Table 1.1 An Overview of the Key Events in the History of Artificial Intelligence

Period	Key Events
Pre-World War II roots	Formal logic Cognitive psychology
The postwar years, 1945–1954 *Pre-AI*	Computers developed H. Simon, *Administrative Behavior* N. Wiener, *Cybernetics* A. M. Turing, "Computing Machinery and Intelligence" Macy Conferences on Cybernetics
The formative years, 1955–1960 *The initiation of AI research*	Growing availability of computers Information Processing Language I (IPL-I) The Dartmouth Summer Seminar on AI, 1956 General Problem Solver (GPS) Information processing psychology
The years of development and redirection, 1961–1970 *The search for general problem solvers*	A. Newell and H. Simon, *Human Problem Solving* LISP Heuristics Satisficing Robotics Chess programs DENDRAL (Stanford)
The years of specialization and success, 1971–1980 *The discovery of knowledge-based systems*	MYCIN (Stanford) HEARSAY II (Carnegie-Mellon) MACSYMA (MIT) Knowledge engineering EMYCIN (Stanford) GUIDON (Stanford) PROLOG Herbert Simon—Nobel Prize
The rush to applications, 1981– *International competition and commercial ventures*	PROSPECTOR (SRI) Japan's Fifth-Generation Project E. Feigenbaum and P. McCorduck, *The Fifth Generation* U.S.'s Microelectronics & Computer Technology Corp. (MCC) INTELLECT (A.I.C.) Various corporate and entrepreneurial AI companies

LECTURE: Knowledge Representation
MIS 680 – Spring 1992

1 Overview

- Types of knowledge: objects, events, meta-knowledge, etc.
- Characteristics of representation: expressive adequacy vs. notational adequacy.
  - scope and grain size.
  - semantic primitives.
  - modularity and understandability.
  - explicit knowledge and flexibility.
  - declarative vs. procedural.
  - closeness to the real world.
  - closeness to the human intelligence.
- A taxonomy of knowledge representation schemes:
  - semantic network.
  - frame/script.
  - logic.
  - production system.
  - Others: direct (analogical) representation, procedural representation, etc.

2 Semantic Network

- Definition: Use nodes and links to represent the knowledge. Nodes indicate objects, concepts, situations, etc. Links represent relationships between objects.
- Important concepts: property inheritance, spreading activation, default reasoning, etc.
- Sample systems: (SCHOLAR, by Carbonell, 1974) (METACAT, by Chen, 1990).
- Advantages: understandability, excellent for information storage and retrieval, human memory-based, and easy to use.
- Disadvantages: lack of formal semantics and terminology, and hard to traverse in large network.

3 Frame/Script

- Definition: Structured representation for stereotypical objects (frame) and events (script).
- Important concepts: frame-based reasoning, slot-filling, procedural attachment, default reasoning, etc.
- Sample systems: (SAM, Schank and Abelson, 1977), (KRL, Bobrow and Winograd, 1977)
- Advantages: similar to semantic network, creates chunks of knowledge.
- Disadvantages: May be suitable for certain applications only.

4 Logic

- Definition: Employ the notions of constant, variable, function, predicate, logical connective and quantifier in order to represent facts as logical formulas.
- Important concepts: resolution proof, logic-based reasoning.
- Sample systems: (STRIPS, SRI, 1972)
- Advantages: availability of inferencing rules for proof, precise semantics, and simplicity of notation.
- Disadvantages: lack of organization principles for facts, difficult to represent procedural knowledge.

5 Production System

- Definition: Use rules and facts to represent knowledge.
- Important concepts: production rules, blackboard architecture, expert systems.
- Sample systems: (MYCIN, Shortliffe, 1977), (AM, Lenat, 1979), (INTERNIST, Popple, 1979).
- Advantages: simulate human knowledge organization, modularity of rules and facts, declarative.
- Disadvantages: For certain applications only, hard to control the flow of logic.

6 Current Issues

- Investigate both the conceptual and implementational issues of knowledge representation.
- Apply hybrid representation schemes for complex, realistic applications.

Semantic Network
MIS 680 – Spring 1992

1 What is Representation?

- Definition: A **representation** is a set of syntactic and semantic conventions that make it possible to describe things.
- Syntax: the symbols that may be used
- Semantics: how meaning is embedded in the symbols
- AI representations: semantic net, frame, production rules, logic, etc.

