รายงานโครงการวิจัยโดยใช้เงินรายได้คณะวิศวกรรมศาสตร์

ประจำปี 2551

การรู้จำตัวอักษรภาษาไทยโดยประยุกต์ใช้เทคนิคโทลีแรนท์ราฟเซต

Printed Thai Character Recognition using Tolerant Rough Sets

รองศาสตราจารย์ ดร. สมศักดิ์ มิตะถา

ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

สารบัญ

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

คอมพิวเตอร์เข้ามามีบทบาทต่อชีวิตมนุษย์เป็นอย่างมาก สามารถช่วยมนุษย์เราในการทำงานที่ยุ่งยากซับซ้อนให้เสร็จภายในเวลาอันสั้น ทำให้มนุษย์เรามีความสะดวกสบายยิ่งขึ้นกว่าแต่ก่อน แต่ความสามารถของเครื่องคอมพิวเตอร์ในปัจจุบันยังไม่สามารถจะนำไปใช้งานได้ดีในงานที่มีลักษณะอาศัยความฉลาดของมนุษย์ในการตัดสินใจ ซึ่งมีความไม่แน่นอนในการแก้ไขปัญหาเข้ามาเกี่ยวข้องด้วย รวมทั้งความเร็วในการแก้ปัญหาของเครื่องคอมพิวเตอร์ต่อปัญหาในลักษณะเช่นนี้ จะใช้เวลาในการแก้ปัญหานานเป็นอันมาก

สำหรับงานทางด้านการรู้จำรูปแบบ (Pattern Recognition) ซึ่งเป็นงานที่เกี่ยวข้องการกับแบ่งกลุ่มของข้อมูลที่สนใจออกเป็นกลุ่ม ๆ และจัดเอาข้อมูลที่ยังไม่ทราบรูปแบบ (Unknown Patterns) เข้าไปใส่ไว้ในกลุ่มซึ่งแบ่งออกนั้น โดยตัวอย่างของงานประเภทนี้ได้แก่ งานด้านการรู้จำตัวอักษร งานการจดจำลายมือ หรืองานทางด้านการแพทย์เช่น การวิเคราะห์เม็ดเลือด หรืองานวิเคราะห์กราฟคลื่นหัวใจ (Electrocardiogram Analysis) เป็นต้น อย่างไรก็ตามวิธีการทางด้านการรู้จำรูปแบบ หรือการจัดกลุ่มรูปแบบสามารถแบ่งออกได้เป็นสามกลุ่มใหญ่ ๆ ได้แก่ วิธีการรู้จำรูปแบบเชิงสถิติ (Statistical Pattern Classification), การจดจำรูปแบบโดยการหาความสัมพันธ์ (Syntactic Pattern Classification) และการรู้จำรูปแบบโดยอาศัยโครงข่ายประสาทเทียม (Neural Network-based Pattern Recognition)

1.2 ความมุ่งหมายและจุดประสงค์ของการศึกษา

งานวิจัยที่ประยุกต์ใช้ทฤษฎี Tolerant Rough Set [1], [2], [3], [4], [5], [6] ในช่วงแรกส่วนใหญ่จะเป็นงานวิจัยที่ไม่เกี่ยวเนื่องกับงานวิจัยทางด้านการจดจำรูปแบบ แต่ในระยะหลังได้เริ่มมีการนำเอาทฤษฎีดังกล่าวเข้ามาประยุกต์ใช้กับงานทางด้านการจดจำรูปแบบมากขึ้น โดยมีเหตุผลหลักคือ ทฤษฎี rough set ถูกออกแบบมาให้สามารถรับมือกับความคลุมเครือ (Vagueness) ของข้อมูลได้ นอกจากนั้นยังได้มีการพัฒนาทฤษฎี rough set ออกไปอีกหลายสาย ทั้งนี้มีจุดประสงค์หลักคือเพื่อให้สามารถจัดการกับปัญหาด้านความคลุมเครือของข้อมูลได้ดียิ่งขึ้น โดยหนึ่งในทฤษฎีที่พัฒนามาโดยใช้ rough set เป็นพื้นฐานนั้นได้แก่ tolerant rough set ซึ่งอาจกล่าวได้ว่าเป็นทฤษฎีที่ค่อนข้างใหม่ทีเดียว

งานวิจัยนี้เป็นงานการนำเอาทฤษฎี tolerant rough set มาประยุกต์ใช้กับงานทางด้านการจดจำรูปแบบ ซึ่งในที่นี้ได้นำเอาข้อมูลตัวอักษรไทย 43 ตัว (ยกเว้นตัว ศ) ซึ่งเป็นตัวอักษรพิมพ์มาใช้เป็นข้อมูลในการฝึกสอน (train) และทดสอบระบบ โดยระบบที่สร้างขึ้นในงานวิจัยนี้จะถูกแบ่งออกเป็น 3 ส่วนหลัก ๆ

โดยส่วนที่ 1 จะเป็นส่วนซึ่งทำหน้าที่ตัดตัวอักษรออกเป็นตัว ๆ แล้วส่งไปให้ส่วนที่ 2 โดยในส่วนที่ 2 นี้จะทำการหาคุณลักษณะเด่นของตัวอักษรนั้น ๆ และส่วนสุดท้าย (ส่วนที่ 3) คือการเรียนรู้ด้วย tolerant rough set

1.3 วัตถุประสงค์ของโครงการวิจัย

- เพื่อศึกษา Tolerant Rough Sets
- เพื่อหาแนวทางในการประยุกต์ใช้งาน Tolerant Rough Sets
- เพื่อพัฒนาระบบการรู้จำตัวอักษรภาษาไทยโดยใช้ Tolerant Rough Sets

1.4 ประโยชน์ที่คาดว่าจะได้รับ

- ความรู้เกี่ยวกับการประยุกต์ใช้งาน Tolerant Rough Sets
- ความเข้าใจปัญหาต่าง ๆ ที่เกิดขึ้นจากการนำเอาทฤษฎี Tolerant Rough Sets มาประยุกต์ใช้
- โปรแกรมต้นแบบ
- บทความทางวิชาการเกี่ยวกับผลการวิจัย

บทที่ 2

ทฤษฎีโทลีแรนท์ราฟเซต

ในบทนี้จะเป็นการอธิบายถึงเนื้อหาของทฤษฎีโทลีแรนท์ราฟเซต (Tolerant Rough Sets) โดย โทลีแรนท์ราฟเซต นี้เป็นราฟเซตประเภทหนึ่งที่ถูกดัดแปลงเพื่อให้สามารถรับมือกับข้อมูลที่มีความคลุมเครือ (Vagueness) และข้อมูลบางประเภท เช่น ข้อมูลซึ่งเป็น non-discrete ซึ่งราฟเซตปกติไม่สามารถจัดการได้อย่างมีประสิทธิภาพ แนวคิดหลักของโทลีแรนท์ราฟเซตคือ เป็นวิธีการทางคณิตศาสตร์แบบใหม่ในการวิเคราะห์ข้อมูลซึ่งไม่มีความแน่นอน คลุมเครือ จุดเริ่มต้นของปรัชญาแห่งทฤษฎีราฟเซตและโทลีแรนท์ราฟเซตมาจากแนวคิดที่จะนำเอา สารสนเทศ (ข้อมูล, ความรู้) ซึ่งเกี่ยวข้องกับทุก ๆ วัตถุในระบบที่สนใจมาเป็นตัวอธิบาย คุณลักษณะของวัตถุนั้น ตัวอย่างเช่น ถ้าวัตถุในที่นี้คือคนไข้ซึ่งป่วยด้วยโรคใดโรคหนึ่ง อาการต่าง ที่เกิดจากโรคนั้น ๆ จะประกอบกันขึ้นเป็นสารสนเทศ (Information) เกี่ยวกับคนไข้คนนั้น โดยจะกล่าวได้ว่าวัตถุนั้นเหมือนกัน ถ้าวัตถุเหล่านั้นมีสารสนเทศที่บรรยายคุณลักษณะของมันเหมือน กัน

2.1 การแสดงค่าความรู้

ในทฤษฎีราฟเซตและโทลีแรนท์ราฟเซตจะประกอบไปด้วยระบบซึ่งเป็นตัวแทนการแสดงความรู้ (Knowledge representation) อยู่ 2 ระบบ ได้แก่ ระบบสารสนเทศ (Information System) และระบบการตัดสิน (Decision System)

2.1.1 ระบบสารสนเทศ

ระบบสารสนเทศ (Information System) คือระบบซึ่งแสดงความรู้ (Knowledge) ในขั้นพื้นฐานที่สุด โดยจะประกอบได้ด้วยค่าคุณสมบัติ (attribute values) โดยระบบสารสนเทศคือคู่ลำดับ A=(U, A) โดย U ไม่เป็นเซตว่างและมีสมาชิกที่แน่นอน (nonempty, finite set) ซึ่งต่อไปนี้จะเรียกว่า "เซตเอกภพสัมพัทธ์" (universe) และ A คือเซตซึ่งไม่เป็นเซตว่างและมีสมาชิกเป็นค่าคุณสมบัติ (attributes) ซึ่งมีจำนวนที่แน่นอน แต่ละ$a \in A$ จะเป็นฟังก์ชัน $a:U \rightarrow V_a$ โดย V_a จะเป็นเซตของค่าของ a เรียกว่า "ช่วงของ a" (range of a) โดยแต่ละ element ของ universe จะถูกเรียกว่าวัตถุ (objects) ในระบบสารสนเทศนี้จะมีเพียงข้อมูลดิบของวัตถุเท่านั้น ไม่มีการแปล หรือการตีความ หรือการตั้งสมมุติฐานเกี่ยวกับวัตถุนั้น ๆ แต่ประการใด ดังแสดงไว้ในตารางที่ 2.1 และตารางที่ 2.2 จะเป็นตัวอย่างของระบบสารสนเทศ โดยวัตถุในที่นี้คือรถยนต์ และมีการวัดค่าคุณลักษณะ 3 ประการคือ color, price และ speed

ตาราง 2.1 รูปแบบของระบบสารสนเทศโดยทั่ว ๆ ไป

	a_1	• • •	a_j	• • •	a_m
x_1	a_{1x1}				a_{mx1}
⋮		⋱			
x_i			a_{jxj}		a_{mxj}
⋮				⋱	
x_n					a_{mxn}

ตารางที่ 2.2 ตัวอย่างของระบบสารสนเทศ ของรถยนต์ โดย x จะเป็นรถยนต์แต่ละคันและ a จะใช้อธิบายลักษณะของรถยนต์แต่ละคัน

	Color (a_1)	Price (a_2)	Speed (a_3)
x_1	Black	Cheap	Slow
x_2	Black	Cheap	Slow
x_3	Green	Cheap	Fast
x_4	White	Expensive	Slow
x_5	White	Affordable	Slow
x_6	Red	Expensive	Fast
x_7	Red	Expensive	Fast

2.1.2 ระบบการตัดสิน

ระบบการตัดสิน (Decision system) นี้จะมีลักษณะเหมือนกับระบบสารสนเทศ แต่จะแตกต่างกันตรงที่จะมีคุณลักษณะอยู่ 2 ประเภทในระบบชนิดนี้ คือคุณลักษณะที่เป็นเงื่อนไข (Condition Attributes) และ คุณลักษณะการตัดสิน (Decision Attributes)

ดังที่ได้กล่าวมาแล้วข้างต้นว่าในระบบสารสนเทศข้อมูลจะไม่ถูกตีความ แต่เมื่อมีการแยกประเภทของวัตถุในระบบโดยผู้เชี่ยวชาญ และมีการกำหนดค่าคุณลักษณะ ซึ่งเกิดจากการแยกประเภทโดยผู้เชี่ยวชาญนั้นให้แก่วัตถุ ก็จะทำให้ระบบสารสนเทศนั้นกลายเป็นระบบการตัดสิน (Decision System)

โดยคำจำกัดความของระบบการตัดสิน (Decision system) คือ ระบบการตัดสิน (Decision system) A=(U, A, {d}) คือระบบสารสนเทศซึ่งมีคุณลักษณะ (Attributes) ได้ถูกแบ่งแยกออกเป็น 2 กลุ่มที่ไม่เกี่ยวเนื่องกัน โดยคุณลักษณะ 2 กลุ่มดังกล่าวได้แก่ คุณลักษณะเงื่อนไข A (Condition Attribute As) และคุณลักษณะตัดสิน d (Decision Attributes d) และ $A \cap \{d\} = \varnothing$

ตัวอย่างของระบบการตัดสิน (Decision System) ได้แสดงไว้ในตารางที่ 2.3

ตารางที่ 2.3 ตัวอย่างของระบบการตัดสิน (Decision system) ของระบบสารสนเทศในตารางที่ 2.2

	Color (a_1)	Price (a_2)	Speed (a_3)	Manufacturer (d)
x_1	Black	Cheap	Slow	Lada
x_2	Black	Cheap	Slow	Lada
x_3	Green	Cheap	Fast	Volvo
x_4	White	Expensive	Slow	Volvo
x_5	White	Affordable	Slow	Volvo
x_6	Red	Expensive	Fast	Ferrari
x_7	Red	Expensive	Fast	Porsche

