บทที่ 1
ปัญญาประดิษฐ์
(Artificial Intelligence:AI)

1.1 AI คืออะไร

AI คือ แนวทางในการทำให้คอมพิวเตอร์คิดและตัดสินใจได้เช่นเดียวกับ มนุษย์ โดยการศึกษาถึงวิธีการที่มนุษย์คิด,ตัดสินใจและแก้ปัญหาเองได้ แล้วจึงนำแนวความคิดดังกล่าวมากำหนดเป็นขั้นตอนให้เครื่องคอมพิวเตอร์ทำงาน และนำไปออกแบบโปรแกรมที่สามารถแก้ปัญหาได้ โดยใช้ขั้นตอนที่คล้ายกับมนุษย์ใช้นั่นเอง ดังนั้น AI จะเป็นวิธีการที่ง่าย และมีรูปแบบที่จะช่วยให้เราสามรถออกแบบโปรแกรมการตัดสินใจที่ยุ่งยากได้

1.2 ความเป็นมาของ AI

สติปัญญา จัดว่าเป็นสิ่งที่สร้างพื้นฐานให้กับเทคโนโลยีทั้งหมดของมนุษย์รวมทั้งความเจริญรุ่งเรืองของมนุษย์ด้วย

การคิดอย่างมีสติปัญญาเริ่มขึ้นในสมัยของ Aristotle โดยเขาได้แบ่งแยกความแตกต่างระหว่างมวลสาร (matter) กับรูปแบบ (form) เช่น รูปปั้นมนุษย์เกิดจากสาร (material) คือบรอนซ์ ส่วนรูปแบบ (form) คือรูปมนุษย์และการเปลี่ยนแปลงเกิดขึ้นกับสารเมื่อมีการนำสารไปปั้นรูปแบบใหม่ขึ้น ซึ่งต่อมาก็เป็นพื้นฐานในเรื่องของการคำนวณเชิงสัญลักษณ์ (symbolic computing) และ data abstraction ต่อมา Aristotle ได้พัฒนาแนวความคิดเรื่อง ตรรก (logic) ซึ่งเขาถือเสมือนว่าเป็น เครื่องมือ (tool) เพราะเขารู้ว่าการศึกษาเกี่ยวกับเรื่องความคิดจะเป็นพื้นฐานเกี่ยวกับความรู้ทั้งหมดโดยการศึกษาเรื่อง logic จะเป็นลักษณะของ proposition นั่นคือความเป็นจริงและเท็จ เช่น "มนุษย์ทุกคนเป็นอมตะ" และ"วิชัยเป็นมนุษย์" ดังนั้นจะสรุปได้ว่า "วิชัยเป็นอมตะ"

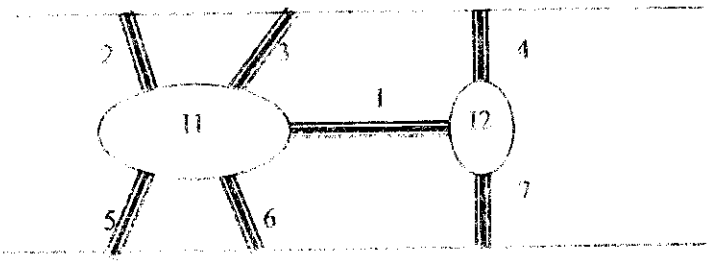
ต่อมาในคริสตวรรษที่ 17 Leibniz ได้เสนอระบบเกี่ยวกับ formal logic และได้สร้างเครื่องคำนวณขึ้น

CT 488 1

เครื่องคำนวณขึ้น

CT 488 บทที่ 1

ในคริสตวรรษที่ 18 Euler ได้เสนอรูปแบบของ การติดต่อกัน (connectedness) ซึ่งแสดงให้เห็นถึงโครงสร้างของความสัมพันธ์ ตัวอย่างคือปัญหา bridges of konigsburg



ซึ่งมีอยู่ว่า เมือง Konigsburg มีแม่น้ำไหลผ่านเมือง โดยแบ่งออกเป็นสองฝั่งคือ ฝั่ง B1 และฝั่ง B2 ที่กลางแม่น้ำมีเกาะ 2 เกาะ ชื่อเกาะ I1 และเกาะI2 โดยมีสะพาน 7 สะพานที่ทำการเชื่อมเกาะ I1 ,เกาะ I2 ฝั่ง B1 และฝั่ง B2 เข้าด้วยกัน (ดังรูป) ปัญหามีอยู่ว่า เราจะสามารถเดินทางไปรอบเมืองโดยข้ามสะพานทั้ง 7 ให้ครบและข้ามได้เพียงครั้งเดียวได้หรือไม่ และต่อมา Euler ได้เสนอแนวคิดในการแก้ปัญหานี้โดยการใช้กราฟ และมีการตั้งทฤษฎีกราฟขึ้น ซึ่งเป็นเครื่องมือด้านความคิดที่สำคัญในเรื่อง AI ในการแสดงรูปแบบการแก้ปัญหากราฟของ Euler เป็นแนวทางของการแก้ปัญหาอย่างเป็นระบบ โดยการค้นหาจากสถานการณ์ (state) ต่างๆ ของปัญหา(ทฤษฎีกราฟของ Euler นี้ต่อมาได้เป็นแนวความคิดเรื่อง state space search)

