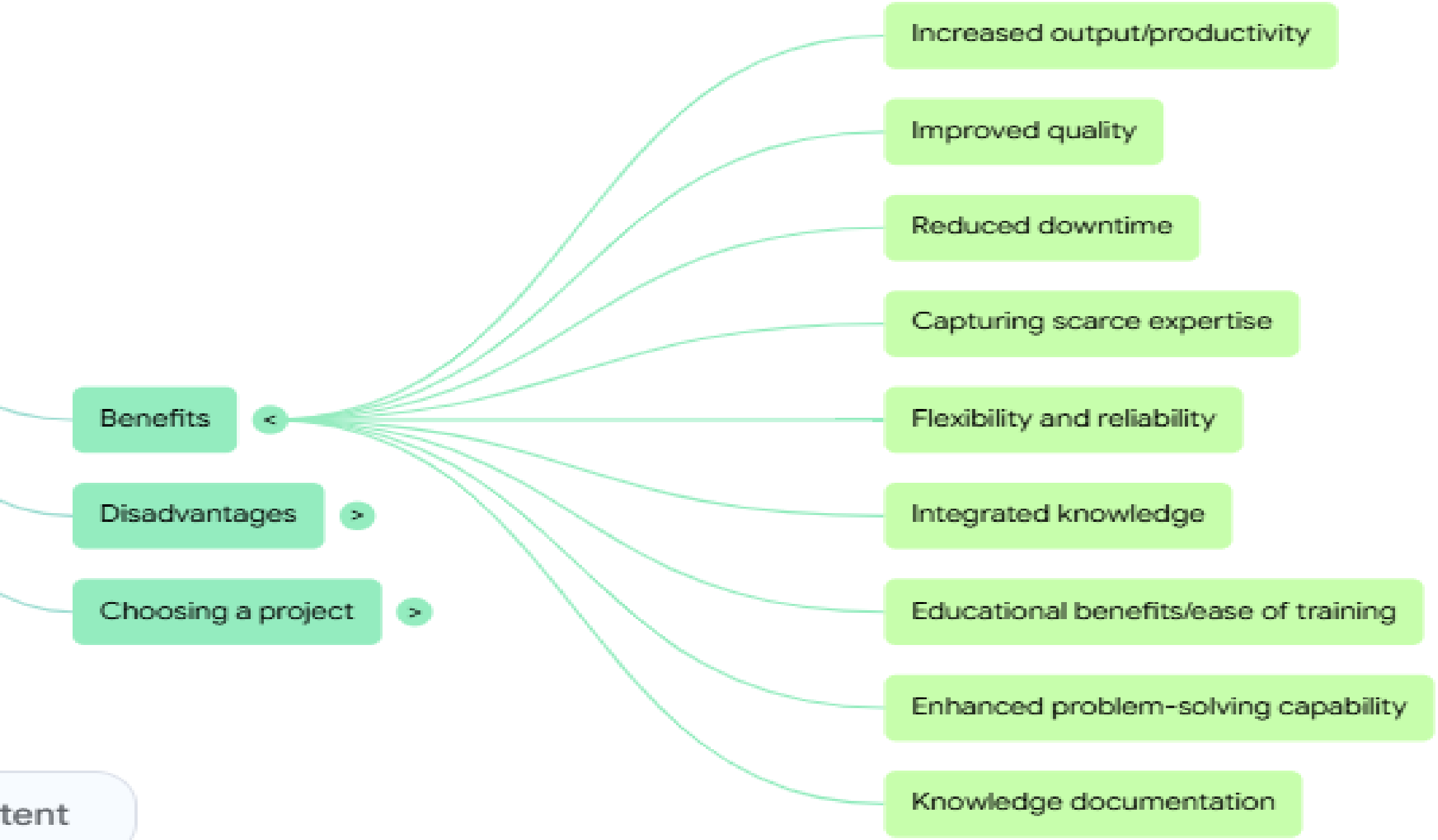
11:15 AM

- Knowledge-Based Systems (KBS)
  - What is a KBS? >
  - Knowledge >
  - Objectives of KBS >
  - KBS vs. Traditional Computer-Based Systems (CBIS) >
  - General Structure of a KBS >
  - Types of KBS <
    - Expert Systems <
      - Pioneer and most popular KBS
      - Replaces one or more experts for problem solving
      - Useful when >
      - Benefits >
      - Disadvantages >
      - Choosing a project >
    - Hypertext Manipulation Systems (Linked Systems)
    - CASE-Based Systems
    - Databases with Intelligent User Interface
    - Intelligent Tutoring Systems
    - Rule-Based Systems
    - Blackboard Systems
    - Classification Systems
  - CommonKADS (Methodology for KBS) >
  - Knowledge Engineering >
  - Knowledge Acquisition >
  - Knowledge Representation >
  - PyKE (Python Knowledge Engine) >
  - Uncertainty Management >
  - Knowledge Validation >
  - Truth Maintenance Systems (TMS) >

Knowledge-Based Systems: Foundations and Applications

Based on 10 sources




Benefits
Disadvantages
Choosing a project
Increased output/productivity
Improved quality
Reduced downtime
Capturing scarce expertise
Flexibility and reliability
Integrated knowledge
Educational benefits/ease of training
Enhanced problem-solving capability
Knowledge documentation

What are Expert System?

