Knowledge Based Systems and Expert Systems

Introduction

- What is a knowledge-based system?
  - A system which is built around a knowledge base. i.e. a collection of knowledge, taken from a human or any other sources, and stored in such a way that the system can *reason* with it.
- What is knowledge?
  - Knowledge is the sort of information that people use to solve problems.

Expert System - definition

- What is an expert system?
  - A particular kind of knowledge-based system, One in which the knowledge, stored in the knowledge base, has been taken from an expert in some particular field.
  - “AI Programs that achieve expert-level competence in solving problems by bringing to bear a body of knowledge [Feigenbaum, McCorduck & Nii 1988]”

Expert Systems Cont'd

- An Intelligent computer program system with large amount of expert level's knowledge and experience, handle problem in this field by using:
  - Heuristic

  Reasoning, judgment and decision making.

  Most part of work and knowledge are Non-mathematical
  - Transparency

  Explanation and answering questions
  - Flexibility

  Ability of Self-learning and improving itself over time

Expert Systems cont’d

- One of the most successful applications of AI reasoning technique using facts and rules

- Arguably, an expert system can, to a certain extent, act as a substitute for the expert from whom the knowledge was taken.

Advantages of ES

1. High efficiency, without tiredness
2. Not influenced by surroundings or emotions
3. No time and space limit
4. Can be used to promote different fields
5. Can be used to gather multi-field expert's knowledge and experience together
6. Great economic and social benefit
7. Promote the whole science and technique level by requiring knowledge to be explicit represented

Types of ES/ Areas that ES have been utilised

- Interpretation
- Prediction
- Diagnosis
- Design
- Planning
- Monitoring
- Control
- Debugging
- Instruction
- repair

Expert system for Interpretation

- Task

To infer situation description from observations i.e ascertain the meaning of some known information by analysis and explanation.

- Features
  - Data very large, imprecise and incomplete
  - Explanation and some hypothesis could be made
  - Long and complicated reasoning process
- Examples

Voice understanding, image analysis

Expert system for Prediction

- Task

To deduce the probable future state by analyzing the past and current state.

- Features
  - Data changed with time, imprecise and incomplete
  - Dynamic model that adapts to the change of time needed, forecast can be obtained, fast response
- Examples

Weather forecast, military forecast, population forecast

Expert system for Diagnosis

- Task

to deduce the cause of fault by observing

- Features
  - Understand the features of object and sub-objects and their relationship
  - Separate one phenomenon from another phenomenon
  - Input: measured data; output: diagnosis
- Examples

Medical diagnosis

Expert system for Design

- Task

To find out assignment satisfying constraint, based on design requirement

- Features
  - Finding design result satisfying multi-constraint
  - Bigger possible solution space
  - Construct possible design, easy to revise
  - Explain the current new design by using already existing ones
- Examples

Digit circuit and integrated circuit design

Computer structure design

Expert system for Planning

- Task

To find out a sequence of motion or step that can lead to target

- Features
  - Target could be dynamic or static, predict for future motion should be made
  - Complicated problem involved
- Examples

Transportation planning

Military commanding planning

Expert system for Monitoring

- Task

Fault detection, by observing, comparing, alarm.

- Features
  - Fast response ability, give alarm before accident
  - Accuracy , no false alarm
- Examples

Safe monitoring for nuclear plant

Expert system for Control

- Task

To manage self-adaptively, a controllable object such that a requirement is satisfied.