ต่อมา Charles Babbage ก็ได้นำแนวคิดอย่างมีระบบมาสร้างเป็นเครื่องคำนวณที่ชื่อว่า "Different engine" กับ "analytical engine" ซึ่งกลายมาเป็นแบบของการสร้างเครื่องคอมพิวเตอร์ในปัจจุบัน

ในคริสตวรรษที่ 19 George Boole ได้เสนอผลงานที่มีชื่อคือ "mathematical formalization of the laws of logic" ซึ่งมีผลอย่างมากต่อวิชาคอมพิวเตอร์ในปัจจุบัน และยังถือว่าเป็นรากฐานของ AI ด้วย

Gottlob Frege กับผลงานชื่อ Foundations of arithmetic ได้เสนอ mathematical specification language ที่ใช้สำหรับอธิบายการประมวลผลทางคณิตศาสตร์ในรูปแบบที่ง่ายและถูกต้อง ซึ่งผลงานของ Frege ต่อมาถูกเรียกเป็น first-order predicate calculus (เป็นเรื่องที่เกี่ยวกับ predicate symbolstheory of functions และ quantitative variables) ซึ่งเป็นภาษาซึ่งอธิบาย ทั้งคณิตศาสตร์ และความมีเหตุมีผลด้วย ทั้งยังเป็นพื้นฐานในการพัฒนาทฤษฎีทางด้าน AI ด้วย เพราะ first-order predicate calculuS เป็นเครื่องมือในการสร้างทฤษฎีเหตุผล คือ ภาษาสำหรับ

แสดงนิพจน์ (expression), ทฤษฎีสำหรับสมมติฐานเกี่ยวกับความหมายของ expression และการวินิจฉัย

ผลงานของ Russel และ Whitehead ชื่อว่า "principia mathematica" ได้เป็นรากฐานสำคัญของ AI เช่นกัน คือการใช้คณิตศาสตร์เป็น formal system ในรูปของ axiom โดยถือว่า axiom คือ string ที่สามารถ พิสูจน์ความเป็นจริงได้

Tarski เป็นนักคณิตศาสตร์อีกผู้หนึ่งซึ่งมีผลงานซึ่งจัดว่าเป็นรากฐานของ AI นั่นคือ "theory of reference" ซึ่งเป็นการอ้างถึง Well-Formed Formular (Wff)

ซึ่งต่อมาผลงานต่างๆ ที่กล่าวมาแล้วนั้น จะเป็นพื้นฐานสำคัญในงานด้าน AI แต่ผลงานด้าน AI ก็ยังไม่ปรากฏเด่นชัด จนกระทั่งมีการใช้ digital computer ซึ่งเป็นเครื่องคอมพิวเตอร์ที่มีความจำสูงและประมวลผลโปรแกรมที่ซับซ้อนได้ จึงมีความเป็นไปได้แล้วที่จะสร้างระบบการให้เหตุผลอย่างมีรูปแบบ และระบบการตรวจสอบให้กับเครื่องคอมพิวเตอร์

digital computer ไม่ได้เป็นเครื่องสำหรับทดสอบทฤษฎีของสติปัญญาเท่านั้น โครงสร้างของเครื่องยังเหมาะแก่การทำงานด้าน AI ด้วย เพราะ **สติปัญญา ก็คือรูปแบบของการประมวลผลสารสนเทศ** (intelligence is a form of information language,formal logic) โดยโปรแกรม AI จะแสดงความรู้ในรูปของ formal language,formal logic ในขณะที่การวิเคราะห์ปัญหาจะแสดงไว้ในรูปทฤษฎีกราฟ

ผลงานสำคัญอีกชิ้นหนึ่งทางด้าน AI คือผลงานของนักคณิตศาสตร์ชื่อ Alan Turing ซึ่งเรียกว่า "computing machinery and intelligence" นั่นคือ การค้นคว้าว่าสามารถสร้างเครื่องจักรที่สามารถคิดได้หรือไม่ โดยมีการทำการทดสองที่ชื่อว่า "turning test" คือ เอาคนและเครื่องจักรมาทดสอบในการตอบคำถาม เพื่อให้ผู้ถามคำถาม (ซึ่งถูกแยกไปอยู่ต่างหาก) บอกว่าคำตอบที่ได้รับมานั้นอันไหนมาจากคน หรือมาจากเครื่อง ผลของการทดสอบนี้ทำให้เกิดแนวการพัฒนาด้าน AI เช่น การสร้างระบบผู้เชี่ยวชาญ (expert system) คือ สร้างโปรแกรมที่เชี่ยวชาญในงานด้านหนึ่งด้านใดเพื่อแก้ปัญหาด้านนั้นโดยเฉพาะ

1.3 งานประยุกต์ด้าน AI

จุดประสงค์หลักของงานด้าน AI คือ

1. การแสดงความรู้ (knowledge representation)
2. การลำดับความรู้ (search)

