บทที่ 2

ทฤษฎีที่ใช้ในการวิจัย

ทฤษฎีที่ใช้ในการวิจัยนี้ ประกอบด้วย การเรียนรู้ของเครื่อง (Machine Learning) การโปรแกรมตรรกะเชิงอุปนัย (Inductive Logic Programming) ระบบงานที่ใช้ในการเรียนรู้ตัวอักษร คือ ระบบ PROGOL และ ทฤษฎีการรู้จำแบบ (Pattern Recognition) รายละเอียดดังต่อไปนี้

<u>การเรียนรู้ของเครื่อง</u> (Machine Learning) [15]

การเรียนรู้ของมนุษย์มีอยู่ด้วยกันหลายรูปแบบ ตั้งแต่ การเรียนรู้จากคำบอกเล่า (learning by being told) จนถึง การเรียนรู้จากการค้นพบ (learning by discovery) ในส่วนแรก การเรียนรู้จากคำบอกเล่า ผู้สอนจะบอกความรู้ทั้งหมดแก่ผู้เรียนอย่างชัดเจน เปรียบได้กับการเขียนโปรแกรมคอมพิวเตอร์ เพื่อให้เครื่องคอมพิวเตอร์รู้ถึงขั้นตอนการทำงานทุกอย่างตามโปรแกรมนั้น ในอีกส่วนหนึ่ง การเรียนรู้จากการค้นพบ ผู้เรียนสามารถค้นพบความรู้ใหม่ได้เอง อาจเกิดจากการสังเกตสภาพแวดล้อม เช่น การเรียนรู้ของเด็ก หรือ อาจเกิดจากการวางแผนและทำการทดลอง เช่น การเรียนรู้ของนักวิทยาศาสตร์

การเรียนรู้จากตัวอย่าง (learning from examples) เป็นการเรียนรู้อีกแบบหนึ่ง ซึ่งอยู่ระหว่างการเรียนรู้ทั้งสองแบบที่กล่าวมาแล้ว โดยผู้สอนจะให้ตัวอย่าง และผู้เรียนจะสรุปแนวคิดที่ได้จากตัวอย่าง โดยมองถึงความสัมพันธ์ที่เหมือนกันของตัวอย่างนั้น และสรุปเป็นความรู้อย่างกว้างๆ (generalization) เพื่อใช้อธิบายตัวอย่างอื่นๆต่อไป ตัวอย่างที่ให้อาจแบ่งออกได้เป็น 2 ประเภท คือ ตัวอย่างที่สอดคล้องกับสิ่งที่ต้องการให้เรียนรู้ เรียกว่า ตัวอย่างบวก และ ตัวอย่างที่ไม่สอดคล้องกับสิ่งที่ต้องการให้เรียนรู้ เรียกว่า ตัวอย่างลบ ความรู้ที่ผู้เรียนสรุปได้ จะเป็นความรู้ที่สามารถอธิบายตัวอย่างบวกได้ และ ไม่ครอบคลุมตัวอย่างลบ ผู้สอนสามารถช่วยผู้เรียนได้ โดยการให้ตัวอย่างที่เหมาะสม และพอเพียงสำหรับการเรียนรู้ วิธีการดังที่กล่าวมาแล้วนี้ เป็นวิธีการเรียนรู้ของเครื่อง ซึ่งนำไปใช้ประโยชน์ในการเรียนรู้หลายด้าน เช่น การตรวจโรคของผู้ป่วย การทำนายคุณสมบัติของส่วนประกอบทางเคมี และ การออกแบบวงจรคอมพิวเตอร์ รวมทั้ง ใช้ในการรู้จำตัวอักษรในงานวิจัยนี้ด้วย เทคนิคการเรียนรู้ของเครื่องนี้เหมาะสมมากกับงานที่ผู้ใช้งานสามารถหาตัวอย่างที่ดีได้ ซึ่งง่ายกว่าที่จะต้องค้นหาความรู้หรือทฤษฎีที่มีความซับซ้อน