2 Important Properties of the Semantic Net?

- **Representation:** Nodes and links.
- **Inheritence:** enables description movement from classes to instances
- **Demons:** enables access to initiate actions or procedures
- **Defaults:** enables assumption in lieu of fact
- **Perspective:** anables purpose to guide access

3 What ISA is? (Brachman, 1983)

- **Generic/generic relations**
  - Subset/superset: "A nuclear submarine is a submarine." == "for every entity x if x is a member of nuclear submarines, then x is a member of submarines."
  - Generation/specialization (predicate): "A barking dog is a good watch dog" == "for every entity x if barking-dog(x) is TRUE then good-watch-dog(x) is TRUE." (predicate: function that maps arguments into TRUE or FALSE values.)
  - AKO: it implies "kind" status for the nodes it connects, whereas generalization relates arbitrary predicates.
  - Conceptual containment: "a triangle is a polygon" == "to be a triangle is to be a polygon with three sides"
  - Role value restriction: "the trunk of an elephant is a cylinder 1.3 meters long" == "the trunk property (slot) of an elephant is a cylinder (type) 1.3 meters long"
- **Generic/individual relations**
  - Set membership: "Clyde is a camel." == "Clyde is a member of the set of camels."
  - Predication: "Camel(Clyde) is TRUE."
  - Conceptual containment: "the king of France is a king" == "the king of France is a conceptual object called king"
  - Abstraction: "the eagle is an endangered species" (a generic type is abstraced into an individual)

4 What ISA isn't?

ISA is not "inheritence of properties".

Frame and Script
MIS 680 – Spring 1992

1 What is Representation?

- Definition: A **representation** is a set of syntactic and semantic conventions that make it possible to describe things.
- Syntax: the symbols that may be used.
- Semantics: how meaning is embedded in the symbols.

2 Frame

- **Definition:** A frame is a collection of semantic network nodes and links that together describe a stereotypical object, act, or event. It was proposed by Minsky in 1975.
- **Structure:** It consists of the heading and the **slots**, properties associated with the frame.
- **Expectation-driven processing:** The organization of the knowledge in frame facilitates expectation-driven processing, looking for things that are expected based on the context one thinks one is in.
- Frame-based reasoning has been used in a lot of domains.

3 Script

- **Definition:** Scripts – frame-like structures specifically designed for representing sequences of events. It was developed by Schank in 1977.
- **Structure:** It indicates sequence of actions in the slots. It organizes the knowledge representation in a way that directs attention and facilitates recall and inference.
- Both frames and scripts have been used in story-understanding programs.

Production Systems
MIS 680 – Spring 1992

1 What is Production System?

- Definition: In the human-modeling world, condition-action rules generally are called **productions** and rule-based systems are called **production systems**.
- A production system consists of:
  - A set of production rules, each consisting of a left hand side (a patter) that determines the applicability of the rule, and a right hand side that describes the action to be performed if the rule is applied.
  - One or more databases that contain whatever information is appropriate for the particular task.
  - A control strategy that specifies the order in which the rules will be compared to the database and a way of resolving the conflicts that arise when several rules match at once.
- Example: OPS5, a language for creating production systems. It consists of formalism for creating production rules, for defining working memory elements (WME), and a conflict resolution strategy.

2 Creating a Production System

- Direct solicitation of knowledge.
- Protocol analysis as a tool for eliciting knowledge.

LECTURE: Logic
MIS 680 – Spring 1992

1 Overview

- Rationale: An attempt to formalize reasoning and meaning.
- Definition: Employ the notions of constant, variable, function, predicate, logical connective and quantifier in order to represent facts as logical formulas.
- Important concepts: resolution proof, logic-based reasoning.
- Sample systems: (STRIPS, SRI, 1972).
- Advantages: availability of inferencing rules for proof, precise semantics, and simplicity of notation.
- Disadvantages: lack of organization principles for facts, difficult to represent procedural knowledge.
- Terms in Logic:
  - Constants, Variables, and Functions.
  - Predicates: functions that return true or false.
  - Atomic formulas: predicates with arguments.
  - Literals: negated or non-negated atomic formulas.
  - Formulas (WFFs): 1) literals, 2) formulas with connectives, and 3) formulas enclosed by quantifiers.
  - Clause: A WFF consisting of disjunction of literals.

2 Syntax

- Connectives have the following properties: communtative, associative, and de Morgan's laws.
- Implications: $p => q = -pvq$.
- Quantifiers: universal and existential.
- Rules of inference: Modus Ponens, Modus Tolens, and Resolution.

3 Rules of Inference

1. Modus ponens: from P => Q and P infer Q.
2. Modus tolens: from P => Q and -Q infer -P.
3. Resolution: from P v Q and -Q v R infer P v R.

4 Resolution Proofs

1. Assume that the negation of the theorem is true.
2. Show that the axioms and the assumed negation of the theorem determine something to be true that cannot be true.
3. Conclude that the assumed negation of the theorem cannot be true since it leads to a contradiction.
4. Conclude that the theorem must be true.

5 Transforming Expressions to Clause Form

The input to resolution proofs must be in **clause form**, which is a disjunction of literals.

1. Eliminate implications.
2. Reduce the scope of negation.
3. Make each universal quantifier bind a unique variable.
4. Move all universal quantifiers to the left.
5. Eliminate existential quantifiers.
6. Drop all universal quantifiers.
7. Convert the formula into a conjunction of disjunts.

6 Characteristics of Logic Based Systems

- Proof is exponential.
- Traditional lgoic is monotonic.
- Theorem is suitable for some problems, but not all.