2.2 ความคล้ายกันของวัตถุ

เมื่อได้กำหนดรูปร่าง ขอบเขตของข้อมูล (วัตถุ) ในระบบแล้ว ในหัวข้อนี้จะกล่าวถึงการแยกแยะวัตถุในระบบ ในบรรดาวัตถุต่าง ๆ ในระบบจะมีทั้งวัตถุซึ่งมีคุณลักษณะที่แตกต่างกัน และวัตถุที่มีคุณลักษณะที่เหมือนกัน โดยการระบุความแตกต่างของวัตถุในระบบนั้นจะยึดเอาค่าคุณลักษณะ (attributes) ของวัตถุเป็นหลัก หรือเป็นข้อมูลในการแยกแยะการเกิดการเหมือนกันระหว่างวัตถุในระบบอันเนื่องมาจากค่าคุณลักษณะที่ถูกนำมาใช้งานนั้น (โดยที่ในความเป็นจริงวัตถุอาจจะไม่มีความเหมือนกันเลย) จะถูกเรียกว่า Indiscernibility โดยความสัมพันธ์แบบ "Indiscernibility" นี้จะทำให้สามารถแบ่งเซตเอกภพสัมพัทธ์ออกเป็นซับเซตย่อย ๆ ซึ่งปราศจากความเกี่ยวเนื่องกัน โดยในแต่ละซับเซตนั้นจะมีสมาชิกเป็นวัตถุซึ่งมีความเหมือนกัน หรือ มีความสัมพันธ์กันแบบ "Indiscernibility" โดยวัดจากค่าคุณลักษณะที่ได้เลือกมาใช้ และความสัมพันธ์แบบ Indiscernibility นี้สามารถนิยามได้ดังนี้ ให้ A = (U, A) เป็นระบบสารสนเทศ ทุก ๆ ซับเซตของค่าคุณลักษณะ (Attributes) ซึ่ง $B \subseteq A$ จะกำหนดความสัมพันธ์เท่าเทียม (Equivalent relation) $IND_A(B)$ ซึ่งต่อไปจะถูกเรียกว่า "ความสัมพันธ์แบบ Indiscernibility" (indiscernibility relation) โดยความสัมพันธ์ดังกล่าวจะสามารถกำหนดได้ดังนี้

$$IND_{\mathcal{A}}(B) = \{(x_i, x_j) \in \mathcal{U}^2 \mid \forall a \in B(\neg discerns(a, a(x_i), a(x_j)))\} \qquad (2.1)$$

โดยฟังก์ชัน *discerns* สามารถนิยามได้ดังนี้

$$discerns(a, a(x_i), a(x_j)) = (a(x_i) \neq a(x_j)) \qquad (2.2)$$

โดย a หมายถึงค่าคุณลักษณะ (Attributes) และ $a(x_i)$ หมายถึงเซตของค่าคุณลักษณะของวัตถุ x_i จะได้ว่าแนวคิดของความสัมพันธ์แบบไม่สามารถแบ่งแยกคือการเลือกเซตของค่าคุณลักษณะ B ซึ่ง $B \subseteq A$ ที่ทำให้เกิดการแบ่งแยกเซตเอกภพสัมพัทธ์ออกเป็นเซตย่อยอย่างสมบูรณ์ โดยที่สมาชิกของแต่ละเซตย่อยนั้นจะไม่สามารถแบ่งแยกได้อีกโดยอาศัยเพียงค่าคุณลักษณะในเซต B และแต่ละเซตย่อยที่เกิดขึ้นจากการแบ่งเซต

เอกภพสัมพัทธ์จะถูกเรียกว่าเซตของกลุ่มที่เท่าเทียม (set of Equivalent Classes) ซึ่งเซตของกลุ่มที่เท่าเทียม (Equivalent Classes) จะสามารถนิยามได้ดังนี้

ให้ $\mathcal{A} = (\mathcal{U}, A)$ เป็นระบบสารสนเทศ และให้ $B \subseteq A$ จะได้ว่า สำหรับ $x \in \mathcal{U}$ ใด ๆ จะได้

$$[x]_B = \{y \in \mathcal{U} \mid (x, y) \in \mathrm{IND}_{\mathcal{A}}(B)\} \tag{2.3}$$

ซึ่งต่อไป Equivalent Classes นี้จะถูกเรียกว่า Object Classes หรือ เรียกอย่างง่าย ๆ ว่า กลุ่ม (classes) โดยที่ Equivalent Classes ที่เกิดจากการใช้ค่าคุณลักษณะ B ซึ่ง $B \subseteq A$ ในการจัดแบ่งแยกจะใช้สัญลักษณ์เป็น $E^B = U/IND_A(B)$ ในขณะที่ Equivalent Classes ของระบบสารสนเทศ หรือ $U/IND_A(A)$ จะมีสัญลักษณ์เป็น E และแต่ละ Equivalent Class สามารถเรียกได้ว่าเป็นเซตปฐมภูมิ (Elementary Set) ซึ่งจะประกอบกันเป็นกลุ่มของความรู้ขั้นพื้นฐาน (basic granule of knowledge) เกี่ยวกับเซตเอกภพสัมพัทธ์

ในบางกรณีปัญหาด้านการแบ่งกลุ่มข้อมูล (data classification) อาจจะอธิบายในรูปแบบของความสัมพันธ์แบบ indiscernible relation ตามสมการ 2.1 หรือ 2.2 ได้ไม่สะดวกนัก ดังนั้นจึงมีผู้เสนอแนวทางอื่นในการวัดความเหมือนของข้อมูลแทนสมการ 2.1 และ 2.2 โดยวิธีการที่ถูกนำเสนอขึ้นมาใหม่นั้นถูกเรียกว่า ความเหมือน (Similarity) หรือ ความสัมพันธ์ในลักษณะที่เรียกว่า Tolerant relation

หลักการสำคัญของความสัมพันธ์ในลักษณะ Tolerant นั้นคือระบบจะยังคงยอมรับวัตถุซึ่งมีค่าคุณลักษณะผิดเพี้ยนไปจากต้นแบบของความรู้ (Knowledge) ในขอบเขตที่ยอมรับได้ให้เป็นวัตถุซึ่งแสดงความรู้นั้น กล่าวโดยง่ายก็คือยอมรับวัตถุที่มีค่าคุณลักษณะแตกต่างไปจากวัตถุอื่น ๆ ในกลุ่มในขอบเขตที่ยอมรับได้ให้อยู่ในกลุ่มนั้นนั่นเอง

สาเหตุหลักของการเสนอหลักการนี้ก็เนื่องมาจากความไม่แน่นอนซึ่งอาจเกิดจากการวัด (Uncertainty in measuring) เป็นต้น ซึ่งโดยมากจะมีสาเหตุมาจากการปฏิบัติ (การวัด เป็นต้น) เป็นหลัก

ความสัมพันธ์ในลักษณะ Tolerant Relation ที่ถูกสร้างขึ้นนี้จะไม่แบ่งแยกเซตเอกภพสัมพัทธ์ U ดังเช่นความสัมพันธ์แบบกลุ่มที่เท่าเทียม (Equivalent Relation) ตามสมการ 3.3 ได้กระทำ แต่ความสัมพันธ์ในลักษณะ Tolerant relation นี้จะเป็นการแสดงเซตของวัตถุซึ่งมีความคล้ายวัตถุ x ภายในขอบเขตที่กำหนด โดยทั่วไป similarity class ของ x จะเขียนแทนด้วย R(x) ซึ่งจะประกอบไปด้วยเซตของวัตถุซึ่งมีความคล้ายคลึงกับวัตถุ x ดังสมการ

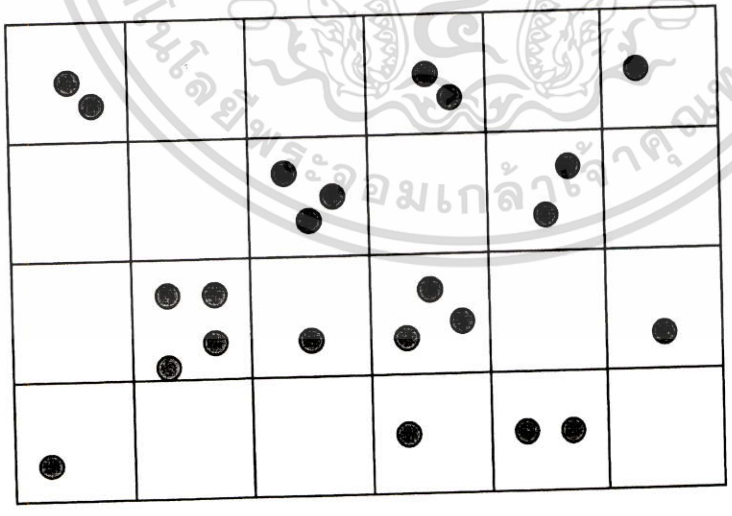
$$R(x) = \{ y \in \mathcal{U} : yRx \} \tag{2.4}$$

โดยในงานวิจัยนี้จะใช้วิธีการวัดค่าความคล้ายโดยใช้การวัดระยะห่างแบบยูคลิเดียน (Euclidean Distance) โดยที่ระยะห่างแบบยูคลิเดียน คือระยะห่างที่สั้นที่สุดระหว่างจุดสองจุดใน space โดยสามารถแสดงได้ดังต่อไปนี้

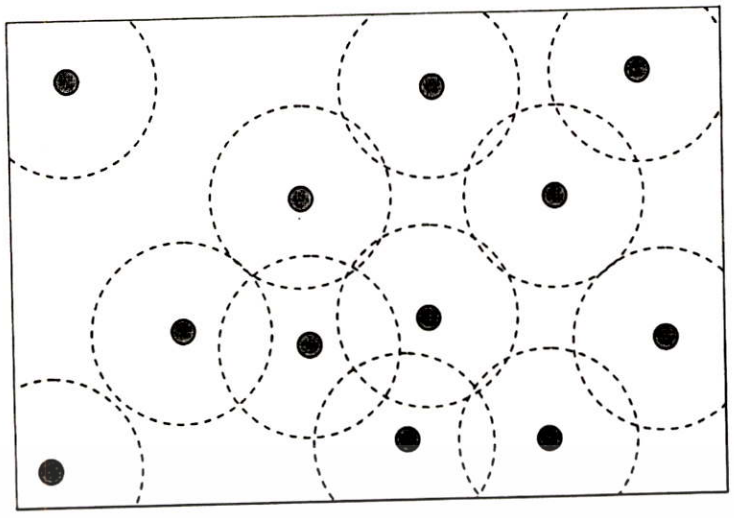
กำหนดให้ $x(x_1, x_2, x_3, \ldots, x_n)$ และ $y(y_1, y_2, y_3, \ldots, y_n)$ เป็นจุดที่พิกัด $(x_1, x_2, x_3, \ldots, x_n)$ และ $(y_1, y_2, y_3, \ldots, y_n)$ ตามลำดับ และ Euclidean distance ε จะสามารถคำนวณได้ตามสมการ (3.5)

$$\varepsilon = \sqrt{\left((x_1 - y_1)^2 + (x_2 - y_2)^2 + \ldots + (x_n - y_n)^2\right)} \qquad (2.5)$$

ความสัมพันธ์ในลักษณะ Tolerant relation นี้ มีคุณสมบัติสมมาตร (Symmetric) และคุณสมบัติการสะท้อน (reflexive) แต่จะไม่มีคุณสมบัติการส่งผ่าน (transitive) หากใช้วิธีการทั้ง 2 แบบ (Indiscernibility Relation และ Tolerant Relation) ดังกล่าวข้างต้นกับวัตถุ x จะได้เซตของวัตถุซึ่งไม่สามารถแยกแยะเป็นอย่างอื่นได้นอกจาก x โดยอาศัยค่าคุณลักษณะของวัตถุในระบบสารสนเทศเป็นหลัก เป็นผลลัพธ์เหมือนกันทั้งคู่ โดยหากใช้วิธีการของ Indiscernibility Relation จะได้กลุ่มของวัตถุซึ่งไม่สามารถแบ่งแยกได้ รวมกลุ่มกันเป็น Equivalent Classes โดยที่แต่ละกลุ่มจะไม่เกี่ยวเนื่องกัน (disjoint) และครอบคลุมทั้งเซตเอกภพสัมพัทธ์ U แต่หากเป็นวิธีการของ Tolerant Relation ผลลัพธ์ที่ได้แม้จะเป็นการสร้างกลุ่ม (class) ซึ่งครอบคลุมทั้งเซตเอกภพสัมพัทธ์เช่นเดียวกัน แต่จะแตกต่างกันตรงที่แต่ละกลุ่ม (class) นั้นจะเกี่ยวเนื่องกัน (not be disjoint) ดังเช่นที่ได้กล่าวมาแล้วว่าความสัมพันธ์ในลักษณะ Tolerant Relation นี้นั้นไม่มีคุณสมบัติการส่งผ่าน (transitive) กล่าวคือ วัตถุ x_1 อาจเหมือนกับวัตถุ x_2 และวัตถุ x_2 อาจเหมือนกับวัตถุ x_3 แต่วัตถุ x_1 อาจเหมือนหรือไม่เหมือนกับวัตถุ x_3 ก็ได้ ซึ่งนั่นหมายถึงว่ามีการซ้อนทับกันของแต่ละกลุ่ม (class) โดยความแตกต่างของ Equivalent Class ที่ถูกสร้างขึ้นโดยความสัมพันธ์ในลักษณะ tolerant relation และความสัมพันธ์แบบไม่อาจแบ่งแยก (Indiscernible Relation) นั้นจะแสดงในรูปที่ 2.1



(ก)



(ข)

รูปที่ 2.1 ความแตกต่างระหว่างกลุ่ม (classes) โดยวัตถุจะแสดงด้วยจุดสีเทาเข้ม

(ก) วิธีการแบบ Indiscernibility Relation

(ข) วิธีการแบบ Tolerant Relation

2.3 ความสามารถแยกแยะในระบบการตัดสิน

ดังที่ได้กล่าวมาแล้วข้างต้นว่ามีความเป็นไปได้ที่จะแบ่งเซตเอกภพสัมพัทธ์ของระบบการ ตัดสิน (Decision system) ออกเป็น Equivalent Class อย่างง่าย ๆ โดยการใช้เซตคุณลักษณะ A แต่อย่างไรก็ตามในหลาย ๆ กรณีจะพบว่ามีความต้องการที่จะหา Equivalent Class ซึ่งมีความเกี่ยวเนื่องกับ Decision Attributes หรืออาจกล่าวได้ว่าใช้ Decision Attributes ชักนำให้เกิดการแบ่งแยกเซต เอกภพสัมพัทธ์ออกเป็น Equivalent Classes โดย classes ที่เกิดจากกรณีนี้จะถูกเรียกว่า Decision Classes มีสัญลักษณ์เป็น X และมีสูตร หรือสมการดังนี้

$$X_i = \{x \in \mathcal{U} \mid d(x) = i\} \tag{2.6}$$

โดยทั่ว ๆ ไปเมื่อใช้ผู้เชี่ยวชาญหลาย ๆ คนในการแยกแยะวัตถุ อาจเกิดกรณีที่วัตถุที่ไม่แตกต่างกันโดยค่าคุณลักษณะถูกจัดให้อยู่คนละกลุ่ม และในทฤษฎี Rough sets จะเรียกเหตุการณ์ทำนองนี้ว่า Indeterminism หรือ Inconsistency ของ Decision System และโดยทั่วไปจะให้นิยามของ Indeterminism/Determinism ได้ดังนี้

กำหนดให้ Decision System A = (U, A, {d}) นั้น Deterministic เมื่อ

for all $E_i \in \mathcal{U}/\mathrm{IND}_{\mathcal{A}}(A)$, *there exists an* $X_j \in \mathcal{U}/\mathrm{IND}_{\mathcal{A}}(\{d\})$ *such that* $E_i \in X_j$ (2.7)

ถ้านอกเหนือไปจาก (2.7) แล้วจะกล่าวได้ว่าระบบนั้น Indeterministic ดังนั้น Decision System A = (U, A, {d}) จะ Indeterministic ถ้าปรากฏ Equivalence Class E_i ซึ่ง $E_i \in U/IND_A(A)$ ทำให้ไม่สามารถแบ่งแยกประเภทที่เฉพาะเจาะจงลงไปได้ ดังนั้นจึงมีการคิดวิธีการที่จะคำนวณหาความเป็นไปได้ในการแบ่งแยกวัตถุต่าง ๆ โดยแนวคิดนั้นก็คือแนวคิดของ generalized decision ซึ่งจะเป็นฟังก์ชันที่จะตรวจหาค่าของการตัดสิน (decision value) สำหรับกลุ่มของวัตถุ (object class) โดย generalized decision มีนิยามดังนี้

สำหรับ Decision System A = (U, A, {d}) และซับเซต B ซึ่งเป็นซับเซตของของเซตค่าคุณลักษณะ (attributes) $B \subseteq A$ จะได้

$$\delta_B(x_i) = \{v \in V_d \mid \exists x_j \in [x_i]_B \wedge d(x_j) = v\} \tag{2.8}$$

และโดยสมการ (2.8) นี้จะช่วยให้สามารถกำหนด Determinism ในเทอมของ generalized decision ได้ โดยที่ Decision System ใด ๆ จะ Deterministic ก็ต่อเมื่อ $|\delta_A(x)| = 1$ สำหรับทุกค่า x ซึ่ง $x \in U$ และมีข้อสังเกตว่า Decision System $A' = (U, A, \{\delta_A\})$ นั้น Deterministic ซึ่งนั่นจะทำให้สามารถแปลงระบบ Decision System ใด ๆ ที่เป็นระบบแบบ Indeterministic ให้เป็นระบบแบบ Deterministic ได้

2.4 กรอบการทำงานของราฟเซต

ในหัวข้อนี้จะกล่าวถึงการทำงานภายใต้กรอบของทฤษฎีราฟเซต (Rough Sets Framework) เพื่อให้สามารถจัดการกับความคลุมเครือของข้อมูล หรือ แนวคิด (vague concept)

กำหนดให้ระบบสารสนเทศ A = (U, A) ให้ $X \subseteq U$ เป็นเซตของวัตถุ $B \subseteq A$ เป็นเซตของค่าคุณลักษณะ และใช้ equivalent classes ที่เกิดขึ้นเป็น building blocks และนำมาประกอบกันเป็นเซต Y โดยที่ $Y \subseteq U$ โดยมีเป้าหมายของการประกอบเซต Y คือให้ Y = X และใช้ building blocks ซึ่งเกิดจาก equivalent classes $E^B = U/IND_A(B)$ และในกรณีที่ X ไม่สามารถกำหนดได้อย่างชัดเจนโดยใช้เซต E^B ก็จะใช้การประมาณเซต X โดยใช้เซต 2 เซต ซึ่งมีชื่อว่า Lower Approximation และ Upper Approximation ของเซต X โดยเซต Lower Approximation และ Upper Approximation นี้สามารถกำหนดได้ดังนี้

กำหนดระบบสารสนเทศ A = (U, A), ให้ $X \subseteq U$ เป็นเซตของวัตถุ และ $B \subseteq A$ เป็นเซตของค่าคุณลักษณะที่ถูกเลือก จะได้ B-Lower Approximation $\underline{B}X$ และ B-Upper Approximation $\overline{B}X$ ของ X ซึ่งเกี่ยวเนื่องกับค่าคุณลักษณะใน B ดังนี้

$$\underline{B}X = \{x \in \mathcal{U} : [x]_B \subseteq X\} \tag{2.9}$$

$$\overline{B}X = \{x \in \mathcal{U} : [x]_B \cap X \neq \varnothing\} \tag{2.10}$$

ซึ่งจะได้ว่าเซต $\underline{B}X$ นั้นเป็นเซตปฐมภูมิ (Elementary set) ทั้งหมดใน U ซึ่งจัดได้ว่าเป็นเซตปฐมภูมิ (Elementary set) ของ X ด้วย โดยการหาสมาชิกของเซตปฐมภูมิ (Elementary Set) แต่ละเซตนั้นจะใช้ค่า

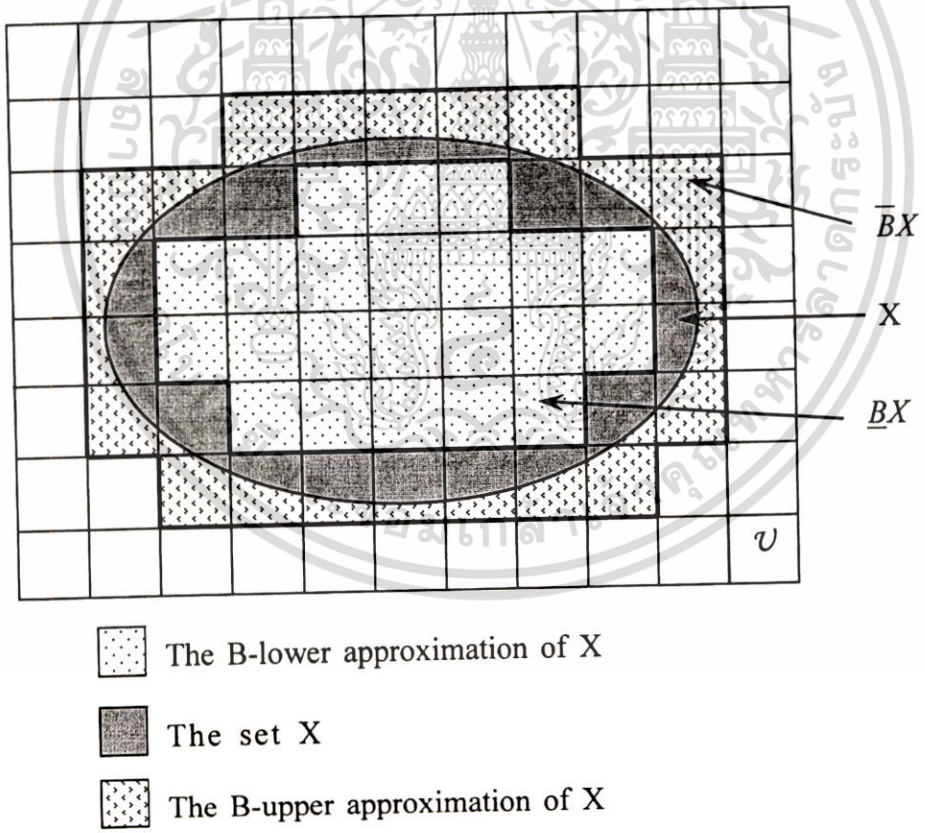
คุณลักษณะในเซต B (ซึ่งเป็นเซตของค่าคุณลักษณะ) ทั้งนี้อาจสามารถเรียกเซตนี้ได้เป็น B-positive region of X มีสัญลักษณ์เป็น $POS_B(X)$ หรืออาจกล่าวอย่างง่าย ๆ ได้ว่า "*วัตถุใดที่ถูกจัดให้อยู่ในเซต $\underline{B}X$ (โดยใช้ค่าคุณลักษณะในเซต B เป็นตัวจัด) จะสามารถบอกได้ว่าวัตถุนั้นเป็นสมาชิกเซต X ด้วยอย่างแน่นอน*"

สำหรับเซต $\overline{B}X$ นั้นจะเป็นเซตปฐมภูมิ (Elementary set) ใน U ซึ่งมีความเป็นไปได้ที่สมาชิกของเซต $\overline{B}X$ จะเป็นสมาชิกของเซต X โดยใช้ค่าคุณลักษณะใน B เป็นตัวจัดแบ่ง หรืออาจกล่าวอย่างง่าย ๆ ได้ว่า "*วัตถุใดก็ตามที่ถูกจัดให้อยู่ในเซต $\overline{B}X$ (โดยใช้ค่าคุณลักษณะในเซต B เป็นตัวจัด) จะสามารถบอกได้เพียงว่าวัตถุนั้นอาจจะเป็นสมาชิกของเซต X*"

จากที่กล่าวมาข้างต้นจะได้ว่า

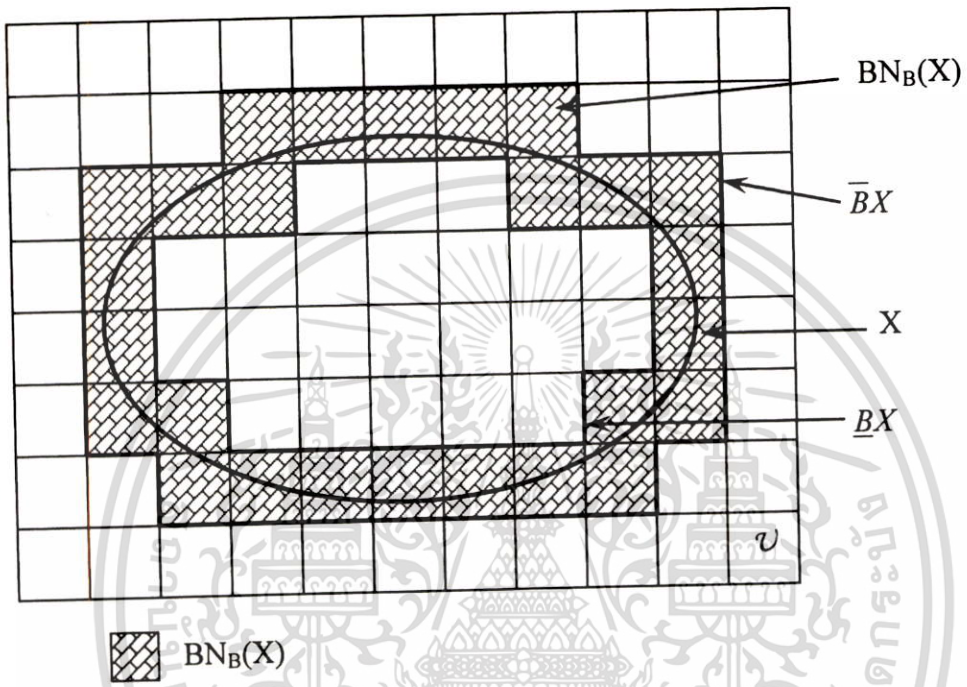
$$BN_B(X) = \overline{B}X - \underline{B}X \tag{2.11}$$

โดยที่บริเวณ $BN_B(X)$ นี้จะถูกเรียกว่า B-boundary of X โดยเซต $BN_B(X)$ นี้จะประกอบไปด้วยเซตปฐมภูมิ (elementary sets) ซึ่งไม่สามารถจะจัดให้อยู่ใน X หรือ อยู่นอก X ได้



รูปที่ 2.2 The B-upper and B-lower approximation of X

ดังนั้นจะได้ว่า Rough set คือ เซต $X \subseteq U$ ซึ่งถูกกำหนดโดย upper approximation และ lower approximation ของมัน ในทางกลับกัน crisp set (หรือก็คือ "เซต" โดยทั่วไปที่เป็นที่รู้จัก) ก็คือเซตซึ่งสามารถนิยามได้โดยใช้ equivalent classes และสามารถถ้านิยามโดยใช้ทฤษฎี Rough set ก็จะได้ว่า crisp set คือเซตซึ่ง upper approximation set มีค่าเท่ากับ lower approximation set ($\overline{B}X = \underline{B}X$)

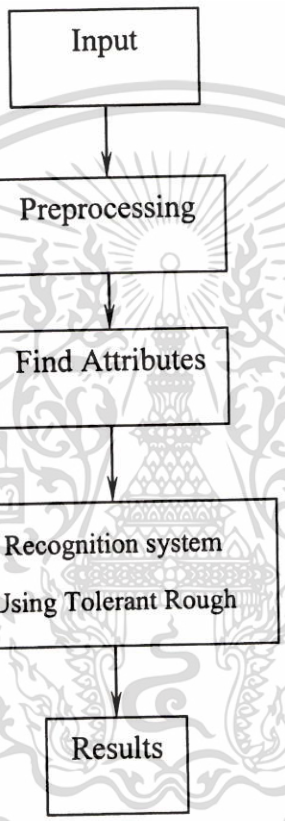


รูปที่ 2.3 แสดงพื้นที่ $BN_B(X)$

บทที่ 3

กระบวนการรู้จำตัวอักษรตัวพิมพ์ภาษาไทยโดยใช้ทฤษฎีราฟเซต

ในบทนี้จะนำหลักการที่ได้กล่าวไว้ในบทต้น ๆ มาประยุกต์ใช้ในการรู้จำตัวอักษรตัวพิมพ์ภาษาไทย โดยใช้ทฤษฎี Tolerant Rough Sets ระบบจะแบ่งออกเป็นสองส่วนหลัก ๆ คือ ส่วนการหาค่าคุณลักษณะเด่นของตัวอักษร และส่วนของการฝึกสอนระบบ โดยภาพรวมของระบบที่ใช้ในงานวิจัยนี้สามารถแสดงได้ดังนี้

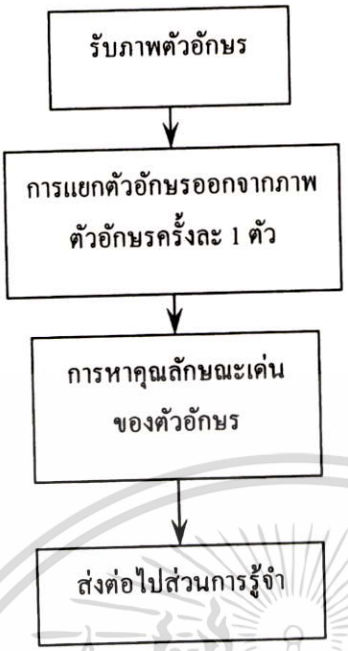


รูปที่ 3.1 ภาพรวมของระบบ

3.1 กระบวนการรับอินพุทภาพตัวอักษร

หน้าที่สำคัญสำหรับสำหรับกระบวนการนี้ก็คือการทำ pre-processing ต่าง ๆ ซึ่งได้แก่การตัดเอาตัวอักษรออกมาจากภาพของตัวอักษรทีละตัวแล้วนำมาหาลักษณะเด่น (Features) ต่าง ๆ เพื่อนำไปเป็นส่วนที่เป็นค่าคุณลักษณะของวัตถุ (Attributes) ของระบบสารสนเทศของ Rough Sets

สำหรับวิธีการที่นำมาใช้ในส่วนที่หนึ่งของงานวิจัยนี้นั้นจะใช้วิธีการประยุกต์เอาวิธีการหาลักษณะเด่นโดยใช้วิธีการแบ่งตัวอักษรออกเป็นส่วน ๆ แล้วทำการนับจุดภาพสีดำในแต่ละส่วนนั้น กระบวนการที่เกิดขึ้นในส่วนที่หนึ่งของระบบจะสามารถแสดงเป็นภาพโดยรวมได้ดังต่อไปนี้

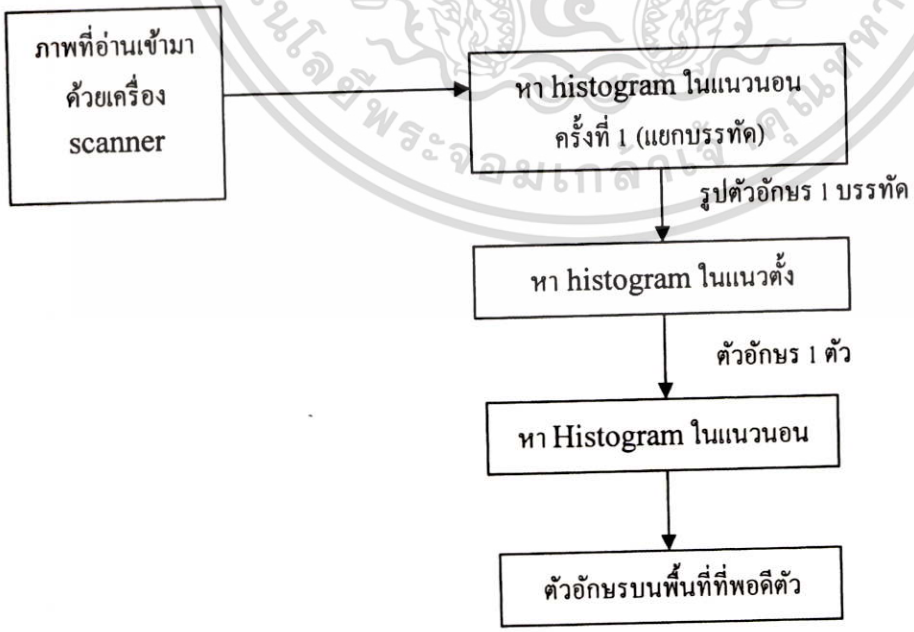


รูปที่ 3.2 ภาพรวม (overview) ของกระบวนการรับข้อมูลภาพตัวอักษร

โดยแต่ละส่วนของกระบวนการในส่วนที่หนึ่งนี้สามารถอธิบายโดยละเอียดได้ดังนี้

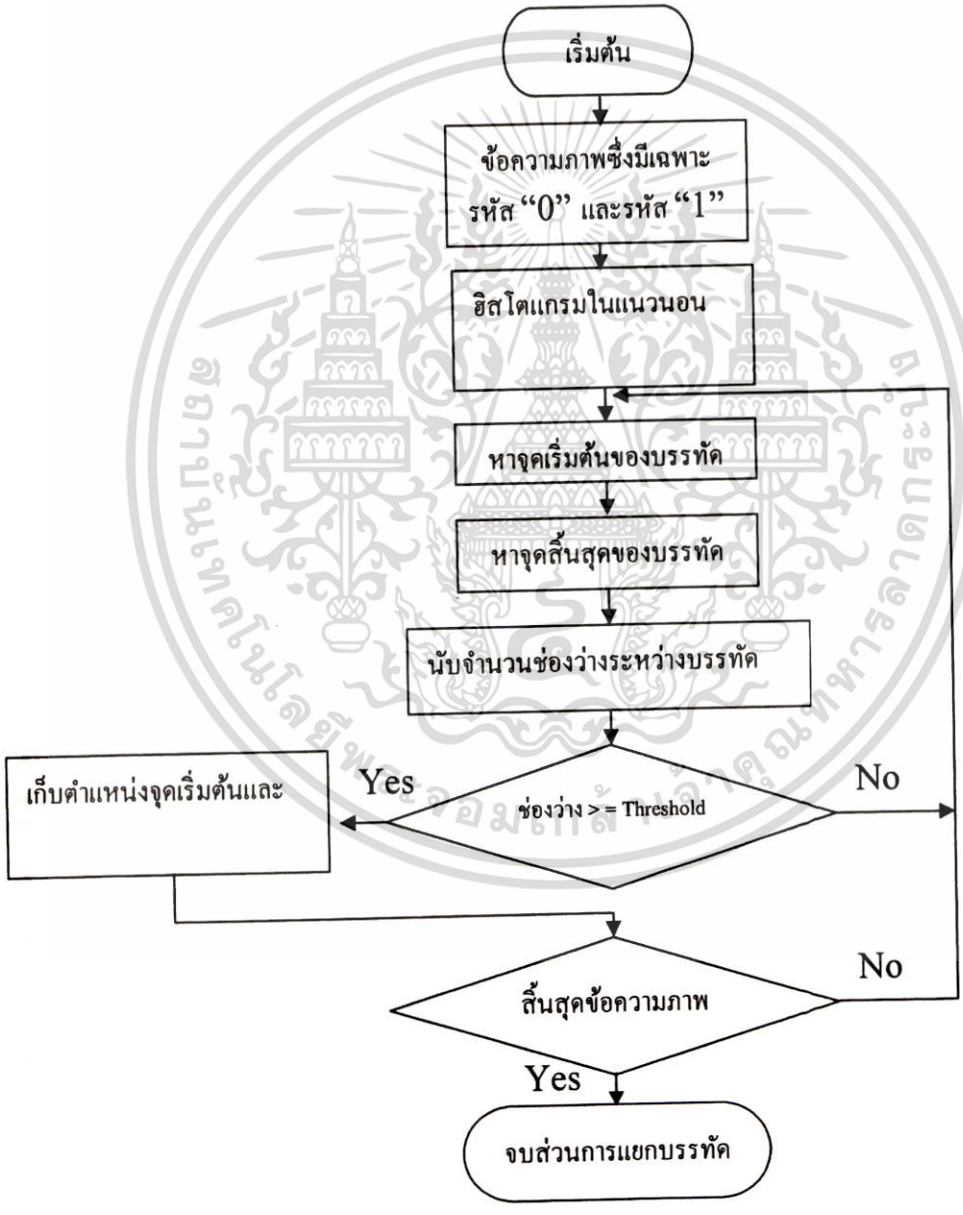
3.1.1 การแยกตัวอักษรออกจากภาพทีละตัวอักษร

ขั้นตอนนี้มีไว้เพื่อทำการแยกภาพตัวอักษรจากที่อ่านเข้ามาซึ่งจะเป็นภาพที่มีตัวอักษรหลายตัวอยู่รวมกันออกมาเป็นภาพตัวอักษรตัวเดียว โดยจะทำการแยกออกมาครั้งละ 1 ตัวอักษร เพื่อจะนำมาใช้ในการหาค่าคุณลักษณะของตัวอักษรตัวนั้นต่อไป

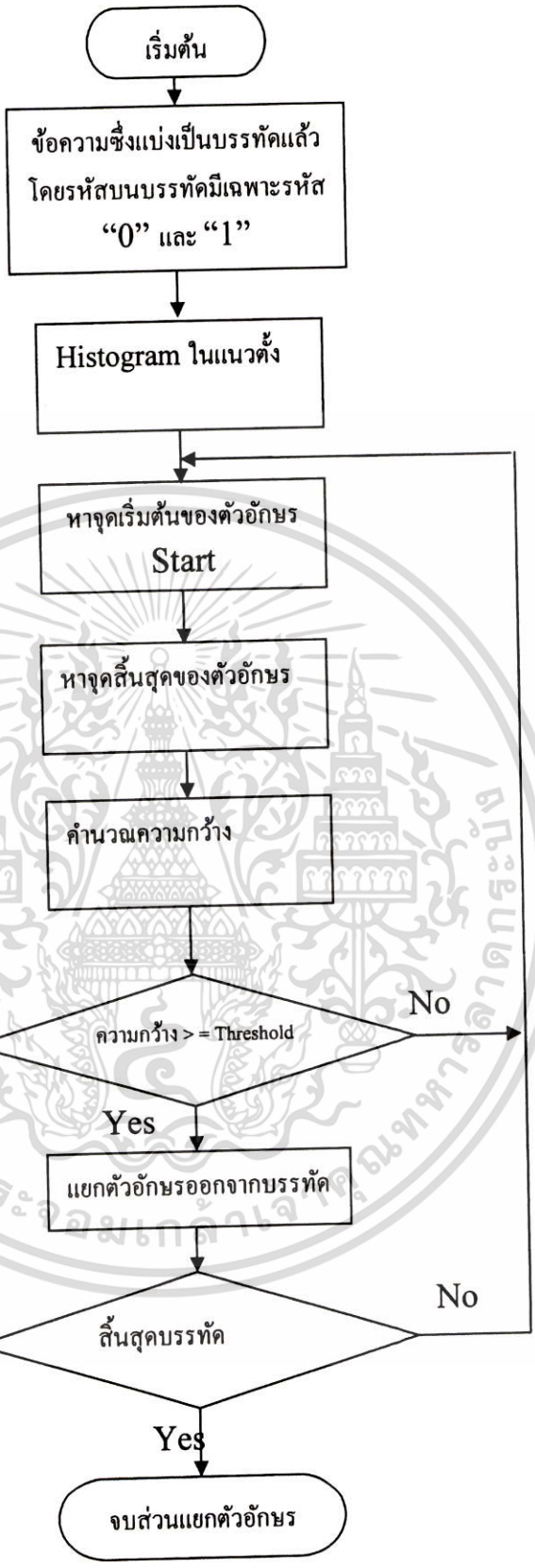


รูปที่ 3.3 ภาพรวม (overview) ของกระบวนการการแยกตัวอักษรออกจากภาพ

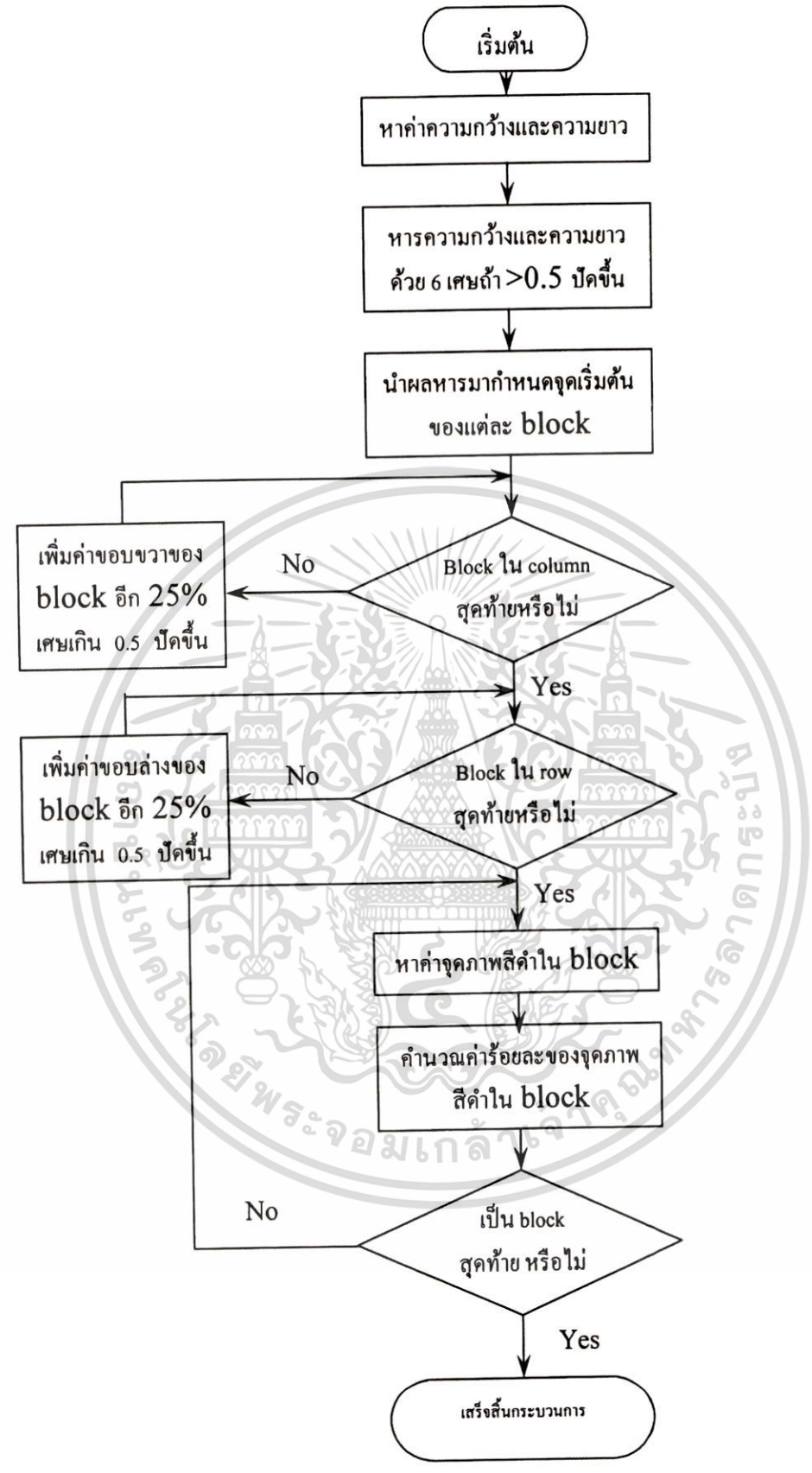
ในงานวิจัยนี้จะทำการแยกตัวอักษรออกเป็นตัว ๆ โดยวิธีการหา Histogram โดยการหา histogram จะทำการหาในสองแนวคือหา histogram ในแนวนอนเพื่อแยกตัวอักษรออกเป็นบรรทัด และเมื่อได้ตัวอักษรในหนึ่งบรรทัดแล้วก็จะนำเอาบรรทัดดังกล่าวมาหา Histogram ในแนวตั้งเพื่อทำการแยกตัวอักษรออกเป็นตัว ๆ และจะทำการหา histogram ในแนวนอนกับภาพตัวอักษรที่แยกได้นั้นอีกครั้งหนึ่งเพื่อตัดเอาเฉพาะส่วนที่เป็นตัวอักษรเท่านั้น ทั้งนี้เนื่องจากในหนึ่งบรรทัดอาจประกอบไปด้วยตัวอักษรที่มีขนาดไม่เท่ากัน กระบวนการต่าง ๆ ดังที่กล่าวมานี้จะแสดงในรูปที่ 3.3 และ flowchart ของการแยกบรรทัดและแยกออกเป็นอักษรตัวเดียวโดยวิธีการ histogram จะแสดงไว้ในรูปที่ 3.4 และ 3.5 และในงานวิจัยนี้จะจำกัดขอบเขตของภาพตัวอักษรที่นำมาทดลองให้เป็นภาพตัวอักษร ก-ฮ โดยที่ตัวอักษรนั้นไม่ตัดกัน และไม่ซ้อนทับกัน



รูปที่ 3.4 Flowchart การตัดแบ่งภาพตัวอักษรออกเป็นบรรทัดโดยวิธีการ histogram



รูปที่ 3.5 Flowchart การตัดแบ่งภาพตัวอักษรออกจากบรรทัดโดยวิธีการ histogram



รูปที่ 3.6 Flowchart การทำงานของการหาค่าลักษณะเด่น

3.1.2 การหาค่าลักษณะเด่นของตัวอักษร

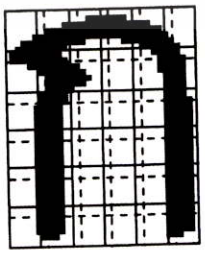
การหาค่าลักษณะเด่นของตัวอักษร (Feature extraction) ในงานวิจัยนี้จะใช้วิธีการแบ่งภาพออกเป็น 36 ส่วน โดยแบ่งทางแนวนอน 6 ส่วน และแบ่งทางแนวตั้ง 6 ส่วน และแต่ละส่วนจะซ้อนทับกัน 25% ทางขอบด้านขวาและขอบด้านบน แล้วนับจำนวนของจุดภาพสีดำที่อยู่ภายในแต่ละส่วนของภาพตัวอักษรนั้น แล้วทำการ normalization ค่าที่หามาได้โดยการคำนวณจำนวนจุดภาพที่เป็นสีดำออกมาเป็นค่าร้อยละของจำนวนจุดภาพสีดำของพื้นที่แต่ละส่วน ซึ่งจะได้ค่าออกมาทั้งหมดเป็น 36 ค่าต่อหนึ่งตัวอักษร รายละเอียดของวิธีการหาคุณลักษณะเด่น ซึ่งใช้ในงานวิจัยนี้มีขั้นตอนดังนี้ (Flowchart ของการหาค่าลักษณะเด่นจะเป็นดังรูปที่ 3.6)

1. ทำการแบ่งตัวอักษรตามแนวตั้งออกเป็น 6 ส่วน และตามแนวนอน 6 ส่วน

2. ทำการขยับขอบทางด้านขวาของแต่ละส่วนให้ซ้อนทับกับขอบทางด้านซ้ายของส่วนที่อยู่ชิดติดกัน และขยับขอบทางด้านล่างของแต่ละส่วนให้ซ้อนทับกับขอบทางด้านบนของอีกส่วนที่อยู่ติดกันทางด้านล่าง โดยระยะของการซ้อนทับกันจะมีความยาวประมาณ 25% ของด้านยาวและด้านกว้างตามลำดับ สาเหตุที่ต้องให้แต่ละส่วนต้องมีการซ้อนทับกันนั้นก็เพื่อแก้ไขปัญหาการเลื่อนซ้าย-ขวา (Left-right shift) และการเลื่อนบน-ล่าง (Upper-lower shift)

3. นับจำนวนจุดของแต่ละส่วนภาพของตัวอักษร แล้วจะได้ข้อมูลซึ่งเป็นลักษณะเด่น 1 ชุด โดยใน 1 ชุดนี้จะประกอบไปด้วยตัวเลข 36 จำนวน จากชุดข้อมูลลักษณะเด่นของตัวอักษรที่ได้จะพบว่าขนาดของตัวอักษรมีผลต่อค่าลักษณะเด่นดังกล่าว แม้ว่าจะเป็นตัวอักษรชนิดเดียวกัน แต่หากมีขนาดต่างกันแล้วก็จะได้ค่าคุณลักษณะที่ต่างกันมาก ดังนั้นเพื่อให้สามารถใช้กับข้อมูลตัวอักษรที่มีขนาดต่าง ๆ กัน จึงต้องมีกระบวนการปรับให้เป็นกลาง (Normalization) ซึ่งในงานวิจัยนี้จะใช้วิธีการแสดงเป็นจำนวนเปอร์เซ็นต์ของจุดภาพสีดำในส่วนของภาพนั้น ๆ

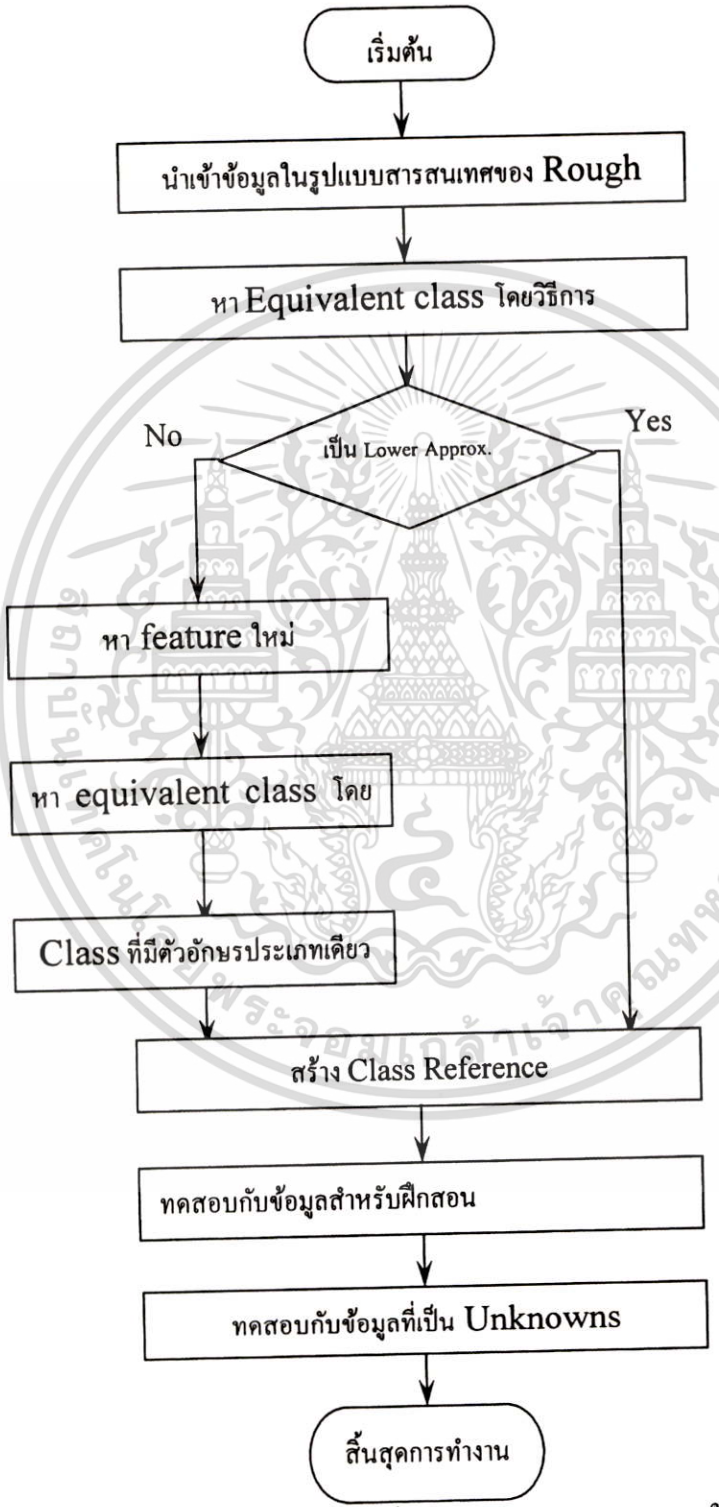
เมื่อได้ค่าต่าง ๆ จากกระบวนการที่แสดงในรูปที่ 3.6 แล้วก็จะกำหนดค่าต่าง ๆ ให้อยู่ในรูปแบบของระบบสารสนเทศใน Rough Sets โดยจะให้ค่า 36 ค่าที่ได้มาเป็นค่าคุณลักษณะ (attributes) ของวัตถุซึ่งในที่นี้ก็คือตัวอักษรซึ่งเป็นเจ้าของค่าทั้ง 36 ค่านั้น โดยจะกำหนดให้เป็นค่าคุณลักษณะ A1 ถึง A36 และกำหนดค่าของ Decision Attribute D ให้กับวัตถุ



รูปที่ 3.7 การซ้อนทับกันของแต่ละส่วน โดยเส้นประแสดงบริเวณที่เกิดการซ้อนทับกัน

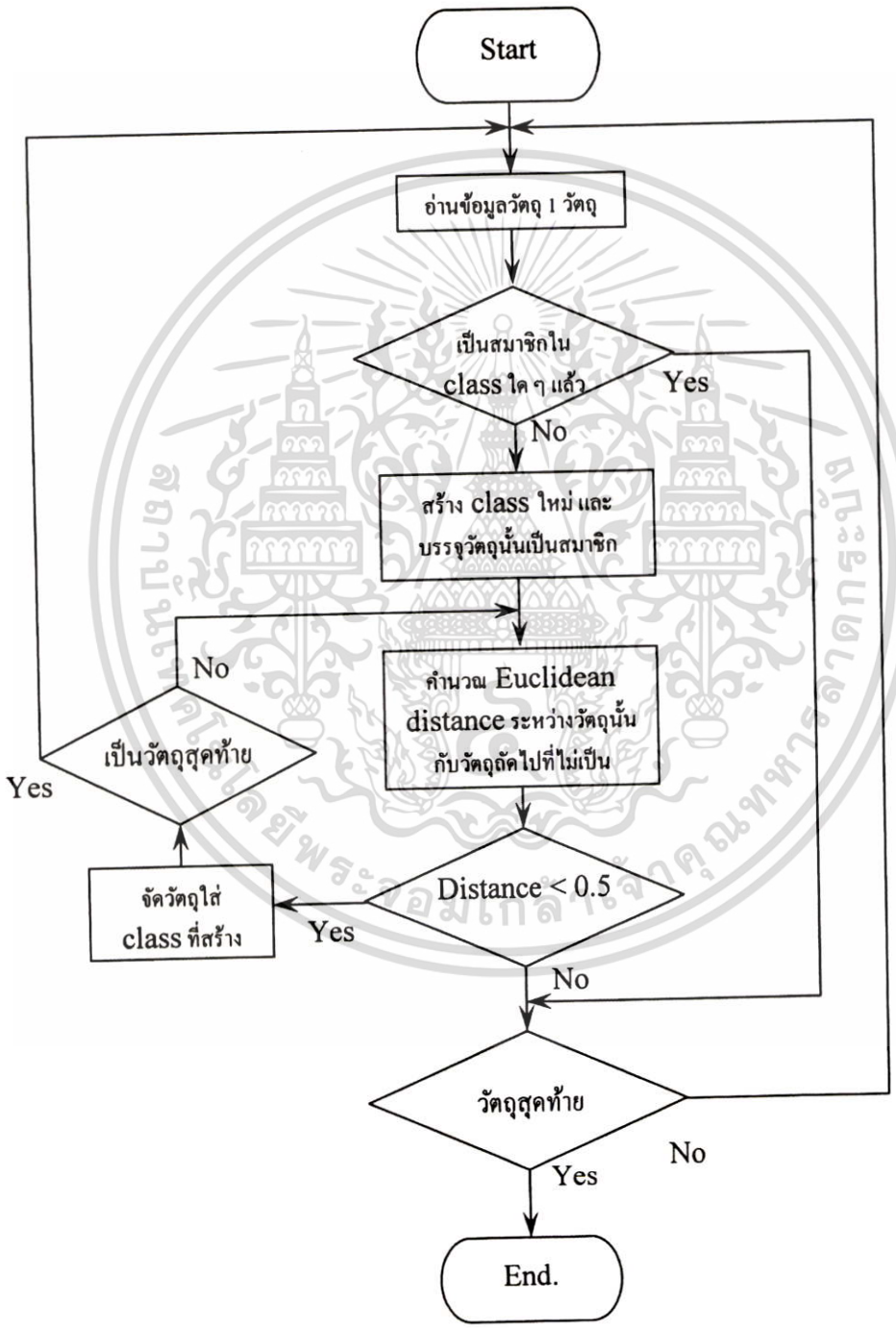
3.2 ระบบของการรู้จำ

ระบบในส่วนนี้คือส่วนที่ทำหน้าที่ในการรู้จำซึ่งในงานวิจัยนี้ได้สร้างระบบในส่วนนี้โดยการประยุกต์ใช้ทฤษฎี Tolerant Rough Sets โดยขั้นตอนการทำงานของระบบสามารถแสดงได้ในรูปแบบของ block diagram ได้ดังรูปที่ 3.8



รูปที่ 3.8 Block diagram ของการประยุกต์ใช้งาน Tolerant Rough Sets ในงานวิจัย

สำหรับการหา equivalent class สำหรับ Tolerant Rough Set ในงานวิจัยนี้นั้นจะใช้วิธีหาความเหมือนของตัวอักษรที่เป็นตัวแทนของกลุ่มกับตัวอักษรที่จะนำมาเรียนรู้ (Training Set) โดยในที่นี้จะนำวิธีการวัดระยะห่างแบบยูคลิเดียน (Euclidean distance) มาประยุกต์ใช้งาน โดยการคำนวณหาระยะห่างแบบ Euclidean นี้จะใช้สมการ (2.5) และขั้นตอนการทำงานของการหา equivalent class ด้วยวิธีการวัดระยะห่างแบบ Euclidean นี้จะสามารถแสดงในรูปแบบ flow chart ได้ดังในรูปที่ 3.9



รูปที่ 3.9 ขั้นตอนการหา equivalent class

สำหรับกลุ่มซึ่งเป็นชนิด Upper Approximation นั้นจะถูกนำมาแบ่งกลุ่มใหม่อีกครั้งโดยใช้ค่าคุณลักษณะชุดใหม่ ซึ่งในการหาคุณลักษณะของตัวอักษรในครั้งนี้จะกระทำกับเฉพาะบางส่วนของตัวอักษรซึ่งเป็นส่วนที่แตกต่างกันของตัวอักษรในกลุ่มนั้น ๆ

โดยขั้นตอนในการแบ่งกลุ่มนั้นจะใช้วิธีการเช่นเดียวกับการแบ่งกลุ่มในครั้งแรก กล่าวคือใช้วิธีการตามใน Flow Chart ในรูปที่ 3.9 และเนื่องจากส่วนของตัวอักษรที่นำมาใช้หาค่าคุณลักษณะในการแบ่งกลุ่มขั้นตอนนี้นั้นแม้จะเป็นส่วนที่แตกต่างกันที่สุดแต่ในภาพตัวอักษรที่มีขนาดเล็กก็ทำให้มีความแตกต่างนั้นลดน้อยลงไป ดังนั้นในงานวิจัยนี้จึงทำการลดค่า distance ลงให้เหลือ 0.2 เพื่อเพิ่มประสิทธิภาพในการรู้จำให้มากยิ่งขึ้น

บทที่ 4

ผลการทดลอง

4.1 ข้อมูลที่ใช้ในการทดสอบ

ข้อมูลที่นำมาทดสอบในงานวิจัยนี้จะเป็นภาพตัวอักษร ก-ฮ โดยพิมพ์และสแกนด้วยเครื่อง scaner โดยตัวอักษรที่ใช้ประกอบไปด้วย 4 ฟอนต์คือ AngsanaUPC, BrowalliaUPC, CordiaUPC, Freesia UPC และแต่ละฟอนต์จะทำการพิมพ์ 4 ขนาด คือขนาด 14, 16, 18, 22 พอยต์ ข้อมูลที่นำมาใช้ในงานวิจัยนี้จะแบ่งออกเป็น 2 ชุด คือ ชุดข้อมูลสำหรับนำไปเรียนรู้ (Training Sets) และชุดข้อมูลที่เป็นตัวไม่ทราบค่า (Unknown Set) สำหรับนำไปทดสอบระบบ โดยข้อมูลสำหรับนำไปเรียนรู้นั้นจะประกอบด้วยฟอนต์ 4 ฟอนต์ 4 ขนาด ดังที่ได้กล่าวมาแล้ว โดยตัวอักษร 1 ชนิด (เช่น ก) จะจัดพิมพ์ทั้งหมด 48 ตัว – 12 ตัวต่อ 1 ฟอนต์ – 3 ตัวต่อ 1 ขนาด รวมทั้งสิ้น 2112 ตัว

สำหรับชุดข้อมูลที่เป็นตัวไม่ทราบค่านั้นจะประกอบไปด้วยฟอนต์ 4 ฟอนต์ 4 ขนาดด้วยเช่นเดียวกัน หากต่างกันที่ตัวอักษร 1 ชนิด (เช่น ก) จะจัดพิมพ์ทั้งหมด 160 ตัว – 40 ตัวต่อ 1 ฟอนต์ – 10 ตัวต่อ 1 ขนาด รวมทั้งสิ้น 7040 ตัว

4.2 ผลการทดลอง

ผลของการทดลองโดยใช้การประยุกต์ Tolerant Rough Sets จะได้ตัวอักษรซึ่ง Equivalent classes ของตัวอักษรนั้นเป็น Equivalent class ที่มีสมาชิกเป็นตัวอักษรประเภทเดียว (Lower approximation set class) ทั้งหมด 33 ตัวได้แก่ ก ค ฅ ฆ ง จ ฉ ฐ ฑ ฒ ถ ท ธ น บ ป ผ ฝ พ ฟ ภ ม ย ร ล ว ศ ษ ส ห อ ฮ ส่วนตัวอักษรที่เหลือคือ ข ฃ ช ซ ฌ ญ ฎ ฏ ณ ด ต นั้นจะมีเพียงบางกลุ่มเป็นกลุ่มที่มีสมาชิกชนิดเดียว (Lower Approximation sets) ส่วนกลุ่มที่เหลือจะมีสมาชิกมากกว่า 1 ชนิด (Upper Approximation sets) ดังแสดงในตาราง 4.1

จากผลในตาราง 4.1 สามารถสรุปออกมาเป็นเปอร์เซ็นต์โดยรวมได้ดังนี้ ระบบการรู้จำที่ใช้ Tolerant Rough Sets เป็นตัวรู้จำนั้นจะให้ค่าเปอร์เซ็นต์ความถูกต้อง 99.01%

ตาราง 4.1 ลักษณะกลุ่มที่แบ่งได้และผลการรู้จำสำหรับระบบการรู้จำโดยใช้ Tolerant Rough Sets

ตัวอักษร	จำนวน	ประเภทของ Class		ผลการรู้จำ		รู้จำผิดพลาดเป็น
	class	Lower	Upper	ถูก	ผิด	
ก	19	19	-	160	-	-
ข	17	17	-	159	1	-
ฃ	17	17	-	159	1	-
ค	17	17	-	160	-	-
ฅ	24	24	-	160	-	-
ฆ	15	15	-	160	-	-
ง	20	20	-	160	-	-
จ	17	17	-	160	-	-
ฉ	17	17	-	160	-	-
ช	19	19	-	158	2	ซ
ซ	18	18	-	160	-	-
ฌ	18	18	-	155	5	ฎ, ณ
ญ	15	15	-	160	-	-
ฎ	17	9	8	141	19	ฏ
ฏ	16	11	5	148	12	ฎ
ฐ	21	21	-	160	-	-
ฑ	14	14	-	160	-	-
ฒ	7	7	-	160	-	-
ณ	18	18	-	159	1	ญ
ด	19	13	6	159	1	ต
ต	19	13	6	159	1	ด
ถ	19	19	-	160	-	-
ท	17	17	-	160	-	-
ธ	20	20	-	160	-	-
น	11	11	-	160	-	-
บ	14	14	-	160	-	-
ป	11	11	-	160	-	-
ผ	23	23	-	160	-	-

ตาราง 4.1 (ต่อ)

ตัวอักษร	จำนวน class	ประเภทของ Class		ผลการรู้จำ		รู้จำผิดพลาดเป็น
		Lower	Upper	ถูก	ผิด	
ฝ	15	15	-	160	-	-
พ	12	12	-	160	-	-
ฟ	10	10	-	160	-	-
ภ	12	12	-	160	-	-
ม	16	16	-	160	-	-
ย	24	24	-	160	-	-
ร	20	20	-	160	-	-
ล	24	24	-	160	-	-
ง	15	15	-	160	-	-
ศ	19	19	-	160	-	-
ษ	15	15	-	160	-	-
ส	19	19	-	160	-	-
ห	16	16	-	160	-	-
ฬ	16	16	-	160	-	-
อ	26	26	-	160	-	-
ฮ	23	23	-	160	-	-

บทที่ 5
บทสรุป

5.1 สรุปผลการทดลอง

จากการทดลองจะพบว่า Tolerant Rough Sets สามารถนำมาใช้ในงานการรู้จำตัวอักษรภาษาไทยได้เป็นอย่างดี โดยให้ความแม่นยำในการรู้จำสูงถึง 99.01%

5.2 ปัญหาและอุปสรรค

สำหรับปัญหาในส่วนของงานวิจัยนั้นจะเป็นที่ตัวอักษรในภาษาไทยบางชนิดมีความคล้ายกันมาก โดยเฉพาะอย่างยิ่งในบางฟอนต์เมื่อผ่านขั้นตอนการพิมพ์และสแกนแล้วยิ่งมีความแตกต่างของตัวอักษรน้อยลงไปอีก ตัวอย่างเช่น ฎ และ ฏ มีลักษณะที่ใกล้เคียงกันมาก ทำให้ค่าคุณลักษณะที่ได้มีการซ้อนทับกันมากทำให้ยากต่อการแบ่งแยก นอกจากนั้นคุณภาพของการพิมพ์รวมทั้งขนาดของตัวอักษรก็ยังส่งผลอย่างมากต่อผลการรู้จำ เนื่องจากการพิมพ์ที่คุณภาพไม่ดีนักเช่นลายเส้นไม่คม หรือเส้นขาด ตัวอักษรขนาดเล็กก็จะทำให้ลายละเอียดบางอย่างของตัวอักษรนั้นเช่น รอยหยักของหัวตัวอักษร หายไปได้หรือหยักน้อยลงจนการค่าลักษณะเด่นด้วยวิธีการที่ในวิทยานิพนธ์ฉบับนี้ให้ค่าออกมาเหมือนไม่มีรอยหยักเป็นต้น

นอกจากนั้นปัญหาในการ scan ที่เกิดจากการวางกระดาษไม่ตรง ทำให้ตัวอักษรเกิดการเอียง ก็มีผลต่อการรู้จำอย่างมาก

ดังนั้นการพัฒนาระบบการทำ preprocessing และการพัฒนาวิธีการหา Feature (Attributes) ของตัวอักษรจะมีส่วนช่วยอย่างมากต่อความสามารถในการรู้จำ

เอกสารอ้างอิง

1. S. K. Pal, A. Skowron, "Rough Fuzzy Hybridization: A New Trend In Decision-Making", Springer, 1999
2. A. Ohrn, "Discernibility and Rough Sets in medicine: Tools and Application," Ph.D. Thesis, Norwegian University of Science and Technology, Department of Computer and Information science, 2000.
3. Z. Pawlak, "Vagueness and uncertainty: a Rough Sets perspective," Computational Intelligence, vol. 11 No. 2, P. 227-232, 1995.
4. Z. Pawlak, "Rough Sets," Rough Sets and data mining: analysis of imprecise data, P. 3-8, 1997.
5. Y. Y. Yao, S. K. Wong and T. Y. Lin, "A Review of Rough Sets models," Rough Sets and data mining: analysis of imprecise data, P. 47-76, 1997.
6. X. Hu, "Knowledge discovery in databases: An attributes-oriented Rough Sets approach," Ph.D. Thesis, University of Regina, 1995.
7. D. Nejman, "Rough sets in handwritten numerals recognition," ICA WUT reports on Rough Sets, 1995.
8. Daijin Kim, Sung-Yang Bang, "A Handwritten Numeral Character Classification Using Tolerant Rough Set", IEEE Trans. On Pattern Analysis and Machine Intelligence, Vol. 22, No. 9, P. 923-937, Sep. 2000.
9. C. Kimpan, A.Itoh, and K. Kawanishi, "Recogniton of printed Thai character using a matching method", IEE Proc., Vol. 130, Pt.E, No. 6, P. 183-188, Nov. 1983.
10. C. Kimpan, "Printed Thai character recognition using topological properties method", INT.J. Electronics, Vol. 60, No. 3, P. 303-329, 1986.
11. C. Kimpan, A.Itoh, and K. Kawanishi, "Fine classification of printed Thai character recognition using the Karhunen-Loeve expansion", Proc. IEE, Vol. 134, Pt.E., No. 5, P. 25764, Sept, 1987
12. C. Kimpan. 1986. "Printed Thai Character Recognition Dissertation of Engineering", Faculty of Engineering, King Mongkut's Institute of Technology Chaokhurn Taharn Ladkrabang.
13. W. Kasemsiri and C. Kimpan, "Printed Thai characters recognition using 2 class levels classification and rough sets", APSBC, 2000
14. S. Mitatha, K. Dejhan, F. Cheevasuvit, B. Chankuang and W. Kasemsiri, "Experimental results of using Rough Sets for printed Thai characters recognition," Proc. IEEE TENCON, 2001.
15. Watjanapong Kasemsiri and Chom Kimpan, "Printed Thai characters recognition using Fuzzy-Rough Sets", Proc. IEEE TENCON, 2001.
16. สุรพันธุ์ เอื้อไพบูลย์ "การเตรียมข้อมูลสำหรับการจำแนกรูปแบบตัวอักษรภาษาไทย", บทความทางวิชาการ Seminar MI, คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าลาดกระบัง กรุงเทพ, พฤศจิกายน 2529. (หน้า 13)
17. สกุล คำนวนชัย, "การรู้จำตัวอักษรคัดลายมือเขียนภาษาไทยประยุกต์ใช้การทรานสฟอร์ม แบบคาร์ฮูเน็นเลิฟเข้ากับโครงข่ายฟัซซี่นิวรอล", วิทยานิพนธ์วิศวกรรมศาสตร์มหาบัณฑิต, สาขาวิศวกรรมไฟฟ้าบัณฑิตวิทยาลัย, สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง,2542.

18. อุกฤษณ์ มารังค์, "การรู้จำอักษรตัวพิมพ์ภาษาไทยโดยใช้เจนเนติก-นิวรอลเน็ตเวิร์ค", วิทยา นิพนธ์วิศวกรรมศาสตร์มหาบัณฑิต, สาขาวิศวกรรมไฟฟ้าบัณฑิตวิทยาลัย, สถาบัน เทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง,2544.

19. P. HIRANVANICHAKORN, T. AGUI, and M. NAKAJIMA, "Recognition Method of Thai Characters", The Transactions of the IECE of Japan, vol. E65, No. 12, December 1982. (P 16)

20. P. HIRANVANICHAKORN, T. AGUI, and M. NAKAJIMA," A Recognition Method of Thai Characters by using local features", The transactions of the IECE of Japan, Vol. E67, No. 8, August 1984. (P 16)

21. http://www.nist.gov/dads/HTML/euclidndstnc.html

22. http://www.wikipedia.org/w/wiki.phtml?title=Euclidean_distance

ภาคผนวก ก.

บทความวิชาการที่ได้รับการตีพิมพ์

I-SEEC 2008

International Workshop and Symposium Science Technology 2008

Accepted Letter

Prof. Dr. Somsak Mitatha

Computer Engineering Department, Faculty of Engineering,
King Mongkut's Institute of Technology Ladkrabang, Bangkok, Thailand.
Telephone: 081-1746469E-mail: kmsomsak@kmitl.ac.th,kmsomsak@hotmail.com

Date : 18.11.2008
Receipt date : 05.11.2008
Reference : ONT002

Prof. Dr. Somsak Mitatha

Title of Paper: Printed Thai Character Recognition using Tolerant Rough Sets

Author(s) : W. Kasemsiri and S. Mitatha

Your paper has been sent to reviewer as reader and if your work has something wrong, please edit your work follow suggestion of reader. We are sincerely with thanks if you please submission and registration fee within 30 November 2008(USD 70). If you register late on 1-15 December 2008, it means that you register in Late Registration Form (USD 80). Please submit your full paper on the registration desk.

With many thanks and best regards

Yours sincerely
Nithiroth Pornsuwancharoen

Dr. Nithiroth Pornsuwancharoen
Editor I-SEEC 2008,
Department of Electrical Engineering Faculty of Industry and Technology
Rajamangala University of Technology Isan, Sakon Nakhon 47160,Thailand
Tel.:(+66)86 300 8617 (+66)42 772 391 Fax:(+66)42 772 392
E-mail:jeewuttinun@gmail.com http://www.i-seec2008.com

PRINTED THAI CHARACTER RECOGNITION USING TOLERANT ROUGH SETS

Watjanapong Kasemsiri[1] and Somsak Mitatha[2]

Computer Engineering Department, Faculty of Engineering,

King Mongkut's Institute of Technology Ladkrabang, Bangkok, Thailand.

ABSTRACT

This paper proposed the method of using the Tolerant Rough Sets for recognition of Printed Thai characters. In our work the inputted images are segmented into 36 pieces and then find the percentage of black pixels in each section. Following this, we use the resulting 36 values as the attributes for each given object. Then we find the center of each class by averaging the attributes of the characters of the same type. The inputted data that have less distance to the class center than threshold value will be the member of that class. The classes which have only one type of character as its member are considered Lower Approximation sets, the others classes which its members consist of many character types are Upper Approximation sets. The samples used in this work are 7040 characters and this proposed system's accuracy is 92.79%.

KEYWORDS: Character Recognition, Rough Sets, Tolerant Rough Sets.

1. INTRODUCTION

The equivalent classes in classical Rough Sets represent the "Equivalence" relation between objects which means objects in equivalent classes must have the same interested attributed. But in real world objects of the same type may not have exactly same attributes. The attempt to handle this problem is to use similarity relation instead of equivalence relation.

In this paper, we apply the tolerant Rough Sets to the application of printed Thai characters recognition. We measure the similarity between a class representation and inputted data by using Euclidean distant.

2. CLASSICAL ROUGH SETS

In this section we will introduce the principle notions of Rough Sets Theory.

2.1 Information System

By an information system S, we mean that S = {U, A, V}, where U is a finite set of object, U = {x_1, x_2, x_3, ..., x_n}, A is a finite set of attributes which further classified into disjointed condition attributes C and decision attributes D, A = C∪D.

$$V = \bigcup V_a,\ a \in A \qquad (1)$$

V_a is the domain of attribute.

F is an information function

$$f: U \times A \rightarrow V \qquad (2)$$

Equation (2) - assigning the value of an attribute for every object $x \in U$ and every attribute $a \in A$.

Each subset of attributes $B \in A$ define an equivalence relation called an indiscernibility relation, denoted $IND(B)$ as follows:

$$IND(B) = \{(x, y) \in U \times U : f(x, a) = f(y, a), for\ every\ a \in B \qquad (3)$$

We say that x and y are indiscernible by the set of attributes B in S if $f(x) = f(y)$ for every $a \in B$. One can check that $IND(B)$ is an equivalence relation set in S. Elementary sets are called atoms of S. Information system S is selective if all atoms in S are one element set, i.e., and A is identifier relation.

2.2 Approximation Space

For the information system S = {U, A, V} and every subset $IND(B) \subset A$ generates an equivalence relation on U. An order pair $AS = (U, IND(B))$ is called an approximation space. For any element of x_i of U the equivalent class of x_i in relation to $IND(B)$ is represented as $U/IND(B)$. Equivalence class of $IND(B)$ are called elementary sets in AS because they represent the smallest discernible groups of objects.

[1] email: kkwatjan@kmitl.ac.th

[2] email: kmsomsak@kmitl.ac.th

Any finite union of elementary sets in AS is called a definable set in AS

With every $x \subset U$ and $B \subset A$, we associate two sets defined as follows:

$$\underline{B}X = \{y \in U/IND(B) | Y \subseteq X\} \quad (4)$$

$\underline{B}X$ is the union of all elementary sets, each of which is contained by X for any $x_i \in \underline{B}X$. It can be certainly classified that x_i belong to X, employing the B set of attributes. And.

$$\overline{B}X = \{y \in U/IND(B) | Y \cap X \neq \phi\} \quad (5)$$

$\overline{B}X$ is the union of elementary sets, each of which has a non-empty intersection with X. For any $x_i \in \overline{B}X$, we can only say that x_i can be possibly classified as elements of X, using attributes of B.

$\overline{B}X - \underline{B}X$ is called the B-doubtful region of B in (U,B). for any $x_i \in U$ if x_i is in $\overline{B}X - \underline{B}X$, it is impossible to determine that x_i belong to X based on the B set of attributes.

$\overline{B}X = \underline{B}X$ is called an B-exact set, otherwise it is called B-rough (with respect to B)

3. TOLERANT ROUGH SETS

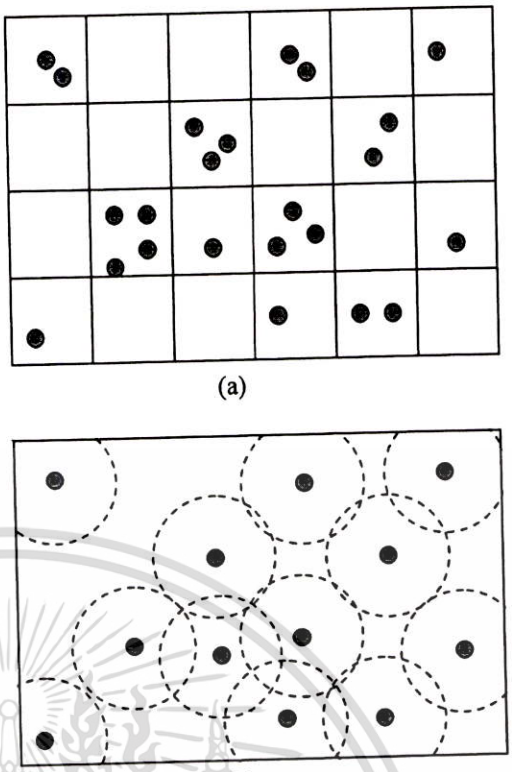
As mention earlier, the objects in real world may not have exactly the same attributes because of uncertainty in measuring and imprecision of data. This model of Rough Sets is developed in attempt to handle that problem.

The Tolerant Rough Sets utilize the similarity or tolerance relation instead of equivalence relation as used in classical Rough Sets.

The similarity relation does not partition the universe U, instead it is used to identify a set of object which are similar to some object x. The similarity class of x, denoted R(x), consists of the set of objects which are similar to x:

$$R(x) = \{y \in U : yRx\} \quad (6)$$

This relation is reflective and symmetric, but not transitive. As the relation is not transitive, object x_1 may be similar to object x_2 and object x_2 may be similar to object x_3, without x_1 similar to x_3. This means that there will be overlap between the classes. The different between the classes created by using equivalence relation and the class that created by using similarity relation is shown in Fig. 1.



(a)

(b)

Fig 1. Different between the classes created using equivalence relation (a) and similarity relation (b)

4. SYSTEM OVERVIEW

The system in this work consists of 3 main steps as follow:

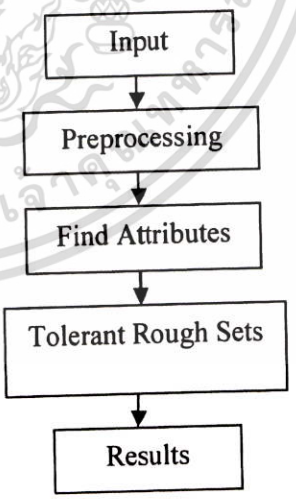


Fig. 2. System's structure

4.1 Preprocessing

After we read the scanned images, we use median filter (with the windows size of 3×3) for removing noise from the character images.

4.2 Attributes Finding

We divide the character image into 36 pieces (Fig. 3). After that we find the percentage of black color pixel in each piece. We use this black color pixel percentage of each piece to form an attribute set of each character image.

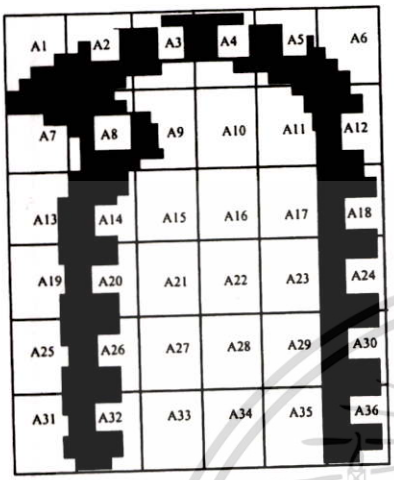


Fig. 3. Character image division.

4.3 Tolerant Rough Sets

We find the class center by averaging the attributes of character of the same type. After that we calculating Euclidean distance between that sample and the class centers. The sample will be classified into the class that has the smallest Euclidean distant.

The classes, which its members are of only one type of character, are considered as lower approximation ($\underline{BX}$). The classes that contain members which are not the same character type are considered as upper approximation ($\overline{BX}$).

For the upper approximation classes, we have to find another set of attributes to classify their member. In our work we will crop only the major different part of the character (Fig. 4.)

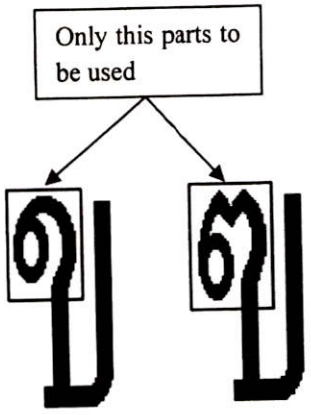


Fig. 4. An example of the character image to be used to classify the character in the upper approximation class

Fig. 5. Example of system's input

5. THE EXPERIMENT AND RESULTS

In our experiments the learning set consists of 2112 printed Thai Characters of 4 fonts and 4 sizes. The unknowns used in this experiment are 7040 Thai characters of 4 fonts and 4 sizes. All of the above are printed with HP LaserJet 6P and scanned with HP ScanJet 6100c in black and white mode. There are 44 classes created, 11 of which are lower approximation class, another 33 are upper approximation class. The example of inputted characters are shown in Fig. 5.

The results of our experiments are as follow:

Table 1. The result of the system

Detail of the unknown	Percent of accuracy
2112 characters (learning set)	99.81%
7040 characters (testing set)	92.79%

6. CONCLUSION

From the experiment, we can conclude that the tolerant relation can handle the problem of classifying data of the same type that slightly different from each other (in this case the percentage of black pixel in each piece of the character images)

The major problem of the experiment is concerned about the orientation of the character's images which is sometimes shift left or shift right. This problem can be eliminated if we use the appropriate features.

REFERENCES

[1] S. Mitatha, K. Dejhan, F. Cheevasuvit, B. Chankuang and W. Kasemsiri. **2001** Experimental results of using Rough Sets for printed Thai characters recognition. *Proceedings IEEE TENCON 2001*. Singapore pp. 331-334

[2] Watjanapong Kasemsiri and Chom Kimpan. **2001** Printed Thai characters recognition using Fuzzy-Rough Sets. *Proceedings IEEE TENCON 2001*. Singapore pp. 326-330

[3] W. Kasemsiri and C. Kimpan. **2000** Printed Thai characters recognition using 2 class levels classification and rough sets. *Proceedings APSBC'2000*, Bangkok, Thailand.

[4] Daijin Kim, Sung-Yang Bang. **2000** A Handwritten Numeral Character Classification Using Tolerant Rough Set. *IEEE Trans. On Pattern Analysis and Machine Intelligence, 22(9)*, 923-937.

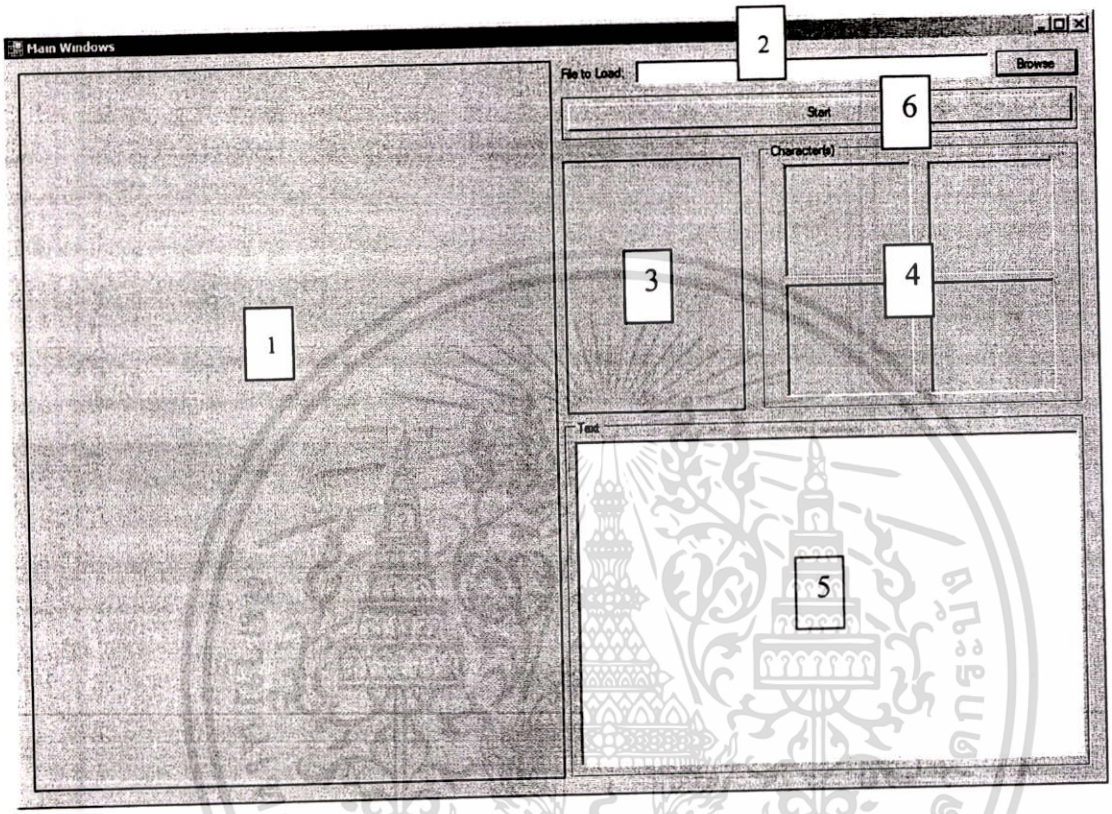
[5] Watjanapong Kasemsiri. 2003 *Printed Thai Character Recognition using Rough Sets*. Master thesis, King Mongkut's Institute of Technology Ladkrabang.

ภาคผนวก ข.

คู่มือการใช้งานโปรแกรม

คู่มือการใช้งานโปรแกรม

1. เรียกใช้งานโปรแกรมได้โดยเรียกใช้โปรแกรม OCR_01.exe เมื่อเรียกใช้งานแล้วจะปรากฏหน้าจอดังแสดงในรูปที่ 1



รูปที่ 1 ภาพหน้าจอหลักของโปรแกรม

โดยหน้าต่างหลักของโปรแกรมจะประกอบไปด้วยส่วนต่าง ๆ ดังนี้

ส่วนที่ 1 เป็นส่วนสำหรับใช้เพื่อแสดงภาพตัวอักษรจากไฟล์ทั้งภาพ

ส่วนที่ 2 เป็นช่องสำหรับแสดงชื่อไฟล์ที่กำลังทำงานด้วย โดยสามารถเปิดไฟล์เพื่อทำงานได้โดยกดปุ่ม Browse

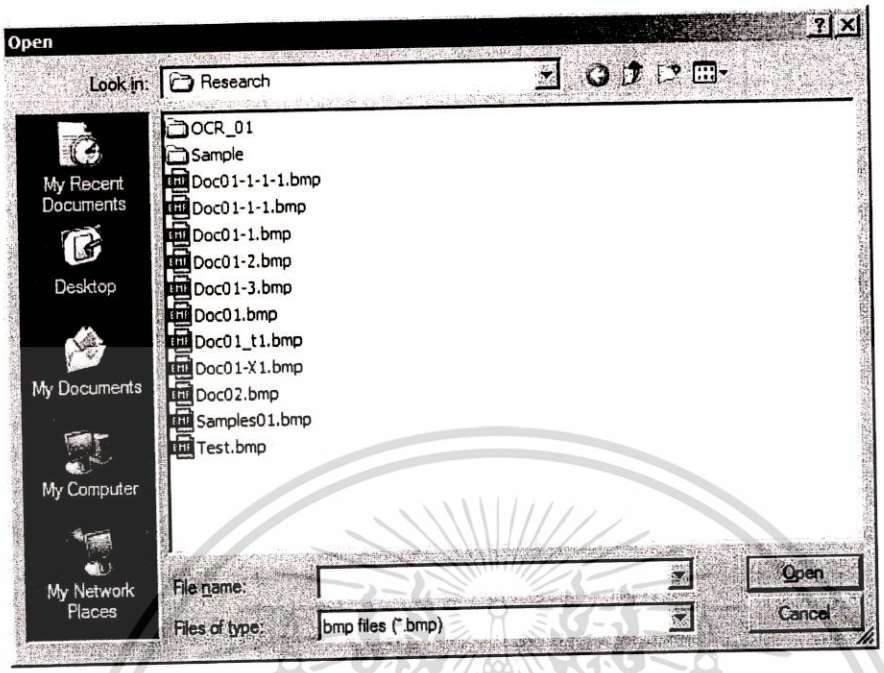
ส่วนที่ 3 เป็นช่องสำหรับแสดงตัวอักษรที่ผ่านการตัด (segment) ครั้งที่ 1 ซึ่งภาพตัวอักษรที่ตัดในครั้งแรกนี้อาจยังมีส่วนที่ไม่ต้องการติดเข้ามาด้วย

ส่วนที่ 4 เป็นช่องสำหรับแสดงภาพตัวอักษรที่ผ่านการตัดครั้งที่ 2 ซึ่งในการตัดครั้งที่ 2 นี้จะได้เฉพาะส่วนที่ต้องการ

ส่วนที่ 5 เป็นช่องสำหรับใช้แสดงผลที่ได้จากการรู้จำ

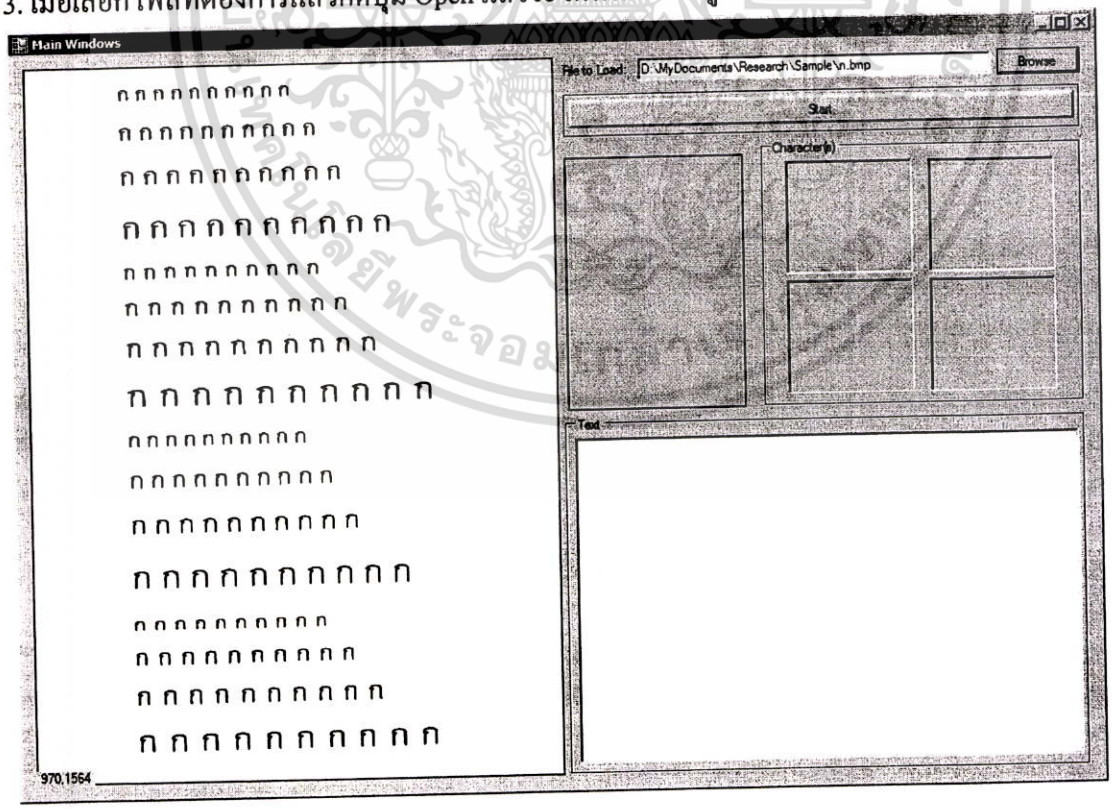
ส่วนที่ 6 ปุ่มสำหรับเริ่มการรู้จำ

2. เมื่อต้องการเริ่มต้นทำงานให้กดปุ่ม Browse จะปรากฏ dialog ดังแสดงในรูปที่ 2



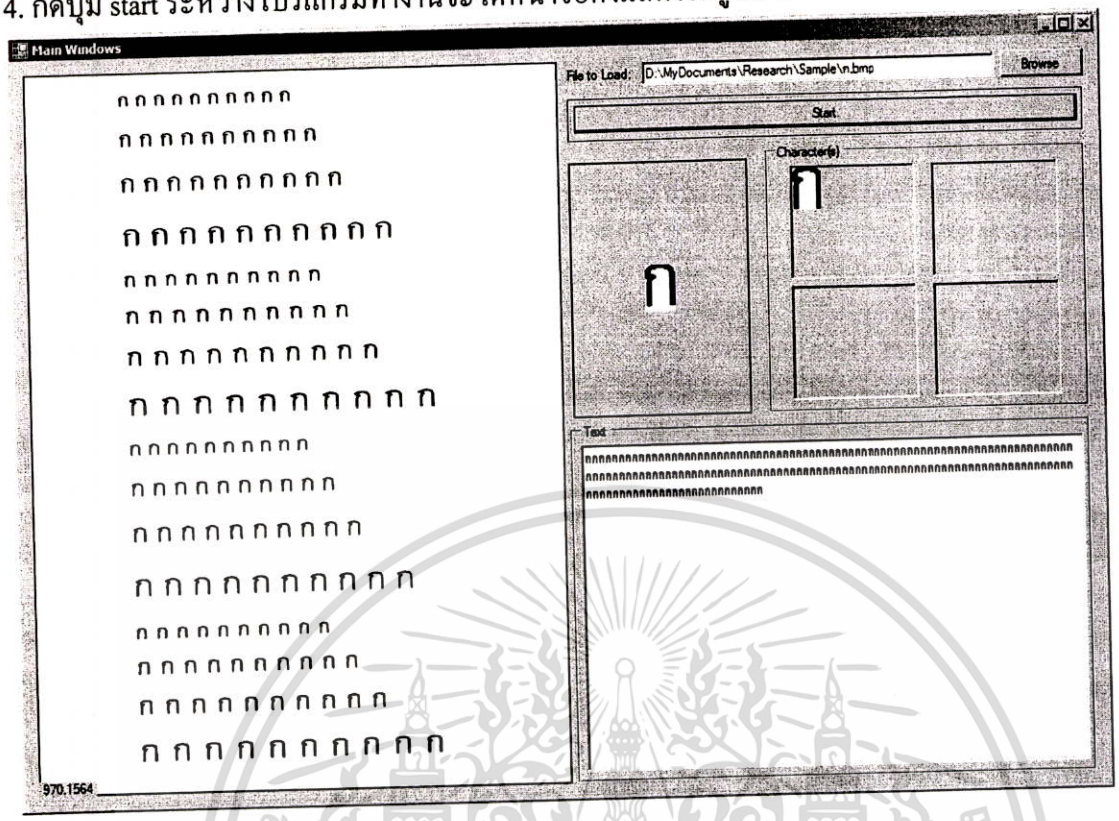
รูปที่ 2 ภาพหน้าต่างสำหรับใช้เลือกเปิดไฟล์

3. เมื่อเลือกไฟล์ที่ต้องการแล้วกดปุ่ม Open แล้วจะได้หน้าจอในรูปที่ 3



รูปที่ 3 หน้าจอเมื่อเปิดไฟล์

4. กดปุ่ม start ระหว่างโปรแกรมทำงานจะได้หน้าจอดังแสดงในรูปที่ 4



รูปที่ 4 หน้าจอขณะทำงาน

โดยผลลัพธ์ของการทำงานจะแสดงไว้ในส่วนที่ 5 ของหน้าจอ