ขั้นตอนการทำงานดังต่อไปนี้ เปรียบได้กับการเรียนรู้จากตัวอย่าง คือ ระบบการเรียนรู้พัฒนาแนวคิดใหม่ (เช่น แนวคิดว่าอะไรที่ทำให้ผู้ป่วยเป็นโรคเบาหวาน) โดยการสร้างความรู้อย่างกว้างๆจากตัวอย่าง (ตัวอย่างได้แก่ ผู้ป่วยโรคเบาหวาน และ ผู้ป่วยที่ไม่เป็นโรคเบาหวานในอดีต) แนวคิดที่ได้จากการเรียนรู้ จะนำไปใช้ในการทำนายตัวอย่างในอนาคต (กรณีนี้ คือ ใช้ในการตรวจวินิจฉัยโรคเบาหวานของผู้ป่วยต่อไป)

<u>การโปรแกรมตรรกะเชิงอุปนัย</u> (Inductive Logic Programming, ILP) [13]

การโปรแกรมตรรกะเชิงอุปนัย หรือเรียกโดยย่อว่า ไอแอลพี (ILP) เป็นงานวิจัยที่รวมคุณลักษณะเด่นของ 2 หลักการ คือ การเรียนรู้ของเครื่อง และ การโปรแกรมตรรกะ (Logic Programming) กล่าวคือ ผู้สอนจะให้ตัวอย่างแก่ระบบไอแอลพีในรูปแบบของการโปรแกรมตรรกะ ระบบไอแอลพีจะใช้หลักการเรียนรู้ของเครื่อง โดยการเรียนรู้จากตัวอย่างที่ได้รับ และ สร้างสมมติฐานขึ้น ซึ่งสมมติฐานที่ได้อยู่ในรูปแบบของการโปรแกรมตรรกะเช่นเดียวกัน นับเป็นข้อดีของระบบไอแอลพีที่เหนือกว่าวิธีการเรียนรู้ของเครื่องแบบอื่น เนื่องจาก การใช้รูปแบบของการโปรแกรมตรรกะ ทำให้ง่ายในการให้ตัวอย่าง และผลการเรียนรู้ที่ได้ก็อยู่ในรูปแบบที่ชัดเจน และสามารถทำความเข้าใจได้ง่าย นอกจากนี้ ผู้สอนยังสามารถให้ความรู้ส่วนหลังแก่ระบบไอแอลพี ในรูปแบบของการโปรแกรมตรรกะด้วย อาจกล่าวได้ว่า เป็นการรวมวิธีการเรียนรู้จากคำบอกเล่า และ การเรียนรู้จากตัวอย่าง ซึ่งระบบไอแอลพีจะใช้ความรู้ส่วนหลังที่ได้รับในการสร้างสมมติฐาน

โดยส่วนมากแล้ว ระบบไอแอลพีจะแบ่งได้เป็น 3 ส่วนประกอบ คือ ตัวอย่าง (E) ความรู้ส่วนหลัง (B) และ สมมติฐาน (H) ซึ่งทั้ง 3 ส่วนมีความสัมพันธ์กันดังนี้

$$B \wedge H \models E$$

โดยที่ B, H และ E คือ โปรแกรมตรรกะ

ทฤษฎีของไอแอลพีอ้างอิงอยู่กับทฤษฎีการพิสูจน์ (proof theory) และทฤษฎีแบบอย่าง (model theory) สำหรับเพรดิเคตอันดับที่หนึ่ง การสร้างสมมติฐานโดยการอุปนัยทำได้โดยใช้เทคนิค ซึ่งประกอบด้วย ไออาร์ (Inverse Resolution, IR) อาร์แอลจีจี (Relative Least General Generalizations, RLGG) อินเวอร์สอิมพลิเคชัน (Inverse Implication) และ ไออี (Inverse Entailment, IE) นอกจากนั้น การขจัดความซ้ำซ้อน และ กระบวนการในการค้นหา ก็เป็นเทคนิคที่มีบทบาทสำคัญในทฤษฎีของไอแอลพีด้วยเช่นกัน ส่วนการวิเคราะห์ผลลัพธ์ที่เกิดจากการเรียนรู้ จะใช้ทฤษฎีการเรียนรู้ที่เกี่ยวกับการคำนวณ (Computational Learning Theory, COLT)