จุดประสงค์แรก คือ การรวบรวมความรู้ทั้งหมดเพื่อสร้างเป็นสติปัญญา ในลักษณะของการใช้ภาษารูปแบบ (formal language) เพื่อเหมาะกับการประมวลผลด้วยคอมพิวเตอร์ ซึ่งในทาง AI จะใช้ predicate calculus เป็น formal language ดังกล่าว

จุดประสงค์ที่สอง คือการสืบค้น (search) คือ เทคนิคในการแก้ปัญหา โดยเป็นวิธีการอย่างมีระบบ (รูปแบบที่ดี) โดยการสำรวจขั้นตอนของปัญหา (problem state) เพื่อหาคำตอบที่เหมาะสมที่สุด เช่น การเล่นหมากรุก ต้องมีการพิจารณาว่าจะเดินไปทางใดได้บ้างและเดินทางใดให้ผลดีที่สุด

AI ยังแบ่งย่อยออกได้หลายๆ แขนง เช่นเดียวกับศาสตร์อื่นๆ โดยที่แต่ละแขนงจะมีแนวทางอย่างเดียวกันคือ การแก้ปัญหา ถึงแม้จะมีการประยุกต์ใช้ต่างกันก็ตามแขนงย่อยของงาน AI ได้แก่

1. การเล่นเกมส์ (game playing)

ในการวิจัยระยะแรกด้วยด้าน AI ในเรื่อง state space search นั้นจะเน้นไปในการเล่นเกมส์ที่ใช้กระดาน เช่น หมากรุก, puzzle ทั้งนี้ก็เพราะเกมส์เหล่านี้มีกฎการเล่นที่แน่นอนรัดกุม ทำให้การค้นหาคำตอบทำได้ง่าย การเล่นเกมส์ จะทำให้เกิด (large search space) เทคนิคที่จะช่วยในการกำหนด problem space ที่เรียกว่า Heuristics search ซึ่งกลายมาเป็นงานวิจัยด้าน AI ที่สำคัญด้านหนึ่ง

2. การให้เหตุผลและการพิสูจน์ทฤษฎีอย่างอัตโนมัติ (automated resoning and theorem proving)

เราสามารถกล่าวได้ว่า การพิสูจน์ทฤษฎีอย่างอัตโนมัติ (automatic theorem proving) จัดว่าเป็นงานแขนงเก่าแก่ที่สุดของ AI คือย้อนไปตั้งแต่ผลงานของ Simon & Mewell ซึ่งเกี่ยวกับ logic และย้อนไปสู่ Ruseell กับ Whitehead ซึ่งพยายามใช้คณิตศาสตร์ในการวิจัยโดยอ้างถึงทฤษฎี

ผลที่ตามมาก็คือ การใช้ predicate calculus กับ formal language ในการแสดงความรู้และการใช้ PROLOG เป็น logic programming language ในการประยุกต์ด้านนี้

3. ระบบผู้เชี่ยวชาญ (Expert Eystems)

งานแขนงสำคัญด้าน AI อีกด้านคือ การสร้างระบบผู้เชี่ยวชาญ นั่นคือการกำหนดขอบเขตความรู้เฉพาะอย่างให้กับระบบความรู้ โดยมีผู้เชี่ยวชาญด้าน AI ที่เรียกว่า knowledge engineer ทำหน้าที่ออกแบบระบบผู้เชี่ยวชาญ และเป็นผู้กำหนดความรู้ให้กับโปรแกรม ซึ่งจะใช้ในระบบผู้เชี่ยวชาญด้วย

4. การเข้าใจภาษาธรรมชาติและการสร้างรูปแบบความหมาย (natural language understanding and semantic modeling)

งานวิจัยด้าน AI ที่น่าสนใจอีกแขนงหนึ่งคือ การสร้างโปรแกรมที่สามารถเข้าใจภาษามนุษย์ และสามารถจะใช้ภาษามนุษย์ได้อย่างมีประสิทธิภาพด้วย ซึ่งงานด้านนี้ได้ผลงานมาบ้างแล้ว แต่ยังไม่เป็นผลสำเร็จขั้นสูงสุดตามจุดประสงค์ที่ตั้งไว้ เพราะการเข้าใจภาษามนุษย์นี้ไม่ใช่เพียงแค่การ parse ประโยคออกเป็นส่วนๆ แล้วตรวจสอบคำแต่ละคำเท่านั้น แต่โปรแกรมจะต้องเข้าใจรูปแบบการใช้ภาษาลักษณะต่างๆ แสลงของภาษาและแก้ไขความคลุมเครือของภาษาได้ด้วย

5. การสร้างรูปแบบการทำงานของมนุษย์ (modeling human performance)

ถึงแม้งาน AI จะพยายามเลียนแบบสติปัญญาของมนุษย์แล้วก็ตาม แต่ไม่มีงาน AI ที่จะเลียนแบบความคิดของมนุษย์ได้ ดังนั้นงาน AI แขนงนี้จึงพยายามเน้นเรื่องการศึกษาถึงแนวความคิดของมนุษย์ซึ่งเทคนิคนี้จะช่วยเพิ่มศัพท์เกี่ยวกับสติปัญญาของมนุษย์ได้ ทั้งยังทำให้เครื่องคอมพิวเตอร์สามารถสร้างความคิดที่ดีขึ้นมาได้ด้วย