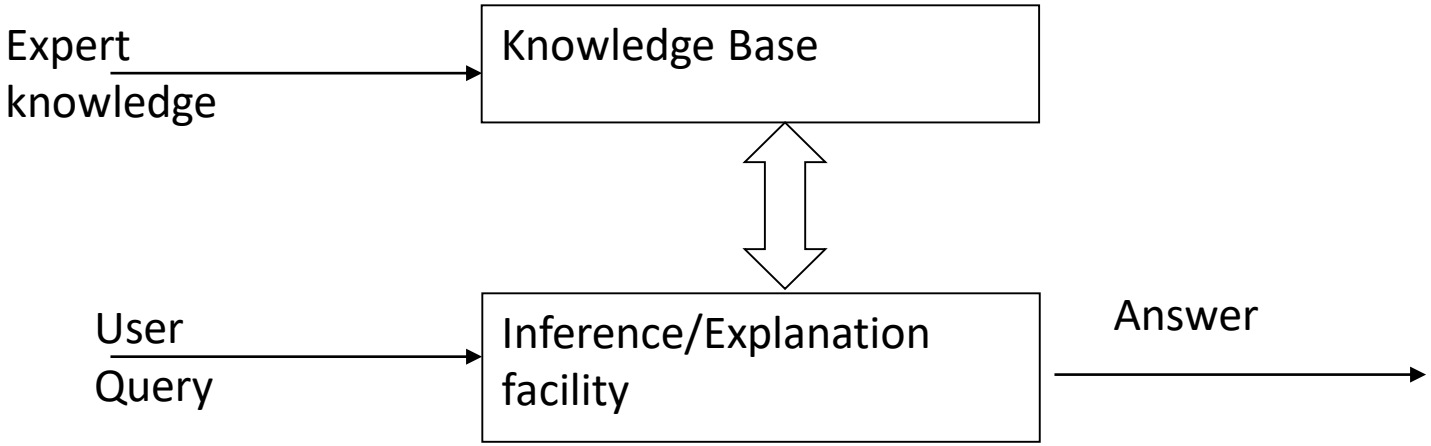
- Features
  - Explanation
  - Forecast
  - Diagnosis
  - Planning
  - Executing
- Examples