การประยุกต์ใช้งานระบบไอแอลพีที่ประสบความสำเร็จในปัจจุบัน ได้แก่ การเรียนรู้กฎเกี่ยวกับสตรัคเจอร์แอคติวิตี้สำหรับการออกแบบยา (learning of structure-activity rules for drug design) กฎเกี่ยวกับการวิเคราะห์และออกแบบไฟไนท์อิลิเมนท์เมช (finite-element mesh analysis design rules) การทำนายโครงสร้างของโปรตีนอันดับหนึ่งและอันดับสอง (primary-secondary prediction of protein structure) กฎเพื่อตรวจสอบข้อผิดพลาดสำหรับดาวเทียม (fault diagnosis rules for satellites) และ การ

เรียนรู้เพื่อใช้ในการแปลงกริยาในภาษาอังกฤษจากกาลปัจจุบันให้เป็นกริยาที่เกิดขึ้นในอดีต (learning to transform English verbs from present to past tense)

ตัวอย่างระบบไอแอลพี ได้แก่ CIGOL, GOLEM, FOIL, PROGOL เป็นต้น

ระบบ PROGOL

การรู้จำตัวพิมพ์อักษรภาษาไทยในงานวิจัยนี้ ระบบที่ใช้ในการเรียนรู้ตัวอักษร คือ ระบบ PROGOL รุ่น 4.1 ซึ่งพัฒนาโดย Stephen Muggleton เหตุผลที่เลือกใช้ระบบ PROGOL เนื่องจาก เป็นระบบการเรียนรู้ที่มีประสิทธิภาพ ผลลัพธ์จากการเรียนรู้อยู่ในรูปแบบของกฎหรือนิยามที่มนุษย์สามารถเข้าใจ และนำไปใช้ประโยชน์ได้ รวมทั้ง ยังเป็นระบบที่สามารถใช้ในงานวิจัยทางด้านการศึกษาได้ โดยไม่เสียค่าใช้จ่าย ในเนื้อหาต่อไปนี้จะกล่าวถึง ความหมายของระบบ PROGOL การใช้งานระบบ PROGOL พร้อมกับแสดงตัวอย่างการใช้งาน ผลลัพธ์ที่ได้จากการเรียนรู้ และขั้นตอนการเจนเนอรัลไรซ์ (generalise) ของ PROGOL สำหรับการติดตั้งระบบ PROGOL และรายละเอียดการใช้งานอื่นๆ จะอยู่ในภาคผนวก ง

1. ความหมายของระบบ PROGOL [15]

1.1 ระบบ PROGOL คือ ระบบการเรียนรู้ของเครื่อง (Machine Learning)

1.2 ระบบ PROGOL เป็นรูปแบบของการโปรแกรมตรรกะเชิงอุปนัย (Inductive Logic Programming)

2. การใช้งานระบบ PROGOL

การใช้งาน PROGOL สามารถทำได้ 2 แบบ คือ ภาวะเชิงโต้ตอบ (Interactive mode) และ ภาวะประมวลผลแบบกลุ่ม (Batch mode) เนื่องจากในงานวิจัยนี้ มีข้อมูลที่ต้องใช้ในการเรียนรู้จำนวนมาก จึงเหมาะกับการใช้งานในภาวะประมวลผลแบบกลุ่ม โดยนำข้อมูลทั้งหมดที่จะใช้ในการเรียนรู้ จัดเก็บในแฟ้มข้อมูลที่ลงท้ายด้วย .pl เช่น min.pl แล้วใช้คำสั่งเริ่มทำงาน ดังนี้

$ progol <ชื่อแฟ้มข้อมูล>

เช่น $ progol min

ข้อมูลที่ใช้ในการเรียนรู้ของ PROGOL ประกอบด้วย 4 ส่วน ดังนี้

2.1 คำสั่งของโปรแกรม ใช้ในการประกาศรูปแบบ (mode declarations) และ การกำหนดค่าพารามิเตอร์ (parameter settings) เพื่อระบุรูปแบบ (mode) ที่เป็นไปได้ในการเรียกใช้ (call) เพรดิเคต ซึ่งในแต่ละเพรดิเคตสามารถประกาศรูปแบบมากกว่าหนึ่งรูปแบบได้ ความหมายของสัญลักษณ์ที่ใช้ คือ

- modeh เป็นการกำหนดรูปแบบของสัญพจน์ส่วนหัว (head literals)

modeh(RecallNumber, HeadTemplete)?