6. การวางแผนและหุ่นยนต์ (planning and robotics)

การวางแผน (planning) เป็นแนวคิดในการออกแบบหุ่นยนต์ ให้สามารถ ปฏิบัติงานได้อย่างมีประสิทธิภาพ planning เป็นปัญหาที่ยุ่งยาก เพราะอาจเกิดการกำหนดแผนงานได้หลายๆ อย่าง ซึ่งโดยปกติ หุ่นยนต์จะถูก plan ให้ทำงานได้เฉพาะอย่าง เช่น เดินหน้า, ถอยหลัง, เลี้ยวซ้าย,

เลี้ยวขวา แต่การให้หุ่นยนต์เดินในห้องที่มีสิ่งของวางตั้งอยู่จากฟากหนึ่งไปยังอีกฟากหนึ่งนั้น จะต้องมีการกำหนดอย่างรัดกุมวิธีการอย่างหนึ่งที่มนุษย์ใช้ในการวางแผน คือ การแตกปัญหาย่อยตามลำดับขั้น (heirarchical problem decomposition) ซึ่งถ้าต้องการให้โปรแกรมทำได้อย่างมนุษย์จะเป็นเรื่องที่ยุ่งยากมาก คือต้องมีการแบ่งเป็นปัญหาย่อย, มีการพิจารณาแผนย่อย เพื่อให้โปรแกรมสามารถบรรลุถึงวัตถุประสงค์ได้

หุ่นยนต์ (robot) จึงต้องสามารถที่จะวางแผนการทำงานได้ ปรับแผนการทำงานให้เหมาะกับสภาพแวดล้อมได้ รวมทั้งสามารถแก้ไขข้อผิดพลาดในแผนที่วางได้ด้วย

7. ภาษาและสภาพแวดล้อมสำหรับ AI (languages and environments for AI)

ผลสำคัญที่ตามมาจากการพัฒนาด้าน AI คือ ความก้าวหน้าด้านภาษาโปรแกรมและการพัฒนาซอฟต์แวร์ นั่นคือจากผลของการหาคำตอบจาก space state ขนาดใหญ่, ขนาดของ AI โปรแกรม ความยากในการคาดการณ์เกี่ยวกับ behavior ของโปรแกรม ทำให้โปรแกรมเมอร์ด้าน AI ต้องมีการพัฒนาเทคนิคการโปรแกรมที่มีประสิทธิภาพขึ้น

การเขียนโปรแกรม AI จึงต้องมีเทคนิคเกี่ยวกับโครงสร้างความรู้เข้ามาเกี่ยวข้องอย่างเช่น object-oriented programming และ expert system framework เช่น LISP,PROLOG ซึ่งเน้นเรื่องการพัฒนาโปรแกรมแบบ module เพื่อควบคุมเรื่องขนาดและความยุ่งยากของโปรแกรม ผลที่ตามมาก็คือการหาเทคนิคใหม่ๆ ด้วย software engineering

8. การเรียนรู้ของเครื่องคอมพิวเตอร์ (machine learning)

การเรียนจัดว่าเป็นปัญหายุ่งยากทางด้านโปรแกรม AI ถึงที่ประสบผลสำเร็จใน การแก้ปัญหาการเรียนรู้เป็นสิ่งสำคัญในเรื่องสติปัญญา expert system อาจจะแก้ปัญหาที่ยุ่งยากได้ดีกว่ามนุษย์ แต่ถ้าถามปัญหานั้นซ้ำอีก expert system ก็ต้องใช้ความพยายามเพื่อหา คำตอบ นั่นคือมันไม่สามารถจำได้ว่า คำตอบคืออะไรในขณะที่มนุษย์สามารถจำได้ และนี่คือสิ่งที่ไม่ใช่สติปัญญา

การเรียนรู้เป็นงานวิจัยที่ยุ่งยาก แต่ก็พบว่าไม่ใช่สิ่งที่เป็นไปไม่ได้

AI เสมือนการนำเสนอและการค้นหา (AI as representation & search)

จากผลของ turing test ทำให้ Simon และ Newell สรุปได้ว่า ความมีสติปัญญาไม่ว่าจะเป็นมนุษย์หรือเครื่องจักรเกิดจากการใช้สิ่งต่อไปนี้

1. symbol pattern ในการแสดงขอบเขตที่สำคัญของปัญหา
2. operation ที่ใช้กับ symbol pattern เพื่อหาวิธีแก้ปัญหาที่เป็นไปได้
3. search เพื่อหาวิธีแก้ปัญหาที่เหมาะสมที่สุด