- An expert system is a computer program designed to emulate the decision-making ability of a human expert in a specific field. These systems are crafted to solve complex problems, provide accurate advice, and make decisions by leveraging a vast knowledge base.
- Expert systems operate by mimicking the way humans think and solve problems. Instead of relying on rigid programming, they use a set of rules and logic to analyze information and arrive at solutions.
- These systems are designed to handle tasks that typically require human expertise, making them invaluable tools in various industries.

Main Elements of an Expert System

A knowledge-based module. This is where the knowledge is stored in a particular representation.

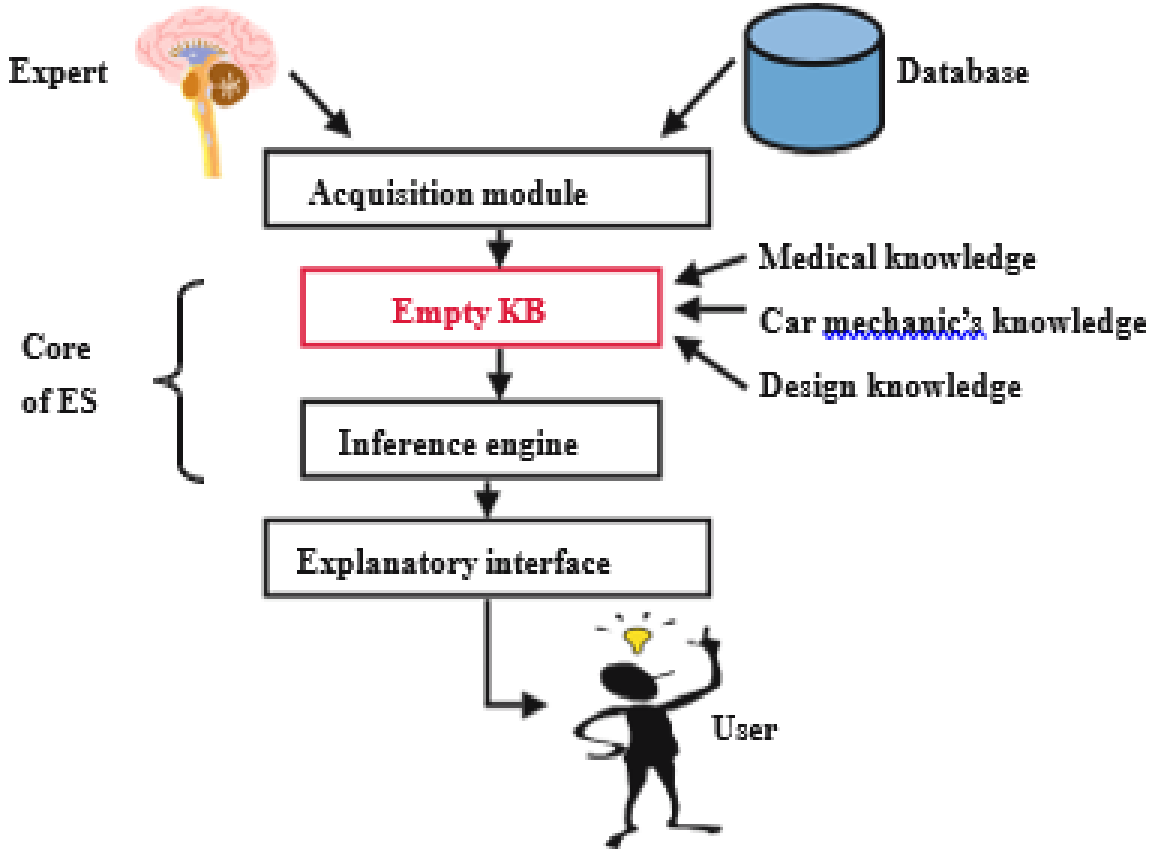


FIGURE 2.2. Elements required in building and using an expert system.

- An explanatory interface with which the human interacts. A knowledge acquisition module that helps when building up new KBs.
- An inference engine. This is a program that uses the knowledge base (KB) to reach conclusions. Clearly, it must understand the format of the KB with which it reasons.