โดยที่ RecallNumber คือ การระบุจำนวนครั้งที่มากที่สุดที่สามารถเรียกใช้เพรดิเคตดังกล่าวได้ โดยอาจระบุเป็นตัวเลข หรือ ใช้เครื่องหมาย * เพื่อระบุจำนวนเต็มค่าหนึ่งที่กำหนดไว้ (ค่าปกติ คือ 100)

HeadTemplete คือ รูปแบบที่ถูกต้องสำหรับสัญพจน์ส่วนหัว

- modeb เป็นการกำหนดรูปแบบของสัญพจน์ส่วนเนื้อความ (body literals)

modeb(RecallNumber, BodyLiteralTemplete)?

โดยที่ Recallnumber มีความหมายดังที่กล่าวมาแล้วข้างต้น

BodyLiteralTemplete คือ รูปแบบที่ถูกต้องสำหรับสัญพจน์ส่วนเนื้อความ

- เครื่องหมาย + แสดงถึง อาร์กิวเมนต์ที่ใช้ในการผ่านค่าให้กับสัญพจน์ (input argument)
- เครื่องหมาย - แสดงถึง อาร์กิวเมนต์ที่ใช้ในการผ่านค่าผลลัพธ์ (output argument)
- เครื่องหมาย # แสดงถึง อาร์กิวเมนต์ที่ใช้ในการผ่านค่า ซึ่งอาจเป็นค่าคงที่ (constant)
- เครื่องหมาย +, - และ # จะตามด้วยชนิดของอาร์กิวเมนต์ที่ใช้ในการผ่านค่า ซึ่งอาจเป็นเพรดิเคตที่ระบุเป็นความรู้ส่วนหลัง หรือ ฟังก์ชันที่กำหนดไว้อยู่แล้วในโปรล็อก (Prolog built-in function) เช่น any(X), float(X), int(X), nat(X) หรือ constant(X)

ข้อความที่เป็นหมายเหตุ (comment) จะปรากฏที่ใดก็ได้ในแฟ้มข้อมูล โดยนำหน้าด้วยเครื่องหมาย %

2.2 ความรู้ส่วนหลัง (background knowledge) เป็นกลุ่มของอนุประโยคตามแบบภาษาโปรล็อก ใช้กำหนดความรู้ให้กับระบบ ซึ่ง PROGOL จะใช้ความรู้ส่วนหลังดังกล่าวในการสร้างอนุประโยคผลลัพธ์

2.3 ตัวอย่างบวก (positive examples) เป็นกลุ่มของอนุประโยค ซึ่งแสดงถึงตัวอย่างที่สอดคล้องกับความสัมพันธ์ที่ต้องการเรียนรู้

2.4 ตัวอย่างลบ (negative examples) เป็นกลุ่มของสัญพจน์ที่แสดงถึง ตัวอย่างที่ไม่สอดคล้องกับความสัมพันธ์ที่ต้องการเรียนรู้ เขียนนำหน้าด้วยเครื่องหมาย :-

ตัวอย่าง แฟ้มข้อมูล min.pl เพื่อให้ PROGOL เรียนรู้การหาสมาชิกที่มีค่าน้อยที่สุดจากสมาชิกทั้งหมดในลิสต์ (list)

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% 1. การประกาศรูปแบบในการเรียกใช้เพรดิเคต
% รูปแบบสัญพจน์ส่วนหัว คือ min(A,B) โดยที่ A คือ ตัวแปรชนิด ilist ผ่านค่าให้กับสัญพจน์ และ B คือ
%       ตัวแปรชนิด int ใช้ผ่านค่าผลลัพธ์ นั่นคือ สมาชิกที่มีค่าน้อยที่สุดในลิสต์ A
        :- modeh(1,min(+ilist,-int))?
% รูปแบบสัญพจน์ส่วนเนื้อความ
        :- modeb(1,min(+ilist,-int))?
        :- modeb(1,least(+int,+int,-int))?
        :- modeb(1,+any= #any)?
        :- modeb(1,+ilist=[-int|-ilist])?
% การกำหนดชนิดของตัวแปร
% กำหนดให้ ilist เป็นตัวแปรแบบลิสต์ และ มีสมาชิกเป็นตัวแปรชนิด int
        ilist([]).
        ilist([H|T]) :- int(H), ilist(T).

