

บทที่ 1

บทนำ

(Introduction)

ปัญญาประดิษฐ์ (Artificial intelligence) นิยมเรียกกันทั่วไปว่า AI จัดเป็นศาสตร์แขนงหนึ่งที่เน้นแนวทางการทำให้คอมพิวเตอร์มีการกระทำอันชาญฉลาด โดยหวังไว้ว่าคอมพิวเตอร์ที่ฉลาดขึ้นจะมีประโยชน์มากขึ้น สามารถตอบสนองในสิ่งที่เราต้องการได้ดีขึ้น และทำงานเป็นที่น่าสนใจมากขึ้น ปัญญาประดิษฐ์เป็นเรื่องที่กล่าวขานกันมานานแล้ว เพราะนับแต่ยุคแรกเริ่มที่มีการประดิษฐ์เครื่องคอมพิวเตอร์ขึ้นมาได้นั้น นักวิทยาศาสตร์พากันคิดฝันอยากให้อคอมพิวเตอร์มีความสามารถในการทำงานได้เช่นเดียวกับมนุษย์ และมีความคิดหรือมีสติปัญญาเฉลียวฉลาดแบบมนุษย์ เพื่อช่วยงานหรือทำงานแทนมนุษย์ได้ ครั้นถึงปัจจุบันเมื่อเทคโนโลยีก้าวหน้าไปมาก แนวคิดหลายอย่างได้พัฒนามาจนมีผลลัพธ์เกิดขึ้นจริง ทั้งในด้านฮาร์ดแวร์ และซอฟต์แวร์ เป็นเหตุให้ศาสตร์แขนงนี้ยังคงมีการศึกษาและก้าวหน้าต่อไปอย่างไม่หยุดยั้ง

1.1 นิยามของปัญญาประดิษฐ์

ปัญญาประดิษฐ์ หรือ AI เป็นสาขาทางวิทยาศาสตร์และวิศวกรรมศาสตร์ที่พยายามทำความเข้าใจกับเรื่องเกี่ยวกับความเฉลียวฉลาด เซอวนปัญญา และสร้างเครื่องจักรที่มีความฉลาดให้ได้จริง โดยเฉพาะอย่างยิ่งคือสร้างโปรแกรมคอมพิวเตอร์ที่ชาญฉลาด

คำว่า “ฉลาด” หรือ “Intelligence” เป็นคำที่คลุมเครือ เพราะบางครั้งบางกรณีการตัดสินใจสิ่งนั้น ๆ มีความฉลาดหรือไม่เป็นเรื่องยาก ดังนั้น AI จึงเป็นสาขาที่จัดว่าไม่มีนิยามที่ดี ความฉลาดนั้นมีหลายระดับ สำหรับปัญญาประดิษฐ์จัดว่าเป็นความสามารถในด้านการคำนวณเพื่อทำงานให้ประสบความสำเร็จตามเป้าหมายที่ตั้งไว้ ถ้าจะถามว่าคอมพิวเตอร์เครื่องนี้ฉลาดหรือไม่ เราคงไม่สามารถให้คำตอบสั้น ๆ เพียงว่าใช่ หรือไม่เช่นนั้น แต่ความฉลาดมีส่วนเกี่ยวข้องกับกลไกการทำงานของเครื่อง ผลงานวิจัยทาง AI ได้ค้นพบวิธีทำให้คอมพิวเตอร์ทำงานบางอย่างได้สำเร็จ แต่ไม่ใช่ทุกอย่าง แต่โปรแกรมคอมพิวเตอร์ดังกล่าวก็ทำงานได้น่าประทับใจและควรที่จะได้รับการพิจารณาว่าฉลาด

โดยทั่วไป AI มักจะมีความหมายว่าเป็นวิศวกรรมซอฟต์แวร์ขั้นสูง ใช้เทคนิคของซอฟต์แวร์ที่ซับซ้อนเพื่อแก้ปัญหายาก ๆ ที่ไม่สามารถใช้วิธีแก้ปัญหาอย่างง่ายได้ อีกความหมายหนึ่งคือเป็นวิธีการที่ไม่ใช้ตัวเลข (Nonnumeric) ในการแก้ปัญหา เนื่องจากคนไม่สามารถแก้ปัญหาตัวเลขได้ดี วิธีที่ไม่ใช้ตัวเลขนี้มักจะเป็นวิธีใช้ “สามัญสำนึก” (Common sense) ซึ่งอาจจะไม่ใช่สิ่งที่ดีที่สุดหรือหนทางที่ดีที่สุด ดังนั้นโปรแกรม AI ปกติแล้วจะไม่สมบูรณ์ และอาจจะผิดพลาดได้อีกด้วย

มีผู้ให้คำนิยามของ AI ไปได้เป็นจำนวนมาก ดังตัวอย่างตามตารางที่ 1.1 ดังนี้

ตารางที่ 1.1 คำนิยามต่าง ๆ ของ AI

ระบบที่คิดเหมือนมนุษย์ - ความพยายามแบบใหม่และน่าทึ่งที่จะทำให้คอมพิวเตอร์คิด และเป็นเครื่องจักรที่มีความคิดและจิตใจ (John Haugeland) - กิจกรรมที่เกี่ยวกับความคิด เช่นการตัดสินใจ การแก้ปัญหา การเรียนรู้ (Richard E. Bellman)	ระบบที่คิดอย่างมีเหตุผล - การศึกษาเรื่องราวทางจิตใจ โดยใช้แบบจำลองของคอมพิวเตอร์ (Eugene Charniak & Drew McDermott) - การศึกษางานคอมพิวเตอร์ที่สามารถรับรู้ (Perceive) ใช้เหตุผล (Reason) และแสดงการกระทำได้ (Act) (Patrick Winston)
ระบบที่กระทำเหมือนมนุษย์ - ศิลปะในการสร้างเครื่องเพื่อทำงานที่เมื่อคนเป็นผู้กระทำ คนต้องใช้ความฉลาด (Ray Kurzweil) - การศึกษาวิธีที่จะทำให้คอมพิวเตอร์ทำสิ่งที่ปกติแล้วคนทำได้ดีกว่า (Elaine Rich & Kevin Knight)	ระบบที่กระทำอย่างมีเหตุผล - ศึกษาการออกแบบ Intelligent agent (David Poole) - สาขาของวิทยาการคอมพิวเตอร์ที่เกี่ยวกับการทำให้เกิดพฤติกรรมอันชาญฉลาด (Nils J. Nilsson)

สรุปได้ว่า นิยามของ AI แบ่งเป็นหมวดหมู่ได้ 4 หมวดดังนี้

1.1.1 ระบบที่คิดเหมือนมนุษย์ (Systems that think like humans)

การจะทำให้คอมพิวเตอร์คิดได้เหมือนกับที่มนุษย์คิดนั้น เริ่มจากปัญหาว่ามนุษย์คิดได้อย่างไร เมื่อทราบการทำงานในจิตใจของมนุษย์ (Human minds) แล้วจึงตั้งทฤษฎีที่แน่ชัดเกี่ยวกับจิตใจขึ้น จากนั้นนำทฤษฎีนี้มาเขียนโปรแกรมคอมพิวเตอร์ ดูว่าโปรแกรมทำงาน

สอดคล้องกับพฤติกรรมของมนุษย์หรือไม่ ถ้าใช่ แสดงว่ากลไกบางอย่างของโปรแกรมมีการทำงานเช่นเดียวกับกลไกของมนุษย์

มีศาสตร์สาขาหนึ่งเรียกว่า Cognitive science ได้นำแบบจำลองคอมพิวเตอร์จากศาสตร์ทาง AI และเทคนิคการทดลองจากจิตวิทยามารวมกัน เช่น ศึกษาการเรียนรู้ของเซลล์สมองในสามมิติ ศึกษาการถ่ายเทประจุไฟฟ้า และวิเคราะห์การเปลี่ยนแปลงทางเคมีไฟฟ้าในร่างกายระหว่างการคิด แล้วนำมาสร้างทฤษฎีการทำงานของจิตใจมนุษย์ มีความแม่นยำและทดสอบได้ AI และ Cognitive science ช่วยเกื้อกูลกันและกัน โดยเฉพาะในด้านของการมองเห็น (Vision) ภาษามนุษยชาติ (Natural language) และการเรียนรู้ (Learning)

1.1.2 ระบบที่กระทำเหมือนมนุษย์ (Systems that act like humans)

การทดสอบว่าคอมพิวเตอร์มีการกระทำเหมือนมนุษย์หรือไม่ ทำได้โดยใช้การทดสอบทัวริง (Turing's Test) ซึ่งเป็นการทดสอบที่ออกแบบโดยนักคณิตศาสตร์ชาวอังกฤษชื่อ อัลัน ทัวริง (Alan Turing) ใช้สำหรับตัดสินว่าเป้าหมายมีความฉลาดในระดับที่น่าพอใจหรือไม่ วิธีทดสอบคือให้คนเป็นผู้สอบถามคอมพิวเตอร์ผ่านเครื่องพิมพ์ระยะไกล (Teletype) เพื่อที่คอมพิวเตอร์จะได้ไม่ต้องได้ยินเสียงหรือรูปร่างของคน ถ้าผู้สอบถามบอกไม่ได้ว่าผู้ตอบคำถามเป็นคนหรือเครื่องกันแน่ ถือว่าเครื่องนั้นผ่านการทดสอบ หลักการของวิธีทดสอบเช่นนี้คือเครื่องคอมพิวเตอร์ที่แสวงหาตนเป็นมนุษย์ หลอกหลวงผู้สอบถามได้สำเร็จ จะได้รับการพิจารณาว่าเป็นเครื่องที่ฉลาด คอมพิวเตอร์ที่ผ่านการทดสอบทัวริงได้นั้น ต้องมีความสามารถต่อไปนี้

1. ประมวลผลภาษามนุษยชาติ (Natural language processing) หมายถึงการรับรู้และเข้าใจในระดับที่สามารถติดต่อสื่อสารกับมนุษย์ได้เป็นอย่างดี ภาษามนุษยชาติที่นิยมใช้กันเป็นส่วนใหญ่ได้แก่ภาษาอังกฤษ

2. การแทนความรู้ (Knowledge representation) เพื่อเก็บข้อมูลที่รับมาทั้งก่อนหน้าและในระหว่างทำการทดสอบ

3. การใช้เหตุผลอัตโนมัติ (Automated reasoning) เพื่อนำข้อมูลที่เก็บไว้มาใช้ตอบคำถาม และสร้างข้อสรุปใหม่ ๆ

4. การเรียนรู้ของเครื่อง (Machine learning) คือมีการเรียนรู้ความรู้ใหม่เพื่อสามารถปรับตัวให้เข้ากับสภาพแวดล้อมใหม่ ๆ และค้นหาแบบแผนข้อมูล

การทดสอบทัวริงนั้นเป็นการทดสอบจากผู้ทดสอบหรือผู้สังเกตการณ์เพียงด้านเดียว เครื่องใดที่ผ่านการทดสอบควรถือว่าฉลาดก็จริง แต่เครื่องจักรบางเครื่องก็ยังสามารถได้รับการพิจารณาว่าฉลาดโดยที่ไม่ต้องพูดคุยรู้เรื่องกับมนุษย์หรือไม่ต้องเลียนแบบมนุษย์ให้เห็น

นักวิจัยด้าน AI หลายคนให้ความสนใจกับการผ่านการทดสอบน้อยมาก แต่หันมาสนใจศึกษาหลักการของความฉลาดมากกว่าจะต้องการเลียนแบบสิ่งที่ฉลาด เช่น การศึกษาด้าน “Artificial flight” สิ้นสุดลงเมื่อพี่น้องตระกูลไรท์ และคนอื่น ๆ เลิกเลียนแบบนก แล้วหันมาศึกษาวิชาความรู้ทางด้านอากาศพลศาสตร์แทน

1.1.3 ระบบที่คิดอย่างมีเหตุผล (Systems that think rationally)

นักปรัชญาชาวกรีกชื่ออริสโตเติลกล่าวว่า การคิดถูกต้อง (Right thinking) คือ “กระบวนการใช้เหตุผลที่หักล้างไม่ได้” และเป็นผู้สร้างกฎการคิด (Laws of thought) ที่มีชื่อเสียงว่าเป็นจุดเริ่มต้นของสาขาวิชาตรรกะในเวลาต่อมา กฎที่มีชื่อเสียงข้อหนึ่งของอริสโตเติลเกี่ยวกับโครงสร้างของประโยคกล่าวว่า ถ้าให้คำกล่าวอ้าง (Premises) ที่ถูกต้อง จะทำให้ได้ข้อสรุป (Conclusions) ที่ถูกต้องเสมอ ตัวอย่างเช่น “Socrates is a man; all men are mortal, therefore Socrates is mortal.”

ปัญหาของการใช้วิธีทางตรรกะคือ การแทนความรู้ในรูปของตรรกะเป็นเรื่องยาก โดยเฉพาะเมื่อความรู้ที่มีความไม่แน่นอนแฝงอยู่ด้วย

1.1.4 ระบบที่กระทำอย่างมีเหตุผล (Systems that act rationally)

การกระทำอย่างมีเหตุผล คือการกระทำเพื่อให้บรรลุเป้าหมายของบุคคลนั้น ๆ ตามความรู้และความเชื่อของบุคคลนั้น

แนวคิดของการสร้างระบบที่กระทำอย่างมีเหตุผลคือการใช้วิธีการแบบ Rational agent approach โดยเอเจนต์ (agent) หมายถึงสิ่งใด ๆ ที่สามารถรับรู้ (Perceive) และมีการกระทำ (Act) เมื่อใช้แนวคิดนี้ AI จะหมายถึงการศึกษาและสร้างเอเจนต์ที่ฉลาดมีเหตุผล (Rational agents)

1.2 ความเป็นมาของ AI

เมื่อคอมพิวเตอร์แบบอิเล็กทรอนิกส์พัฒนาขึ้นเมื่อปี ค.ศ.1941 นั้น ถือได้ว่าเทคโนโลยีที่ช่วยสร้างเครื่องจักรอันชาญฉลาด (Intelligent machine) ได้เกิดขึ้นแล้ว แต่คอมพิวเตอร์แรกสร้างมีขนาดใหญ่ ยากลำบากต่อการโปรแกรม จนกระทั่งมีการพัฒนาคอมพิวเตอร์ที่เก็บตัวโปรแกรมเอาไว้ได้ในปี ค.ศ.1949 การใส่โปรแกรมลงเครื่องจึงง่ายขึ้น ความก้าวหน้าในทฤษฎีจึงก้าวไปสู่วิทยาการคอมพิวเตอร์ และสุดท้ายก็ถึงปัญญาประดิษฐ์ กล่าวได้ว่าวิธีประมวลผลข้อมูลแบบอิเล็กทรอนิกส์เป็นสื่อให้ AI เกิดขึ้นได้

หลังสงครามโลกครั้งที่ 2 มีคนจำนวนมากเริ่มสนใจทำงานวิจัยเพื่อสร้างเครื่องจักรอันชาญฉลาด อาจกล่าวได้ว่า อัลัน ทัวริง เป็นคนแรกที่บรรยายเรื่องนี้ในปี ค.ศ.1947 และเป็นคนแรกที่ตัดสินใจจะวิจัย AI ได้ดีที่สุดในเมื่อใช้วิธีโปรแกรมคอมพิวเตอร์ มากกว่าจะใช้วิธีสร้างเครื่องจักรช่วย ปลายทศวรรษ 1950 จึงมีผลงานวิจัยทาง AI เกิดขึ้นเป็นจำนวนมาก โดยที่ส่วนใหญ่ใช้วิธีโปรแกรมคอมพิวเตอร์

ต้นทศวรรษ 1950 นอร์เบิร์ต ไวนอร์ ชาวอเมริกัน ตั้งข้อสังเกตเกี่ยวกับผลย้อนกลับ และได้ตั้งทฤษฎีผลย้อนกลับ (Feedback theory) ขึ้น ตัวอย่างของทฤษฎีนี้ได้แก่ โปรแกรมที่สังเกตอุณหภูมิจริงในบ้านเปรียบเทียบกับอุณหภูมิที่ต้องการ แล้วตอบสนองโดยการปรับอุณหภูมิความร้อนขึ้น-ลง จากการวิจัยของไวนอร์ในด้านนี้ เขาตั้งทฤษฎีขึ้นว่า พฤติกรรมอันชาญฉลาดทั้งหลายเป็นผลมาจากกลไกของผลย้อนกลับ เครื่องจักรหรือคอมพิวเตอร์สามารถจำลองกลไกเช่นนี้ได้ การค้นพบนี้มีผลอย่างมากต่อการพัฒนา AI ในยุคต้น

ในปี ค.ศ. 1955 นิวเวลและไซมอนพัฒนาโปรแกรมชื่อ The Logic Theorist ถือกันว่าเป็นโปรแกรม AI โปรแกรมแรก ใช้แก้ปัญหาคณิตศาสตร์เรื่องตัวแบบต้นไม้ (Tree model)

ในปี ค.ศ. 1956 จอห์น แมคคาร์ธีย์ ได้จัดการประชุมขึ้นที่มหาวิทยาลัยดาร์ตมัธ เชิญผู้เชี่ยวชาญที่สนใจในการพัฒนาเครื่องจักรอันชาญฉลาด นักวิจัยทางทฤษฎีออโตมาตา ช่างเส้นประสาท (Neural net) รวมทั้งผู้สนใจจากมหาวิทยาลัยพรินซ์ตัน เอ็มไอที (Massachusetts Institute of Technology) บริษัทไอบีเอ็ม รวมหลายฝ่ายมาช่วยกันระดมสมอง การประชุมนี้ใช้ชื่อว่าโครงการวิจัยเกี่ยวกับปัญญาประดิษฐ์ และจากนั้นเป็นต้นมา สาขาวิชาทางด้านนี้ก็ถูกเรียกว่าปัญญาประดิษฐ์ (Artificial intelligence – AI)

หลังการประชุมครั้งนี้ AI เริ่มเป็นที่สนใจมากขึ้น มีการก่อตั้งศูนย์วิจัย AI ขึ้นที่มหาวิทยาลัยคาร์เนกี-เมลลอน และเอ็มไอที

ปี ค.ศ. 1958 จอห์น แมคคาร์ธีย์ย้ายจากมหาวิทยาลัยดาร์ตมัธมาอยู่ที่เอ็มไอที และได้พัฒนาภาษาลิสป์ (LISP) ขึ้นเพื่อใช้ในงานด้าน AI ชื่อของลิสป์มาจากคำว่า LIST Processing และไม่นานหลังจากนั้น ภาษานี้ได้นำมาใช้กันอย่างแพร่หลายในหมู่ผู้พัฒนา AI

ในปี ค.ศ. 1963 เอ็มไอทีได้รับทุนจากรัฐบาลสหรัฐ 2.2 ล้านดอลลาร์เพื่อวิจัยเกี่ยวกับการทำให้คอมพิวเตอร์สามารถรับรู้ได้ ซึ่งเป็นส่วนหนึ่งของ AI และได้ทุนจากหน่วยงานด้านการวิจัยของกระทรวงกลาโหม (Defense Advanced Research Projects Agency, DARPA) โครงการเช่นนี้ทำให้ AI ก้าวหน้าขึ้นมาก มีนักวิทยาการคอมพิวเตอร์มาจากทั่วโลกมาร่วมงาน ระบบผู้เชี่ยวชาญหลายระบบได้รับการพัฒนามาใช้ในการแปลผลทางสถิติ งานพยากรณ์ตลาดหุ้น ช่วยหมอ

วินิจฉัยโรค ช่วยชาวเหมืองค้นพบตำแหน่งแร่ และยังเกิดทฤษฎีใหม่ในด้านการมองเห็น คอมพิวเตอร์สามารถแยกแยะระดับสีของภาพ รูปร่าง กรอบ และพื้นผิว มูลค่าการซื้อขาย ฮาร์ดแวร์และซอฟต์แวร์ทางด้าน AI ในระหว่างช่วงปี ค.ศ.1980 จึงสูงมากถึง 425 ล้านดอลลาร์ ส่วนใหญ่จะเน้นเรื่องของระบบผู้เชี่ยวชาญ ส่วน AI สาขาอื่นที่เข้าสู่ตลาดในระหว่างนั้นคือเรื่องของการมองเห็น เช่นการใช้กล้องถ่ายภาพกับคอมพิวเตอร์ในสายงานประกอบผลิตภัณฑ์ แม้ว่าจะยังทำงานได้อย่างหยاب ๆ แต่ก็สามารถแยกแยะความแตกต่างของรูปร่างของวัตถุได้

แต่ในปี ค.ศ. 1986-87 ความต้องการระบบ AI กลับลดน้อยลง ธุรกิจด้านนี้เสียหายกว่าห้าพันล้านดอลลาร์ การสนับสนุนที่ได้รับจากหน่วยงาน DARPA ถูกตัดออก และโครงการพัฒนาหุ่นยนต์เพื่อใช้ในสนามรบก็ถูกยกเลิกทุนสนับสนุนที่เคยได้รับจากกระทรวงกลาโหมสหรัฐ

แม้ว่าจะมีเหตุการณ์ที่ไม่เอื้ออำนวยเกิดขึ้นมาก แต่ AI ก็ฟื้นตัวขึ้นมาได้อย่างช้า ๆ ตัวอย่างเช่นประเทศสหรัฐอเมริกาเป็นประเทศที่บุกเบิกงานด้านฟัซซีลอจิก (Fuzzy logic) ขึ้นเป็นครั้งแรก ฟัซซีลอจิกเป็นตรรกะที่สามารถตัดสินใจภายใต้ความไม่แน่นอนได้ นอกจากนี้ยังมีโครงข่ายเส้นประสาท (Neural network) ที่ได้รับการพิจารณาว่าจะเป็นตัวนำสู่ความสำเร็จของ AI เทคโนโลยีเหล่านี้มีส่วนช่วยให้ AI ก้าวไปข้างหน้าต่อไป

ปัจจุบัน เทคโนโลยีที่ใช้ AI จำนวนมาก ได้เข้ามาแทรกอยู่ในชีวิตประจำวันของประชาชนในศตวรรษที่ 21 ทางทหารสหรัฐอเมริกา นำฮาร์ดแวร์ที่มีพื้นฐานของ AI ไปทดลองใช้ในสงครามอ่าว (ปี ค.ศ.1991) เทคโนโลยีทาง AI ถูกนำมาใช้ในระบบจรวด การแสดงภาพ และอื่น ๆ ตลอดจนองค์การนาซาใช้หุ่นยนต์สำรวจดาวอังคาร คอมพิวเตอร์แอปเปิลแมคอินทอช และเครื่องคอมพิวเตอร์ตระกูลไอบีเอ็มมีความสามารถในการรับรู้เสียงและอักขระ เทคโนโลยี AI ได้สร้างเครื่องแคมคอร์ดเดอร์ (Camcorder) โดยใช้ฟัซซีลอจิกแบบง่าย ๆ และปัจจุบัน ญี่ปุ่นสร้างของเล่นหุ่นยนต์ เช่นหุ่นยนต์สุนัข โดยอาศัยเทคโนโลยีของ AI

เมื่อ AI มีความก้าวหน้ามากขึ้น สามารถแก้ปัญหาต่าง ๆ ได้หลากหลายขึ้น นักวิจัย AI ในยุคนี้จึงหันกลับมาสนใจปัญหาการสร้างสิ่งที่มีความฉลาดที่เรียกชื่อทั่วไปว่าเอเจนต์ฉลาด (Intelligent agent) ขึ้นมาอีกครั้งหนึ่ง โดยให้ความสนใจกับการทำงานของเอเจนต์ภายในสภาพแวดล้อมจริงตามที่เอเจนต์รับรู้ได้จากอินพุตทางการรับรู้ (Sensory input) สภาพแวดล้อมที่สำคัญและมักจะใช้อยู่เสมอได้แก่อินเทอร์เน็ต ระบบงานปัญญาประดิษฐ์จึงมักจะเป็นแอปพลิเคชันบนเว็บ (Web-based application) และเทคโนโลยีที่ใช้ใน AI ก็ต้องอาศัยเครื่องมือบนเว็บมาช่วย เช่น Search engine และเครื่องมือสร้างเว็บไซต์ เป็นต้น

อย่างไรก็ตาม AI ต้องการใช้ความรู้หลายด้านเข้ามาผสมผสาน จึงจะได้ผลลัพธ์ที่ดี เพราะถึงปัจจุบันนี้ ระบบการรับรู้ เช่น การมองเห็น โซนาร์ เครื่องรับรู้คำพูด ยังไม่อาจให้ข้อมูลที่นำเชื่อถือเกี่ยวกับสภาพแวดล้อมได้สมบูรณ์แบบ ดังนั้นจึงต้องนำระบบการใช้เหตุผล และระบบวางแผนมาช่วยเสริมเพิ่มเติม เพื่อแก้ไขความไม่แน่นอนที่อาจเกิดขึ้น

1.3 ปัญหาที่ต้องใช้ AI

งานด้าน AI ในสมัยแรกเน้นไปทางด้านการเล่นเกม และการพิสูจน์ทฤษฎี มีผู้เขียนโปรแกรมเล่นหมากรุกที่不仅能够เล่นกับคู่ต่อสู้เท่านั้น แต่ยังสามารถใช้ประสบการณ์ในเกมที่ผ่านมาปรับปรุงความสามารถให้ดีขึ้นได้อีกด้วย ส่วนโปรแกรม The Logic Theorist เป็นโปรแกรม AI ที่พัฒนาขึ้นเพื่อพิสูจน์ทฤษฎีทางคณิตศาสตร์ จะเห็นว่าบุคคลใดที่เล่นเกม และพิสูจน์ทฤษฎีได้เก่ง มักถูกมองว่าเป็นคนฉลาด โปรแกรมก็เช่นกัน ในสมัยก่อนคิดกันว่าการที่คอมพิวเตอร์ทำงานประเภทนี้ได้ดี เป็นเพราะใช้ความเร็วในการเลือกเส้นทางแก้ปัญหาที่ดีที่สุดจากเส้นทางที่มีอยู่เป็นจำนวนมาก และดูเหมือนจะไม่ได้ใช้ความรู้ใด ๆ ในกระบวนการทำงาน จึงน่าจะเขียนโปรแกรมได้ง่าย แต่ต่อมา ข้อสันนิษฐานเช่นนี้ได้รับการพิสูจน์ว่าผิด เพราะไม่มีคอมพิวเตอร์ใดที่เร็วพอที่จะเลือกเส้นทางแก้ปัญหาที่มีอยู่เป็นจำนวนมหาศาลได้

ในสมัยแรก AI มุ่งเน้นวิธีแก้ปัญหาตามแบบฉบับที่มนุษย์ใช้ตัดสินใจในการทำงานประจำวัน เรียกว่าการใช้เหตุผลตามหลักสามัญสำนึก (Commonsense reasoning) รวมถึงการใช้เหตุผลเกี่ยวกับสิ่งของ และความสัมพันธ์ระหว่างสิ่งของนั้น ใช้เหตุผลเกี่ยวกับการกระทำและผลที่ตามมา (เช่น ถ้าปล่อยของบางอย่าง ของนั้นจะตกลงสู่พื้น และอาจแตกได้) แต่ก็ไม่มีใครสร้างโปรแกรมที่ประกอบด้วยองค์ความรู้ขนาดใหญ่เกี่ยวกับปัญหานั้น มีแต่เลือกทำโปรแกรมสำหรับงานง่าย ๆ เท่านั้น

เมื่อการวิจัยทาง AI ก้าวหน้าขึ้น เริ่มมีการพัฒนาเทคนิคเพื่อจัดการกับองค์ความรู้ในโลก ซึ่งมีจำนวนมากมายมหาศาล เริ่มมีงานใหม่ ๆ เช่น ในด้านการรับรู้ การมองเห็น การพูด ความเข้าใจในภาษาธรรมชาติ และการแก้ปัญหาเฉพาะทาง เช่น การวินิจฉัยโรค และการวิเคราะห์ทางเคมี งานต่าง ๆ ทางด้าน AI แสดงได้ดังรูปที่ 1.1

AI มักจะเป็นผู้บุกเบิกหลายสิ่งในวิทยาการคอมพิวเตอร์ ภาษาระดับสูงหลายภาษา อินเทอร์เน็ตของคอมพิวเตอร์ และเวิร์ดโปรเซสเซอร์ ต่างก็เป็นผลมาจากการวิจัย AI ทฤษฎีและสิ่งที่เกิดขึ้นจาก AI กลายเป็นแนวโน้มในอนาคตของคอมพิวเตอร์ ผลิตภัณฑ์หลายสิ่งในปัจจุบันเกิดขึ้นเพราะ AI เห็นได้ว่า ความก้าวหน้าของ AI จะยังคงให้ผลกระทบอย่างต่อเนื่องไปสู่การงาน การศึกษา และชีวิตของคนเราต่อไปอย่างไม่หยุดยั้ง

งานทั่วไป

การรับรู้

- การมองเห็น
- การพูด

ภาษาธรรมชาติ

- เข้าใจ
- สร้างประโยค
- แปล

การใช้เหตุผลตามสามัญสำนึก

การควบคุมหุ่นยนต์

งานเฉพาะ

เกม

- หมากรุก
- แบ็กแกมมอน
- โกะ
- หมากระดาน

คณิตศาสตร์

- เรขาคณิต
- ตรรกะ
- แคลคูลัสเชิงปริพันธ์ (Integral calculus)
- พิสูจน์คุณสมบัติของโปรแกรม

งานผู้เชี่ยวชาญ

วิศวกรรม

- ออกแบบ
- หาที่ผิดพลาด (Fault finding)
- แผนการผลิต

วิเคราะห์ทางวิทยาศาสตร์

วินิจฉัยโรค

วิเคราะห์การเงิน

รูปที่ 1.1 งานด้านต่างๆ ของ AI

ผู้ที่รู้วิธีทำงาน AI ในรายการของรูปที่ 1.1 ต้องเรียนรู้ทักษะที่จำเป็นตามลำดับที่เป็นมาตรฐาน ก่อนอื่นคือเรียนรู้ทักษะของการรับรู้ ภาษา และสามัญสำนึก ต่อมาจึงเรียนทักษะที่ต้องการความเชี่ยวชาญเฉพาะ เช่น วิศวกรรมศาสตร์ การแพทย์ หรือการเงิน อาจดูเหมือนว่าทักษะประเภทแรกจะง่ายกว่า และสามารถนำลงสู่คอมพิวเตอร์ได้ง่ายกว่าทักษะของเรื่องเฉพาะ แต่ที่จริงแล้ว ปรากฏว่าเป็นตรงกันข้าม แม้ว่าทักษะของความเชี่ยวชาญเฉพาะจะต้องการความรู้ในเรื่องที่คนส่วนใหญ่ไม่มี แต่ความรู้ที่ต้องการใช้จะมีน้อยกว่าที่ทักษะทั่วไปต้องการเป็นอย่างมาก และความรู้เหล่านั้นก็นำมาไว้ในโปรแกรมได้ง่ายกว่าอีกด้วย

ดังนั้น งานที่ง่ายที่สุด และทำให้เกิดเป็นรูปธรรมได้มากที่สุด คืองานที่ต้องการความชำนาญเฉพาะทางที่ไม่ต้องอาศัยความรู้ตามสามัญสำนึก ปัจจุบันมีโปรแกรมสนับสนุนโปรแกรม เรียกว่าระบบผู้เชี่ยวชาญ (Expert system) ใช้ปฏิบัติงานประจำวันในอุตสาหกรรม และรัฐบาลหลายด้าน แต่ละระบบพยายามแก้ไขปัญหา อาจแก้ไขเป็นบางส่วน หรือทั้งหมดก็ได้ แต่มักจะใช้กับปัญหาสำคัญที่ก่อนหน้านี้ต้องใช้นักผู้เชี่ยวชาญมาทำ

1.4 การศึกษาด้านต่าง ๆ ของ AI

ศาสตร์ทางด้านนี้แยกออกเป็นสาขาย่อย ทั้งด้านทั่วไป (เช่นการรับรู้ การใช้เหตุผล) จนถึงเฉพาะทาง (เช่นการเล่นหมากรุก การพิสูจน์ทฤษฎีทางคณิตศาสตร์ การวินิจฉัยโรค การแต่งบทกวี)

AI เน้นการศึกษาในแต่ละสาขาดังนี้

1. ภาษาธรรมชาติ (Natural language) ทำให้คอมพิวเตอร์สื่อสารกับคนได้โดยใช้ภาษาธรรมชาติ เช่น ภาษาอังกฤษ เป็นต้น การสื่อสารนี้มีทั้งทางการพิมพ์คำสั่งผ่านเทอร์มินัล หรือการที่จะสื่อสารให้เข้าใจคำพูดที่คนพูดออกไป (Speech)
2. การสรุปความ (Inference) ทำให้คอมพิวเตอร์จดจำข้อเท็จจริง (Fact) ที่สัมพันธ์กันอย่างสลับซับซ้อนได้ และสามารถหาข้อสรุปจากข้อเท็จจริงเหล่านั้น
3. การวางแผน (Planning) ทำให้คอมพิวเตอร์วางแผนการทำงานอย่างต่อเนื่องจนกระทั่งบรรลุเป้าหมาย
4. ระบบผู้เชี่ยวชาญ (Expert system) ทำให้คอมพิวเตอร์สามารถให้คำแนะนำแก่เราในสถานการณ์ต่าง ๆ โดยอาศัยกฎเกณฑ์อันซับซ้อนที่มีอยู่
5. การมองเห็น (Vision) ทำให้คอมพิวเตอร์มองเห็นผ่านทางกล้องถ่ายรูป และแยกแยะสิ่งของต่าง ๆ ได้

6. วิทยาการหุ่นยนต์ (Robotics) ทำให้คอมพิวเตอร์เคลื่อนไหวหรือเคลื่อนย้ายตัวเองและสิ่งของอื่นไปมาได้

หัวข้อที่เป็นแกนหลักของวิชา AI ได้แก่การสรุปความ การวางแผน และระบบผู้เชี่ยวชาญ ส่วนอีก 3 หัวข้อที่เหลือนั้นค่อนข้างจะเป็นเรื่องเฉพาะทาง แต่ละหัวข้อเหล่านี้เป็นเรื่องที่ต้องอาศัยเวลาในการทำวิจัยหลายปี และยิ่งยากจะวัดว่าประสบความสำเร็จเท่าใด เนื่องจากต้องพิจารณาเปรียบเทียบการทำงานของโปรแกรมกับพฤติกรรมของมนุษย์ ว่าเหมือนกันหรือต่างกันมากน้อยเพียงใด ซึ่งเรื่องนี้เป็นเรื่องที่น่าสนใจได้ยาก

1.5 เทคนิคของ AI

เทคนิคและแนวคิดของ AI ดูเหมือนว่าเป็นสิ่งที่เข้าใจได้ยากกว่าเรื่องอื่นอีกมากมายในศาสตร์ของคอมพิวเตอร์ เหตุผลประการหนึ่งคือ มีรายละเอียดมากมายที่เราต้องคำนึงถึง AI จะทำงานได้ดีที่สุดกับปัญหาที่ซับซ้อน ซึ่งใช้แก้โดยวิธีทั่วไปไม่ได้ ปัญหาเช่นนี้อาจเขียนได้ยาวหลายหน้า และแตกต่างจากปัญหาทางคณิตศาสตร์ที่เห็นกันอยู่ดาษดื่น

ความซับซ้อนนี้เป็นเครื่องจำกัดไม่ให้โปรแกรมเมอร์สามารถเข้าใจได้ถ่องแท้ว่า เกิดอะไรขึ้นในโปรแกรม AI บ่อยครั้งที่โปรแกรมทำตัวเป็นแบบจำลอง โดยโปรแกรมเมอร์กำหนดเงื่อนไขของพฤติกรรมของโปรแกรมไว้แล้ว แต่ทันทีที่เริ่มต้นทำงาน เราไม่อาจทราบได้เลยว่าจะเกิดอะไรขึ้น ลักษณะของโปรแกรมเช่นนี้แตกต่างจากโปรแกรมที่เขียนด้วยภาษาระดับสูงเช่นฟอร์แทรน ปาสคาล หรือเอด้า ซึ่งเราสามารถติดตามและรู้ว่าโปรแกรมกำลังจะทำอะไรอย่างละเอียดทุกขั้นตอน แต่เทคนิคของ AI มักจะเป็นวิธีเดียวที่ใช้แก้ปัญหาทำได้สำเร็จ แม้ว่าจะติดตามรายละเอียดได้ลำบากก็ตาม

ปัญหาปัญญาประดิษฐ์มีขอบเขตที่กว้างขวาง แต่ละเรื่องมีความแตกต่างกันไปจนแทบไม่มีสิ่งใดร่วมกัน แต่ยังคงมีเทคนิคที่เหมาะสมอยู่จำนวนหนึ่งที่น่าสนใจกับปัญหาอันหลากหลายเหล่านี้ได้ เทคนิคที่ AI ต้องการนำมาใช้จะมีคุณสมบัติอย่างไรบ้างนั้น ต้องพิจารณาจากงาน AI ทั่ว ๆ ไปอย่างกว้าง ๆ ก่อน

ผลจากการวิจัยทาง AI ในระยะเวลาเกือบ 30 ปี พบว่า ความฉลาด (Intelligence) ต้องการใช้ความรู้ (Knowledge) เป็นเครื่องช่วย แต่ความรู้มีคุณสมบัติที่ไม่พึงปรารถนาหลายข้อ เช่น

1. ความรู้มีปริมาณมาก
2. การบอกคุณลักษณะของความรู้ให้ชัดเจนเป็นเรื่องที่ทำได้ยาก
3. ความรู้มักมีการเปลี่ยนแปลงอย่างต่อเนื่อง

4. ความรู้ต่างจากข้อมูล ตรงที่การรวบรวมความรู้จะรวบรวมไปตามลักษณะของวิธีใช้ความรู้

กล่าวได้ว่าเทคนิคของ AI คือวิธีการนำความรู้มาใช้ประโยชน์ โดยที่ความรู้นั้นต้องถูกนำเสนอในลักษณะต่อไปนี้

1. ความรู้ต้องแสดงความเป็นทั่วไป ไม่เฉพาะเจาะจง (Generalization) นั่นคือ ไม่จำเป็นต้องแสดงถึงสถานการณ์แต่ละอย่างแยกจากกัน สถานการณ์ใดที่มีคุณสมบัติสำคัญเหมือนกัน ควรถูกรวมเข้าด้วยกัน แทนที่จะแยกความรู้ออกไป เพราะไม่เช่นนั้น ความรู้จะต้องใช้เนื้อที่เก็บจำนวนมาก และต้องปรับปรุงอยู่เสมอ ซึ่งนั่นเป็นลักษณะของข้อมูลมากกว่าความรู้

2. ผู้ที่จัดหาความรู้ต้องมีความเข้าใจในรู้นั้นอย่างถ่องแท้ ไม่ใช่เพียงแค่จัดหามาให้ได้ปริมาณมาก แต่ต้องเป็นสิ่งที่สำคัญ และเป็นที่เข้าใจของผู้จัดหา

3. ปรับปรุงแก้ไขที่ผิดพลาดได้ง่าย และสามารถสะท้อนความเปลี่ยนแปลงของโลกได้ นั่นคือ สามารถรับรู้สถานการณ์ที่เปลี่ยนไปได้ทันที

4. ใช้ได้หลายสถานการณ์ แม้ว่าจะไม่ถูกต้องแม่นยำ หรือสมบูรณ์แบบเต็มที่

5. นำมาใช้เป็นแนวทางช่วยชี้แนะการแก้ปัญหาให้มีช่วงเป็นไปได้ที่แคบลง

แม้ว่าเทคนิคของ AI จะได้รับการออกแบบให้เหมาะกับปัญหาของ AI แต่ก็ยังมีบางครั้งที่ปัญหากับเทคนิคการแก้ปัญหาเหล่านี้ไปด้วยกันไม่ได้หรือไม่เหมาะสม ปัญหาทาง AI หลายอย่างแก้ไขได้โดยไม่ต้องใช้เทคนิคเหล่านี้เลย (แต่อาจจะได้คำตอบที่ไม่ดีนัก) และในทำนองเดียวกัน เราอาจประยุกต์ใช้เทคนิคเหล่านี้กับการแก้ปัญหาที่ไม่ใช่ AI ได้ด้วย

พิจารณาตัวอย่างของการแก้ปัญหาทั้งโดยการใช้และไม่ใช้เทคนิคของ AI ได้จากตัวอย่างโปรแกรมเล่นเกมทิก-แทค-โท ดังนี้

โปรแกรม 1

โครงสร้างข้อมูล

กระดาน แทนด้วยเวกเตอร์ 9 สมาชิก (9-element vector) โดยสมาชิกแต่ละตัวมีตำแหน่งบนกระดานดังนี้

1	2	3
4	5	6
7	8	9

ค่าที่เก็บในสมาชิกเป็น 0 ถ้าช่องนั้นว่างเปล่า มีค่าเป็น 1 ถ้าช่องนั้นเต็ม X และจะมีค่าเป็น 2 ถ้าช่องนั้นเต็ม O

ตารางการเดินแต่ละตา (Movetable)

กระดานที่ผู้เล่นเดินแต่ละตาจะมีลักษณะแตกต่างกันได้ถึง 19,683 กระดาน (3^9) แต่ละกระดานแทนได้ด้วยเวกเตอร์ 9 สมาชิก (นั่นคือตารางการเดินแต่ละตาจะเก็บเวกเตอร์ไว้ 19,683 เวกเตอร์)

อัลกอริทึม

ในการเดินแต่ละตา มีขั้นตอนดังนี้

1. มองกระดานที่เป็นเวกเตอร์ให้เป็นเลขฐาน 3 (สมาชิกในกระดานมีค่า 0-2) แล้วเปลี่ยนเป็นเลขฐาน 10
2. ใช้ตัวเลขที่คำนวณได้จากข้อ 1 เป็นดัชนีชี้ไปยังตารางการเดินแต่ละตา (Movetable) เพื่อไปหาเวกเตอร์ที่เก็บไว้
3. เวกเตอร์ที่เลือกได้จากข้อ 2 จะแสดงลักษณะของกระดานที่เกิดขึ้นหลังจากการเดินในตา นั้น ต่อจากนี้ให้เซตกระดานให้เป็นไปตามเวกเตอร์ที่เลือกไว้

ข้อสังเกต

โปรแกรมนี้มีประสิทธิภาพมากในด้านเวลา และในทางทฤษฎีแล้วสามารถเล่นเกมทิก-แทค-โทที่ตีที่สุด แต่มีข้อเสียหลายข้อคือ

1. ใช้เนื้อที่จำนวนมากในการเก็บตารางการเดินแต่ละตา (ซึ่งแทนด้วยเวกเตอร์ 9 สมาชิก)
2. การจัดทำตารางการเดินต้องใช้เวลามากกว่าจะสร้างตารางเก็บข้อมูลเสร็จ
3. อาจเกิดข้อผิดพลาดในการป้อนข้อมูลตารางได้เพราะในส่วนนี้ต้องใช้คนเป็นผู้กระทำ
4. ถ้าต้องการขยายเกมออกไป เช่นเป็นเกม 3 มิติ ใช้เทคนิคนี้ไม่ได้ เพราะ 3^{27} เป็นจำนวนสมาชิกที่มากเกินไปกว่าจะเก็บในหน่วยความจำของคอมพิวเตอร์ปัจจุบัน

เทคนิคที่ใช้ในโปรแกรมนี้อย่างไม่เป็นไปตามลักษณะการนำความรู้มาใช้ตามเทคนิคของ AI ต่อไปพิจารณาวิธีที่ดีกว่านี้

โปรแกรม 2

โครงสร้างข้อมูล

กระดาน แทนด้วยเวกเตอร์ 9 สมาชิก เหมือนกับในโปรแกรม 1 แต่แทนที่จะใช้ตัวเลข 0, 1, 2 ในแต่ละสมาชิก ให้เก็บค่า 2 (แทนว่างเปล่า), ค่า 3 (แทน X) และค่า 5 (แทน O)

ตาเดิน ใช้ตัวเลข 1-9 แทนลำดับของการเดินแต่ละตาที่กำลังจะเดิน 1 หมายถึงการเดินตาแรก และ 9 หมายถึงการเดินตาสุดท้าย (เกมนี้เดินได้ 9 ตา เพราะลงได้ 9 ช่องเท่านั้น)

อัลกอริทึม

อัลกอริทึมหลักประกอบด้วยกระบวนการย่อย (Subprocedure) 3 กระบวนการ คือ

Make2 ให้ค่าเป็น 5 ถ้าช่องกลางกระดานว่าง (นั่นคือ ถ้ากระดาน [5] = 2) ไม่เช่นนั้น ฟังก์ชันนี้จะให้ค่าเป็นหมายเลขช่องที่ไม่ใช่มุมและว่าง เช่น (2, 4, 6, หรือ 8)

หมายเหตุ กระดาน [5] หมายถึง ช่องหมายเลข 5 ของกระดาน

Posswin(p) ให้ค่าเป็น 0 ถ้าผู้เล่น p ไม่สามารถชนะในตาต่อไปของเขาได้ ไม่เช่นนั้น ฟังก์ชันนี้จะให้ค่าเป็นหมายเลขของช่องที่สามารถทำให้ชนะได้ ฟังก์ชันนี้ทำให้โปรแกรมเอาชนะได้ และยังกันคู่ต่อสู้ได้อีกด้วย ในการทำงานแต่ละครั้ง Posswin จะตรวจสอบแต่ละแถวบน แถวตั้ง แถวทแยง และจากวิธีการที่ค่าในสมาชิกถูกตั้งเป็นตัวเลขไว้ (เป็นเลข 2, 3, 5) จึงสามารถเช็คทั้งแถว (แนวตั้ง แนวบน และแนวทแยง) เพื่อดูว่ามีโอกาสชนะได้หรือไม่ โดยการคูณค่าทุกช่องเข้าด้วยกัน ถ้าผลคูณเป็น 18 ($3 \times 3 \times 2$) X จะชนะ แต่ถ้าผลคูณเป็น 50 ($5 \times 5 \times 2$) O จะชนะ ถ้าพบแถวที่จะชนะได้ เราก็ตัดสินใจได้ว่าสมาชิกใดเป็นช่องว่าง และคืนค่าเป็นหมายเลขของช่องนั้น

Go(n) เดินไปที่ช่องหมายเลข n กระบวนการนี้จะเช็คค่ากระดาน[n] เป็น 3 ถ้าตาเดินเป็นเลขคู่ และจะตั้งค่าให้เป็น 5 ถ้าเป็นเลขคู่ จากนั้นจะบวกค่าตาเดินเพิ่มขึ้นอีก 1

อัลกอริทึมใช้กลยุทธ์ที่มีอยู่ในตัวเองกับการเดินแต่ละตา ถ้าเล่น X จะเป็นการเดินในเลขคู่ ถ้าเล่น O จะเป็นการเดินในเลขคู่ กลยุทธ์ในแต่ละตาเดินเป็นดังนี้

ตาเดิน = 1	Go(1) (มุมบนซ้าย)
ตาเดิน = 2	ถ้า กระดาน[5] ว่าง Go(5) ไม่เช่นนั้น Go(1)
ตาเดิน = 3	ถ้า กระดาน[9] ว่าง Go(9) ไม่เช่นนั้น Go(3)
ตาเดิน = 4	ถ้า Posswin(X) ไม่ใช่ 0 แล้ว Go(Posswin(X)) [นั่นคือกันไม่ให้คู่ต่อสู้ชนะ] ไม่เช่นนั้น Go(Make2)
ตาเดิน = 5	ถ้า Posswin(X) ไม่ใช่ 0 แล้ว Go(Posswin(X)) [นั่นคือชนะ] ไม่เช่นนั้น ถ้า Posswin(O) ไม่ใช่ 0 แล้ว Go(Posswin(O)) [กันไม่ให้ชนะ] ไม่เช่นนั้น ถ้า กระดาน[7] ว่าง Go(7) ไม่เช่นนั้น Go(3) [ณ จุดนี้ โปรแกรม พยายามแทรกเข้ามาแยกแนวคู่ต่อสู้]
ตาเดิน = 6	ถ้า Posswin(O) ไม่ใช่ 0 แล้ว Go(Posswin(O)) ไม่เช่นนั้น ถ้า Posswin(X) ไม่ใช่ 0 แล้ว Go(Posswin(X)) ไม่เช่นนั้น Go(Make2)
ตาเดิน = 7	ถ้า Posswin(X) ไม่ใช่ 0 แล้ว Go(Posswin(X)) ไม่เช่นนั้น ถ้า Posswin(O) ไม่ใช่ 0 แล้ว Go(Posswin(O)) ไม่เช่นนั้น ไปที่ใดก็ได้ที่ว่าง
ตาเดิน = 8	ถ้า Posswin(O) ไม่ใช่ 0 แล้ว Go(Posswin(O)) ไม่เช่นนั้น ถ้า Posswin(X) ไม่ใช่ 0 แล้ว Go(Posswin(X)) ไม่เช่นนั้น ไปที่ใดก็ได้ที่ว่าง
ตาเดิน = 9	เหมือนตาเดินที่ 7

ข้อสังเกต

โปรแกรมนี้ไม่ค่อยมีประสิทธิภาพด้านเวลาเท่ากับโปรแกรมแรก เพราะต้องเช็คเงื่อนไขมากก่อนจะลงตาเดินแต่ละตา แต่มีประสิทธิภาพมากกว่าในด้านเนื้อที่หน่วยความจำ การทำความเข้าใจกับกลยุทธ์ที่โปรแกรมใช้เป็นเรื่องง่าย หรือแม้แต่จะปรับเปลี่ยนกลยุทธ์ก็เช่นกัน เพราะผู้เขียนโปรแกรมได้แสดงกลยุทธ์โดยรวมไว้ล่วงหน้าแล้วในกระบวนการทั้ง 3 กระบวนการนั้น จากโปรแกรมนี้ เรายังไม่สามารถทำความรู้ของโปรแกรมให้เป็นเรื่องทั่วไปได้มากพอที่จะนำไปใช้กับสถานการณ์ที่ต่างออกไปเช่น ทิค-แทค-โท แบบ 3 มิติได้

โปรแกรม 2'

โปรแกรมนี้คล้ายกับโปรแกรม 2 ยกเว้นแต่มีการเปลี่ยนแปลงการแทนค่ากระดาน กระดานยังคงแทนได้ด้วยเวกเตอร์ 9 สมาชิก แต่ตำแหน่งของช่องในกระดานจะถูกกำหนดขึ้นใหม่ ไม่ได้เรียงจาก 1-9 เหมือนในโปรแกรม 2 ดังนี้

8	3	4
1	5	9
6	7	2

ตำแหน่งของช่องในกระดานก่อให้เกิดสี่เหลี่ยมมหัศจรรย์ (Magic square) นั่นคือ แต่ละแถวทั้งแนวนอน แนวตั้ง และแนวทแยง มีผลรวมเท่ากันคือ 15 ทำให้กระบวนการตรวจสอบโอกาสของการชนะทำได้ง่ายขึ้น แทนที่จะกำหนดค่าในช่องกระดานขณะที่เดินไปแต่ละตา (ว่าเป็น 3 หรือ 5) ก็ใช้วิธีเก็บหมายเลขช่องเดินของผู้เล่นแต่ละคนเอาไว้ นำหมายเลขช่องของผู้เล่นมาคู่หนึ่ง คำนวณโดยนำ 15 ตั้ง ลบด้วยผลบวกของหมายเลขช่องคู่หนึ่ง ถ้าผลลัพธ์ไม่เป็นเลขบวก หรือมากกว่า 9 แสดงว่าช่องทั้งคู่ไม่เป็นแนวตรงกัน ไม่ต้องสนใจ ไม่เช่นนั้น ดูว่าช่องหมายเลขเดียวกันกับผลลัพธ์นั้นว่างหรือไม่ ถ้าว่าง การลงตาเดินในช่องนั้นจะทำให้ชนะ เนื่องจากผู้เล่นคนหนึ่ง ๆ จะมีช่องเดินอยู่ในความครอบครองได้ไม่เกิน 4 ช่องในการเล่นแต่ละครั้ง การตรวจสอบเช่นนี้จึงมีจำนวนช่องให้ตรวจสอบไม่มากเมื่อเทียบกับการตรวจสอบในโปรแกรมที่ 2 จะเห็นว่าการเลือกการแทนค่าที่เหมาะสมมีผลต่อประสิทธิภาพของการแก้ปัญหาของโปรแกรมเป็นอย่างมาก

ข้อสังเกต

การเปรียบเทียบระหว่างโปรแกรม 2 และ 2' ก่อให้เกิดคำถามที่น่าสนใจข้อหนึ่งเกี่ยวกับความสัมพันธ์ระหว่างวิธีที่คนแก้ปัญหา กับวิธีที่คอมพิวเตอร์แก้ปัญหา เพราะเหตุใดคนจึงพบว่าการกราดตรวจแต่ละแถวเป็นเรื่องง่ายกว่า ในขณะที่คอมพิวเตอร์พบว่าการใช้วิธีนับจำนวนมีประสิทธิภาพกว่า เราไม่รู้อย่างถ่องแท้ว่า คนทำได้อย่างไรจึงตอบคำถามดังกล่าวได้ครบถ้วนสมบูรณ์ คำตอบข้อหนึ่งอาจเป็นเพราะว่าคนใช้โปรเซสเซอร์แบบขนาน (Parallel processor) และสามารถมองกระดานหลาย ๆ ส่วนได้ในเวลาเดียวกัน (มองรวดเดียวทั้งกระดาน) ขณะที่คอมพิวเตอร์ต้องมองกระดานไปที่ละช่อง บางครั้ง การสืบเสาะหาวิธีที่คนแก้ปัญหาก็มีประโยชน์

เพื่อให้คอมพิวเตอร์หันมาใช้วิธีแบบเดียวกัน แต่บางครั้งความแตกต่างด้านฮาร์ดแวร์กับกายภาพของคนมีมากจนทำให้แต่ละฝ่ายใช้กลยุทธ์แตกต่างกันไป

โปรแกรม 3

โครงสร้างข้อมูล

ตำแหน่งช่องในกระดาน โครงสร้างประกอบด้วย เวกเตอร์ 9 สมาชิก แทนกระดาน
รายการของตำแหน่งช่องในกระดานที่มีผลต่อตาเดินต่อไป และตัวเลขที่
แทนตำแหน่งช่องในกระดานที่กะประมาณว่าจะนำไปสู่ชัยชนะในตาเดิน
ของผู้เล่น

อัลกอริทึม

การตัดสินใจว่าจะลงตาต่อไปอย่างไร ทำได้โดยมองตำแหน่งกระดานล่วงหน้าหลังจากลง
ตาที่เป็นไปได้แต่ละตา แล้วดูว่าตำแหน่งไหนดีที่สุด เดินไปในเส้นทางที่จะทำให้ได้กระดาน
ตำแหน่งนั้น และกำหนดแต้มต่อของตาเดินดีที่สุดนั้นให้กับตำแหน่งกระดานปัจจุบัน

การตัดสินใจว่าตำแหน่งกระดานตำแหน่งใดดีที่สุด มีขั้นตอนดังนี้

1. ดูว่าทำให้ชนะหรือไม่ ถ้าใช่ ก็มอบตำแหน่งดีที่สุดโดยกำหนดแต้มต่อสูงสุดให้
2. ไม่เช่นนั้น พิจารณาทุกตาเดินที่คู่ต่อสู้จะเดินเป็นตาต่อไป ดูว่าตาไหนที่จะทำให้เรา
ลำบากที่สุด (โดยเรียกกระบวนการนี้มาทำซ้ำ ๆ แบบ Recursive) ให้สมมุติว่าคู่ต่อสู้จะเลือกเดิน
ตานั้น กำหนดแต้มต่อให้กับตาเดินที่เรากำลังพิจารณาอยู่
3. ตาเดินที่ดีที่สุดจะเป็นอันที่แต้มต่อดีที่สุด

อัลกอริทึมนี้มองไปข้างหน้าถึงลำดับ (Sequence) ของตาเดินหลาย ๆ ตา เพื่อค้นหาลำดับ
ที่จะนำไปสู่ชัยชนะ พยายามจะทำให้มีโอกาสชนะสูงสุด ขณะเดียวกันก็ตั้งสมมติฐานว่าคู่ต่อสู้
จะพยายามลดโอกาสชนะของเราให้มากที่สุดด้วยเช่นกัน อัลกอริทึมเช่นนี้เรียกว่า Minimax
procedure

ข้อสังเกต

โปรแกรมนี้อาจใช้เวลามากกว่าโปรแกรมอื่น เพราะต้องค้นหาในโครงสร้างแบบต้นไม้ที่ชี้แทน
ลำดับของตาเดินทุกตาที่เป็นไปได้ก่อนที่จะลงตาเดินทุกตา แต่มีข้อดีเหนือโปรแกรมอื่น ๆ อยู่มาก
กล่าวคือสามารถขยายวิธีจัดการกับเกมให้กว้างขวางซับซ้อนขึ้นมากกว่าเกมทิก-แทค-โท ธรรมชาติ
ได้ ในขณะที่โปรแกรมอื่นทำไม่ได้ สามารถขยายไปสู่การใช้ความรู้แบบอื่น ๆ หลากหลายแบบ
รวมถึงวิธีเล่นเกม ตัวอย่างเช่น แทนที่จะพิจารณาตาเดินต่อไปทุกตาเท่าที่จะเป็นไปได้

ก็อาจพิจารณาเฉพาะเซตย่อยของตาเดินที่ดูแล้ว (ด้วยอัลกอริทึมง่าย ๆ) เห็นว่ามีเหตุผลสมควร และแทนที่จะทำตามชุดของการเดินไปจนชนะ ก็อาจสามารถค้นหาและประเมินผลตำแหน่งของ กระดานไว้ก่อน โดยหาฟังก์ชันบางอย่างมาช่วยตัดสิน

โปรแกรมนี้เป็นตัวอย่างของการใช้เทคนิคของ AI ในปัญหาที่มีขนาดเล็กมาก การใช้เทคนิค อาจจะมีประสิทธิภาพน้อยกว่าการใช้วิธีตรงไปตรงมา แต่เทคนิคเหล่านี้ใช้ได้ดีกับสถานการณ์ที่วิธี ตรง ๆ ใช้ไม่ได้ผล

โดยสรุป เทคนิคสำคัญที่ AI มักใช้อยู่เสมอ ได้แก่

1. ค้นหา เป็นวิธีแก้ปัญหที่ตรงไปตรงมาที่สุด
2. ใช้ความรู้มาช่วย เป็นวิธีแก้ปัญหที่ซับซ้อน โดยอาศัยโครงสร้างของสิ่งที่เกี่ยวข้อง
3. นามธรรม (Abstraction) เป็นวิธีที่แยกลักษณะสำคัญและความหลากหลายต่าง ๆ ออกมาจากสิ่งที่ไม่ค่อยสำคัญอื่น ๆ

ในการแก้ปัญหายาก ๆ โปรแกรมที่ใช้เทคนิคเหล่านี้จะได้เปรียบกว่าโปรแกรมที่ไม่ใช้ การใช้ ความรู้ในโปรแกรมเป็นสิ่งที่เข้าใจได้ง่าย และเทคนิคเหล่านี้มักใช้ได้ดีกับปัญหขนาดใหญ

จากการนำ AI มาใช้แก้ปัญหในงานด้านต่าง ๆ สามารถประเมินความสำเร็จของ AI ไว้ กว้าง ๆ 4 ระดับ ดังนี้

1. ดีที่สุด (Optimal) คือระดับที่ไม่สามารถทำงานได้ดีไปกว่านี้อีกแล้ว
2. เหนือมนุษย์อย่างยิ่ง (Strong super-human) คือระดับที่ทำงานได้ดีกว่ามนุษย์ทั้งหลาย
3. เหนือมนุษย์ (Super-human) คือระดับที่ทำงานได้ดีกว่ามนุษย์ส่วนใหญ่
4. ต่ำกว่ามนุษย์ (Sub-human) คือระดับที่ทำงานได้แย่กว่ามนุษย์ส่วนใหญ่

ตัวอย่างเช่น โปรแกรมเกมในปัจจุบันนี้ พบว่าจากการที่นำมาใช้แข่งขันกับผู้เล่นที่เป็นมนุษย์ ถือว่าสมรรถนะของโปรแกรมหมากฮอสอยู่ในระดับ “ดีที่สุด” เนื่องจากสามารถเอาชนะแชมป์หมาก ฮอสของโลกในการแข่งขันได้ สมรรถนะของโปรแกรมหมากฮอสอยู่ในระดับ “เหนือมนุษย์” แต่ก็เกือบ ไกลแค่ียงระดับ “เหนือมนุษย์อย่างยิ่ง” สมรรถนะของเกมโกะอยู่ในระดับ “ต่ำกว่ามนุษย์” ส่วน สมรรถนะของโปรแกรมหรือเอเจนต์ที่ทำงานประจำวันของมนุษย์ส่วนใหญ่อยู่ในระดับ “ต่ำกว่ามนุษย์”

อย่างไรก็ตาม AI มีการพัฒนาและก้าวหน้าไปอย่างไม่หยุดยั้ง ในอนาคตจะมีการออกแบบ และสร้างเอเจนต์ต่าง ๆ ที่ทำงานแทนมนุษย์ และแก้ปัญหายาก ๆ ได้สำเร็จมากขึ้นเรื่อย ๆ เพราะ ศาสตร์ด้านนี้กว้างขวางครอบคลุมได้หลายสาขา ตั้งแต่ของเล่นเด็ก การช่วยงานในบ้านและใน สำนักงาน ไปจนถึงการควบคุมการยิงจรวดให้เข้าเป้าหมายในสงคราม หรือการส่งยานอวกาศไป สสำรวจดาวเคราะห์อื่นในระบบสุริยจักรวาล ล้วนแต่มี AI เป็นส่วนร่วมในการแก้ปัญหาทั้งสิ้น

แบบฝึกหัดบทที่ 1

1. จงศึกษาว่างานที่ใช้ AI ในข้อใดต่อไป่นี้สามารถแก้ปัญหาได้แล้วโดยใช้คอมพิวเตอร์
 - 1.1 เล่นปิงปองในระดับดี
 - 1.2 ขับรถไปสนามหลวง
 - 1.3 ชื่อของที่ชุปเปอร์มาร์เก็ต
 - 1.4 ชื่อของบนเว็บ
 - 1.5 เล่นไพ่บริดจ์ในระดับแข่งขันได้
2. จงศึกษาโปรแกรม Eliza (โดยค้นหาข้อมูลจากอินเทอร์เน็ต) แล้วอธิบายว่าโปรแกรมนี้จะผ่านการทดสอบทัวริงได้หรือไม่ เพราะเหตุใด
3. จงยกตัวอย่างการใช้แอปพลิเคชันที่อาศัยความสามารถในเรื่องต่อไปนี้
 - 3.1 ภาษามือ
 - 3.2 การสรุปความ
 - 3.3 การวางแผน
 - 3.4 ระบบผู้เชี่ยวชาญ
 - 3.5 การมองเห็น
 - 3.6 วิทยาการหุ่นยนต์
4. อ่านเรื่องต่อไปนี้ แล้วพิจารณาว่าในสาขาทั้ง 6 ด้านของ AI มีสาขาใดบ้างที่ถูกนำมาใช้ พร้อมทั้งยกหลักฐานสนับสนุนให้เห็นจริง

"จากตอนหนึ่งของการ์ตูนเรื่อง *บ๊ักส์บันนี่ กระต่ายแสนกล* เอลเมอร์ พัดจ์ ได้สั่งซื้อหุ่นยนต์มากำจัดบ๊ักส์บันนี่ และได้นำรูปภาพกระต่ายมาให้หุ่นยนต์ดู หุ่นยนต์จดจำรูปร่างนั้นไว้ แล้วออกเดินไปเรื่อย ๆ เพื่อหาว่ากระต่ายอยู่ที่ไหน เมื่อไปพบลาตัวหนึ่งยืนอยู่ในคอก หุ่นยนต์เห็นลามียูยาวคล้ายกับกระต่ายในรูปก็ยกปืนขึ้นยิงลาตัวนั้น เอลเมอร์วิ่งมาร้องบอกว่า ไม่ใช่ ๆ กระต่ายต้องมียูยาวแบบนี้ ว่าแล้วก็เอาหูกระต่ายมาติดที่หูตัวเอง แล้วทำท่ากระโดดสี่ขาเลียนแบบกระต่ายให้ดู หุ่นยนต์มองเอลเมอร์ แล้วก็ชักปืนออกมายิง"