Main Elements of an Expert System

1. Knowledge Base:

- At the heart of every expert system lies a meticulously curated knowledge base.
- This repository encapsulates the expertise of human professionals in a specific field.
- It includes factual information, rules, heuristics, and experiential knowledge that the system draws upon when faced with a problem or query

Main Elements of an Expert System

2. Inference Engine:

- Think of the inference engine as the cognitive powerhouse of the expert system.
- This component processes the information stored in the knowledge base, applying logical reasoning and rule-based analysis to generate solutions or recommendations.
- It mimics the decision-making process of a human expert, adapting its approach based on the nature of the problem.

Main Elements of an Expert System

3.User Interface:

- A user-friendly interface serves as the bridge between the expert system and its users.
- This component allows individuals to input queries, receive recommendations, and interact with the system seamlessly.
- A well-designed interface enhances the user experience and ensures effective communication between the user and the expert system.

Main Elements of an Expert System

4. Explanation Module:

- Transparency is key in the realm of expert systems.
- The explanation module provides users with insights into the system's decision-making process.
- It clarifies the rationale behind the recommendations, making the system more trustworthy and user-friendly.

Use cases

1. Medical Diagnosis:

- In the field of healthcare, expert systems play a crucial role in assisting medical professionals with diagnosis and treatment recommendations. These systems leverage vast medical knowledge to analyze symptoms, medical history, and test results, aiding doctors in making accurate and timely decisions.

2. Financial Planning:

- Expert systems are employed in the financial sector to provide personalized investment advice, risk assessment, and financial planning. By analyzing market trends, economic indicators, and individual financial goals, these systems assist individuals in making informed financial decisions.

3. Manufacturing and Quality Control:

- Industries benefit from expert systems in optimizing manufacturing processes and ensuring product quality. These systems can identify potential issues in real-time, recommend adjustments, and contribute to efficient production and quality assurance.

4. Customer Support:

- In the realm of customer service, expert systems are employed to provide instant support and solutions. These systems can understand customer queries, troubleshoot problems, and offer relevant information, enhancing the overall customer experience.

5. Educational Assistance:

- Expert systems find application in education by providing personalized learning experiences. They can assess individual student performance, tailor educational content, and offer recommendations for improvement, adapting to the unique needs of each learner.

Activity 1: LO 2.1: Design expert system and intelligent agents and identify their components along with a brief assessment of their social and economic impact.
(PI 3.2.2) (PI 6.3.2)

Identify the components for the give use case

Use Case: Educational Assistance System

Purpose: To provide personalized learning recommendations by analyzing student performance data and tailoring content accordingly.

1. Knowledge Base (What the system knows)

- Stores domain knowledge, pedagogical rules, and performance benchmarks.
- Contents: Subject-wise curriculum (Math, Science, etc.)
- Concept difficulty levels (Easy, Moderate, Hard)
- Rules like:
  - IF student score < 40% AND attempts = 1 THEN suggest basics
  - IF score > 85% THEN suggest advanced level material
- Learning styles (Visual, Auditory, Kinesthetic)
- Common student misconceptions and remediation paths

Use Case: Educational Assistance System

Purpose: To provide personalized learning recommendations by analyzing student performance data and tailoring content accordingly.

2. Inference Engine (How the system thinks)

- Applies reasoning to match student data with suitable content.
- Operations: Applies rules:
  - IF student repeatedly fails on concept X → suggest visual tutorial
- Prioritizes learning content based on gaps identified
- Matches learning style + performance to content types

e.g., For a visual learner weak in Geometry → video-based modules

- Chooses whether to recommend revision, new topic, or test

Use Case: Educational Assistance System

Purpose: To provide personalized learning recommendations by analyzing student performance data and tailoring content accordingly.

Component	Example in Educational Assistant System
Knowledge Base	Curriculum, performance rules, learning style database, question banks
Inference Engine	Matches performance to content, applies remedial or advanced rules
User Interface	Student dashboard for inputs and feedback, teacher panel for overrides
Explanation Module	Gives feedback: "Why was this content recommended?" – linked to rules and past performance

Use Case: Educational Assistance System

Purpose: To provide personalized learning recommendations by analyzing student performance data and tailoring content accordingly.

4. Explanation Module (Why the system made a decision)

Explains to the user why certain content was recommended.🗣

Examples:

- ❑You were suggested remedial algebra videos because your last 3 quiz scores were below 50%.
- ❑Advanced topic in Physics unlocked due to your consistent performance above 90%.
- ❑Visual learning content is prioritized based on your success rate with that format.

❑This builds student trust, encourages learning, and helps teachers explain system behavior to parents or stakeholders.

Practice Set1:

Exercise 1: Sports

- **Scenario:**

  An expert system provides personalized training and recovery advice for athletes based on performance data and fatigue levels.

Practice Set 1 :

Component	**What should it contain?** *(Student to fill)*
Knowledge Base	
Inference Engine	
User Interface	
Explanation Module	

Practice set1:Solution

Component	What should it contain? *(Student to fill)*
Knowledge Base	Example: Rules about heart rate thresholds, recovery techniques, training plans for different sports
Inference Engine	Example: Detects overtraining if heart rate remains elevated post-session; suggests rest or light activity
User Interface	Example: Athlete logs daily stats; sees weekly training plan and fatigue alerts
Explanation Module	Example: "Recovery session advised due to HR 15% above baseline for 3 consecutive days."

Practice Set 2:

Agriculture – Crop Recommendation System

- **Scenario:**

  A system suggests suitable crops for a given farm location based on soil type, rainfall, and temperature.

Practice Set2

Component	**What should it contain?** *(Student to fill)*
Knowledge Base	
Inference Engine	
User Interface	
Explanation Module	

Practice Set2:Solution

Component	**What should it contain?** *(Student to fill)*
Knowledge Base	Example: Crop-soil suitability rules, rainfall and temperature ranges per crop
Inference Engine	Example: Matches farm data to crop rules to recommend 2–3 best options
User Interface	Example: Farmer enters soil pH, rainfall, location
Explanation Module	Example: "Wheat recommended due to loamy soil, avg. rainfall, and mild climate in your region."

MyCin:1970, First Expert System

Coding the Cognitive Engines of Tomorrow

Computer-Based Medical Consultations: Mycin

Book • 1976

Author:
Edward Hance Shortliffe

↓ About the book

Browse this book

By table of contents

Purchase book

Share this book

Book description

Computer-Based Medical Consultations: MYCIN focuses on MYCIN, a novel computer-based expert system designed to assist physicians with clinical decisions concerning the selection of ... read full description

Book chapter · No access
Chapter 1 - Introduction
Pages 1-61
Purchase View chapter >

Book chapter · No access
Chapter 2 - Design Considerations for MYCIN
Pages 63-78
Purchase View chapter >

Book chapter · No access
Chapter 3 - Consultation System
Pages 79-158
Purchase View chapter >

Book chapter · No access
Chapter 4 - Model of Inexact Reasoning in Medicine
Pages 159-194
Purchase View chapter >

Book chapter · No access
Chapter 5 - Explanation System
Pages 195-204
Purchase View chapter >

Book chapter · No access
Chapter 6 - Future Directions for MYCIN
Pages 205-232
Purchase View chapter >

Book chapter · No access
Chapter 7 - Conclusion

Mycin

Mycin is a rule-based expert system for diagnosing infections and selecting antibiotics.

What is Mycin?

Mycin is a computer program that was developed in the 1970s at Stanford University. It was one of the first expert systems, and was designed to diagnose and treat infections in humans. Mycin was written in the Lisp programming language, and used a rule-based system to make decisions.

Mycin was able to make diagnoses by asking questions about a patient's symptoms, and then comparing the answers to a database of known

https://en.akinator.com/

Comparison of KBS

Aspect	Expert Systems	Intelligent Agents	Data Mining Systems	Case-Based Reasoning	Genetic Algorithms
Knowledge Acquisition	From experts via interviews, rules				
Knowledge Validation	Expert review, test cases				
Knowledge Representation	Rules, frames, semantic networks				
Inference Engine	Rule-based reasoning				
Explanation Module	Rule tracing				
Use Case	Medical diagnosis				
Example System	MYCIN, DENDRAL				
Pros	Consistent, domain-specific				
Cons	Hard to acquire knowledge				
Best For	Complex decision-making in specific domain				

Activity for LO 2.1: Design expert system and intelligent agents and identify their components along with a brief assessment of their social and economic impact. (PI 3.2.2) (PI 6.3.2)

Intelligent Agents

Intelligent Agent

📌 Definition

- An **Intelligent Agent (IA)** is a **software or hardware-based entity** that:
- **Perceives** its environment through sensors or input data.
- **Processes and reasons** about that information.
- **Acts** autonomously to achieve specified goals.
- **Learns** and adapts based on feedback.

💡 Think of them as **digital decision-makers** — they don't just follow fixed instructions, they choose **what to do next** based on the current state of the environment.

Intelligent Agent

🔍 Key Characteristics

- ❑**Autonomy** – Can operate without constant human intervention.
- ❑**Reactivity** – Responds to changes in the environment.
- ❑**Proactiveness** – Takes initiative to achieve goals.
- ❑**Social Ability** – Can communicate with other agents or humans.