ซึ่งทำให้เกิดเป็นรากฐานที่เรียกว่า physical sysbol system hypothesis ซึ่งเป็นสิ่งที่ทำให้เราพยายามสร้างเครื่องจักรที่มีสติปัญญาขึ้น (รวมทั้งงานวิจัยด้าน AI ด้วย)

physical symbol system hypothesis แยกความแตกต่างระหว่างรูปแบบ (pattern) ซึ่งสร้างขึ้นมาจากสัญลักษณ์ (symbol) กับตัวกลาง (medium) ที่ใช้อธิบายรูปแบบ (pattern) เราพบว่าถ้าสติปัญญาสร้างขึ้นมาจากโครงสร้างของ symbol แต่เพียง อย่างเดียว medium ใดๆ ที่สามารถจัดเก็บรูปแบบ (implement pattern) นั้นได้ก็จะถือว่าเป็นเครื่องที่มีสติปัญญา เราพบว่า digital computer ที่ถูกโปรแกรมอย่างเหมาะสมมีลักษณะเป็นเช่นว่านี้ คือสามารถจัดการจัดเก็บสัญลักษณ์ (implement symbolic pattern) ได้ ดังนั้นนั่นคือ digital computer เป็นเครื่องที่มีสติปัญญา (intelligence machin) ด้วย

physical symbol system hypothesis สามารถใช้กำหนดโครงสร้างสัญลักษณ์และรูปแบบ (symbol,structure) และการปฏิบัติการจำเป็นที่ใช้ในการแก้ปัญหาและยังสามารถพัฒนา search ที่ถูกต้องและมีประสิทธิภาพได้ ซึ่งไปสัมพันธ์กับ knowledge representation และ search ซึ่งเป็นหัวใจสำคัญของงานวิจัยใหม่ๆ ด้าน AI

มีนักวิชาการบางคนคัดค้านเรื่อง physical symbol hypothesis โดยอ้างว่าสติปัญญานั้นเป็นผลทางชีวภาพ และเกิดขึ้นเองตามธรรมชาติ คือไม่สามารถจำลองขึ้นมาได้ใน รูปแบบของสัญลักษณ์ได้ ผลการคัดค้านนี้ก่อให้เกิดแนวทางการวิจัยด้าน AI ทั้งทางทฤษฎี และทางปฏิบัติโดยเฉพาะในเรื่อง expert systems,planing และภาษาธรรมชาติ

การแสดงความรู้ (knowledge representation)

การทำงานของรูปแบบการแสดง (representation scheme) ของปัญหาใดๆ คือ การกำหนดขอบเขตที่สำคัญของปัญหา และการสร้างข้อมูลที่จะนำมาใช้ ในการกำหนดขั้นตอนเพื่อแก้

CT 488

ปัญหานั้น ดังนั้นในภาษาการแสดง (representation language) จะต้องอนุญาตให้โปรแกรมเมอร์แสดงความรู้เพื่อการนำไปใช้แก้ปัญหาได้และที่สำคัญคือโปรแกรมที่สร้างขึ้นจะต้องมีประสิทธิภาพด้านการคำนวณด้วย

ในการพิจารณาถึงภาษาแสดงความรู้ (knowledge representation) จะเน้น ในเรื่องการแสดงออก (expressions) และประสิทธิภาพ (efficiency) ภาษาที่มีความสามารถในการแสดงออก (expressiveness) สูงอาจจะทำให้เกิดปัญหาในเรื่อง efficiency ในการนำไปใช้แก้ปัญหาบางอย่าง ดังนั้นบางครั้งเราต้องยอมเสียเรื่อง expressiveness เพื่อให้เกิด efficience ของภาษา แต่จะต้องไม่เป็นการจำกัดขอบเขตของการแสดงความสามารถในการแสดงถึงความรู้ในการแก้ปัญหา

ภาษาแสดงความรู้ (knowlegde representation language) เป็นเครื่องมือซึ่งช่วยมนุษย์ในการแก้ปัญหา ดังนั้นการแสดงออก (representation) จะต้องให้ความเป็นธรรมชาติในการแสดงความรู้ในการแก้ปัญหา และต้องให้ความรู้นั้นแก่คอมพิวเตอร์ได้ รวมทั้งต้องช่วยให้โปรแกรมเมอร์จัดรูปแบบความรู้ได้

ตัวอย่างของการแสดงความรู้ในเครื่องคอมพิวเตอร์ ได้แก่ การเก็บค่าทศนิยมโดยปกติ ค่าทศนิยมอย่างเช่นค่า ¶ เป็นค่าไม่รู้จบ คือ 3.141592... ซึ่งเป็นค่าที่ไม่สามารถเก็บได้ในอุปกรณ์ที่มีความจำกัด ดังนั้นเพื่อให้คอมพิวเตอร์เก็บค่าของ ¶ ได้จึงมีการแบ่งค่านี้ออกเป็น 2 ส่วน คือส่วนของ mantissa และ exponent โดยการแสดงค่า ¶ ในคอมพิวเตอร์จะเป็น

31416	1
mantissa	exponent

การแสดงค่าดังกล่าวนี้ ทำให้ expressiveness หมดไปคือ ค่าที่แท้จริงจะถูกประมาณเป็น 3.1416 ทั้งนี้ก็เพื่อสนับสนุนการเป็น efficiency คือทำให้สามารถจัดเก็บในหน่วยความจำได้นั่นเอง