Business management

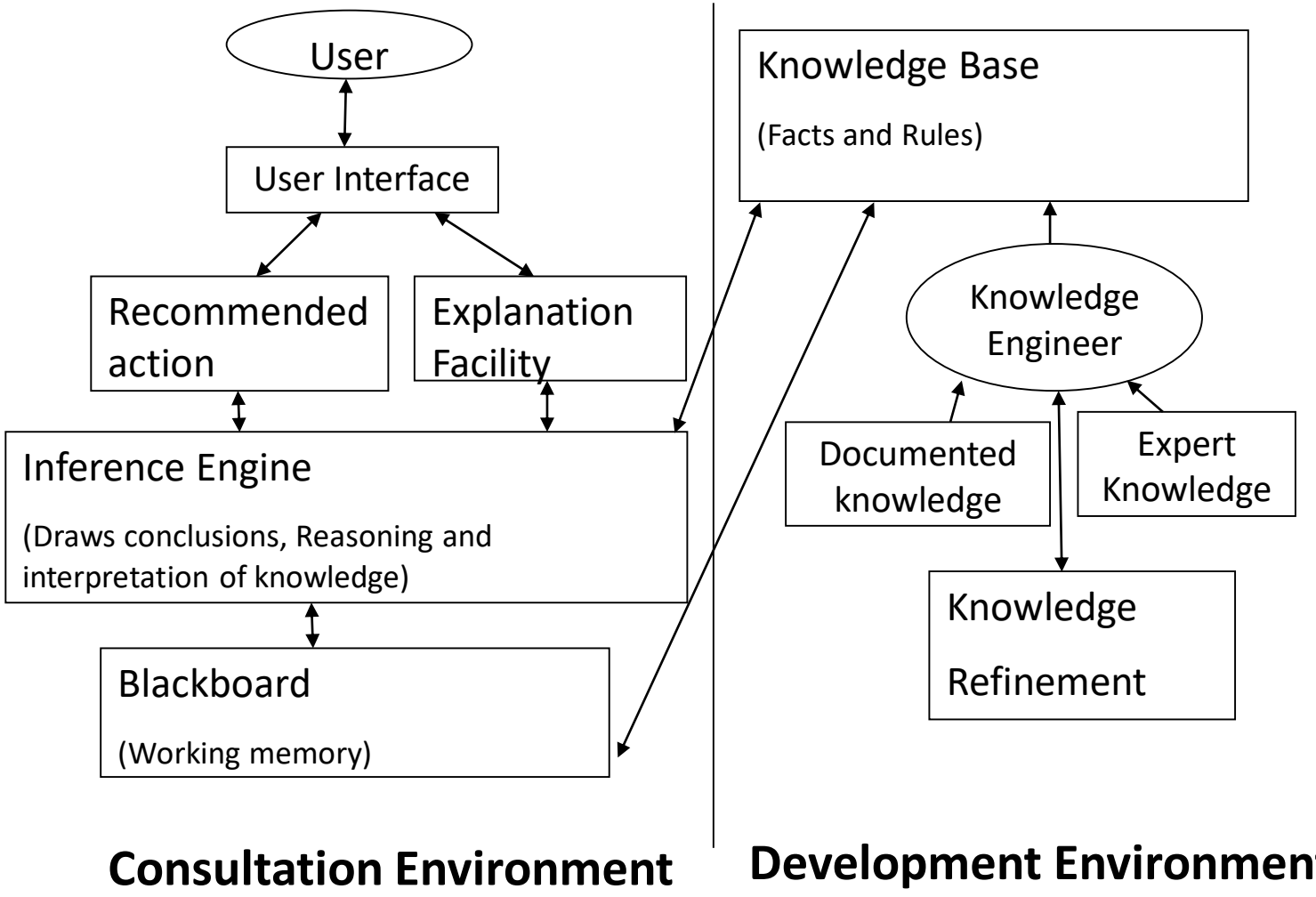
Production process control

Production quality control

Simplified Architecture



Detailed ES Architecture



Knowledge base

- Contains the knowledge needed for understanding, formulating and solving problems
- Includes facts and theory in the problem area
- Structures used are called knowledge representation formalisms and include: rules, semantic nets, frames, propositions and predicates

Inference Engine

- It’s the brain of the ES
- Also called control structure or rule interpreter
- Provides methodology and implementation of the reasoning mechanism that uses the knowledge in the KB and blackboard to formulate conclusions
- It develops the agenda that organizes and controls the steps taken to solve problems during consultation

Inference Engine Cont'd

It consists of:

- Interpreter – that executes selected agenda items by applying relevant rules from KB
- Schedule – maintains control over the agenda. Uses priorities and other criteria to estimate effects of applying rules on the agenda
- Consistency enforcer – maintains a consistent representation of the emerging soln

User interface

- Enables the users and builders to submit their items to ES and also ES to respond to their queries

Blackboard

- Area in working memory where current problem are described, data is entered and intermediate results kept
- May contain a plan of how the problem should be solved, an agenda of potential actions awaiting execution and solutions indicating candidate hypothesis and alternate courses of action

Explanation Subsystem

- Also known as a justifier
- It traces responsibility for conclusions to their sources
- It may explain:
  - How some conclusions are reached
  - Why some alternatives are rejected
  - The plan used to reach a solution etc

Knowledge refining system

- It enables the analysis and use of knowledge so that learning may take place and improvements may be made

Participants in ES

- The Expert – person who has special knowledge, judgement, experience or methods used to solve problems or advice on a given class of problems
- The Knowledge Engineer – responsible for extracting and structuring knowledge from sources such as human experts etc. Many times he/she may be the ES builder also

Participants in ES

- The User – uses the ES during the consulting seeking for advice or solutions to a problem

Development cycle of a knowledge-based system

- 1. Plan knowledge base (the content of the knowledge base, relevant inputs and outputs, strategy for testing, knowledge dictionary, concepts etc. are identified.)
- 2. Select domain experts and knowledge sources
- 3. Acquire (elicit) knowledge
- 4. Formulate and represent knowledge (knowledge is formulated in the form suitable for inference)
- 5. Implement knowledge base (knowledge is encoded in machine-readable form.
- 6. Test knowledge base depending on the results: continue with knowledge acquisition or go to 7.
- 7. Systems test

Software for building ES

- It ranges from software used for building ES to programs that can aid the knowledge acquisition process.

It includes:

- Development software – specialized AI languages (PROLOG and LISP) as well as general purpose languages; shells (e.g CLIPS, JESS) etc

Software for building ES

- Development support tools – I/O facilities; debugging aids; explanation facilities; editors
- Systems building tools – e.g. rule induction engines

KBS Comparison

Human Expert	KBS	Conventional programs
Use knowledge in the form of heuristics to solve problems in a narrow domain	Process knowledge expressed in the form of rules and uses symbolic reasoning to solve problems in a narrow domain	Process data and use algorithms, a series of well –defined operations, to solve general numerical problems
In a human mind knowledge exists in a compiled form	Provides a clear separation of knowledge from its processing	Do not separate knowledge from control structures to process the knowledge
Capable of explaining a liine of reasoning and providing the details	Traces the rules fired during a problem-solving session and explains how a particular conclusion was reached and why specific data was needed	Do not explain how a particular result was obtained and why input data was needed
Use inexact reasoning and can deal with incomplete, uncertain and fuzzy information	Permit inexact reasoning and can deal with incomplete, uncertain and fuzzy data	Work only on problems where data is complete and exact

Knowledge acquisition

- It’s the process of extracting knowledge(facts, procedures, rules) from human experts, books, documents, sensors or computer files and converting it into a form that can be stored and manipulated by the computer for purposes of problem solving

Knowledge acquisition process

- Identification: identify the problem including data, criteria for solution etc
- Conceptualization: determine the key concepts and relationships by characterizing the data, flow of information, the domain structure etc
- Formalization – understand underlying search space, uncertainty issues etc

Knowledge acquisition process

- Implementation – translate acquired knowledge into a program
- Testing – validate and verify

Knowledge Elicitation

- Implies that knowledge acquisition is accomplished from a human expert

Sources of knowledge

- Written sources – books, manuals etc
- Experts
- Observation of actual process etc

Knowledge acquisition methods

There are two basic strategies:

- Starting from general concepts leading the expert to elicit details of a topic – top-down (or deductive) methods
- Starting from details of specific cases and helping the expert establish and derive general concepts from the specific examples – bottom-up (or inductive) methods

Top-Down Methods

Can be grouped into four categories:

- Questioning methods
- Object-oriented methods
- Quantitative methods
- Inventive methods

Questioning methods

- KE interviews the expert in a series of meetings or ask the expert to fill out a questionnaire
- Strategies involve:
  - Structured interviews – systematic and goal oriented
  - Unstructured interviews – informal and can be eye opener to the KE
  - Questionnaires

Object-oriented methods

- KE focuses the interview session on discovering the objects within the domain
- Expert is made to group actual objects in the field in order to form a class of objects that has a common set of attributes

Quantitative methods

- These methods were developed in cognitive science and decision analysis for eliciting the degree of a decision maker's preferences and utilities, and in grouping various objects and attributes
- Used to measure and determine:
  - Extent of relationships among objects (or objects)
  - Degree of uncertainty about the domain knowledge

Inventive methods

Expert is allowed a more active part in the process in one of the following roles:

- Expert as a teacher
- Expert as a partner in systematic innovation – e.g. discovering contradictions and solns of removing them
- Expert as KE

Bottom-Up methods

- Focuses on specific cases, and experts absract the decision for resolving a specific case to a more generalized rule or concept and includes:
  - Example-based methods
  - Protocol analysis
  - Observations of the experts decision making process

Example-based methods

- Constitutes foundation *of case-based learning and learning by analogy*
- Knowledge engineer and expert work on a number of representative cases or examples by:
  - Grouping examples
  - Walk-through examples
  - Quantitative analysis of examples (statistical or inductive methods)

Protocol analysis

- Expert is asked *to think aloud and verbalize his/her thought process while solving a set of actual (or simulated) problems and making decisions*
- *KE records the process and later analyzes information produced to develop general rules expert uses in solving problems*

Observation

- Observe expert while solving a problem
- Useful for procedural solutions

Knowledge acquisition modes

- Manual – direct interaction between KE and expert
- Automated – using some software to elicit knowledge either by expert interacting with a computer or a system fed with examples to produce rules or decision trees
- Combination of the two

Validation and Verification

Collectively known as evaluation techniques

- Validation – deals with performance of the system i.e. how well it is making conclusions
- Verification – deals with building the system to correct specification; reliability, breadth and depth, adequacy, adaptability sensitivity etc