Intelligent Agent

📥 Knowledge Acquisition

- Intelligent Agents gather their knowledge from multiple sources:
- **Sensor inputs** (robotics: cameras, GPS, LiDAR)
- **Online databases** (e.g., weather APIs, traffic data)
- **User interaction** (commands, preferences, feedback)
- **Learning from historical data** (machine learning models)

Intelligent Agent

🎁 Knowledge Representation

- The most common models are:
- **BDI Model (Belief-Desire-Intention)**
  - *Beliefs* → Information the agent knows about the world.
  - *Desires* → Goals or objectives.
  - *Intentions* → Plans the agent commits to achieving.
- **Logic-based models** (predicate logic, production rules)
- **Neural network embeddings** (for learned decision-making)

Intelligent Agent

Inference & Reasoning

- The agent decides **what to do next** based on:
  - Goal prioritization (urgent vs. optional tasks)
  - Rule-based reasoning (IF condition THEN action)
  - Machine learning predictions (next likely best move)
  - Planning algorithms (e.g., A*, Dijkstra, reinforcement learning)

Intelligent Agent

📌 Example Applications

- ❑**Web Crawlers** – Navigate and index the web for search engines.
- ❑**Personal Assistants** – Siri, Alexa, Google Assistant.
- ❑**Robotics** – Autonomous drones, delivery robots.
- ❑**Finance** – Automated stock trading agents.
- ❑**Healthcare** – Remote patient monitoring bots.

Activity for LO 2.1: Design expert system and intelligent agents and identify their components along with a brief assessment of their social and economic impact. (PI 3.2.2) (PI 6.3.2)

Mapping Real-World Intelligent Agents to Knowledge Engineering Activities (Skills: Analytical Thinking , Application of theory)

Domain / Application	Knowledge Acquisition	Knowledge Validation	Knowledge Representation	Inference	Explanation
Web Crawlers (Search Engine Indexing)					
Personal Assistants (Siri, Alexa, Google Assistant)					
Robotics (Autonomous Drones, Delivery Robots)					
Finance (Automated Stock Trading)					
Healthcare (Remote Patient Monitoring)					

Mapping Real-World Intelligent Agents to Knowledge Engineering Activities (Skills: Analytical Thinking , Application of theory)

Domain / Application	Knowledge Acquisition	Knowledge Validation	Knowledge Representation	Inference	Explanation
Web Crawlers (Search Engine Indexing)	Collects web pages via links, sitemaps	Removes duplicates, irrelevant, or broken pages	Inverted index, web graph structure	Determines page importance and ranking (e.g., PageRank)	"Result shown due to keyword match and relevance score"
Personal Assistants (Siri, Alexa, Google Assistant)	Captures user voice, retrieves from APIs and apps	Confirms speech recognition accuracy, validates sources	Semantic model / knowledge graph	Matches intent to action using NLP + rules	"Here's your weather based on current forecast data"
Robotics (Autonomous Drones, Delivery Robots)	Uses cameras, GPS, LiDAR, sensors	Cross-checks sensor data with maps and GPS	Spatial maps, 3D grids	Plans routes avoiding obstacles	"Changed route due to detected obstacle at location X"
Finance (Automated Stock Trading)	Monitors market prices, news feeds, reports	Filters unreliable or outdated data	Time-series data, technical indicators	Applies trading algorithms and predictions	"Bought stock X because moving average crossed upward"
Healthcare (Remote Patient Monitoring)	Reads wearable device data (heart rate, BP, glucose)	Checks for anomalies or device errors	Time-series patient health records	Detects abnormal patterns, triggers alerts	"Alert due to BP above safe threshold for 3 consecutive readings"

Intelligent Agent

🖥 Example Systems in Use

- **Google Assistant** – Understands speech, fetches data, executes tasks.
- **Roomba Robot Vacuum** – Navigates rooms, adapts cleaning patterns.

https://www.youtube.com/watch?v=XIPzSmwClJ8

- **Amazon Scout** – Small autonomous delivery robot.

https://www.youtube.com/watch?v=peaKnkNX4vc

Intelligent Agent

- Pros
  - Real-time adaptability to dynamic environments.
  - Reduces human workload in repetitive or hazardous tasks.
  - Can integrate diverse data sources for informed decisions.
- Cons
  - Dependent on training data quality (bias, incompleteness).
  - Privacy concerns due to continuous data collection.
  - Vulnerable to hacking or manipulation.
  - Decision-making may be non-transparent (black-box models).

From Theory to Practice: Mapping Modern AI Tools to KBS Architecture

Question1: Is chatbot in a pharma mobile app (like a Tata 1mg chatbot) or a website navigation chatbot can be considered an Intelligent Agent???

From Theory to Practice: Mapping Modern AI Tools to KBS Architecture

- Quick Demo of chatbots:
  - https://www.hdfcbank.com/
  - https://icnte.fcrit.ac.in/
- Cleverbot (Casual Conversational Agent)
- ServiceNow Virtual Agent (Enterprise Support Agent)
- LiveChatAI Demo Bots (Domain-Specific Assistants)
  - **What it offers**: Ready-made AI bots trained on websites like Stripe or Notion for support tasks.
  - **Why useful**: Ideal for students to interact with realistic, goal-oriented agents.
  - **Try it**: Explore demo bots at [LiveChatAI Demo]Live Chat AI.
- Replika (Emotional Companion AI)
  - **What it does**: A conversational agent designed to form emotional connections with users.
  - **Why useful**: Excellent example of personalization and long-term interaction behavior in agents.

From Theory to Practice: Mapping Modern AI Tools to KBS Architecture

- **Key point for students:**
  Not all chatbots are Intelligent Agents. If the chatbot is just showing **predefined answers** without **reasoning or adapting**, it's not truly "intelligent." The intelligent tag applies when it **autonomously processes, reasons, and improves**.

Optional Extension: Think of developing a chatbot.....

From Theory to Practice: Mapping Modern AI Tools to KBS Architecture

Question2: Is chatgpt is an intelligent agent???

From Theory to Practice: Mapping Modern AI Tools to KBS Architecture

Answer: Yes — **ChatGPT qualifies as an Intelligent Agent**, but it's a **special type** called a **conversational AI agent** or **Large Language Model (LLM)-powered agent.**

From Theory to Practice: Mapping Modern AI Tools to KBS Architecture

1. Knowledge Acquisition
ChatGPT is trained on **a vast dataset** from books, websites, research papers, and code repositories (pre-training phase).
In use, it acquires **context** from the user's input (your prompts, instructions).

2. Knowledge Representation
Internally, ChatGPT doesn't store knowledge as a rule base or database.
Instead, it represents knowledge in **billions of parameters** (weights) in a deep neural network — a compressed statistical representation of patterns in language.

3. Inference Engine
Uses **probabilistic reasoning** to predict the most likely next words or sentences.
Adapts to your intent based on **context window** (recent conversation history).

4. User Interface
Interacts through a chat interface (text or voice in some versions).
Can be integrated into apps, websites, or other systems

5. Learning / Adaptation
- ❑ The **base model** doesn't learn from your single conversation (no memory by default), but OpenAI fine-tunes models over time using aggregated, anonymized data.
- ❑ Some versions (like ChatGPT with custom instructions) adapt to your preferred style.
- ❑ In tools like Auto-GPT or LangChain, ChatGPT can be embedded into **agents that continuously learn** during a session.

From Theory to Practice: Mapping Modern AI Tools to KBS Architecture

Feature	**ChatGPT (LLM-based Intelligent Agent)**	**Classic Expert System**
Knowledge Acquisition	Learned from massive datasets (books, articles, web content) using machine learning; can adapt to new inputs in real time during a session.	Acquired from domain experts via interviews, manuals, or structured data entry.
Knowledge Representation	Stored as **neural network weights** (billions of parameters) representing statistical relationships in language.	Stored as **rules**, **facts**, and **ontologies** in a structured knowledge base.
Inference Engine	Uses probabilistic reasoning and deep learning to predict the next best output.	Uses logical reasoning (forward chaining, backward chaining) to apply rules and facts.
Transparency / Explainability	Reasoning is a **black box** — difficult to explain exact decision path.	Reasoning is transparent — can trace rules used to reach the conclusion.
Adaptability	Can handle open-ended, multi-domain queries; adapts its style and scope in real time.	Domain-specific; needs manual updates to handle new knowledge or rules.
Learning	Improves over time through fine-tuning and reinforcement learning from user feedback.	Does not learn automatically; requires explicit rule updates by knowledge engineers.
User Interface	Natural language chat interface (text/voice); conversational.	Often uses forms, menus, or text prompts.
Applications	Education, coding help, content creation, brainstorming, tutoring, customer support.	Medical diagnosis, equipment troubleshooting, legal advice, financial planning.
Strengths	Highly flexible, handles ambiguous queries, multi-domain knowledge.	Precise, explainable reasoning in well-defined domains.
Limitations	Can produce inaccurate or fabricated info (hallucinations); lacks inherent truth verification.	Limited to knowledge entered by experts; struggles outside its domain.

From Theory to Practice: Mapping Modern AI Tools to KBS Architecture

Question3: Is **Google NotebookLM** — Google's AI-powered note-taking and research assistant is an intelligent agent ???

From Theory to Practice: Mapping Modern AI Tools to KBS Architecture

Answer: Yes, **NotebookLM can also be seen as an Intelligent Agent**, but it's **task-focused** rather than general-purpose like ChatGPT.

From Theory to Practice: Mapping Modern AI Tools to KBS Architecture

Component	How NotebookLM Does It
Knowledge Acquisition	Takes input from your uploaded documents, notes, and references. Can also pull data from Google Docs or Drive.
Knowledge Representation	Organizes the ingested content into an internal text-based representation for reasoning.
Inference Engine	Uses a Large Language Model (LLM) to summarize, answer questions, and make connections between your uploaded materials.
User Interface	Simple web interface where you can upload docs, ask questions, and get summaries.
Learning/Adaptation	Adapts its answers to your specific uploaded materials — works within that knowledge scope.

From Theory to Practice: Mapping Modern AI Tools to KBS Architecture

KBS Component	**Rule-Based Chatbot** (e.g., website FAQ bot)	**ChatGPT** (LLM-based Intelligent Agent)	**Google NotebookLM** (Document-based Intelligent Agent)
Knowledge Acquisition	Manually entered scripts and question–answer pairs by developers.	Trained on massive datasets from books, articles, web data; updated via fine-tuning and user feedback.	Acquires knowledge from user-uploaded documents, notes, and linked Google Docs.
Knowledge Validation	Manually tested for accuracy; fixed scripts ensure no deviation.	Indirect validation through training data curation and RLHF (Reinforcement Learning from Human Feedback).	Cross-checks only within uploaded documents, reducing hallucinations.
Knowledge Representation	Structured as decision trees, keyword lists, or if–then rules.	Encoded as billions of parameters in a neural network — statistical relationships between tokens.	Represents uploaded content in text embeddings and indexed form for retrieval.
Inference Engine	Matches user input to pre-defined rules or templates.	Uses probabilistic reasoning to predict next best token and generate answers dynamically.	Uses retrieval-based LLM reasoning over uploaded content to answer queries.
Explanation Module	Limited — may only display "Here's the answer I found."	No transparent reasoning path — answers are generated without showing steps.	Can cite and highlight text from uploaded docs as justification.
Adaptability	Low — only works for programmed questions; changes require manual updates.	High — can handle multi-domain, open-ended queries in real time.	Medium — adapts to the scope of provided documents; domain-limited but accurate.
Example Use Case	Banking FAQ bot answering branch hours or service info.	Coding help, brainstorming, tutoring, multi-domain Q&A.	Summarizing lecture notes, creating study guides, generating quizzes from documents.

Comparison of KBS

Aspect	Expert Systems	Intelligent Agents	Data Mining Systems	Case-Based Reasoning	Genetic Algorithms
Knowledge Acquisition	From experts via interviews, rules	Sensor data, environment feedback			
Knowledge Validation	Expert review, test cases	Environment feedback loops			
Knowledge Representation	Rules, frames, semantic networks	BDI models, logic, states			
Inference Engine	Rule-based reasoning	Goal-directed reasoning			
Explanation Module	Rule tracing	Action justification			
Use Case	Medical diagnosis	Virtual assistants			
Example System	MYCIN, DENDRAL	Google Assistant			
Pros	Consistent, domain-specific	Real-time adaptability			
Cons	Hard to acquire knowledge	Limited by perception/training			
Best For	Complex decision-making in specific domain	Autonomy & interaction			

Data Mining

Data mining Systems

- **Definition**
- Data Mining Systems are **Knowledge-Based Systems** that **automatically discover hidden patterns, correlations, trends, and relationships** from large datasets using statistical, machine learning, or AI techniques.

Data mining Systems

Component	How It Works in Data Mining Systems	Example
Knowledge Acquisition	Data is collected from large datasets (databases, data warehouses, online logs, IoT sensor data). Data mining algorithms like classification, clustering, association rule mining, regression, and anomaly detection are applied to discover patterns.	Weka importing CSV and applying association rule mining to find “People who buy X also buy Y.”
Knowledge Validation		
Knowledge Representation		
Inference		
Explanation Module		

Data mining Systems

Component	How It Works in Data Mining Systems	Example
Knowledge Acquisition	Data is collected from large datasets (databases, data warehouses, online logs, IoT sensor data). Data mining algorithms like classification, clustering, association rule mining, regression, and anomaly detection are applied to discover patterns.	Weka importing CSV and applying association rule mining to find “People who buy X also buy Y.”
Knowledge Validation	Discovered patterns are validated using statistical tests (e.g., confidence, lift, p-value) or tested against a validation dataset to ensure reliability and relevance.	Testing fraud detection model on historical transaction data to ensure <5% false positives.
Knowledge Representation		
Inference		
Explanation Module		

Data mining Systems

Component	How It Works in Data Mining Systems	Example
Knowledge Acquisition	Data is collected from large datasets (databases, data warehouses, online logs, IoT sensor data). Data mining algorithms like classification, clustering, association rule mining, regression, and anomaly detection are applied to discover patterns.	Weka importing CSV and applying association rule mining to find “People who buy X also buy Y.”
Knowledge Validation	Discovered patterns are validated using statistical tests (e.g., confidence, lift, p-value) or tested against a validation dataset to ensure reliability and relevance.	Testing fraud detection model on historical transaction data to ensure <5% false positives.
Knowledge Representation	Patterns can be represented as **decision trees**, **association rules** (If X → Y), **clusters** (grouping similar data points), or **predictive models** (e.g., regression formulas).	Decision tree for predicting customer churn.
Inference		
Explanation Module		

Data mining Systems

Component	How It Works in Data Mining Systems	Example
Knowledge Acquisition	Data is collected from large datasets (databases, data warehouses, online logs, IoT sensor data). Data mining algorithms like classification, clustering, association rule mining, regression, and anomaly detection are applied to discover patterns.	Weka importing CSV and applying association rule mining to find “People who buy X also buy Y.”
Knowledge Validation	Discovered patterns are validated using statistical tests (e.g., confidence, lift, p-value) or tested against a validation dataset to ensure reliability and relevance.	Testing fraud detection model on historical transaction data to ensure <5% false positives.
Knowledge Representation	Patterns can be represented as **decision trees**, **association rules** (If X → Y), **clusters** (grouping similar data points), or **predictive models** (e.g., regression formulas).	Decision tree for predicting customer churn.
Inference	Uses discovered patterns to make predictions or classify new data points. In ML-based systems, the model infers outcomes based on learned relationships.	Predicting whether a transaction is fraudulent in real time.
Explanation Module		

Data mining Systems

Component	How It Works in Data Mining Systems	Example
Knowledge Acquisition	Data is collected from large datasets (databases, data warehouses, online logs, IoT sensor data). Data mining algorithms like classification, clustering, association rule mining, regression, and anomaly detection are applied to discover patterns.	Weka importing CSV and applying association rule mining to find “People who buy X also buy Y.”
Knowledge Validation	Discovered patterns are validated using statistical tests (e.g., confidence, lift, p-value) or tested against a validation dataset to ensure reliability and relevance.	Testing fraud detection model on historical transaction data to ensure <5% false positives.
Knowledge Representation	Patterns can be represented as **decision trees**, **association rules** (If X → Y), **clusters** (grouping similar data points), or **predictive models** (e.g., regression formulas).	Decision tree for predicting customer churn.
Inference	Uses discovered patterns to make predictions or classify new data points. In ML-based systems, the model infers outcomes based on learned relationships.	Predicting whether a transaction is fraudulent in real time.
Explanation Module	Explains which variables and patterns influenced a decision — often using visualizations (heatmaps, decision tree paths, cluster diagrams).	Market basket analysis chart showing top product combinations.

Data Mining systems

- **Applications**
  - **Market Basket Analysis** – Retailers find product-purchase relationships.
  - **Fraud Detection** – Banking systems detect unusual transaction patterns.
  - **Healthcare Analytics** – Identifying risk factors for diseases based on patient history.
  - **Customer Segmentation** – Grouping customers for targeted marketing.

Data Mining systems

- **Applications**
  - **Market Basket Analysis** – Retailers find product-purchase relationships.
  - **Fraud Detection** – Banking systems detect unusual transaction patterns.
  - **Healthcare Analytics** – Identifying risk factors for diseases based on patient history.
  - **Customer Segmentation** – Grouping customers for targeted marketing.

Data mining Systems

Component	How It Works in Data Mining Systems	Example
Knowledge Acquisition	Data is collected from large datasets (databases, data warehouses, online logs, IoT sensor data). Data mining algorithms like classification, clustering, association rule mining, regression, and anomaly detection are applied to discover patterns.	Weka importing CSV and applying association rule mining to find “People who buy X also buy Y.”
Knowledge Validation	Discovered patterns are validated using statistical tests (e.g., confidence, lift, p-value) or tested against a validation dataset to ensure reliability and relevance.	Testing fraud detection model on historical transaction data to ensure <5% false positives.
Knowledge Representation	Patterns can be represented as **decision trees**, **association rules** (If X → Y), **clusters** (grouping similar data points), or **predictive models** (e.g., regression formulas).	Decision tree for predicting customer churn.
Inference	Uses discovered patterns to make predictions or classify new data points. In ML-based systems, the model infers outcomes based on learned relationships.	Predicting whether a transaction is fraudulent in real time.
Explanation Module	Explains which variables and patterns influenced a decision — often using visualizations (heatmaps, decision tree paths, cluster diagrams).	Market basket analysis chart showing top product combinations.

Comparison of KBS

Aspect	Expert Systems	Intelligent Agents	Data Mining Systems	Case-Based Reasoning	Genetic Algorithms
Knowledge Acquisition	From experts via interviews, rules	Sensor data, environment feedback	From datasets via algorithms		
Knowledge Validation	Expert review, test cases	Environment feedback loops	Statistical validation		
Knowledge Representation	Rules, frames, semantic networks	BDI models, logic, states	Decision trees, clusters, patterns		
Inference Engine	Rule-based reasoning	Goal-directed reasoning	Statistical/ML reasoning		
Explanation Module	Rule tracing	Action justification	Pattern explanation		
Use Case	Medical diagnosis	Virtual assistants	Fraud detection		
Example System	MYCIN, DENDRAL	Google Assistant	Weka		
Pros	Consistent, domain-specific	Real-time adaptability	Finds hidden patterns		
Cons	Hard to acquire knowledge	Limited by perception/training	Dependent on data quality		
Best For	Complex decision-making in specific domain	Autonomy & interaction	Data-driven decision-making		