รูปแบบการแสดง (representational scheme) ที่ดีควรมีลักษณะดังนี้

1. เพียงพอที่จะแสดงข้อมูลที่สำคัญได้ทั้งหมด
2. สนับสนุนการทำงานอย่างมีประสิทธิภาพของรหัสที่ใช้แสดงความรู้
3. ให้รูปแบบที่เป็นธรรมชาติสำหรับแสดงความรู้ที่ต้อง การ

Mcdermont และคนอื่นๆ ได้กล่าวไว้ว่า สิ่งสำคัญในการเขียนโปรแกรมแสดงความรู้ที่ให้ผลดีนั้นขึ้นอยู่กับการเลือกใช้ representational tool ที่เหมาะสม ภาษาโปรแกรมเช่น BASIC และ FORTRAN จะเป็นภาษาโปรแกรมที่ไม่เหมาะกับการสร้างระบบผู้เชี่ยวชาญ เพราะระดับของภาษา

ค่อนข้างต่ำในการให้ความสามารถในการแสดงความรู้ และไม่ให้รูปแบบ modularity ซึ่งจำเป็นสำหรับโปรแกรมฐานความรู้

AI เกี่ยวกับการแก้ปัญหาเชิงคุณภาพมากกว่าการแก้ปัญหาเชิงปริมาณ นั่นคือ เป็นเรื่องๆของการใช้เหตุผลมากกว่าการคำนวณ และเกี่ยวข้องกับการจัดรูปแบบของความรู้มากกว่าการกำหนด algorithm ที่ดี และนี่คือสาเหตุที่ทำให้ AI เป็น representation language ต้องมี คุณสมบัติ

A. จัดความรู้เชิงคุณภาพได้

B. สามารถสร้างความรู้ใหม่ได้จากความรู้และเหตุผลที่มีอยู่ได้

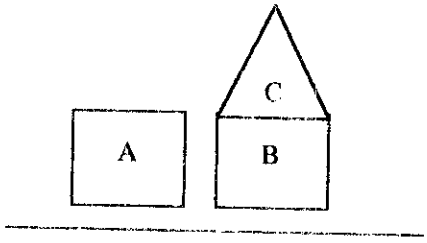
C. ให้การแสดงออกสำหรับกฎเกณฑ์ทั่วไปและสถานการณ์เฉพาะอย่างได้

D. แสดงความหมายที่ซับซ้อนได้

E. สามารถให้เหตุผลที่เป็น meta-level reasoning ได้

A. การจัดการความรู้เชิงคุณภาพ

การเริ่มโปรแกรมด้วย AI ต้องการสิ่งที่เกี่ยวข้องกับการรับรู้และการให้เหตุผลในเชิงคุณภาพสำหรับปัญหาที่เกิดขึ้น ตัวอย่างเช่น การจัดกล่องบนโต๊ะ (ดังรูป)



วิธีหนึ่งที่จะแสดงการจัดกล่องได้คือ การใช้ catesian coordinate system โดยใช้แกน X และ Y กำหนดจุดยอดของกล่องแต่ละกล่อง แต่ปรากฏว่าวิธีนี้จะไม่สามารถแสดงได้ว่า กล่องแต่ละใบจัดวางไว้อย่างไร (บอกแต่ตำแหน่งที่อยู่) คือบอกไม่ได้ว่า กล่องใบไหนอยู่บนกล่องใบไหน ดังนั้นวิธีการแสดงการจัดกล่อง อาจทำได้โดยการใช้ predicate calculus กำหนดกฎเกณฑ์ในการจัดกล่อง ซึ่งจะได้ผลดังนี้

CLEAR(C)

CLEAR(A)

ONTABLE(A)

ONTABLE(B)

ON(C,B)

CUBE(B)

CUBE(A)

PYRAMID(C)

รูปแบบที่ใช้ แบ่งออกเป็น 2 ส่วนคือ

1. ส่วนของ predicate (คือ คำที่แสดงไว้ในรูปแบบ เช่น CLEAR,ON เป็นต้น) แสดงถึงคุณสมบัติหรือความสัมพันธ์ที่เกิดขึ้นของสิ่งที่เราสนใจ

2. argument คือ สัญลักษณ์ที่แสดงถึงสิ่งที่อยู่ในขอบเขตที่กำลังกล่าวถึง

ส่วนประโยคที่เป็นเหตุผลทั้งหมดที่แสดงไว้ จะอธิบายถึงคุณสมบัติที่สำคัญ หรือความสัมพันธ์ที่เกิดขึ้นระหว่างสิ่งที่เราสนใจ จะเห็นได้ว่า predicate calculus สามารถทำให้โปรแกรมเมอร์ด้าน AI แสดงเหตุผลเชิงคุณภาพได้ด้วยภาษาที่ถูกกำหนดไว้เป็นอย่างดีแล้ว (well-defined language) นอกจากนี้ยังสามารถนำไปเป็นพื้นฐานสำหรับ formal method อื่นๆ ในการแสดงความรู้