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% 2. ความรู้ส่วนหลัง
% least(A,B,C) คือ การรับค่า A และ B เพื่อเปรียบเทียบหาค่าที่น้อยที่สุด และให้ค่ากับ C
        least(X,Y,Z) :- X=<Y, Z=X.
        least(X,Y,Z) :- Y=<X, Z=Y.

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% 3. ตัวอย่างบวก
        min([3,1,2],1).
        min([1,2,3],1).
        min([5,2,4],2).
        min([2,4,5],2).
        min([5,3,2],2).
```

```
min([2,5,3],2).
min([5,3,3],3).
min([3,5,7],3).
min([3,2,1],1).
min([8,3,2],2).
min([2,8,3],2).
% การให้ตัวอย่างบวกในลักษณะของอนุประโยค
min([X],X).
min([X,Y],X) :- X=<Y.
min([X,Y],Y) :- Y=<X.

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% 4. ตัวอย่างลบ
:- min([1,2,3],3).
:- min([3,2,1],2).
:- min([2,1],2).
:- min([1,1],2).
:- min([1,2],2).
:- min([1,2],0).
```

3. ผลลัพธ์ที่ได้จากระบบ PROGOL

ผลลัพธ์ที่ได้จะอยู่ในรูปของอนุประโยคฮอร์น นอกจากนั้น ผู้ใช้งานสามารถตามรอย (trace) การเรียนรู้ของระบบได้ โดยระบบจะแสดงขั้นตอนต่างๆในระหว่างการเรียนรู้ภายในเครื่องหมาย [ ] ในการค้นหา ระบบจะแสดงทุกอนุประโยคที่ระบบทำการทดลอง โดยแสดงตัวเลข 4 ตัว ก่อนหน้าการแสดงอนุประโยคดังกล่าว คือ

f = จำนวนตัวอย่างบวกที่สอดคล้องกับอนุประโยค -
จำนวนตัวอย่างลบที่สอดคล้องกับอนุประโยค -
จำนวนสัญพจน์ในเนื้อความของอนุประโยค -
ค่าประมาณจำนวนสัญพจน์ที่จำเป็น

p = จำนวนตัวอย่างบวกที่สอดคล้องกับอนุประโยค

n = จำนวนตัวอย่างลบที่สอดคล้องกับอนุประโยค

h = ค่าประมาณจำนวนสัญพจน์ที่จำเป็น

ตัวอย่าง อนุประโยคฮอร์นที่ได้จากการเรียนรู้ของ PROGOL โดยใช้แฟ้มข้อมูล min.pl