B. การสร้างความรู้ใหม่จากเหตุผลที่มีอยู่

ความสามารถในการอนุมานความรู้ใหม่จากความรู้ที่มีอยู่ จัดเป็นสิ่งสำคัญของสติปัญญา จากตัวอย่างเรื่องการจัดกล่อง เราสามารถกำหนดการทดสอบเพื่อพิจารณาว่ากล่องนั้นวางอยู่โดยไม่มีกล่องใบอื่นมาทับ (คือ CLEAR นั่นเอง) ได้ซึ่งเป็นสิ่งสำคัญในการสั่งให้แขนกลหยิบกล่องนั้นขึ้นมา หรือวางกล่องอื่นซ้อนลงไป โดยเราไม่จำเป็นต้องกำหนดกฎขึ้นมาใหม่ คือเราสามารถสร้างกฎที่จะอธิบายเหตุการณ์นี้ได้จากกฎที่มีอยู่แล้วโดยใช้ predicate calculus เราจะได้ว่า

$$\forall X \neg \exists Y \cdot ON(Y,X) \Rightarrow CLEAR(X)$$

ซึ่งอ่านได้ว่า "สำหรับทุก X,X จะเป็น CLEAR ถ้าไม่มี Y ใดๆ เลย ที่ซึ่งอยู่บน X"

C. การแสดงออกสำหรับกฎเกณฑ์ทั่วไปและสถานการณ์เฉพาะอย่าง

นอกจากจะแสดงให้เห็นถึงการอนุมานความรู้ใหม่จากเหตุผลที่มีอยู่แล้วในตัวอย่างเรื่องการจัดกล่อง ตัวอย่างนี้ยังเสนอถึงการใช้ตัวแปรใน predicate calculus เนื่องจาก AI จะต้องมีรูปแบบเป็นแบบทั่วๆ ไปให้มากที่สุดเท่าที่จะเป็นไปได้ จึงต้องมีการใช้ตัวแปร ซึ่งการใช้ ตัวแปรด้าน AI จะแตกต่างจากการใช้ตัวแปรในการเขียนโปรแกรมทั่วๆ ไป คือตัวแปรด้าน AI จะมีการกำหนดแบบ dynamic มากกว่าตัวแปรในโปรแกรมทั่วไป เพื่อเน้นถึงการให้เหตุผล

D. การแสดงความหมายที่ซับซ้อน

ขอบเขตปัญหาด้าน AI ส่วนใหญ่จะเกี่ยวกับความต้องการความรู้ที่สัมพันธ์กันแบบโครงสร้างจำนวนมาก เช่น การอธิบายถึงรถยนต์ จะพูดถึงแต่เพียงส่วนประกอบแต่ละส่วนเท่านั้นไม่ได้ คือ เราจะต้องพูดถึงความสัมพันธ์ที่เกิดขึ้นระหว่างส่วนประกอบเหล่านั้นด้วย โครงสร้างของความสัมพันธ์จะมีความสำคัญต่อการกำหนดระดับของเหตุการณ์ เช่น การจัดแบ่งพืช ตามสกุล (genus) และ ชนิด (species) ในตัวอย่างเรื่องกล่อง การติดต่อกันระหว่าง predicate ที่กำหนดไว้ คือ รายละเอียดการจัดวางกล่อง

ความสัมพันธ์ด้านความหมายเป็นสิ่งสำคัญในการอธิบายถึงความสัมพันธ์ระหว่างเหตุการณ์ที่เกิดขึ้นในระยะเวลาหนึ่ง ดังนั้นจึงเป็นเรื่องที่สำคัญในการที่เราต้องเข้าใจ คำบรรยายนั้นได้ง่าย เช่น นิทานเด็ก หรือ การแสดงแผนการเคลื่อนไหวของ robot ในลักษณะของขั้นตอนของ atomic acton ซึ่งต้องทำงานตามลำดับ

ในการแสดงความหมายนี้อาจจะอยู่ในรูปแบบของสัญลักษณ์ระดับสูง เช่น การอธิบายถึงความรู้เกี่ยวกับ "BLUEBIRD" ว่าเป็น "นกสีน้ำเงินตัวเล็ก และนกเป็นสัตว์มีกระดูกสันหลัง มีขนที่บินได้" เราสามารถอธิบายความหมายของ BLUEBIRD ได้ด้วย logical predicate calculus ดังนี้

SIZE (BLUEBIRD,SMALL)

HASCOVERING (BIRD,FEATHERS)

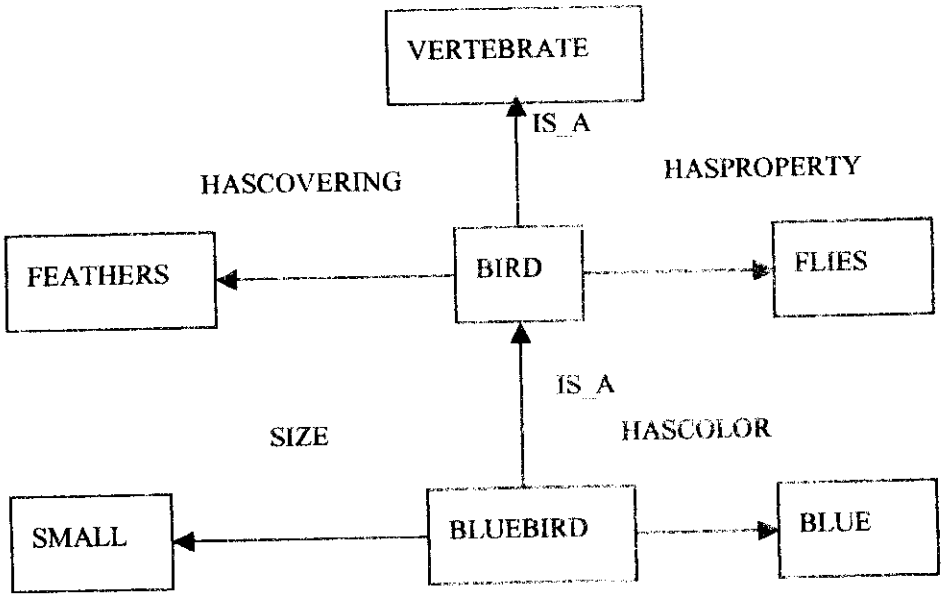
HASCOLOR(BLUEBIRD,BLUE)

HASPROPERTY(BIRD,FLIES)

IS_A(BLUEBIRD,BIRD)

IS_A(BIRD,BERTEBRATE)

ซึ่งคำอธิบายนี้ยังสามารถแสดงในรูปของ graph แทนการใช้ predicate ได้เช่นกัน คือ



เราเรียกรูปแบบนี้ว่า semantic network ซึ่งเป็นเทคนิคพื้นฐานในการแสดงการให้ความหมาย

การแสดงความหมายในรูปกราฟนี้ ทำให้การหาเหตุผลในสิ่งที่เราสนใจง่ายขึ้น คือโดยการไล่ไปตาม link ที่เชื่อมจุดในกราฟเข้าด้วยกัน ตัวอย่างเช่น เราไล่ตาม link 2 เส้นทาง เราจะได้ว่า BLUEBIRD เป็นสัตว์มีกระดูกสันหลัง ซึ่งการค้นนี้ จะง่ายกว่าการไปค้นในฐานข้อมูลของ predicate calculus ที่อธิบายถึงรูปแบบของ IS_A(X,Y)

นอกจากนี้ ความรู้ยังสามารถถูกจัดให้สะท้อนให้เห็นถึงรูปแบบที่ธรรมชาติของขอบเขตในสิ่งที่เรากำลังกล่าวถึง เช่น การ link บางอย่างใน semantic network เช่น link ของ ISA แสดงถึงสมาชิกของกลุ่ม และคุณสมบัติของกลุ่มที่สามารถถ่ายทอด (inherit) ไปยังสมาชิกทุกคนในกลุ่ม ซึ่งกลไกลการถ่ายทอดนี้ ถูกสร้างให้กับภาษา และทำให้ความรู้ถูกจัดเก็บไว้ในระดับของนามธรรม (abstraction) ขั้นสูง การถ่ายทอดนี้เป็นเครื่องมือสำคัญในการสร้างรูปแบบความสัมพันธ์อย่างมีโครงสร้าง และสนับสนุนในแง่ที่ว่าสมาชิกของกลุ่ม (class) เดียวกันจะมีคุณสมบัติอย่างเดียวกัน รูปแบบที่กล่าวมาแล้วนี้จะเสนอถึง efficiency และ expressiveness แบบธรรมชาติ และยังมีทฤษฎีกราฟที่ดีและเหมาะแก่การนำมาใช้ให้เหตุผล จึงมีการใช้ semantic network แทนการใช้ predicate calculus ด้วย

E. การให้เหตุผลที่เป็น meta-level reasoning

ในระบบความเฉลียวฉลาดนั้น การรู้เพียงอย่างเดียวนั้นไม่พอ คือ ระบบควรจะรู้ว่ามีความรู้อะไรบ้าง นั่นคือไม่ควรรู้วิธีแก้ปัญหาแต่เพียงอย่างเดียว แต่ควรจะอธิบายได้ว่าแก้ปัญหานั้นได้อย่างไร และทำไมจึงตัดสินใจแก้ปัญหาด้วยวิธีนี้ ทั้งยังควรสามารถอธิบายความรู้ได้โดยทั่วไปและโดยเฉพาะเจาะจงได้ สามารถรู้ขอบเขตความรู้ของตัวได้ และสามารถเรียนรู้จากการติดต่อกับโลกภายนอกได้

"การรู้เกี่ยวกับความรู้ของตน" เป็นภาษาระดับสูงที่เรียกว่า meta knowledge ซึ่งถือว่าเป็นระบบสติปัญญาที่แท้จริง

การค้นหาวิธีการแก้ปัญหา (problem solving as search)

แนวทางที่สองของ Simon และ Newell ในเรื่อง physical symbol system hypothesis คือ การแก้ปัญหาโดยการค้นหาวิธีแก้ปัญหาจากหลายๆ ทางเลือก ซึ่งก็คือวิธีที่มนุษย์เราใช้อยู่นั่นเอง ตัวอย่างเช่น การเล่นหมากรุก แล้วเลือกวิธีที่ดีที่สุดมาใช้ ซึ่งวิธีดังกล่าวนี้เป็นแนวทางของเทคนิคในการแก้ปัญหา ที่เรียกว่า state space search