```
min([X],X).
min([A|B],C) :- min(B,D), least(D,A,C).
```

4. ขั้นตอนการเจนเนอรัลไรซ์อนุประโยคของ PROGOL

ขั้นตอนที่ 1. PROGOL เลือกอนุประโยคแรก เพื่อทำการเจนเนอรัลไรซ์ ซึ่งโดยปกติจะเลือกตัวอย่างแรกก่อน

ขั้นตอนที่ 2. สำหรับแต่ละตัวอย่างที่ได้จากขั้นตอนที่ 1 PROGOL จะสร้างอนุประโยคที่เฉพาะมากที่สุด (the most specific clause) ที่อธิบายตัวอย่างนั้นๆ

ขั้นตอนที่ 3. สำหรับแต่ละอนุประโยคที่ได้จากขั้นตอนที่ 2 PROGOL จะค้นหาแลตทิซ (lattice) เพื่อหาอนุประโยคที่เจนเนอรัล (general) มากขึ้น ซึ่งจะต้องครอบคลุมตัวอย่างลบได้ไม่เกินค่าที่กำหนดในพารามิเตอร์ "noise"

ขั้นตอนที่ 4. PROGOL จะลบตัวอย่างทั้งหมดที่สอดคล้องกับอนุประโยคที่ได้จากขั้นตอนที่ 3

ขั้นตอนที่ 5. PROGOL จะกระทำซ้ำตั้งแต่ขั้นตอนที่ 1 ถึง 4 จนกว่าจะไม่เหลือตัวอย่างอีก

การสร้างอนุประโยคจากตัวอย่าง PROGOL จะทำการค้นหาแบบเจนเนอรัลไปสู่สเปคซิฟิค (general-to-specific) PROGOL ต่างจากระบบไอแอลพีอื่นๆหลายระบบ เนื่องจากแลตทิซที่ค้นหาโดย PROGOL นั้น จะถูกกำหนดขอบเขตของส่วนบน และ ส่วนล่าง ส่วนบนของแลตทิซ คือ อนุประโยคว่าง (empty clause) และส่วนล่าง คือ อนุประโยคที่เฉพาะมากที่สุด PROGOL ใช้หลักการของเอ็มดีไออาร์ (Mode-Directed Inverse Resolution, MDIR) ระบบ PROGOL จะทำการค้นหาแบบอันดับแรกที่ดีที่สุด (best-first search) ระหว่าง 2 จุดนี้ในแลตทิซ โดยใช้การวัดการบีบอัด (compression) การบีบอัดของอนุประโยคจะคำนวณได้จาก P - N - C โดยที่ P คือ จำนวนของตัวอย่างบวกที่สอดคล้องกับอนุประโยค, N คือ จำนวนของตัวอย่างลบที่สอดคล้องกับอนุประโยค และ C คือ จำนวนของสัญพจน์ในอนุประโยค

การรู้จำแบบ (Pattern Recognition) [9]

การรู้จำแบบ คือ การค้นหารูปแบบที่เก็บอยู่ในฐานความรู้ที่เหมือนกับรูปแบบที่ต้องการรู้จำ ส่วนประกอบของระบบการรู้จำแบบ ประกอบด้วย

1. ส่วนของการนำเข้า (input transducer) ทำหน้าที่ในการแปลงรูปแบบที่ต้องการวิเคราะห์ให้เป็นสัญญาณอิเล็กทรอนิกส์ อุปกรณ์นำข้อมูลเข้าที่นิยมใช้ ได้แก่ กล้องวีดีโอ (video cameras) เครื่องอ่านพิกัดภาพ (image digitizers) เครื่องกวาดตรวจ (scanners) และ ไมโครโฟน (microphones)

i 18016741

2. ส่วนการประมวลผลขั้นต้น (preprocessor) เป็นการกระทำกับสัญญาณตามเงื่อนไขที่เพิ่มเติมขึ้น และ อาจรวมถึงหน้าที่ในการขยายสัญญาณ สเปเชียลฟิลเตอร์ริง (spatial filtering) การวิเคราะห์สเปคตรัม และ การแปลงสัญญาณแอนะล็อกไปเป็นสัญญาณดิจิตอล

3. ส่วนของการดึงลักษณะ (feature extractor) หรือเรียกว่า ส่วนของการแสดงให้เห็นความแตกต่าง (discriminators) ทำหน้าที่ เช่น การจับคู่ต้นแบบ (template matching) สเปเชียลโคอินซิเดนท์ (spatial coincidence) และ ดีซิชันเซอเฟส (decision surfaces)

4. ส่วนของการเลือกผลลัพธ์ (response selector) เป็นขั้นตอนการเลือกรูปแบบที่เก็บไว้ ซึ่งเป็นรูปแบบที่เหมือนกับรูปแบบที่ต้องการรู้จำมากที่สุด เทคนิคที่ใช้ประกอบด้วย การค้นหา การเรียงลำดับ และ การวิเคราะห์ลิสสแควร์ (least-squares analysis)

5. ส่วนการแสดงผลลัพธ์ (output systems) ระบบการแสดงผลลัพธ์อาจใช้ ตัวกำเนิดเสียง รูปภาพ หรือ จอภาพ เป็นสื่อในการแสดงผล

การประยุกต์ใช้หลักการรู้จำแบบมีอยู่เป็นจำนวนมาก ได้แก่ การรู้จำเสียง (voice recognition) การบ่งชี้ลายนิ้วมือ (fingerprint identification) การบ่งชี้หน้าตา (face identification) การบ่งชี้ลายมือเขียน (handwriting identification) การรู้จำตัวอักษร (character recognition) การวิเคราะห์สไลด์ทางชีววิทยา (biological slide analysis) การวิเคราะห์พลังงานสูงในทางฟิสิกส์ (high energy physics analysis) การวิเคราะห์ข้อมูลทางอุตุนิยมวิทยา (meteorological data analysis) การมองเห็นของหุ่นยนต์ (robot vision) เป็นต้น ซึ่งการรู้จำตัวอักษรพิมพ์ภาษาไทยในงานวิจัยนี้ ก็เป็นงานอีกด้านหนึ่งที่ใช้หลักการรู้จำแบบ

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย