ใบรับรองวิทยานิพนธ์

บัณฑิตวิทยาลัย มหาวิทยาลัยเกษตรศาสตร์

วิทยาศาสตรมหาบัณฑิต (วิทยาการคอมพิวเตอร์)

ปริญญา

วิทยาการคอมพิวเตอร์	วิทยาการคอมพิวเตอร์
สาขา	**ภาควิชา**

เรื่อง การตรวจข้อสอบอัตนัยภาษาไทยแบบอัตโนมัติ

Automated Thai-Language Essay Scoring

นามผู้วิจัย นายสมมาตร อังคเศรณีกุล

ได้พิจารณาเห็นชอบโดย

อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก

(รองศาสตราจารย์ชุลีรัตน์ จรัสกุลชัย, D.Sc.)

หัวหน้าภาควิชา

(ผู้ช่วยศาสตราจารย์ศิริกร จันทร์นวล, M.S.)

บัณฑิตวิทยาลัย มหาวิทยาลัยเกษตรศาสตร์รับรองแล้ว

(รองศาสตราจารย์กัญจนา ธีระกุล, D.Agr.)

คณบดีบัณฑิตวิทยาลัย

วันที่ ..... **เดือน** ..... **พ.ศ.** .....

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

วิทยานิพนธ์

เรื่อง

การตรวจข้อสอบอัตนัยภาษาไทยแบบอัตโนมัติ

Automated Thai-Language Essay Scoring

โดย

นายสมมาตร อังคเศรณีกุล

เสนอ

บัณฑิตวิทยาลัย มหาวิทยาลัยเกษตรศาสตร์
เพื่อความสมบูรณ์แห่งปริญญาวิทยาศาสตรมหาบัณฑิต (วิทยาการคอมพิวเตอร์)
พ.ศ. 2555

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

สมมาตร อังคเศรณีกุล 2555: การตรวจข้อสอบอัตนัยภาษาไทยแบบอัตโนมัติ ปริญญาวิทยาศาสตรมหาบัณฑิต (วิทยาการคอมพิวเตอร์) สาขาวิทยาการคอมพิวเตอร์ ภาควิชาวิทยาการคอมพิวเตอร์ อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก: รองศาสตราจารย์ชุลีรัตน์ จรัสกุลชัย, D.Sc. 116 หน้า

การวิจัยด้านการเรียนรู้ของเครื่องจักรถูกนำประยุกต์ใช้ในชีวิตประจำวันมากขึ้น รวมทั้งการตรวจข้อสอบอัตนัยแบบอัตโนมัติ แนวคิดเบื้องต้นในการตรวจข้อสอบอัตโนมัติคล้ายกับการจัดกลุ่มเอกสาร หรือการจำแนกเอกสารตามหมวดหมู่ของคำตอบ กระบวนการในการตรวจข้อสอบต้องจัดเตรียมชุดคำตอบเพื่อเป็นข้อมูลในการสอนระบบ โดยปกติการเตรียมข้อมูลสอนระบบจะเตรียมเฉพาะคำตอบที่ถูกต้อง และไม่สามารถเตรียมข้อมูลสอนระบบให้กับข้อมูลทุกประเภท อีกทั้งการตรวจข้อสอบแต่ละครั้งไม่สามารถกำหนดจำนวนกลุ่มคำตอบได้แน่นอน

งานวิจัยนี้นำเสนอขั้นตอนวิธีการจัดกลุ่มคำตอบโดยที่ไม่ต้องกำหนดจำนวนกลุ่มแน่นอน คืออัลกอริทึม EM และ Cobweb และศึกษาปัจจัยที่ช่วยในการคิดคะแนน โดยให้คะแนนคำตอบแต่ละกลุ่มด้วยการวัดค่าความคล้ายของคำตัวแทนแต่ละกลุ่มคำตอบกับคำเฉลยจำนวน 1 ชุดด้วยสมการ Cosine ร่วมกับเทคนิค 2 เทคนิคที่นำมาใช้ในงานวิจัยคือ เทคนิคการขยายคำค้นใช้ในการแก้ปัญหาคำเหมือน และเทคนิค LSA ใช้ในการลดขนาดเวกเตอร์ชุดคำตอบ และหาความหมายที่ซ่อนอยู่จากความสัมพันธ์ของคำในชุดคำตอบ จากนั้นนำผลการทดลองมาเปรียบเทียบกับผลที่ได้จากอัลกอริทึมการเรียนรู้แบบมีผู้สอนคือ อัลกอริทึม K-NN, ANN และ Bayesian

การทดสอบประสิทธิผลทดลองกับข้อสอบวิชาการโปรแกรมเชิงวัตถุจำนวน 7 ข้อ มีชุดคำตอบของนักศึกษา 55 คน และชุดเฉลย 1 ชุด ผลการทดลองพบว่า วิธีการให้เกรดด้วยอัลกอริทึมการเรียนรู้แบบมีผู้สอนกับวิธีการให้เกรดด้วยอัลกอริทึมการเรียนรู้แบบไม่มีผู้สอนได้ค่าความแม่นยำเฉลี่ยรวมใกล้เคียงกัน

_____ _____ ___ / ___ / _____

ลายมือชื่อนิสิต ลายมือชื่ออาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

Sommart Aungkaseraneekul 2012: Automated Thai-Language Essay Scoring. Master of Science (Computer Science), Major Field: Computer Science, Department of Computer Science. Thesis Advisor: Associate Professor Chuleerat Jaruskulchai, D.Sc. 116 pages.

Machine learning research has been applied in the daily life including automatic essay scoring. The concept of automatic essay grading is comparable to the documents clustering or classification according to a set of essays. However, the preparation of the training data needs to include every category of the answer sets which is impractical. The training data usually includes the appropriate essays and it might be available only for some types of data. Additionally, the number of clusters may not be known in advantage.

In this research, EM and Cobweb algorithms are used to discover automatically the number of clusters for each question. The similarity score of each of clusters was calculated using cosine function between each representative of clusters and its one solution. We used query expansion and LSA techniques in our work; query expansion technique was deployed in order to solve the problem of synonyms and LSA technique was used to find a latent semantic meaning and reduced data dimension. The results from tests were compared with the learning algorithm such as K-NN, ANN and Bayesian.

The effectiveness of the algorithms is assessed with seven questions of object-oriented programming test with total of fifty-five students' answers and applied only a single solution for assigning score. The experimental results showed that the total average accuracy rate between unsupervised algorithm and supervise algorithm are very close.

_____ _____ ___ / ___ / _____

Student's signature Thesis Advisor's signature

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จสมบูรณ์ได้ด้วยความเมตตาและความช่วยเหลือเป็นอย่างดีจาก รองศาสตราจารย์ ดร.ชุลีรัตน์ จรัสกุลชัย ประธานกรรมการที่ปรึกษาวิทยานิพนธ์ ที่กรุณาให้ความรู้ คำแนะนำ ชี้แนะแนวทาง ตลอดจนตรวจสอบและแก้ไขข้อบกพร่องต่างๆ เพื่อให้วิทยานิพนธ์มีความถูกต้องสมบูรณ์และมีคุณค่าทางวิชาการ ผู้วิจัยรู้สึกซาบซึ้งในความกรุณาที่ได้รับเป็นอย่างยิ่ง จึงขอกราบขอบพระคุณอย่างสูงไว้ ณ โอกาสนี้

ขอกราบขอบพระคุณคณาจารย์ภาควิชาวิทยาการคอมพิวเตอร์ทุกท่าน ที่ได้อบรมสั่งสอนและมอบความรู้อันเป็นประโยชน์อย่างยิ่งในการนำไปใช้ประโยชน์ต่อไป และขอขอบคุณเจ้าหน้าที่ ภาควิชาวิทยาการคอมพิวเตอร์ทุกท่าน ที่ได้ให้ความช่วยเหลือและให้คำแนะนำต่างๆ เป็นอย่างดี

ขอกราบขอบพระคุณ ดร.วาสนา วิรัตน์โยสินทร์ และ ดร.อิทธิพันธ์ เมธเศรษฐ จากสำนักงานพัฒนาวิทยาศาสตร์และเทคโนโลยีแห่งชาติ (สวทช.) ที่ให้ความรู้และคำแนะนำต่างๆ ที่เป็นประโยชน์ในงานวิจัย

ขอขอบคุณ คุณกิตติ ขุนสนิท นิสิตปริญญาเอกสาขาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยเกษตรศาสตร์ ที่คอยให้คำปรึกษาในการเรียน และการแก้ปัญหาในสถานการณ์ต่างๆ ตลอดจนให้การสนับสนุนให้ผู้วิจัยได้เผยแพร่ผลงานและตีพิมพ์ในการประชุมวิชาการระดับนานาชาติได้สำเร็จ

ท้ายสุดนี้ผู้วิจัยขอกราบขอบพระคุณบิดามารดาที่ให้การอบรมเลี้ยงดู ให้โอกาสทางการศึกษา ให้กำลังใจและให้การสนับสนุนในทุกๆ ด้านตลอดมาอย่างดียิ่ง ขอขอบคุณญาติพี่น้อง และนิสิตร่วมรุ่นปริญญาโท และนิสิตปริญญาเอกสาขาวิทยาการคอมพิวเตอร์ทุกท่าน ตลอดจนเพื่อนๆ ที่มิได้เอ่ยนามที่ให้ความช่วยเหลือและสนับสนุนตลอดมาทำให้วิทยานิพนธ์ฉบับนี้เสร็จสมบูรณ์ในที่สุด

สมมาตร อังคเศรณีกุล

พฤษภาคม 2555

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

สารบัญ

หน้า

สารบัญ	(1)
สารบัญตาราง	(2)
สารบัญภาพ	(6)
คำอธิบายสัญลักษณ์และคำย่อ	(7)
คำนำ	1
วัตถุประสงค์	4
การตรวจเอกสาร	6
อุปกรณ์และวิธีการ	20
อุปกรณ์	20
วิธีการ	20
ผลและวิจารณ์	36
ผล	36
วิจารณ์	58
สรุปและข้อเสนอแนะ	61
สรุป	61
ข้อเสนอแนะ	62
เอกสารและสิ่งอ้างอิง	63
ภาคผนวก	70
ภาคผนวก ก รายละเอียดผลการทดลอง	71
ภาคผนวก ข ผลงานตีพิมพ์	87
ภาคผนวก ค ตัวอย่างโปรแกรมตรวจข้อสอบอัตนัยภาษาไทย	113
ประวัติการศึกษา และการทำงาน	116

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

สารบัญตาราง

ตารางที่		หน้า
1	ตัวอย่างแท็ก POS	18
2	สถิติของข้อมูล	24
3	ตัวอย่างคำตอบของนักเรียน 3 คน	26
4	การแทนคำตอบในรูปความถี่ของคำ	26
5	การแทนคำตอบในรูปความถี่ของคำและถ่วงน้ำหนักโดยใช้ TF-IDF	26
6	การแทนคำตอบในรูปความถี่ของคำแบบใหม่ โดยใช้พจนานุกรมเข้ามาช่วยแก้ปัญหากรณีมีคำเหมือน	28
7	ตัวอย่างผลลัพธ์ที่ได้จากการประมวลผลสำหรับเกรด A	34
8	ค่าความแม่นยำในแต่ละข้อในการจำแนกด้วยอัลกอริทึม K-NN	36
9	ค่าความแม่นยำในแต่ละข้อกรณีที่ใช้พจนานุกรม ในการจำแนกด้วยอัลกอริทึม K-NN	37
10	ค่าความแม่นยำในแต่ละข้อในการจำแนกด้วยอัลกอริทึม K-NN ร่วมกับเทคนิค LSA	37
11	ค่าความแม่นยำในแต่ละข้อกรณีที่ใช้พจนานุกรม ในการจำแนกด้วยอัลกอริทึม K-NN ร่วมกับเทคนิค LSA	38
12	ค่าความแม่นยำในแต่ละข้อในการจำแนกด้วยอัลกอริทึม ANN	39
13	ค่าความแม่นยำในแต่ละข้อกรณีที่ใช้พจนานุกรม ในการจำแนกด้วยอัลกอริทึม ANN	39
14	ค่าความแม่นยำในแต่ละข้อในการจำแนกด้วยอัลกอริทึม ANN ร่วมกับเทคนิค LSA	40
15	ค่าความแม่นยำในแต่ละข้อกรณีที่ใช้พจนานุกรม ในการจำแนกด้วยอัลกอริทึม ANN ร่วมกับเทคนิค LSI	41
16	ค่าความแม่นยำในแต่ละข้อในการจำแนกด้วยอัลกอริทึม Bayesian	42
17	ค่าความแม่นยำในแต่ละข้อกรณีที่ใช้พจนานุกรม ในการจำแนกด้วยอัลกอริทึม Bayesian	42
18	ค่าความแม่นยำในแต่ละข้อในการจำแนกด้วยอัลกอริทึม Bayesian ร่วมกับเทคนิค LSA	43

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

สารบัญตาราง (ต่อ)

ตารางที่		หน้า
19	ค่าความแม่นยำในแต่ละข้อกรณีที่ใช้พจนานุกรม ในการจำแนกด้วยอัลกอริทึม Bayesian ร่วมกับเทคนิค LSA	44
20	ค่าความแม่นยำเฉลี่ยรวมข้อ 1-7 ของวิธีการคิดเกรดแบบดั้งเดิม และอัลกอริทึมการ แบ่งกลุ่มทั้ง 2 เรียงตามช่วงค่า Threshold A แบบปกติ	49
21	ค่าความแม่นยำเฉลี่ยรวมข้อ 1-7 ของวิธีการคิดเกรดแบบดั้งเดิม และอัลกอริทึมการแบ่งกลุ่มทั้ง 2 เรียงตามช่วงค่า Threshold A แบบปกติร่วมกับพจนานุกรมแก้ปัญหาคำเหมือน	50
22	ค่าความแม่นยำเฉลี่ยรวมข้อ 1-7 ของวิธีการคิดเกรดแบบดั้งเดิม และอัลกอริทึมการแบ่งกลุ่มทั้ง 2 เรียงตามช่วงค่า Threshold A แบบใช้เทคนิค LSA	51
23	ค่าความแม่นยำเฉลี่ยรวมข้อ 1-7 ของวิธีการคิดเกรดแบบดั้งเดิม และอัลกอริทึมการแบ่งกลุ่มทั้ง 2 เรียงตามช่วงค่า Threshold A แบบใช้เทคนิค LSA ร่วมกับพจนานุกรมแก้ปัญหาคำเหมือน	52
24	ประสิทธิผลการจัดกลุ่มด้วยอัลกอริทึม EM	53
25	ประสิทธิผลการจัดกลุ่มด้วยอัลกอริทึม Cobweb	54
26	ค่าความแม่นยำและค่า MSE เฉลี่ยรวมในแต่ละอัลกอริทึม ด้วยวิธีการคิดเกรดแบบต่างๆ	56

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

สารบัญตาราง (ต่อ)

ตารางผนวกที่		หน้า
ก1	ค่าความแม่นยำของแต่ละช่วง Threshold A ของข้อที่ 1-7 ในวิธีการให้เกรดแบบดั้งเดิม	72
ก2	ค่าความแม่นยำของแต่ละช่วง Threshold B ของข้อที่ 1-7 ในวิธีการให้เกรดแบบดั้งเดิม	73
ก3	ค่าความแม่นยำของแต่ละช่วง Threshold C ของข้อที่ 1-7 ในวิธีการให้เกรดแบบดั้งเดิม	74
ก4	ค่าความแม่นยำของแต่ละช่วง Threshold D ของข้อที่ 1-7 ในวิธีการให้เกรดแบบดั้งเดิม	75
ก5	ค่าความแม่นยำของแต่ละช่วง Threshold F ของข้อที่ 1-7 ในวิธีการให้เกรดแบบดั้งเดิม	76
ก6	ค่าความแม่นยำของแต่ละช่วง Threshold A ของข้อที่ 1-7 ในวิธีการแบ่งกลุ่มด้วยอัลกอริทึม EM	77
ก7	ค่าความแม่นยำของแต่ละช่วง Threshold B ของข้อที่ 1-7 ในวิธีการแบ่งกลุ่มด้วยอัลกอริทึม EM	78
ก8	ค่าความแม่นยำของแต่ละช่วง Threshold C ของข้อที่ 1-7 ในวิธีการแบ่งกลุ่มด้วยอัลกอริทึม EM	79
ก9	ค่าความแม่นยำของแต่ละช่วง Threshold D ของข้อที่ 1-7 ในวิธีการแบ่งกลุ่มด้วยอัลกอริทึม EM	80
ก10	ค่าความแม่นยำของแต่ละช่วง Threshold F ของข้อที่ 1-7 ในวิธีการแบ่งกลุ่มด้วยอัลกอริทึม EM	81
ก11	ค่าความแม่นยำของแต่ละช่วง Threshold A ของข้อที่ 1-7 ในวิธีการแบ่งกลุ่มด้วยอัลกอริทึม Cobweb	82
ก12	ค่าความแม่นยำของแต่ละช่วง Threshold B ของข้อที่ 1-7 ในวิธีการแบ่งกลุ่มด้วยอัลกอริทึม Cobweb	83
ก13	ค่าความแม่นยำของแต่ละช่วง Threshold C ของข้อที่ 1-7 ในวิธีการแบ่งกลุ่มด้วยอัลกอริทึม Cobweb	84

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

สารบัญตาราง (ต่อ)

ตารางผนวกที่ **หน้า**

ก14 ค่าความแม่นยำของแต่ละช่วง Threshold D ของข้อที่ 1-7 ในวิธีการแบ่งกลุ่มด้วยอัลกอริทึม Cobweb 85

ก15 ค่าความแม่นยำของแต่ละช่วง Threshold F ของข้อที่ 1-7 ในวิธีการแบ่งกลุ่มด้วยอัลกอริทึม Cobweb 86

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

สารบัญภาพ

ภาพที่		หน้า
1	โครงสร้าง Perceptron	11
2	แผนภาพขั้นตอนการสร้างระบบการตรวจข้อสอบอัตนัยภาษาไทย แบบอัตโนมัติวิธีการที่ 1	21
3	แผนภาพขั้นตอนการสร้างระบบการตรวจข้อสอบอัตนัยภาษาไทย แบบอัตโนมัติวิธีการที่ 2	22
4	ตัวอย่างเมทริกที่ได้หลังจากการแตกเป็นเมทริกย่อยในเทคนิค LSA	27
5	ตัวอย่างโครงข่ายโมเดล Naïve Bayes	29
6	ตัวอย่างโครงสร้างแบบ Multi-Layer Perceptron	30
7	กราฟความสัมพันธ์ของค่าความคล้ายเฉลี่ยกับจำนวนคำเอกลักษณ์	33
8	กราฟค่าความแม่นยำเฉลี่ยรวมข้อ 1-7 ของวิธีการคิดเกรดแบบดั้งเดิม และอัลกอริทึมการแบ่งกลุ่มทั้ง 2 เรียงตามช่วงค่า Threshold A แบบปกติ	45
9	กราฟค่าความแม่นยำเฉลี่ยรวมข้อ 1-7 ของวิธีการคิดเกรดแบบดั้งเดิม และอัลกอริทึมการแบ่งกลุ่มทั้ง 2 เรียงตามช่วงค่า Threshold A แบบปกติร่วมกับพจนานุกรมแก้ปัญหาคำ เหมือน	46
10	กราฟค่าความแม่นยำเฉลี่ยรวมข้อ 1-7 ของวิธีการคิดเกรดแบบดั้งเดิม และอัลกอริทึมการแบ่งกลุ่มทั้ง 2 เรียงตามช่วงค่า Threshold A แบบใช้เทคนิค LSA	47
11	กราฟค่าความแม่นยำเฉลี่ยรวมข้อ 1-7 ของวิธีการคิดเกรดแบบดั้งเดิม และอัลกอริทึมการแบ่งกลุ่มทั้ง 2 เรียงตามช่วงค่า Threshold A แบบใช้เทคนิค LSA ร่วมกับพจนานุกรมแก้ปัญหาคำเหมือน	48

ภาพผนวกที่

ค1	โปรแกรมตรวจข้อสอบอัตนัยภาษาไทยด้วยวิธี Classical method	114
ค2	โปรแกรมตรวจข้อสอบอัตนัยภาษาไทยด้วยอัลกอริทึม EM	115

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

คำอธิบายสัญลักษณ์และคำย่อ

®	=	Copyright
LSA	=	Latent Semantic Analysis
ANN	=	Artificial Neural Network
BBN	=	Bayesian Belief Network
K-NN	=	K-Nearest Neighbor
CPT	=	Conditional Probability Table
MSE	=	Mean Squared Error

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

การตรวจข้อสอบอัตนัยภาษาไทยแบบอัตโนมัติ

Automated Thai-Language Essay Scoring

คำนำ

การเรียนการสอนในปัจจุบัน ไม่ว่าในระดับชั้นประถมศึกษา ระดับชั้นมัธยมศึกษา และระดับอุดมศึกษาต่างๆ ยอมรับว่าการออกข้อสอบแบบอัตนัยเพื่อใช้ในการวัดประสิทธิผลการเรียนรู้ของผู้เรียนมีศักยภาพมากกว่าการออกข้อสอบแบบปรนัย เนื่องจากข้อสอบแบบอัตนัยเป็นการเขียนคำตอบ ทำให้ผู้เรียนสามารถแสดงทักษะทางความคิด ความจำ และความสามารถในการเขียนสังเคราะห์หรือการวิเคราะห์ต่างๆ ได้มากกว่าข้อสอบแบบปรนัย แต่อย่างไรก็ตามเมื่อจำนวนข้อสอบมีปริมาณมากขึ้น ผู้ตรวจจะพบปัญหาที่หลีกเลี่ยงไม่ได้คือ เวลาที่ใช้ในการตรวจมากขึ้น เกิดความเมื่อยล้าของผู้ตรวจ และขาดความเที่ยงตรงในการตรวจ เพราะมีอารมณ์และความรู้สึกต่างๆมาเกี่ยวข้อง เป็นต้น จากปัญหาที่กล่าวมานี้ ทำให้นักวิจัยพยายามคิดสร้างระบบการตรวจข้อสอบอัตนัยแบบอัตโนมัติด้วยคอมพิวเตอร์ขึ้นโดยเบื้องต้นได้จำลองแนวคิดและวิธีการตรวจของมนุษย์ เช่น ให้คะแนนโดยดูจากคำสำคัญ คำเฉลย และจากลักษณะคำตอบแบบต่างๆของผู้ตอบในแต่ละระดับคะแนน เป็นต้น เนื่องจากในปัจจุบันคอมพิวเตอร์สามารถทำงานได้รวดเร็วกว่าแต่ก่อนมาก ทำให้ประหยัดเวลา ประหยัดทรัพยากร และลดปัญหาความไม่เที่ยงตรงในการตรวจ เพราะคอมพิวเตอร์ไม่มีเรื่องของอารมณ์ ความรู้สึก และความเมื่อยล้าที่เกิดขึ้นในมนุษย์มาเกี่ยวข้อง

ระเบียบวิธีการตรวจข้อสอบอัตนัยแบบอัตโนมัติจะคล้ายกับการจำแนกเอกสาร โดยจำแนกกลุ่มคำตอบตามระดับความคล้ายของแต่ละระดับเกรดในชุดข้อมูลสอนระบบ ซึ่งจำเป็นที่จะต้องมีชุดข้อมูลสอนระบบขนาดใหญ่เพื่อใช้สำหรับสร้างแบบจำลองการทำนายคะแนน ตัวอย่างระบบการตรวจข้อสอบในเริ่มแรก เช่น ระบบ Project Essay Grade (PEG) ถูกพัฒนาขึ้นโดย Page (Page, 1966) โดยใช้ลักษณะคำตอบที่ผ่านการตรวจด้วยมนุษย์ในแต่ละระดับเกรด เช่น คำที่ใช้ในคำตอบ จำนวนคำในคำตอบ ความยาวประโยค และเครื่องหมายวรรคตอน เป็นต้น มาใช้เป็นตัวแปรในแบบจำลอง Multiple linear regressions เพื่อใช้ในการให้คะแนนในแต่ละคำตอบ ต่อมาได้มีการคิดระบบ Electronic Essay Rater (E-Rater) (Attali and Burstein, 2006) เป็นการประยุกต์ใช้แบบจำลองของ Multiple linear regressions เพื่อทำนายคะแนน โดยแทนชุดคำตอบด้วยลักษณะ

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

หรือรูปแบบของประโยค และไวยากรณ์ เป็นต้น และระบบ Intelligent Essay Assessor (IEA) (Landauer *et al.*, 1998) ใช้เทคนิค Latent Semantic Analysis (LSA) ซึ่งเป็นเทคนิคที่มีการใช้อย่างแพร่หลายทางด้านการประมวลผลภาษาธรรมชาติ มีพื้นฐานมาจาก Singular Value Decomposition (SVD) ซึ่งเป็นวิธีการทางพีชคณิต เพื่อแทนความสัมพันธ์ระหว่างชุดคำตอบกับคำในคำตอบ และลดขนาดของเมทริกเดิมที่ใช้แทนคำในแต่ละคำตอบ แต่ยังคงลักษณะที่สำคัญไว้ เรียกเมทริกนี้ว่า พื้นที่เชิงความหมาย (Semantic space) และใช้เมทริกใหม่นี้ในการทำนายหรือคำนวณคะแนนคำตอบ ทำให้การให้คะแนนคำตอบถูกต้องมากขึ้น โดยอัลกอริทึมที่นำมาใช้ในระบบการตรวจข้อสอบจะเป็นอัลกอริทึมทางด้าน Machine learning แบ่งตามประเภทการเรียนรู้ คือ อัลกอริทึมการเรียนรู้แบบมีผู้สอน และอัลกอริทึมการเรียนรู้แบบไม่มีผู้สอน

อัลกอริทึมการเรียนรู้แบบมีผู้สอน เช่น อัลกอริทึม Artificial Neural Network (ANN) (Loraksa and Peachavanish, 2007) อัลกอริทึม Bayesian (Rudner and Liang, 2002) และอัลกอริทึม K-NN (Bin *et al.*, 2008) เป็นต้น จำเป็นต้องใช้ชุดข้อมูลสอนระบบเพื่อสร้างแบบจำลองสำหรับการให้คะแนน อัลกอริทึมที่นำมาใช้ทางด้านการจัดกลุ่มข้อความ (Test-categorization) เช่น Bayesian independence (Larkey, 1998) ถูกนำมาประยุกต์ใช้ในการแยกกลุ่มคำตอบที่ถูกและผิดออกจากกัน โดยใช้แบบจำลอง Linear regression ในการให้คะแนนคำตอบ และใช้เทคนิค LSA มาวิเคราะห์รูปแบบหรือความสัมพันธ์ในประโยคร่วมด้วย แต่อย่างไรก็ตามเทคนิค LSA ยังมีจุดด้อย คือ ไม่มีส่วนของการวิเคราะห์โครงสร้างหรือลำดับของคำในคำตอบ ดังนั้นเทคนิค Probabilistic Latent Semantic Analysis (PLSA) (Tuomo *et al.*, 2005) และ Syntactically Enhanced Latent Semantic Analysis (SELSA) (Kanejiya *et al.*, 2003) ถูกนำเสนอขึ้นเพื่อแก้ปัญหาดังกล่าว

อัลกอริทึมการเรียนรู้แบบไม่มีผู้สอน เช่น อัลกอริทึม K-means (Chen *et al.*, 2010) ถูกนำมาประยุกต์ใช้ในคำถามปลายเปิด (Open-ended question) เพื่อวัดความเข้าใจหรือการแสดงความคิดเห็นของผู้ตอบ ซึ่งคำตอบของคำตามประเภทนี้ไม่จำเป็นต้องใช้ชุดข้อมูลสอนระบบหรือชุดเฉลยในการให้คะแนนคำตอบ แต่จะขึ้นอยู่กับระดับความคล้ายของคำตอบในอัลกอริทึม Voting โดยขั้นตอนเริ่มต้นจะกำหนดคะแนนเริ่มต้น (Initial score) จากจำนวนคำและคำที่ใช้ร่วมกันในคำตอบ และคะแนนสุดท้าย (Final score) คำนวณได้จากอัลกอริทึม Voting

งานวิจัยนี้นำเสนอระบบการตรวจข้อสอบอัตนัยภาษาไทย โดยใช้ชุดคำตอบจากลักษณะคำถามแบบปลายปิด ความยาวคำตอบแบบสั้น ทดลองกับอัลกอริทึมการเรียนรู้แบบมีผู้สอน คือ อัลกอริทึม K-NN ANN และ Bayesian และอัลกอริทึมการเรียนรู้แบบไม่มีผู้สอน คือ อัลกอริทึม

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

EM และ Cobweb โดยใช้เทคนิคการขยายคำค้น (Query Expansion) และเทคนิค LSA ร่วมด้วยในการวิเคราะห์ผล เพื่อเปรียบเทียบข้อดีข้อเสีย และสรุปผลที่ได้ในอัลกอริทึมการเรียนรู้แต่ละแบบ

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

วัตถุประสงค์

1. เพื่อนำอัลกอริทึม K-NN ANN และ Bayesian มาใช้ในการทำนายคะแนนคำตอบ โดยใช้ชุดข้อมูลสอนระบบหรือชุดคำตอบที่ผ่านการตรวจจากมนุษย์

2. เพื่อนำอัลกอริทึม EM และ Cobweb มาใช้ในการแบ่งกลุ่มคำตอบ โดยใช้ชุดเฉลย 1 ชุด ในการให้คะแนนในแต่ละกลุ่มคำตอบ

3. เพื่อนำเทคนิคการขยายคำขอ (Query Expansion) ด้วยพจนานุกรมอังกฤษ–ไทย และเทคนิค LSA มาช่วยเพิ่มประสิทธิภาพในการให้คะแนนคำตอบแบบอัตโนมัติ

4. เพื่อสร้างโปรแกรมต้นแบบการตรวจข้อสอบอัตนัยภาษาไทยแบบอัตโนมัติ โดยใช้ลักษณะคำถามแบบปลายปิด และเป็นคำตอบแบบสั้น

ประโยชน์ที่คาดว่าจะได้รับ

1. ช่วยตรวจข้อสอบอัตนัยภาษาไทยและให้คะแนนแทนมนุษย์อย่างยุติธรรมและเที่ยงตรง

2. ลดความเมื่อยล้าในการตรวจให้กับมนุษย์

3. สะดวก รวดเร็ว และประหยัดเวลาในการตรวจ ทำให้สามารถตรวจข้อสอบของนักเรียนได้ในปริมาณมากๆ

ขอบเขตและข้อจำกัด

1. ใช้ข้อสอบอัตนัยวิชาการโปรแกรมเชิงวัตถุด้วยภาษาจาวา จำนวน 7 ข้อ มีคำตอบนักเรียน 55 คน โดยลักษณะคำตอบเป็นแบบปลายปิด ความยาวคำตอบแบบสั้น

2. ใช้ชุดข้อมูลสอนระบบ และชุดเฉลยในการให้คะแนนคำตอบและวัดประสิทธิผลในแต่ละอัลกอริทึม

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

3. วิเคราะห์ในระดับคำในแต่ละคำตอบของนักเรียน ด้วยวิธีการตัดคำด้วยโปรแกรมตัดคำภาษาไทย โดยไม่คำนึงถึงลำดับของคำในประโยค

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

การตรวจเอกสาร

ทฤษฎีที่เกี่ยวข้องกับงานวิจัย

ทฤษฎีที่นำมาใช้ในระบบการตรวจข้อสอบอัตนัย แบ่งเป็น 2 หัวข้อ คือ ทฤษฎีทางด้านเทคนิคที่ใช้ในการปรับเปลี่ยนข้อมูลเข้ารูปแบบเวกเตอร์ และทฤษฎีทางด้านปัญญาประดิษฐ์ (Artificial Intelligence) มีรายละเอียดดังนี้

1. ทฤษฎีทางด้านเทคนิคที่ใช้ในการปรับเปลี่ยนข้อมูลเข้ารูปแบบเวกเตอร์

1.1 Latent Semantic Analysis (LSA)

Latent Semantic Analysis (LSA) (Furnas, *et al.*, 1988) เป็นเทคนิคที่นำมาใช้ทางด้าน NLP (Natural Language Processing) ในระบบการค้นคืนสารสนเทศ (Information Retrieval System) มีจุดประสงค์เพื่อลดขนาดของชุดข้อมูลหรือชุดเอกสารที่มีขนาดใหญ่ และใช้ประโยชน์จากการลดขนาดนี้ในการหาโครงสร้างทางความหมายที่ซ่อนอยู่ในชุดเอกสาร โดยวิเคราะห์ความสัมพันธ์ระหว่างคำในชุดเอกสาร มีพื้นฐานมาจากพีชคณิตเชิงเส้นที่ชื่อว่า Singular Value Decomposition (SVD) ซึ่งเริ่มต้นด้วยเมทริกของ Term-Document หรือความถี่ของคำต่างๆในแต่ละเอกสาร (ส่วนใหญ่จะแทนในรูปของ TF-IDF เพื่อให้ความสำคัญในแต่ละคำ รายละเอียดอยู่ในขั้นตอนเริ่มต้นในหัวข้ออุปกรณ์และวิธีการ) และแตกเป็นเมทริกย่อย 3 ส่วน คือ U, S, และ V ดังสมการ (1) โดยเรียกเมทริกย่อย 3 ส่วนนี้ว่าพื้นที่เวกเตอร์ (Vector Space) หรือพื้นที่เชิงความหมาย (Semantic Space) และมีรูปแบบการวัดค่าความคล้าย 3 รูปแบบ คือ วัดค่าความคล้ายระหว่างเวกเตอร์เอกสารกับเวกเตอร์เอกสาร เวกเตอร์คำกับเวกเตอร์คำ และเวกเตอร์เอกสารกับเวกเตอร์คำขอ (Query) โดยทั่วไปจะใช้สมการ Cosine เพื่อหามุมระหว่าง 2 เวกเตอร์ ค่าที่ได้ถ้าเข้าใกล้ 1 แสดงถึงเวกเตอร์ทั้ง 2 คล้ายกันมาก ในทางตรงกันข้ามถ้าค่าเข้าใกล้ 0 แสดงถึงเวกเตอร์ทั้ง 2 คล้ายกันน้อย

$$A_{m\times n} = U_{m\times k} S_{k\times k} V^{T}_{k\times n} \qquad (1)$$

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

โดยที่ $A_{m\times n}$ คือ เมทริกของ Term-Document หรือความถี่ของคำในแต่ละคำตอบ มีขนาด m×n โดย m คือจำนวนแถว และ n คือจำนวนคอลัมน์

$U_{m\times k}$ คือ เมทริกที่คำนวณได้จาก Eigenvectors ของเมทริก AA^T มีขนาด m×k เมื่อ k คือขนาดของเมทริกเชิงความหมาย

$S_{k\times k}$ คือ เมทริกค่าเอกพจน์ (Singular values) แบบ Diagonal ของ A มีขนาด k×k

$V_{k\times n}$ คือ เมทริกที่คำนวณได้จาก Eigenvectors ของเมทริก A^TA มีขนาด k×n

ในงานวิจัยนี้จะใช้สมการ Cosine ในการคำนวณค่าความคล้ายระหว่างเวกเตอร์เอกสารกับเวกเตอร์เอกสารในอัลกอริทึม EM และ Cobweb โดยเวกเตอร์เอกสารหนึ่งเป็นเวกเตอร์ความถี่ของคำเฉลี่ยในกลุ่มคำตอบ และอีกเอกสารหนึ่งเป็นเวกเตอร์ความถี่ของคำในเฉลย ซึ่งจำเป็นต้องทำการแปลงเวกเตอร์ทั้ง 2 ให้อยู่ในรูปแบบเดียวกัน ดังสมการที่ (2)

$$d_{new} = d_{old}^{T} \cdot U \cdot S^{-1} \text{ และ } q_{new} = q_{old}^{T} \cdot U \cdot S^{-1} \qquad (2)$$

โดยที่ d_{old} คือ เวกเตอร์ความถี่ของคำในคำตอบแบบเดิม

d_{new} คือ เวกเตอร์ความถี่ของคำในคำตอบแบบใหม่

q_{old} คือ เวกเตอร์ความถี่ของคำในเฉลยแบบเดิม

q_{new} คือ เวกเตอร์คำเฉลยแบบใหม่

LSA มีการนำมาประยุกต์ใช้ในงานวิจัยต่างๆ เช่น ใช้ในโมเดลการค้นคืนข้อมูล (Du *et al.*, 2008) การจัดกลุ่มซอฟแวร์ (Maletic and Valluri, 1999) และใช้ในระบบการตรวจข้อสอบอัตนัยแบบอัตโนมัติ เพื่อคิดคะแนน (Kanejiya *et al.*, 2003; Loraksa and Peachavanish, 2007; Tuomo *et al.*, 2005) เป็นต้น

1.2 การขยายคำขอ (Query Expansion)

การขยายคำขอ (Xu and Croft, 1996) เป็นการเพิ่มคำหรือเทอมในคำขอของระบบการค้นคืนสารสนเทศที่มีความหมายใกล้เคียงกัน เพื่อปรับปรุงประสิทธิภาพในการค้นคืน โดยทั่วไป

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

วิธีการขยายคำขอมี 2 แบบ คือ การขยายคำขอด้วยมือและแบบอัตโนมัติ (Gauch and Wang, 1997; Gauch *et al*., 1999) สำหรับการขยายคำขอด้วยมือ ผู้ใช้จะเป็นคนตัดสินใจว่า จะเพิ่มคำขอใหม่หรือไม่ โดยเลือกคำขอใหม่จากเอกสารที่ส่งคืนกลับมาจากคำขอเดิม และการขยายคำขอแบบอัตโนมัติ เอกสารลำดับแรกที่ได้จากผลการค้นหาถูกสมมติว่า คำในเอกสารนั้นมีความสัมพันธ์กับคำขอ ดังนั้นคำต่างๆที่ได้จากคำในเอกสารลำดับแรกจะถูกเพิ่มเข้าไปเป็นคำขอใหม่เพื่อปรับปรุงประสิทธิภาพในการค้นคืนเอกสาร โดยผู้ใช้จะไม่เกี่ยวข้องกับกระบวนการเพิ่มหรือปรับเปลี่ยนคำขอนี้ ซึ่งมีงานวิจัยต่างๆทำการเพิ่มประสิทธิภาพการขยายคำขอแบบอัตโนมัติเพื่อให้สามารถนำไปใช้ได้จริงมากขึ้น โดยการเพิ่มแหล่งข้อมูลภายนอกสำหรับการขยายคำขอ เช่น การใช้พจนานุกรม (Sadat *et al*., 2001) และ การใช้ Search engine (Yin *et al*., 2009) เป็นต้น

2. ทฤษฎีทางด้านปัญญาประดิษฐ์ (Artificial Intelligence)

ทฤษฎีทางด้านปัญญาประดิษฐ์ เป็นทฤษฎีที่จำลองกระบวนการเรียนรู้ การตัดสินใจ และการแก้ปัญหาของมนุษย์ ยกตัวอย่างเช่น การนำชุดข้อมูลสอนระบบป้อนให้เครื่องจักรเรียนรู้ (Machine Learning) ซึ่งเป็นศาสตร์ด้านหนึ่งในปัญญาประดิษฐ์ เป็นต้น โดยการเรียนรู้ของเครื่องจักรสามารถแบ่งออกเป็น 2 แบบ แบบแรกคือ การเรียนรู้แบบมีผู้สอน (Supervised Learning) เป็นการเรียนรู้จากชุดข้อมูลสอนระบบแบบมีเฉลยที่มนุษย์เป็นผู้จัดเตรียมไว้ (Training set) เพื่อหารูปแบบเฉพาะหรือเอกลักษณ์เฉพาะในชุดข้อมูลสอนระบบนั้นๆ เปรียบเสมือนการเรียนรู้ของมนุษย์ที่มีครูคอยแนะนำหรือป้อนความรู้ และใช้ชุดข้อมูลทดสอบ (Testing set) วัดประสิทธิผลการเรียนรู้ของอัลกอริทึม จุดประสงค์ของการเรียนรู้แบบนี้คือใช้ในการพยากรณ์หรือจำแนกประเภทข้อมูล (Classification) ต่างๆเฉพาะเรื่องใดเรื่องหนึ่งเพื่อหารูปแบบเฉพาะในข้อมูลแต่ละประเภท แบบที่ 2 คือ การเรียนรู้แบบไม่มีผู้สอน (Unsupervised Learning) เป็นการเรียนรู้แบบไม่มีผู้แนะนำหรือไม่มีชุดข้อมูลสอนระบบ มีจุดประสงค์เพื่อใช้จัดหมวดหมู่หรือแบ่งกลุ่มข้อมูล (Clustering) ตามความคล้ายของรูปแบบ โครงสร้างและลักษณะต่างๆที่ได้จากข้อมูลที่นำมาทดสอบ อัลกอริทึมที่นำใช้ในงานวิจัยมีดังนี้

2.1 อัลกอริทึมการเรียนรู้แบบมีผู้สอน (Supervised Learning Algorithm)

2.1.1 อัลกอริทึม K-Nearest Neighbor (K-NN) (Cover and Hart, 1967) เป็นอัลกอริทึมที่ใช้สำหรับจำแนกข้อมูลตามลักษณะความคล้ายของข้อมูลในชุดข้อมูลสอนระบบ โดยจะเริ่ม

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

คำนวณค่าความคล้ายใหม่ทุกครั้งเมื่อป้อนข้อมูลที่ต้องการจำแนก ดังนั้นจึงไม่มีการสร้างโมเดลการจำแนกไว้ก่อน เรียกการเรียนรู้แบบนี้ว่า การเรียนรู้แบบเกียจคร้าน (Lazy learning)

วิธีการของอัลกอริทึม K-NN จะคำนวณค่าความคล้ายระหว่างข้อมูลที่ต้องการทำนายกับชุดข้อมูลสอนระบบทั้งหมด และจัดเรียงชุดข้อมูลสอนระบบที่คล้ายกับข้อมูลที่ต้องการทำนายมากที่สุดจำนวน K ข้อมูล และทำนายผลลัพธ์ด้วยวิธีการโหวตจากเสียงข้างมาก (Majority vote) จากตัวแปรคลาสใน K ข้อมูล โดยที่ค่า K ควรกำหนดให้เป็นเลขคี่เพื่อหลีกเลี่ยงกรณีที่ผลโหวตเสียงเท่ากัน และจำเป็นต้องทำการทดลองเพื่อหาค่า K ที่เหมาะสม สำหรับชุดข้อมูลที่นำมาทดสอบจะแทนในรูปแบบตัวเลขเพื่อให้สามารถนำไปคำนวณค่าความคล้ายได้ เช่น แทนเป็นระยะทางของถนน (Shin *et al.*, 2010) และแทนรูปเวกเตอร์ความถี่ของคำ (Bin *et al.*, 2008) เป็นต้น ตัวอย่างสูตรที่นิยมใช้วัดค่าความคล้ายคือ สมการ Euclidean distance ดังสมการ (3) และ สมการ Cosine ดังสมการ (4)

$$\mathrm{d}(\mathbf{p},\mathbf{q}) = \sqrt{(p_1-q_1)^2+(p_2-q_2)^2+\cdots+(p_n-q_n)^2} = \sqrt{\sum_{i=1}^{n}(p_i-q_i)^2} \qquad (3)$$

โดยที่ $d(p,q)$ คือ ฟังค์ชันของค่าความคล้ายของ p และ q

n คือ จำนวนคุณลักษณะ (Attribute) ทั้งหมด

p_i คือ ค่าคุณลักษณะที่ i ของ p

q_i คือ ค่าคุณลักษณะที่ i ของ q

$$S(d_i,d_j) = \cos(\alpha) = \frac{\sum_{k=1}^{m} w_{ki}\cdot w_{kj}}{\sqrt{\sum_{k=1}^{m} w_{ki}^2}\cdot\sqrt{\sum_{k=1}^{m} w_{kj}^2}} \qquad (4)$$

โดยที่ $S(d_i,d_j)$ คือ ฟังค์ชันของค่าความคล้ายของ d_i และ d_j

w_{ki} คือ ค่าน้ำหนักในคุณลักษณะที่ k ของ i

w_{kj} คือ ค่าน้ำหนักในคุณลักษณะที่ k ของ j

อัลกอริทึม K-NN ถูกนำมาประยุกต์ใช้ในงานวิจัยต่างๆ เช่น ใช้ในการจัดกลุ่มข้อความ (Text categorization) ในเอกสาร (Baoli *et al.*, 2004; Soucy and Mineau, 2001; Zhang *et al.*, 2009) การหาเส้นทางที่สั้นที่สุดในการเดินทางบนฐานข้อมูลโครงข่ายถนน (Shin *et al.*, 2010) และระบบการตรวจข้อสอบอัตนัยด้วย K-NN (Bin *et al.*, 2008) เป็นต้น

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

2.1.2 อัลกอริทึม Bayesian เป็นโมเดลทางสถิติ ซึ่งเป็นทฤษฎีที่เกี่ยวข้องกับความน่าจะเป็นของเหตุการณ์ต่างๆ นำมาใช้ในการวิเคราะห์หรือทำนายเหตุการณ์ที่อาจจะเกิดขึ้นในอนาคต สูตรที่นำมาใช้ในการคำนวณดังสมการ (5)

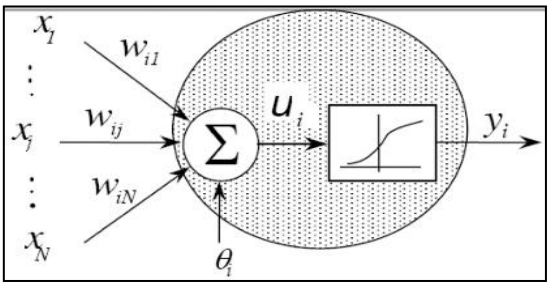
$$P(c \mid x) = \frac{P(x \mid c) \cdot P(c)}{P(x)} \qquad (5)$$

โดยที่ P(c) คือ ความน่าจะเป็นที่จะเกิดเหตุการณ์ c
P(x) คือ ความน่าจะเป็นที่จะเกิดเหตุการณ์ x
P(x|c) คือ ความน่าจะเป็นที่เกิดเหตุการณ์ x เมื่อ c เกิดขึ้นแล้ว
P(c|x) คือ ความน่าจะเป็นที่เกิดเหตุการณ์ c เมื่อ x เกิดขึ้นแล้ว

จุดเด่นของอัลกอริทึม Bayesian คือ เป็นโมเดลที่ใช้กับสถานการณ์ที่มีความไม่แน่นอนและมีความเป็นเหตุและผลที่สอดคล้องกัน โดยสามารถแสดงเป็นรูปแบบภาพของโครงข่าย ซึ่งแทนสถานการณ์ต่างๆด้วยรูปวงรี เรียกว่าโหนด และความสัมพันธ์ด้วยลูกศรชี้ โดยลูกศรจะชี้จากสถานการณ์หนึ่งไปอีกสถานการณ์หนึ่งที่ได้รับผลโดยตรง และมีตารางเก็บค่าความน่าจะเป็นของเหตุการณ์ต่างๆที่เรียกว่า ตารางความน่าจะเป็นแบบมีเงื่อนไข (Conditional Probability Table (CPT) หรือ Probability Distribution Table) โดยโครงข่ายของอัลกอริทึม Bayesian อาจจะเป็นแบบ Naïve Bayes ก็ได้คือทุกโหนดเป็นอิสระต่อกัน หรือแบบ Bayesian Network ที่แต่ละโหนดมีความสัมพันธ์กัน เป็นต้น ด้วยเหตุนี้อัลกอริทึม Bayesian ถูกนำมาประยุกต์ใช้อย่างกว้างขวาง เช่น นำมาใช้ในการทำนายคุณภาพและข้อผิดพลาดของซอฟแวร์ด้วยอัลกอริทึม Bayesian Network (Amasaki *et al.*, 2003; Fenton *et al.*, 2008; Stefan, 2009) ใช้พยากรณ์อากาศด้วยอัลกอริทึม Bayesian Network (อุษานาฎ และคณะ, 2552) อีกทั้งถูกนำมาประยุกต์ใช้ในระบบการตรวจข้อสอบอัตนัยแบบอัตโนมัติด้วยอัลกอริทึม Bayesian (Rudner and Liang, 2002) เป็นต้น

2.1.3 อัลกอริทึม Artificial Neural Network (ANN) หรือเครือข่ายประสาทเทียม เป็นแบบจำลองการทำงานของระบบประสาทส่วนกลางในสมองมนุษย์ที่มีโครงสร้างเป็นลักษณะของเครือข่ายที่เชื่อมโยงกันระหว่างหน่วยประมวลผล เรียกแต่ละหน่วยที่เชื่อมโยงกันว่า โหนด (Node) ซึ่งเป็นได้ทั้งโหนดรับเข้า (Input node) โหนดซ่อน (Hidden node) และโหนดส่งออก (Output node) โดยแต่ละโหนดสามารถที่จะรับรู้ข้อมูลและปรับตัวเข้ากับสถานการณ์หรือสิ่งแวดล้อมที่เผชิญอยู่ มีวัตถุประสงค์เพื่อใช้ในการจดจำรูปแบบ และทำนายผลที่อาจจะเกิดขึ้นในอนาคต ตัวอย่าง ANN อย่างง่าย คือ Perceptron มีโครงสร้างดังภาพที่ 1

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์



ภาพที่ 1 โครงสร้าง Perceptron

โดยที่ $x = [x_1, x_2, ..., x_N]^T$ คือ โหนดรับเข้า

$w = [w_{i1}, w_{i2}, ..., w_{iN}]^T$ คือ ค่าน้ำหนักของเส้นเชื่อมในแต่ละโหนดรับเข้า

θ_i คือ ฟังค์ชันการรวมค่าของผลรวมของโหนดรับเข้าคูณกับค่าน้ำหนักของเส้นเชื่อมในแต่ละโหนดรับเข้า

u_i คือ ค่าผลรวมที่ได้จาก θ_i

y_i คือ ค่าผลลัพธ์

โครงสร้าง Perceptron ในภาพที่ 1 ประกอบด้วยชุดของโหนด ซึ่งเป็นได้ทั้งโหนดรับเข้า x ที่ได้จากค่าคุณลักษณะของข้อมูลต่างๆหรือเซนเซอร์ที่วัดค่าต่างๆ และโหนดส่งออกหรือโหนดผลลัพธ์ y_i โดยกำหนดค่าน้ำหนัก w กำกับอยู่ในเส้นเชื่อมทุกเส้นแบบสุ่ม ในตอนเริ่มต้นทำงานจะกำหนดค่าให้กับโหนดรับเข้า ซึ่งค่าเหล่านี้ได้มาจากชุดข้อมูลสอนระบบที่มนุษย์เตรียมไว้ จากนั้นโหนดรับเข้าจะส่งค่าที่ได้รับไปตามเส้นเชื่อมขาออก โดยค่าที่ส่งออกไปจะนำไปคูณกับค่าน้ำหนักของเส้นเชื่อม โหนดในชั้นถัดไปจะรับค่าที่ได้จากผลรวมจากโหนดต่างๆแล้วนำไปคำนวณผลลัพธ์ด้วยฟังค์ชั่นกระตุ้น (Activation function) เพื่อทำให้ผลลัพธ์มีอยู่ในบรรทัดฐานเดียวกัน โดยทั่วไปฟังก์ชันกระตุ้นที่นิยมใช้ คือฟังก์ชันซิกมอยด์ (Sigmoid function) สูตรดังสมการ (6) และส่งค่าที่ได้จากการคำนวณไปยังโหนดส่งออก y_i ซึ่งเป็นผลลัพธ์การทำนายในอัลกอริทึมนี้ หลังจากนั้นจะวัดประสิทธิผล ถ้าประสิทธิผลที่ได้ยังไม่เป็นที่ต้องการหรือยังมีความผิดพลาด (Error) เกินกว่าที่ยอมรับได้ จะเริ่มกระบวนการใหม่โดยทำการปรับค่าน้ำหนักใหม่ดังสมการที่ (7) และ (8) จากนั้นทำกระบวนการเดิมซ้ำจนกว่าจะได้ค่าน้ำหนักที่เหมาะสม หรือไม่เปลี่ยนแปลง

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

$$y_i = g(u_i) = \frac{1}{1+\exp^{-\beta u_i}} \qquad (6)$$

โดยที่ β คือ ค่าความชัน

$$w_{ij}^{new} = w_{ij}^{old} + LR \bullet e_j \bullet x_i \qquad (7)$$

$$e_j = y_i \bullet (1-y_i) \bullet (d_j - y_i) \qquad (8)$$

โดยที่ w_{ij}^{old} คือ ค่าน้ำหนักของเส้นเชื่อมรอบก่อนหน้า

w_{ij}^{new} คือ ค่าน้ำหนักของเส้นเชื่อมรอบใหม่

LR คือ อัตราการเรียนรู้

d_j คือ ค่าตัวแปรคลาส

ตัวอย่างในการนำ ANN มาประยุกต์ใช้ในงานด้านต่างๆ เช่น การจดจำและวิเคราะห์เสียงพูด (Botros *et al.*, 1993) การตรวจจับการปลอมแปลงภาพ (Gopi *et al.*, 2006) และระบบการตรวจข้อสอบอัตนัยแบบอัตโนมัติ (Loraksa and Peachavanish, 2007) เป็นต้น

2.2 อัลกอริทึมการเรียนรู้แบบไม่มีผู้สอน (Unsupervised Learning Algorithm)

2.2.1 อัลกอริทึม Expectation Maximization (EM algorithm) เป็นกระบวนการวนซ้ำเพื่อปรับค่าตัวแปรหรือพารามิเตอร์ในแต่ละรอบ ทำจนกระทั่งได้ค่า log likelihood ของโมเดลความน่าจะเป็น (Gaussian mixture Model) มีค่าใกล้พอกับค่า log likelihood ของโมเดลรอบปัจจุบันบนชุดข้อมูลที่มีตัวแปรสูญหาย (Missing values) หรือตัวแปรที่สังเกตไม่ได้อยู่ (Dellaert, 2002; Dempster *et al.*, 1977) การทดลองนี้ใช้อัลกอริทึม EM ในโปรแกรม Weka มาประยุกต์ใช้ในการจัดกลุ่มคำตอบ โดยสามารถกำหนดจำนวนกลุ่มแบบอัตโนมัติด้วยการทดสอบ cross validation มีขั้นตอนดังนี้

ขั้นตอนที่ 1: กำหนดจำนวนกลุ่มเป็น 1

ขั้นตอนที่ 2: แบ่งชุดคำตอบแบบสุ่มเป็น 10 folds

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

ขั้นตอนที่ 3: คำนวณค่า log likelihood ดังสมการที่ (15) ในแต่ละ folds

ขั้นตอนที่ 4: หาค่า log likelihood เฉลี่ยใน 10 folds

ขั้นตอนที่ 5: ถ้าค่า log likelihood เฉลี่ยเพิ่มขึ้น จำนวนกลุ่มจะเพิ่มขึ้น 1 แล้วทำขั้นตอนที่ 2 ต่อ

เมื่อได้จำนวนกลุ่มที่ต้องการแบ่งกลุ่มคำตอบแล้ว จะทำการสุ่มตัวแทนจากชุดทดสอบเพื่อเป็นตัวแทนในแต่ละกลุ่มเป็นตัวตั้งต้น และเข้าสู่ขั้นตอนของอัลกอริทึม EM เพื่อจัดคำตอบที่เหลือเข้ากลุ่มด้วยความน่าจะเป็น โดยอัลกอริทึม EM มีขั้นตอนการทำงาน 2 ขั้นตอนดังนี้

1) E-step เป็นขั้นตอนการประเมินค่าความคาดหวังของคำตอบที่อยู่ในแต่กลุ่มของโมเดลรอบปัจจุบัน มีสูตรการคำนวณดังสมการที่ (9) – (11)

$$\Pr(c \mid x) = \frac{W_c^j \cdot \Pr^j(x \mid c)}{\Pr^j(x)}, c = 1,2,...,k \qquad (9)$$

$$\Pr^j(x \mid c) = \frac{1}{\sqrt{(2\pi)^d \left|\Sigma^c\right|}} \cdot \exp\left(\frac{\left(x - m^c\right)^T \cdot \left(x - m^c\right)}{2 \cdot \Sigma^c} \right) \qquad (10)$$

$$\Pr^j(x) = \sum_{c=1}^{k} W_c^j \cdot \Pr(x \mid c) \qquad (11)$$

โดยที่ $\Pr(c|x)$ คือ ค่าความน่าจะเป็นของกลุ่มที่ c เมื่อ x เกิดขึ้น

$\Pr^j(x|c)$ คือ ค่าความน่าจะเป็นของ x เมื่อกลุ่มที่ c เกิดขึ้นในรอบที่ j

$\Pr^j(x)$ คือ ค่าความน่าจะเป็นของ x ในรอบที่ j

W_c^j คือ ค่าสัมประสิทธิ์ความสอดคล้องของกลุ่ม c

x คือ ค่าตัวแปรคำตอบในแต่ละแถวของชุดคำตอบทั้งหมด หรือ $x \in D$

m คือ ค่าเฉลี่ยค่าตัวแปรคำตอบในชุดคำตอบ

d คือ มิติของ Gaussian probability distribution

$\sum^c$ คือ ค่าความแปรปรวนของกลุ่ม c

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

2) M-step เป็นขั้นตอนการปรับปรุงค่าพารามิเตอร์ต่างๆของโมเดล และตรวจสอบเงื่อนไขการหยุดกระบวนการดังสมการ (12) – (14)

$$W_c^{j+1} = \sum_{x \varepsilon D} \Pr(c \mid x) \qquad (12)$$

$$m^{j+1} = \frac{\sum_{x \varepsilon D} x \cdot \Pr(c \mid x)}{\sum_{x \varepsilon D} \Pr(c \mid x)} \qquad (13)$$

$$\Sigma^j = \frac{\sum_{x \varepsilon D} \Pr(c \mid x) \cdot \left(x - m^{j+1}\right) \cdot \left(x - m^{j+1}\right)^T}{\sum_{x \varepsilon D} \Pr(c \mid x)}, c = 1,2,...,k \qquad (14)$$

$$E^j = \sum_{x \varepsilon D} \log\left(\sum_{c=1}^{k} W^j \cdot \Pr^j(x \mid c)\right) \qquad (15)$$

ตรวจสอบเงื่อนไขถ้า $|E^j - E^{j+1}| \leq \epsilon$ ให้หยุดกระบวนการ ถ้าไม่ใช่ให้ j=j+1และเริ่มกระบวนการซ้ำ โดยที่ E^j คือค่า log likelihood ของโมเดล Gaussian mixture รอบที่ j ดังสมการที่ (15) และ ϵ คือ เกณฑ์ที่ใช้ในการหยุดกระบวนการ

ตัวอย่างการนำอัลกอริทึม EM มาใช้ เช่น นำอัลกอริทึม EM มาประยุกต์ใช้ในการเพิ่มประสิทธิภาพของอัลกอริทึม Naïve Bay ที่ใช้ในการจำแนกข้อความจากเอกสารที่มีจำนวนชุดข้อมูลสอนระบบจำนวนน้อย โดยการรวมข้อมูลที่ไม่ทราบคลาสเข้ากับข้อมูลที่ทราบคลาสด้วยอัลกอริทึม EM เพื่อทำให้ชุดข้อมูลสอนระบบมีจำนวนมากขึ้น ส่งผลให้ผลการจำแนกเอกสารถูกต้องมากขึ้น (Nigam *et al.*, 2000) และใช้อัลกอริทึม EM ในการแก้ปัญหาในกรณีที่ตัวแปรคลาสของชุดข้อมูลมีความไม่ชัดเจนหรืออาจไม่ถูกต้อง (Yuri *et al.*, 2001) เป็นต้น

2.2.2 อัลกอริทึม Cobweb ใช้ในการจัดกลุ่มข้อมูลบนพื้นฐานของการจัดกลุ่มเชิงแนวคิด (Conceptual clustering) กล่าวคือ เป็นการหาคุณลักษณะของกลุ่มด้วยวิธีทางสถิติ เช่น ค่าความน่าจะเป็น และใช้เทคนิคการจัดกลุ่มแบบลำดับชั้น (Hierarchical) ซึ่งผลลัพธ์จะเป็นต้นไม้การจำแนก (classification tree) เริ่มต้นด้วยการใส่ object จากโหนดรากเข้ามาทีละหนึ่ง object และสำรวจจากบนลงล่าง มีวิธีดำเนินการ 4 ขั้นตอนคือ การเพิ่ม การสร้าง การรวม และการแยกโหนด โดยใช้ค่าที่คำนวณได้จากฟังค์ชัน Category utility (Fisher, 1987; Mirkin, 2001) ในสมการที่ (16)

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

และ (17) เพื่อสร้างต้นไม้การจำแนก โดยจะใช้วิธีดำเนินการใดใน 4 วิธีนั้น ขึ้นอยู่กับค่า Category utility ในวิธีดำเนินการแบบใดที่ให้ค่ามากที่สุด

$$CU(C) = \frac{1}{n}\sum_{i} \Delta(C, A_i) \qquad (16)$$

$$\Delta(C,|A_i) = \sum_{k=1}^{n} P(C_k) \sum_{j} P(A_i = V_{ij} \mid C_k)^2 - \sum_{j} P(A_i = V_{ij})^2 \qquad (17)$$

โดยที่ CU(C) คือ ฟังค์ชัน category utility ในการแบ่งกลุ่ม C

C_k คือ กลุ่มที่ k

A_i คือ คุณลักษณะ (Attribute) ในชุดข้อมูล

V_{ij} คือ ค่าของแต่ละคุณลักษณะในชุดข้อมูล

n คือ จำนวนกลุ่มในชุดข้อมูล

ตัวอย่างการนำอัลกอริทึม Cobweb มาใช้เช่น จัดกลุ่มเอกสารข้อความ (Sahoo *et al.*, 2006) และจำแนกเอกสารบน (Gregory *et al.*, 2006) เป็นต้น

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

งานวิจัยที่เกี่ยวข้อง

งานวิจัยที่เกี่ยวข้องในงานวิจัยนี้ แยกออกเป็น 3 หัวข้อ คือ งานวิจัยที่ยืนยันการนำไปใช้จริงของระบบการตรวจข้อสอบ งานวิจัยที่ประยุกต์ใช้กับลักษณะข้อสอบแบบต่างๆ ด้วยอัลกอริทึมทางด้านการเรียนรู้ด้วยเครื่อง (Machine Learning) ในระบบการตรวจข้อสอบ และงานวิจัยที่พบในประเทศไทยของระบบการตรวจข้อสอบ มีรายละเอียดดังนี้

1. งานวิจัยที่ยืนยันการนำไปใช้จริงของระบบการตรวจข้อสอบ

ระบบการตรวจข้อสอบอัตนัยแบบอัตโนมัติถูกสร้างและใช้เป็นเครื่องมือสำหรับมนุษย์มาหลายปีแล้ว จากงานวิจัยต่างๆได้ยืนยันผลการตรวจข้อสอบด้วยเครื่องกับผลการตรวจข้อสอบด้วยมนุษย์ โดยวัดด้วยค่าสหสัมพันธ์ (Correlation) ซึ่งเป็นค่าการเปรียบเทียบระหว่างผลลัพธ์ที่ได้จากคอมพิวเตอร์ตรวจกับมนุษย์ผู้เชี่ยวชาญตรวจได้ผลที่สอดคล้องกัน (Landauer *et al.*, 1998; Palmer *et al*., 2002; Semire, 2006; Valenti *et al.*, 2003) และได้มีการจดสิทธิบัตรในประเทศอเมริกาแล้ว (Andreyev *et al.*, 2005; Burstein *et al*., 2000)

2. งานวิจัยที่ประยุกต์ใช้กับลักษณะข้อสอบแบบต่างๆ ด้วยอัลกอริทึมทางด้านการเรียนรู้ด้วยเครื่อง (Machine Learning) ในระบบการตรวจข้อสอบ

ระบบการตรวจข้อสอบอัตนัยแบบอัตโนมัติถูกนำมาใช้ในการตรวจข้อสอบตามลักษณะข้อสอบแบบต่างๆ ดังนี้

2.1 ระบบการตรวจข้อสอบที่ใช้ประเมินคุณภาพในการเขียน

ระบบการตรวจข้อสอบที่ใช้ประเมินคุณภาพในการเขียนของผู้ตอบ มีวัตถุประสงค์เพื่อประเมินแนวความคิด ความมีเหตุมีผลของผู้ตอบ ซึ่งไม่เน้นความถูกต้องของเนื้อหาในคำตอบ ลักษณะของคำตอบแบบนี้ส่วนมากจะมีความยาวของคำตอบค่อนข้างมาก ทำให้มีความหลากหลายของคำตอบมากตามไปด้วย ดังนั้นในการตรวจคำตอบลักษณะนี้ จึงจำเป็นต้องจัดเตรียมชุดข้อมูลสอนระบบปริมาณมาก เพื่อให้สอดคล้องกับลักษณะของคำตอบที่มีความหลากหลาย อัลกอริทึมการเรียนรู้แบบมีผู้สอนจึงถูกหยิบขึ้นมาทดลองใช้ในระบบการตรวจข้อสอบในงานวิจัยต่างๆ เช่น อัลกอริทึม Bayesian (Rudner and Liang, 2002) ประยุกต์ใช้ในข้อสอบวิชาชีววิทยา ให้คะแนน

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

คำตอบด้วยมนุษย์ 2 คน และนำเฉพาะคำตอบที่มนุษย์ทั้ง 2 ให้คะแนนตรงกันมาใช้เป็นชุดข้อมูลสอนและทดสอบระบบ โดยจำนวนคำตอบที่นำมาใช้ในระบบมีทั้งหมด 542 คำตอบ แบ่งเป็นชุดข้อมูลสอนระบบจำนวน 462 ชุด และชุดทดสอบระบบจำนวน 80 ชุด มีจำนวนคำเอกลักษณ์ (Unique words) 1,208 คำ และวลีเอกลักษณ์ (Unique phrases) จำนวน 8,326 วลี และลดขนาดของคำเอกลักษณ์และวลีเอกลักษณ์โดยการตัดคำและวลีที่มีความถี่น้อยกว่า 20 คำออก ได้ค่าความแม่นยำมากสุดประมาณ 80% อัลกอริทึม Artificial Neural Network (Sargur *et al.*, 2008) ประยุกต์ใช้กับข้อสอบที่ทดสอบความเข้าใจในการอ่านของผู้ตอบ (Reading comprehension) โดยรับข้อมูลมาเป็นภาพและแปลงมาเป็นตัวอักษรภาษาอังกฤษ ใช้ชุดข้อมูลสอนระบบจำนวน 150 คำตอบ และชุดข้อมูลทดสอบระบบจำนวน 150 คำตอบ มีระดับคะแนน 5 ระดับ และอัลกอริทึม K-NN (Bin *et al.*, 2008) ประยุกต์ใช้กับข้อสอบ CET4 ใน Chinese Learner English Corpus (CLEC) หัวเรื่อง “Global shortage of Fresh Water” ซึ่งเป็นข้อสอบทดสอบการเขียนภาษาอังกฤษของคนจีนตามหัวเรื่องที่กำหนด ชุดคำตอบที่นำมาทดลองมีทั้งหมด 271 ชุด โดยแบ่งเป็นชุดข้อมูลสอนระบบจำนวน 221 ชุด และชุดข้อมูลทดสอบจำนวน 50 ชุด ผลการทลองได้ค่าความแม่นยำประมาณ 76% เป็นต้น

อัลกอริทึมการเรียนรู้แบบมีผู้สอนมีข้อจำกัดในเรื่องของจัดเตรียมชุดข้อมูลสอนระบบเพื่อสร้างแบบจำลองการให้คะแนนคำตอบขนาดใหญ่ ซึ่งใช้เวลามากและไม่ยืดหยุ่นในการนำไปใช้งานจริง ดังนั้น Chen *et al.* (2010) ได้นำเสนอระแบบการตรวจข้อสอบด้วยอัลกอริทึมการเรียนรู้แบบไม่มีผู้สอนขึ้นกับอัลกอริทึม Voting โดยใช้หลักการแบ่งกลุ่มคำตอบด้วยอัลกอริทึม K-means ซึ่งเป็นอัลกอริทึมการแบ่งกลุ่มที่ใช้การเรียนรู้แบบ Unsupervised คือไม่ต้องเตรียมชุดข้อมูลสอนระบบ โดยใช้แบ่งกลุ่มคำตอบตามความคล้ายของคำตอบ ในชุดคำตอบแบบคำถามปลายเปิดจำนวน 689 คำตอบ แบ่งระดับคะแนนเป็น 6 ระดับ ในหัวข้อ “Recess at School” โดยกำหนดคะแนนเริ่มต้น (Initial score) ตามจำนวนคำเอกลักษณ์ (Unique word) และความสัมพันธ์ของคำในคำตอบ และทำขั้นตอนวนซ้ำโดยคะแนนของแต่ละคำตอบถูกโหวตโดยคำตอบอื่นๆ ในขั้นตอนนี้คะแนนจะถูกคำนวณใหม่บนพื้นฐานของค่าความคล้ายของคำที่ใช้ร่วมกันในแต่ละคำตอบกับคะแนนก่อนหน้า ทำซ้ำจนคะแนนในแต่ละคำตอบไม่มีการเปลี่ยนแปลง

งานวิจัยที่กล่าวมาข้างต้น ยังขาดส่วนของการวิเคราะห์ความสัมพันธ์ระหว่างชุดคำตอบกับคำในคำตอบที่เป็นชุดของแนวคิดที่สัมพันธ์กับคำตอบ ดังนั้นระบบ E-rather (Attali and Burstein, 2006) และเทคนิค LSA ถูกนำมาประยุกต์ใช้ในระบบการตรวจข้อสอบเพื่อแก้ปัญหานี้ (Tuomo and Sutinen, 2003) โดยระบบ E-rater ถูกนำมาใช้ใน Educational Testing Services (ETS)

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

มหาวิทยาลัย Princeton เพื่อตรวจข้อสอบ Graduate Management Admission Test ® Analytical Writing Assessment (GMAT ® AWA) ตั้งแต่ปี ค.ศ. 1999 ในเวอร์ชันแรกทดลองบนชุดคำตอบที่ผ่านการตรวจให้คะแนนโดยมนุษย์เพื่อใช้เป็นชุดข้อมูลสอนระบบในการสร้างโมเดล Multiple linear regression สำหรับการให้คะแนนคำตอบ ประกอบด้วย 2 แอพพลิเคชั่นหลัก คือ E-rater engine และ ส่วนของการป้อนข้อมูลคำแนะนำหรือบทวิจารณ์ในการเขียนคำตอบกลับไปยังผู้ตอบ ต่อมาได้ทำการปรับปรุงต่อจากเวอร์ชันแรก โดยการใช้เทคนิคทางสถิติและการประมวลผลภาษาธรรมชาติ (Neural Language Processing (NLP)) เพื่อใช้ในการสกัดลักษณะของคำตอบ เช่น เนื้อหาของคำตอบ ข้อมูลทางโครางสร้าง และข้อมูลเชิงแนวคิด เป็นต้น เพื่อคิดคะแนนคำตอบ ทำให้ได้ประสิทธิที่ผลที่ดีมากขึ้น สำหรับเทคนิค LSA ใช้ในการลดขนาดของชุดข้อมูลและหาความสัมพันธ์ของคำในชุดคำตอบ ทำให้การให้คะแนนหรือจำแนกคำตอบถูกต้องมากขึ้น แต่เทคนิค LSA ยังมีข้อด้อยหลายประการ เช่น ขาดการวิเคราะห์ลำดับของคำหรือโครงสร้างของประโยคและการกำหนดขนาดของเมทริกที่ต้องการลด ดังนั้น เทคนิค SELSA ถูกนำเสนอในระบบการตรวจข้อสอบ (Kanejiya *et al.*, 2003) เพื่อแก้ปัญหาของการวิเคราะห์ลำดับของคำหรือโครงสร้างของประโยค โดยการเพิ่มส่วนของ POS tag เข้าไปหน้าคำในขั้นตอนเริ่มต้น (Preprocessing) เพื่อใช้ข้อมูลโครงสร้างคำในประโยค ตัวอย่าง POS tag ดังตารางที่ 1 และเทคนิค PLSA ถูกนำเสนอในระบบการตรวจข้อสอบ (Tuomo *et al.*, 2005) เพื่อแก้ปัญหาการกำหนดขนาดของเมทริกที่เหมาะสมสำหรับการลดขนาด เทคนิคนี้จะใช้อัลกอริทึม EM ในการสร้างแบบจำลองเพื่อค้นหาขนาดของเมทริกที่เหมาะสม และกำหนดการแจกแจงความน่าจะเป็นที่เหมาะสมในเอกสารบนพื้นฐานทางสถิติ โดยที่เทคนิคนี้ให้ประสิทธิผลเท่ากับหรือดีกว่าเมื่อจำนวนตัวแปรแฝง (Latent variable) มีมากขึ้น

ตารางที่ 1 ตัวอย่างแท็ก POS

คำ	**แท็ก POS**
This	PRON
is	V
an	ART
artificial	ADJ
test	N
example	N
.	PUNC

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

2.2 ลักษณะข้อสอบที่ใช้ประเมินความถูกต้องของเนื้อหา

ระบบการตรวจข้อสอบที่ตรวจความถูกต้องของคำตอบ คำตอบลักษณะนี้จะมีความยาวของคำตอบแบบสั้น หรืออธิบายสั้นๆและเป็นคำตอบที่ชัดเจน ดังนั้นจึงมีรูปแบบของการตอบค่อนข้างน้อย ตัวอย่างงานวิจัย เช่น ระบบการตรวจข้อสอบโดยใช้เทคนิคการจัดกลุ่มข้อความ (Larkey, 1998) ในงานวิจัยนี้ ใช้อัลกอริทึม Bayesian สำหรับแบ่งกลุ่มคำตอบออกเป็น 2 กลุ่ม คือ กลุ่มถูก และกลุ่มผิด และใช้แบบจำลอง Linear regression ในการให้คะแนนคำตอบ ทดลองบนชุดข้อสอบวิชา สังคม ฟิสิกส์ และกฎหมาย ซึ่งต้องการคำตอบที่ชัดเจน เป็นต้น

3. งานวิจัยที่พบในประเทศไทยของระบบการตรวจข้อสอบ

งานวิจัยแรกที่พบในประเทศไทยในระบบการตรวจข้อสอบด้วยภาษาไทยถูกนำเสนอในปี ค.ศ. 2007 โดยใช้ อัลกอริทึม Backpropagation Neural Network (BNN) และเทคนิค LSA (Loraksa and Peachavanish, 2007) ทดลองบนชุดข้อสอบวิชาประวัติศาสตร์ไทย จำนวน 40 คำตอบ ความยาวคำตอบแบบสั้นเพื่อใช้ประเมินคุณภาพการเขียนคำตอบ ซึ่งให้คะแนนแต่ละคำตอบโดยมนุษย์ 3 คนและเฉลี่ยคะแนนคำตอบ การเลือกลักษณะคำ (Feature section) อยู่บนพื้นฐานของการตัดคำและการถ่วงน้ำหนัก IDF ซึ่งยังไม่มีการวิเคราะห์เกี่ยวกับลักษณะของชุดข้อมูลสอนระบบ เป็นเพียงการพิสูจน์ประสิทธิผลว่า การนำอัลกอริทึม BNN ร่วมกับเทคนิค LSA ได้ผลที่ดีกว่าการนำอัลกอริทึม BNN มาใช้เพียงอย่างเดียว

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

อุปกรณ์และวิธีการ

งานวิจัยฉบับนี้ ทำการทดลองหาวิธีในการตรวจให้คะแนนชุดคำตอบภาษาไทยแบบอัตโนมัติความยาวคำตอบไม่เกิน 3-4 บรรทัด โดยมีอุปกรณ์และวิธีการวิจัยดังต่อไปนี้

อุปกรณ์

1. เครื่องคอมพิวเตอร์ส่วนบุคคล หน่วยประมวลผล AMD Althlon(tm) 64 X2 Dual Core Processor 3800+ 2.00 GHz, 3.00 GB of RAM Physical Address Extension

2. โปรแกรม Weka เวอร์ชัน 3.5.7

3. โปรแกรม Eclipse Version 3.3.2

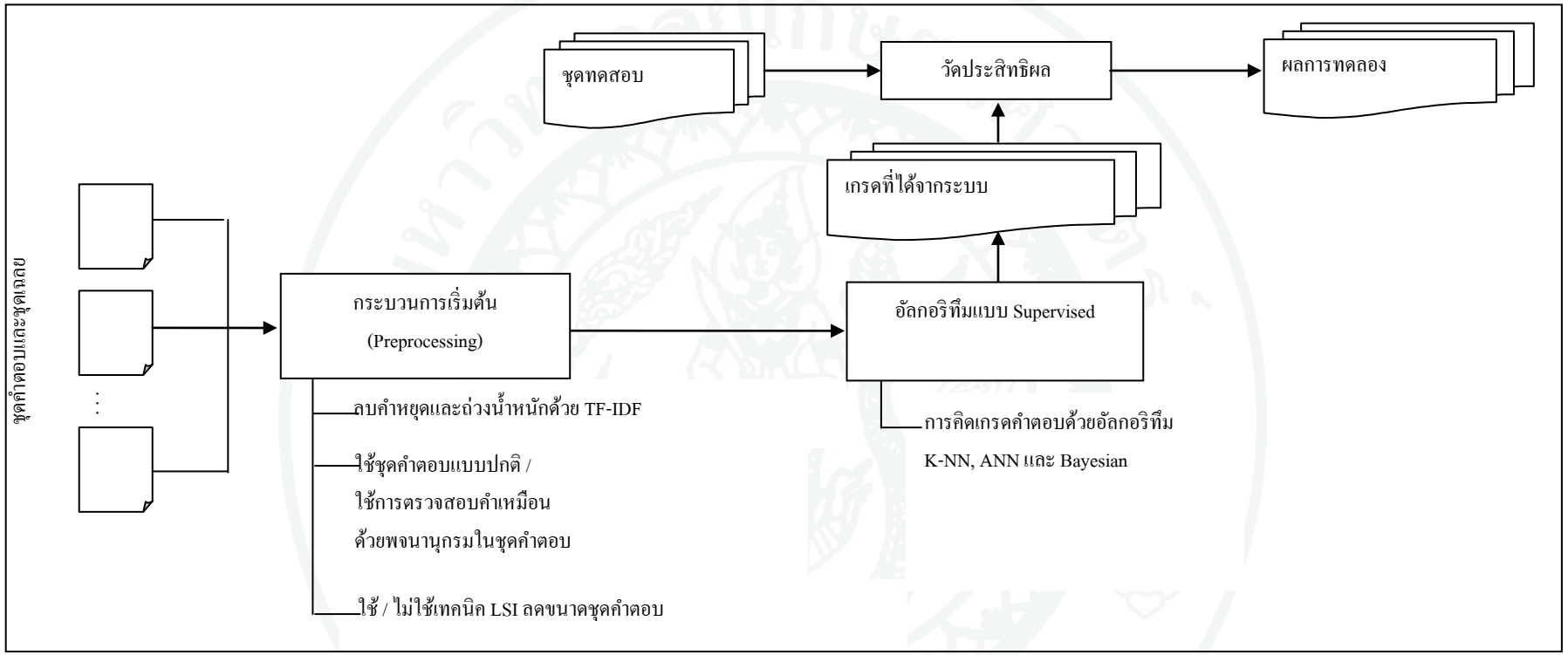
4. โปรแกรม SWATH เพื่อใช้ในการตัดคำภาษาไทย

5. ระบบปฏิบัติการ Microsoft Windows XP Professional

วิธีการ

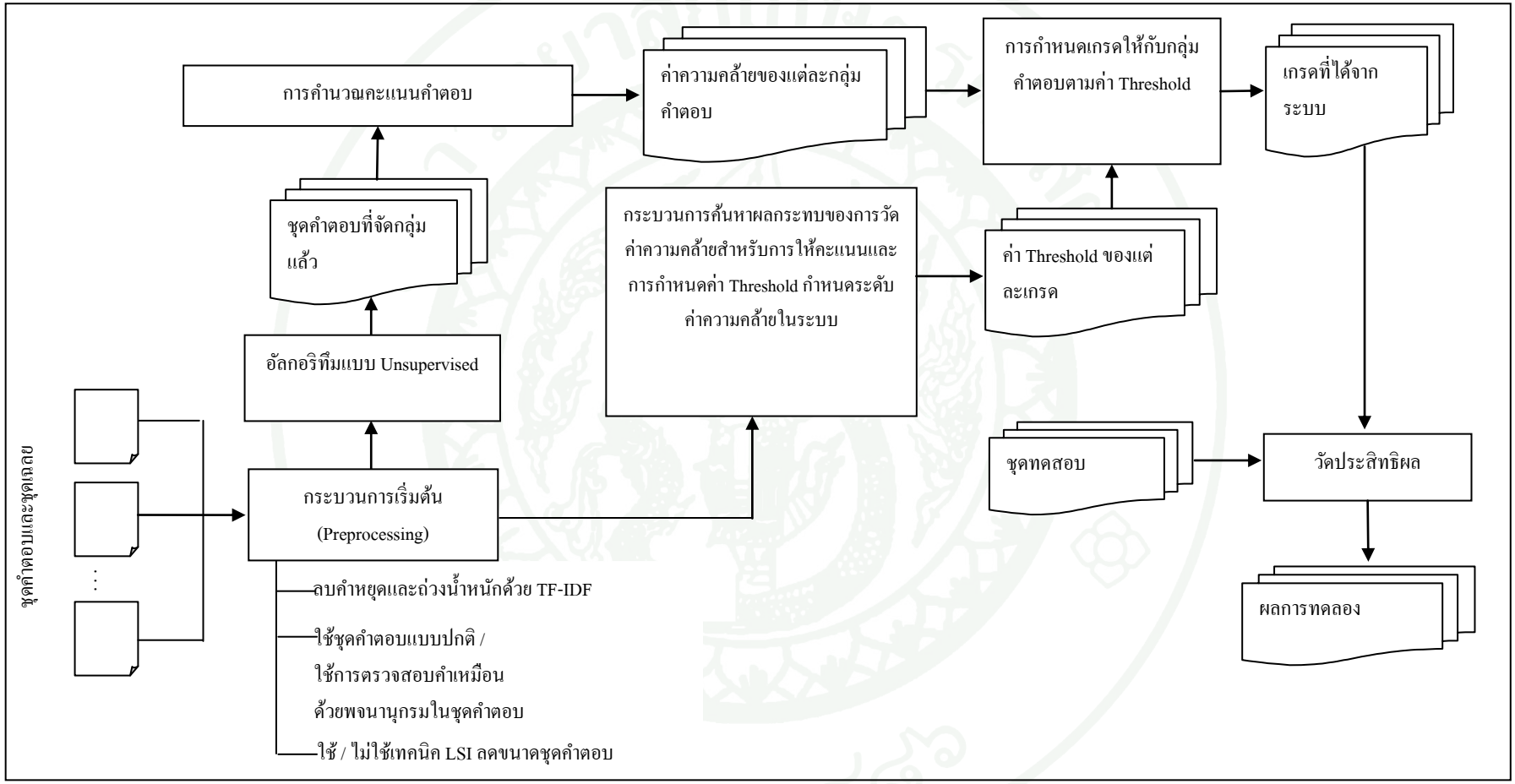
ขั้นตอนในการสร้างระบบการตรวจข้อสอบอัตนัยภาษาไทยแบบอัตโนมัติในงานวิจัยนี้ แบ่งเป็น 2 วิธีการ วิธีการแรกทดลองบนอัลกอริทึมการเรียนรู้แบบมีผู้สอนระบบ และวิธีการที่สองทดลองบนอัลกอริทึมการเรียนรู้แบบไม่มีผู้สอนระบบ ดังภาพที่ 2 และ 3

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์



ภาพที่ 2 แผนภาพขั้นตอนการสร้างระบบการตรวจข้อสอบอัตนัยภาษาไทยแบบอัตโนมัติวิธีการที่ 1

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์



ภาพที่ 3 แผนภาพขั้นตอนการสร้างระบบการตรวจข้อสอบอัตนัยภาษาไทยแบบอัตโนมัติวิธีการที่ 2

จากภาพที่ 2 และ 3 มีรายละเอียดในแต่ละขั้นตอนการสร้างระบบการตรวจข้อสอบอัตนัยภาษาไทยแบบอัตโนมัติในแต่ละวิธีดังนี้

1. ข้อมูลและเครื่องมือที่ใช้ในการทดลอง

ข้อมูลที่ใช้ในการทดลองจะใช้ข้อสอบวิชา “การโปรแกรมเชิงวัตถุด้วยภาษาจาวา” มีชุดเฉลย 1 ชุด และชุดคำตอบทั้งหมดผ่านการตรวจให้เกรดจากผู้ตรวจ โดยแบ่งกลุ่มระดับคะแนนเป็น 5 กลุ่ม คือ A B C D และ F โดยที่เกรด A เป็นกลุ่มที่มีคะแนนความถูกต้องมากสุด ส่วนเกรด B C D และ F มีคะแนนความถูกต้องลดลงมาตามลำดับ คัดเลือกมา 7 ข้อ จาก 17 ข้อ จำนวนคำตอบนักเรียนทั้งหมด 55 คน โดยทำการแบ่งประเภทของคำถามตามหลัก Wh-questions ในไวยากรณ์ภาษาอังกฤษ มีทั้งหมด 7 ประเภท ในแต่ละประเภทมีรายละเอียดดังนี้

1) Who (ใคร) เป็นคำถามที่ต้องการคำตอบเป็นบุคคลและบุคคลนั้นจะต้องทำหน้าที่เป็นประธานของประโยค

2) What (อะไร) เป็นคำถามที่ต้องการคำตอบเป็นสิ่งของ เวลา ความคิด ฯลฯ

3) Which (สิ่งไหน อันไหน) เป็นคำถามที่ต้องการคำตอบที่เฉพาะเจาะจงเพราะจะมีคำตอบให้เลือกอย่างใดอย่างหนึ่ง

4) Why (ทำไม) เป็นคำถามที่ต้องการคำตอบในเชิงให้เหตุผล

5) Where (ที่ไหน) เป็นคำถามที่ต้องการคำตอบเป็นสถานที่

6) When (เมื่อไร) เป็นคำถามที่ต้องการคำตอบเป็น วัน และเวลา

7) How (อย่างไร) เป็นคำถามที่ต้องการคำตอบเป็นอาการ หรือวิถีทาง

ข้อมูลที่ใช้ในการทดลอง จะทำการบันทึกใหม่และจัดเก็บอยู่ในรูปแบบไฟล์ข้อความ (ไฟล์ Text) สถิติของข้อมูลคำตอบ และประเภทคำถาม แสดงดังตารางที่ 2

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

ตารางที่ 2 สถิติของข้อมูล

ข้อ	จำนวนคำเอกลักษณ์ (Unique words)	จำนวนคำสูงสุดในคำตอบนักเรียน	จำนวนคำต่ำสุดในคำตอบนักเรียน	จำนวนคำเฉลี่ยในคำตอบนักเรียน	จำนวนคำในเฉลย	ประเภทคำถาม
1	161	70	2	22.2041	38	What
2	285	78	1	31.8810	43	What
3	217	85	2	23.9756	41	How
4	124	68	10	33.0833	33	How
5	243	78	6	30.9744	16	What
6	173	70	1	14.0408	35	Which
7	199	89	1	28.5500	53	How

จากตารางที่ 2 แสดงจำนวนคำสูงสุด ต่ำสุด จำนวนคำเฉลี่ย และประเภทคำถาม โดยจำนวนคำหาได้จากการนับความถี่ของคำที่ไม่ซ้ำหรือคำเอกลักษณ์ในแต่ละคำตอบ

เครื่องมือที่ใช้ในการทดลองจะใช้ Library ภาษาโปรแกรมจาวาของโปรแกรม Weka เวอร์ชัน 3.5.7 โดยนำมาเฉพาะ Library ของอัลกอริทึมที่ใช้ในการทดลองในงานวิจัยนี้ และในส่วนของเทคนิค LSA จะใช้ Library ภาษาโปรแกรมจาวาของ LingPipe (Alias-i., 2008)

2. กระบวนการเริ่มต้น (Preprocessing)

2.1 การแทนคำตอบและเฉลยด้วย TF-IDF

เริ่มต้นด้วยการแปลงรูปแบบคำตอบในแต่ละข้อให้อยู่ในรูปแบบที่สามารถนำไปประมวลผลในคอมพิวเตอร์ได้ ในการทดลองจะแทนแต่ละคำตอบและเฉลยของแต่ละข้อในรูปแบบของเวกเตอร์ดังสมการ (18)

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

$$d_j = (w_{1j}, w_{2j}, w_{3j}, \ldots, w_{ij}) \qquad (18)$$

โดยที่ d_j คือ คำตอบของนักเรียนคนที่ j

w_{ij} คือ ค่าน้ำหนักของคำที่ i ในคำตอบของนักเรียนคนที่ j ดังสมการ (19)

$$w_{ij} = TF - IDF = TF(i, j) \cdot \log \frac{N}{DF(i)} \qquad (19)$$

โดยที่ TF(i,j) คือ ค่าความถี่ของคำที่ i ในคำตอบของนักเรียนคนที่ j

N คือ จำนวนคำตอบทั้งหมดที่นำมาใช้ในการทดลอง

DF(i) คือ จำนวนคำตอบของนักเรียนที่ปรากฏคำที่ i อยู่

ขั้นตอนในการแปลงรูปแบบคำตอบ ดังนี้

1) แปลงข้อมูลโดยการบันทึกใหม่เพื่อจัดในรูปอิเล็กทรอนิกส์ไฟล์ ตัวอย่างคำตอบของนักเรียน 3 คนดังตารางที่ 3

2) นำไฟล์ที่ได้จากการบันทึก เข้าโปรแกรมตัดคำภาษาไทยแล้วบันทึกไฟล์ใหม่

3) ทำการลดขนาดของเวกเตอร์ โดยการลบคำที่ไม่มีประโยชน์ออก เนื่องจากขนาดของเวกเตอร์มีผลต่อการคำนวณคะแนน และเวลาในการประมวลผล ในการทดลองจะทำการลบคำหยุด (Stop word) เช่น คำว่า “และ”, “หรือ” เป็นต้น

4) เขียนโปรแกรมนับความถี่ของคำที่เหลือในคำตอบ หลังจากการลบคำหยุดออก ตัวอย่างดังตารางที่ 4

5) นำระบบน้ำหนักเข้ามาแปลงค่าความถี่ของคำใหม่ โดยใช้สมการ TF-IDF ดังสมการ (19) ผลที่ได้ดังตารางที่ 5

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

ตารางที่ 3 ตัวอย่างคำตอบของนักเรียน 3 คน

นักเรียนคนที่	คำตอบ
1	1. การสืบทอด หรือการสืบต่อ (abstract) 2. polymorphism
2	สร้าง class ขึ้น โดยสามารถอ้างถึง class นั้นได้โดยไม่ต้องสร้างใหม่
3	ผู้ใช้สามารถสร้างตัวแปรประเภท object ขึ้นมาเองได้ โดย object ที่สร้างขึ้นมามีคุณสมบัติของ object อย่างชัดเจน เช่นมี Attribute method สามารถถ่ายทอดกัน

ตารางที่ 4 การแทนคำตอบในรูปความถี่ของคำ

นักเรียนคนที่	การสืบทอด	การสืบต่อ	abstract	class	polymorphism	object	ตัวแปร
1	1	1	1	0	1	0	0
2	0	0	0	2	0	0	0
3	1	0	0	0	0	2	1

ตารางที่ 5 การแทนคำตอบในรูปความถี่ของคำและถ่วงน้ำหนักโดยใช้ TF-IDF

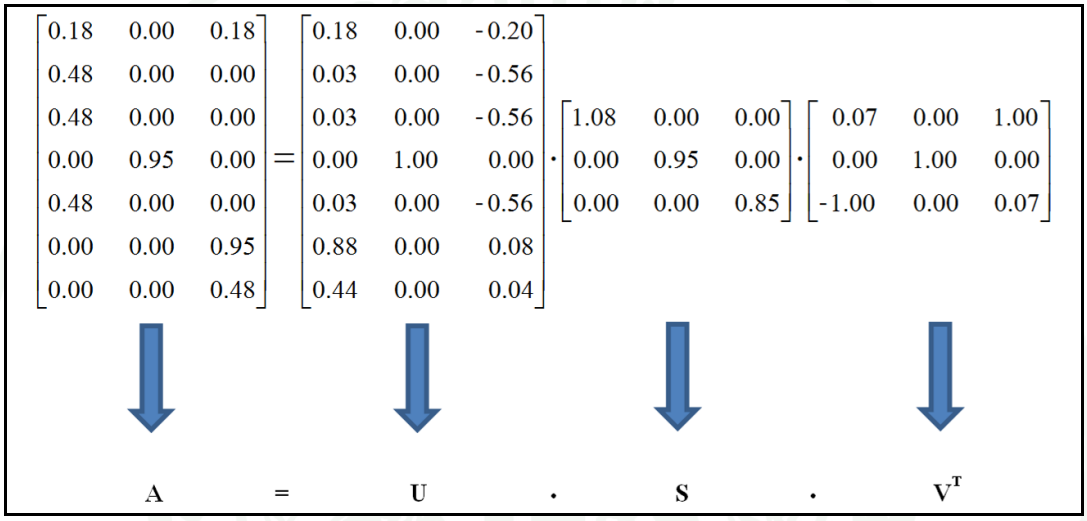
นักเรียนคนที่	การสืบทอด	การสืบต่อ	abstract	class	polymorphism	object	ตัวแปร
1	0.18	0.48	0.48	0.00	0.48	0.00	0.00
2	0.00	0.00	0.00	0.95	0.00	0.00	0.00
3	0.18	0.00	0.00	0.00	0.00	0.95	0.48

ในการเลือกคำตอบที่นำมาใช้ในการทดลอง จะคัดแยกคำตอบที่ผู้ตอบไม่ได้ตอบ และคำตอบที่คำนวณค่าความคล้ายน้อยกว่าหรือเท่ากับ 0.01 ออกก่อน แล้วกำหนดเกรดคำตอบเหล่านี้เป็นเกรด F เนื่องจากเป็นคำตอบที่ไม่สามารถนำไปใช้ประโยชน์ได้ในการจัดกลุ่ม

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

2.2 การแทนคำตอบและเฉลยด้วยเทคนิค LSA

เป็นการนำชุดคำตอบที่ถูกแปลงเป็นค่า TF-IDF ในรูปแบบของเมทริกแล้ว มาทำการแตกเป็นเมทริกย่อย 3 เมทริกดังสมการ (1) ซึ่งเรียกทั้ง 3 ส่วนนี้ว่าพื้นที่เชิงความหมาย (Semantic space) โดยกำหนดขนาดของเมทริก Singular เป็น 3 แสดงดังภาพที่ 4 ซึ่งใช้ตัวอย่างข้อมูลจากตารางที่ 5



ภาพที่ 4 ตัวอย่างเมทริกที่ได้หลังจากการแตกเป็นเมทริกย่อยในเทคนิค LSA

2.3 การใช้พจนานุกรมเข้ามาช่วยแก้ปัญหากรณีมีคำเหมือน

เทคนิคการขยายคำค้น สามารถนำมาประยุกต์เพื่อแก้ปัญหากรณีเกิดคำเหมือน เป็นกรณีของการที่คำตอบบางคำตอบใช้คำที่แตกต่างกัน เช่น ส่งเสริม กับสนับสนุน ทั้ง 2 คำนี้เขียนต่างกันแต่มีความหมายเดียวกัน โดยจะใช้ทั้งพจนานุกรมอังกฤษ-ไทย ในกรณีที่เป็นคำภาษาอังกฤษ และไทย-อังกฤษในกรณีที่เป็นคำภาษาไทย พัฒนาโดย NECTEC ซึ่งใช้พจนานุกรมที่มีชื่อว่า Lexitron มาแก้ปัญหา เนื่องจากคำตอบเป็นได้ทั้งภาษาไทย และภาษาอังกฤษผสมกัน โดยจะนำคำในคำตอบแต่ละคำไปค้นหาในพจนานุกรมว่ามีคำเหมือนหรือไม่ ถ้ามีจะทำการจับรวมคำเหมือนเหล่านี้เป็น 1 กลุ่มคำ ส่งผลให้การวัดค่าความคล้ายระหว่างคำตอบกับเฉลยมีค่ามากขึ้น ตัวอย่างดังตารางที่ 6

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

ตารางที่ 6 การแทนคำตอบในรูปความถี่ของคำแบบใหม่ โดยใช้พจนานุกรมเข้ามาช่วยแก้ปัญหากรณีมีคำเหมือน

นักเรียนคนที่	การสืบทอด/การสืบต่อ	abstract	class	polymorphism	object	ตัวแปร
1	**2**	1	0	1	0	0
2	0	0	2	0	0	0
3	**1**	0	0	0	2	1

จากตารางที่ 6 นักเรียนคนที่ 1 และ 3 ไม่ว่าจะใช้คำว่า “การสืบทอด” หรือ “การสืบต่อ” จะได้ความถี่ของคำ “สืบทอด / การสืบต่อ” เพิ่มขึ้นใน Attribute เดียวกัน อีกทั้งยังทำให้จำนวน Attribute ลดลง ทำให้ใช้หน่วยความจำน้อยลงและประหยัดเวลาในการประมวลผล

3. กระบวนการให้คะแนนคำตอบด้วยอัลกอริทึมการเรียนรู้แบบ Supervised (วิธีการที่ 1)

3.1 การเตรียมข้อมูลสอนและทดสอบระบบ

ในการทดลองจะใช้ชุดคำตอบที่ให้เกรดโดยผู้ตรวจในแต่ละระดับเกรด (A B C D และ F) ของชุดคำตอบทั้งหมดที่นำมาทดลอง โดยแบ่งตามระดับเกรด เกรดละ 3 คำตอบ รวมเป็น 15 คำตอบ ซึ่งบางข้ออาจเตรียมไม่ได้ทุกระดับเกรดสืบเนื่องมาจากข้อจำกัดของการเตรียมชุดข้อมูลสอนระบบ เช่น ในข้อนั้นนักเรียนทุกคนตอบถูกหมด ดังนั้นผู้ตรวจจะให้เกรดคำตอบทุกคำตอบเป็นเกรด A ส่วนเกรดอื่นๆจะไม่มี อีกทั้งในการเตรียมชุดคำตอบที่เป็นระดับเกรด F ไม่สามารถเตรียมได้เนื่องจากคำตอบผิดมีได้หลากหลายรูปแบบ เป็นต้น โดยการเตรียมชุดข้อมูลสอนระบบเสมือนเป็นการแทนคำเฉลยจากผู้ตรวจในแต่ละระดับเกรด เพื่อสร้างโมเดลการทำนายเกรด และชุดคำตอบที่เหลือใช้เป็นชุดทดสอบสำหรับวัดประสิทธิผลของโมเดล

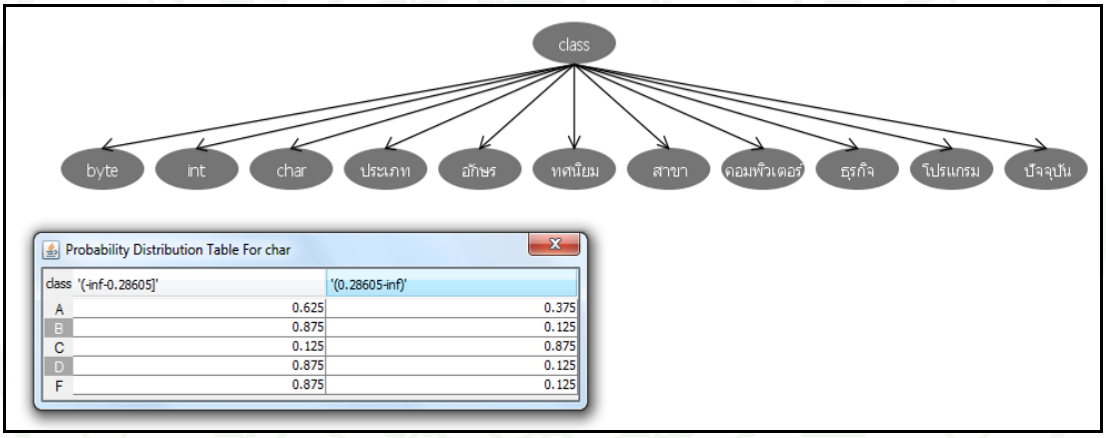
3.2 การคิดเกรดคำตอบด้วยอัลกอริทึม K-NN, Bayesian และ ANN

การกำหนดพารามิเตอร์ต่างๆในแต่ละอัลกอริทึม มีรายละเอียดดังนี้

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

3.2.1 การกำหนดค่าพารามิเตอร์ในอัลกอริทึม K-NN จะกำหนดค่า K = 3 และใช้สมการ Euclidean distance เป็นสมการวัดค่าความคล้ายในอัลกอริทึม

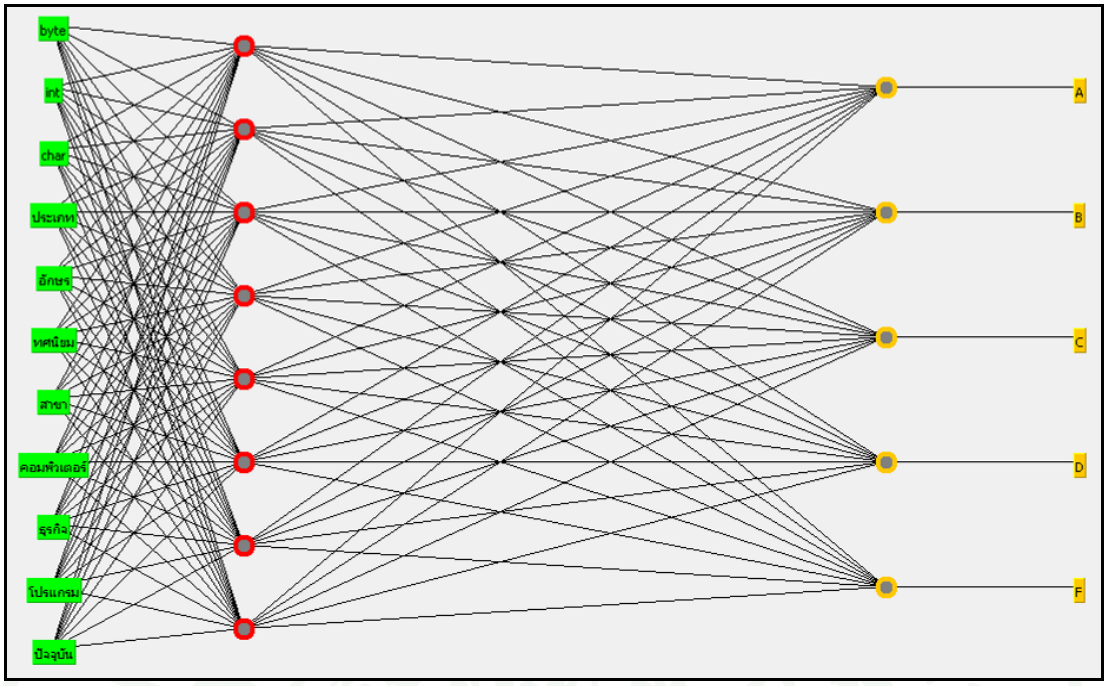
3.2.2 การกำหนดค่าพารามิเตอร์ในอัลกอริทึม Bayesian จะทดลองในอัลกอริทึม BayesNet ในโปรแกรม Weka และใช้ค่า Default โดยกำหนด estimator แบบ SimpleEstimator ค่า alpha หรือค่าประเมินค่าความน่าจะเป็นในตาราง CPT เท่ากับ 0.5 ใช้อัลกอริทึมการค้นหาแบบ K2 และกำหนดค่า initAsNaiveBayes เป็น True ทำให้โครงสร้างของ Bayesian ที่ได้จะเป็นแบบ Naïve Bayes ดังภาพที่ 5 แสดงตัวอย่างเหตุการณ์ของการเกิดขึ้นของคำต่างๆในคำตอบ ในแต่ละคำจะมีตาราง CPT ที่ได้จากการเรียนรู้ในชุดข้อมูลสอนระบบ และมีเส้นเชื่อมระหว่างโหนด class ไปยังโหนดคำต่างๆ เห็นได้ว่าตัวแปรคลาส (class) ขึ้นตรงต่อคำต่างๆ โดยที่คำต่างๆเป็นอิสระต่อกัน



ภาพที่ 5 ตัวอย่างโครงข่ายโมเดล Naïve Bayes

3.2.3 การกำหนดค่าพารามิเตอร์ในอัลกอริทึม ANN ในการทดลองจะใช้การเรียนรู้แบบ Back-propagation โดยใช้จำนวนโหนดนำเข้าเป็นโหนดความถี่ของแต่ละคำ ซึ่งโจทย์ทั้ง 7 ข้อมีจำนวนโหนดนำเข้าแตกต่างกันในแต่ละข้อดังนี้ 161 285 217 124 243 173 และ 199 ตามลำดับ (เรียงลำดับจากข้อ 1 ถึง 7) ส่วนในชุดข้อมูลที่ใช้เทคนิค LSA จะมีโหนดนำเข้าเพียง 3 โหนด (กำหนดขนาดของเมทริก Singular เป็น 3) และโหนดผลลัพธ์มี 5 โหนด คือเกรด A B C D และ F งานวิจัยนี้จะใช้อัลกอริทึม ANN แบบ Multi-Layer Perceptron ซึ่งเป็นการเชื่อมต่อกันของ Perceptron หลายๆตัว โดยการกำหนดค่าพารามิเตอร์ต่างๆจะใช้ค่า Default ในโปรแกรม Weka ตัวอย่างดังภาพที่ 6 โดยโหนดคำต่างๆทางซ้ายคือโหนดเข้า โหนดตรงกลางคือโหนดซ่อน และโหนดถัดมาคือโหนดผลลัพธ์มี 5 โหนดตามระดับเกรด

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์



ภาพที่ 6 ตัวอย่างโครงสร้างแบบ Multi-Layer Perceptron

4. กระบวนการให้คะแนนคำตอบด้วยอัลกอริทึมการเรียนรู้แบบ Unsupervised (วิธีการที่ 2)

จากภาพที่ 3 ในขั้นตอนของกระบวนการเริ่มต้นจะเหมือนกับวิธีการที่ 1 ขั้นตอนต่อไปจะส่งค่าเวกเตอร์ TF-IDF ในแต่ละคำตอบ เป็นข้อมูลนำเข้า เพื่อทำการแบ่งกลุ่มคำตอบด้วยอัลกอริทึมการเรียนรู้แบบ Unsupervised ซึ่งจะตัดตัวแปรคลาสออก โดยจะจัดคำตอบที่มีลักษณะของคำคล้ายๆกันให้อยู่ในกลุ่มเดียวกัน ในงานวิจัยนี้จะทำการแยกคำตอบที่มีค่าความคล้ายกับชุดเฉลยต่ำกว่า 0.01 ออกก่อนที่จะนำไปจัดกลุ่ม และกำหนดเกรดเป็น F เนื่องจากเป็นคำตอบที่วัดได้ค่าความคล้ายต่ำมากจึงไม่ควรนำมาใช้ในการจัดกลุ่ม แล้วนำคำตอบที่เหลือไปจัดกลุ่มตามความคล้าย ทำให้การแบ่งกลุ่มมีความถูกต้องมากขึ้น และเนื่องจากในความเป็นจริงไม่สามารถทราบจำนวนกลุ่มที่จัดได้ ดังนั้นในงานวิจัยนี้จึงเลือกอัลกอริทึม EM และ Cobweb มาทำการทดลอง และนำกระบวนการให้คะแนนคำตอบด้วยอัลกอริทึมการเรียนรู้แบบ Unsupervised นี้ไปสร้างโปรแกรมต้นแบบโดยแยกเป็น 2 ส่วนคือ ส่วนของการให้คะแนนคำตอบแบบรายคำตอบ (Classical method) กับส่วนของการแสดงผลการจัดกลุ่มและให้คะแนนกลุ่มคำตอบด้วยอัลกอริทึม EM และ Cobweb ตัวอย่างโปรแกรมต้นแบบดังภาพผนวก ค โดยรายละเอียดของการทดลองในแต่ละอัลกอริทึมมีดังนี้

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

4.1 การแบ่งกลุ่มด้วยอัลกอริทึม EM

อัลกอริทึม EM มีความสามารถในการค้นหาจำนวนกลุ่มที่จะทำการแบ่งกลุ่มแบบอัตโนมัติ โดยการกำหนดพารามิเตอร์จะทดลองหาค่าพารามิเตอร์ที่เหมาะสม โดยดูจากค่าการกระจายเฉลี่ยหลังการจัดกลุ่ม โดยใช้พารามิเตอร์ที่ให้ค่าการกระจายน้อยที่สุดมาใช้ในการทดลอง ในการกำหนดค่าพารามิเตอร์สำหรับการค้นหาจำนวนกลุ่มแบบอัตโนมัติ จะกำหนดพารามิเตอร์ numCluster เป็น -1 จำนวนรอบของการวนซ้ำไม่เกิน 100 (maxIterations = 100) ค่าการกระจายที่สามารถยอมรับได้ขั้นต่ำคือ 0.001 (minStdDev = 0.001) และรูปแบบ (Pattern) ของความถี่คำในคำตอบกำหนดเป็น 119 ครั้ง (seed = 119)

4.2 การแบ่งกลุ่มด้วยอัลกอริทึม Cobweb

การแบ่งกลุ่มด้วยอัลกอริทึม Cobweb พารามิเตอร์ต่างๆที่กำหนดในการทดลองจะใช้ค่า Default ของโปรแกรม Weka โดยค่าส่วนเบี่ยงเบนมาตรฐานที่สามารถยอมรับได้มีค่าเท่ากับ 1.0 (acuity = 1.0) ค่า Threshold ของฟังค์ชัน Category utility ที่ใช้สำหรับการสร้างต้นไม้การจำแนกกำหนดเป็น 0.0028 (cutoff = 0.0028) และรูปแบบ (Pattern) ของความถี่คำในคำตอบกำหนดเป็น 42 ครั้ง (seed = 42)

4.3 การคำนวณคะแนนคำตอบในแต่ละกลุ่ม

หลังจากจัดกลุ่มคำตอบด้วยอัลกอริทึมการแบ่งกลุ่มแล้ว ขั้นตอนต่อไปจะทำการคำนวณคะแนนในแต่ละกลุ่มคำตอบ โดยคำนวณค่าความคล้ายของตัวแทนกลุ่มคำตอบกับเฉลยด้วยสมการ Cosine ซึ่งตัวแทนแต่ละกลุ่มคำตอบหาได้จากค่าเฉลี่ยของข้อมูลในกลุ่มคำตอบนั้นๆ ถ้าตัวแทนกลุ่มคำตอบกับเฉลยใช้คำคล้ายกัน ค่าความคล้ายจะเข้าใกล้ 1 ซึ่งตามวิธีการให้คะแนนคำตอบของมนุษย์ ผู้ตรวจจะให้คะแนนคำตอบตามความถูกต้องของเนื้อหา หรือลดคะแนนลงเมื่อเนื้อหามีความถูกต้องน้อยลง นอกจากนี้การให้คะแนนคำตอบโดยทั่วไป คะแนนที่มากกว่า 80% ของคะแนนเต็มจะกำหนดเกรดเป็น A ส่วนคะแนนที่ต่ำกว่า 50% ของคะแนนเต็มจะให้เกรดเป็น F รายละเอียดของอัลกอริทึมสำหรับการให้คะแนนคำตอบดังนี้

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

อัลกอริทึมสำหรับการให้คะแนนคำตอบ

$S = \{S_1, S_2, \ldots, S_n\}$ // S แทนชุดเวกเตอร์คำเฉลยแต่ละข้อ จำนวน n ข้อ

$G = \{G_1, G_2\ldots., G_k\}$ // G แทนชุดเวกเตอร์ตัวแทนกลุ่มคำตอบแต่ละกลุ่ม จำนวน k กลุ่ม

// คำนวณค่าความคล้ายให้กับแต่ละกลุ่มคำตอบ (groupSimilarity) ระหว่างเวกเตอร์ตัวแทนกลุ่มคำตอบแต่ละกลุ่มกับเวกเตอร์คำเฉลยในแต่ละข้อด้วยฟังค์ชัน Cosine (Cosine_Sim)

```
FOR numCount = 1 to n  // n คือจำนวนข้อ
    FOR numGroupCount = 1 to k // k คือจำนวนกลุ่ม
        groupSimilarity[num,numGroup] = CALL Cosine_Sim(G[k], S[n])
    END FOR
END FOR
```

Threshold ∈ {0.8, 0.7, 0.6, 0.5} // Threshold คือเกณฑ์ที่ใช้สำหรับแบ่งระดับความคล้ายใน 5 ระดับตามเกรด A B C D และ F โดยการกำหนดเกณฑ์จะแตกต่างกันไปตามการทดลอง

Grade ∈ {A, B, C, D, F} // เกรดแต่ละระดับความคล้ายตามค่า Threshold

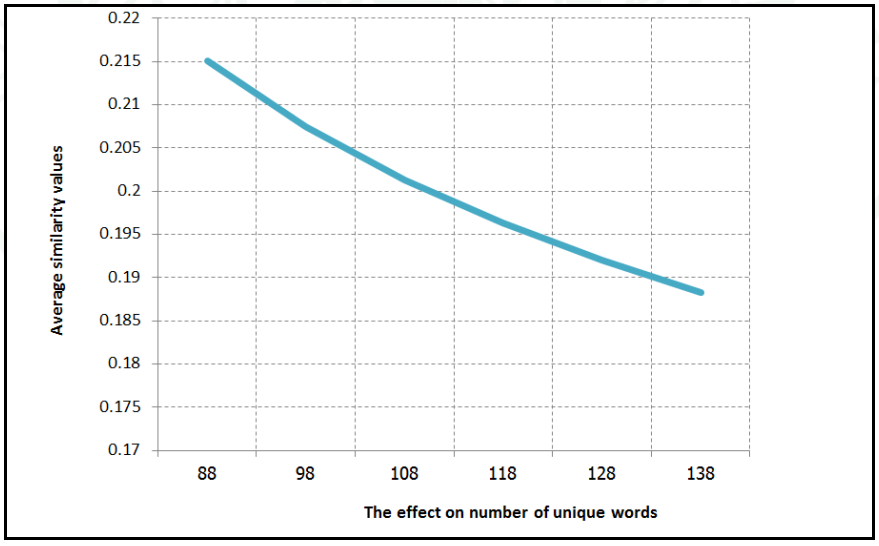
// กำหนดเกรดให้กับแต่ละกลุ่มคำตอบ (groupGrade) ตามค่า Threshold ที่กำหนด

```
FOR numCount2 = 1 to n
    FOR numGroupCount2 = 1 to k
        FOR index = 1 to 4  // Grade  A, B, C, D
            IF groupSimilarity[numCount2][numGroupCount2] >= Threshold[index]
                groupGrade[numCount2][numGroupCount2] = Grade[index]
            END IF
        END FOR
        IF groupSimilarity[numCount2][numGroupCount2] < Threshold[4]  // Grade F
            groupGrade[numCount2][numGroupCount2] = Grade[4]
        END IF
    END FOR
    END FOR
```

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

4.4 กระบวนการค้นหาผลกระทบของการวัดค่าความคล้ายสำหรับการให้คะแนนคำตอบและการกำหนดค่า Threshold

ในขั้นตอนการคำนวณคะแนนของกลุ่มคำตอบ โดยคำนวณค่าความคล้ายระหว่างเวกเตอร์ตัวแทนกลุ่มคำตอบกับเวกเตอร์คำเฉลย มีปัจจัยที่ส่งผลกระทบต่อค่าความคล้าย เช่น เวกเตอร์ความถี่ของคำที่ถูกแปลงมาจากข้อความ ทำให้ข้อมูลบางส่วนขาดหายไป การเขียนหรือสะกดคำผิด และการใช้คำที่ตอบนอกเหนือจากคำในเฉลย (ในงานวิจัยนี้จะยังไม่คำนึงถึงคำที่เพิ่มเข้ามาถูกหรือผิด) จะส่งผลกระทบต่อการคำนวณค่าความคล้ายในทางลบ กล่าวคือถ้าจำนวนคำเอกลักษณ์ (Unique word) มากขึ้น ขนาดของเวกเตอร์จะใหญ่ขึ้นด้วย ซึ่งถ้าคำเอกลักษณ์ที่เพิ่มเข้ามาไม่สัมพันธ์กับคำในเฉลย คะแนนของคำตอบนั้นๆจะมีค่าลดลงตามจำนวนคำเอกลักษณ์ที่เพิ่มเข้ามาดังภาพที่ 7 อย่างไรก็ตาม ขนาดของเวกเตอร์ที่เพิ่มขึ้นเนื่องจากคำเอกลักษณ์ที่มากขึ้น จะไม่ส่งผลกระทบต่อเวกเตอร์คำตอบที่ตอบถูกอยู่แล้ว เช่น $V_1 = \{1,3,2,1\}$ และ $V_2 = \{1,3,2,1,0,0,0\}$ ใน V_2 มีขนาดเวกเตอร์มากกว่า V_1 อยู่ 3 คอลัมน์ แต่เมื่อนำไปวัดค่าความคล้ายกับเวกเตอร์คำเฉลย ค่าความคล้ายที่ได้มีค่าเท่าเดิม เป็นต้น ในงานวิจัยนี้จะทดลองกำหนด Threshold ของเกรด A เริ่มที่ 0.5, 0.55, 0.6, …, 1.0 ตามลำดับเพื่อทดลองหาช่วงค่า Threshold ที่เหมาะสมที่ทำให้ค่า Accuracy มากที่สุด



ภาพที่ 7 กราฟความสัมพันธ์ของค่าความคล้ายเฉลี่ยกับจำนวนคำเอกลักษณ์

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

5. การวัดประสิทธิผล

ในการวัดประสิทธิผล จะใช้ตัววัด 2 ค่า คือ ค่าความแม่นยำ (Accuracy) และ Mean Squared Error (MSE) มีรายละเอียดดังนี้

5.1 ค่าความแม่นยำ (Accuracy)

การคำนวณค่าความแม่นยำ จะนับจำนวนคำตอบที่ผลเฉลยให้เกรดตรงกับเกรดที่ระบบให้ แล้วทำการคิดเป็นอัตราร้อยละของแต่ละเกรดนั้นแล้วหาค่าเฉลี่ยรวม ซึ่งสูตรที่ใช้ในการคำนวณดังสมการที่ (20) และตัวอย่างผลลัพธ์ที่ได้จากการประมวลผลสำหรับเกรด A ดังตารางที่ 7

$$Acc = \frac{TP + TN}{TP + FP + FN + TN} \qquad (20)$$

โดยที่ TP คือ จำนวนคำตอบที่ระบบให้ผลเป็นจริงและผลเฉลยให้ผลเป็นจริงด้วย (True Positive)

TN คือ จำนวนคำตอบที่ระบบให้ผลเป็นเท็จและผลเฉลยให้ผลเป็นเท็จด้วย (True Negative)

FP คือ จำนวนคำตอบที่ระบบให้ผลเป็นจริงแต่ผลเฉลยให้ผลเป็นเท็จ (False Positive)

FN คือ จำนวนคำตอบที่ระบบให้ผลเป็นเท็จแต่ผลเฉลยให้ผลเป็นจริง (False Negative)

ตารางที่ 7 ตัวอย่างผลลัพธ์ที่ได้จากการประมวลผลสำหรับเกรด A

ข้อที่ 1	**ผลลัพธ์ที่ได้จากระบบ**		
ผลลัพธ์ที่ได้จากเฉลย		**จำนวนคำตอบที่ได้เกรด A**	**จำนวนคำตอบที่ไม่ได้เกรด A**
	จำนวนคำตอบที่ได้เกรด A	7 (TP)	2 (FN)
	จำนวนคำตอบที่ไม่ได้เกรด A	5 (FP)	41 (TN)
ค่า Accuracy(%) = 87.27			

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

ตารางที่ 7 แสดงตัวอย่างผลลัพธ์ที่ได้จากการประมวลผลสำหรับเกรด A โดย TP มีจำนวน 7 คำตอบ TN จำนวน 41 คำตอบ FN มีจำนวน 2 คำตอบ และ FP มีจำนวน 5 คำตอบ ได้ค่า Accuracy ของเกรด A เป็น 87.27%

5.2 ค่า Mean Squared Error (MSE)

ค่า MSE ใช้สำหรับวัดความคลาดเคลื่อนของค่าจริงกับค่าที่พยากรณ์ได้ โดยที่ค่า MSE ถ้ามีค่าต่ำ แสดงถึงค่าจริงกับค่าที่พยากรณ์ได้ใกล้เคียงกัน ซึ่งแสดงถึงการพยากรณ์มีความแม่นยำสูง สูตรที่ใช้ในการคำนวณดังสมการที่ (21) และ (22)

$$MSE = \frac{\sum_{i=1}^{n} e_i^2}{n} \qquad (21)$$

$$e_i = X_i - T_i \qquad (22)$$

โดยที่ MSE คือ ค่า Mean Squared Error

e_i คือ ค่าความคาดเคลื่อนของค่าจริงหรือผลเฉลยกับค่าที่พยากรณ์ได้ของคำตอบที่ i

n คือ จำนวนคำตอบ

X_i คือ ค่าพยากรณ์ของคำตอบที่ i

T_i คือ ค่าจริงหรือผลเฉลยของคำตอบที่ i

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

ผลและวิจารณ์

ผล

ผลการทดลองแบ่งออกเป็น 2 ส่วน คือ ผลการทดลองบนอัลกอริทึมการเรียนรู้แบบมีผู้สอน และผลการทดลองบนอัลกอริทึมการเรียนรู้แบบไม่มีผู้สอน มีรายละเอียดดังนี้

1. ผลการทดลองบนอัลกอริทึมการเรียนรู้แบบมีผู้สอน

การวัดประสิทธิผลด้วยค่าความแม่นยำบนอัลกอริทึมการเรียนรู้แบบมีผู้สอน แบ่งเป็น 3 อัลกอริทึม คือ อัลกอริทึม K-NN ANN และ Bayesian แยกตามเทคนิคที่นำมาใช้ในการทดลอง โดยผลการทดลองแสดงดังตารางที่ 8 - 19

ตารางที่ 8 ค่าความแม่นยำในแต่ละข้อในการจำแนกด้วยอัลกอริทึม K-NN

ข้อที่	เกรด A (%)	เกรด B (%)	เกรด C (%)	เกรด D (%)	เกรด F (%)	เฉลี่ย (%)
1	80.00	62.50	47.50	87.50	76.36	70.77
2	97.56	9.76	82.93	65.85	69.09	65.04
3	97.50	92.50	92.50	30.00	61.82	74.86
4	100.00	97.92	95.83	100.00	94.55	97.66
5	77.78	35.56	97.78	93.33	54.55	71.80
6	95.35	95.35	37.21	58.14	90.91	75.39
7	100.00	83.33	19.05	71.43	72.73	69.31
			Total			**74.98**

ตารางที่ 8 แสดงค่าความแม่นยำในแต่ละข้อในการจำแนกด้วยอัลกอริทึม K-NN โดยในข้อที่ 1 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 70.77% ข้อที่ 2 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 65.04% ข้อที่ 3 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 74.86% ข้อที่ 4 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 97.66% ข้อที่ 5 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 71.80% ข้อที่ 6 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 75.39% ข้อที่ 7 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 69.31% และค่าความแม่นยำเฉลี่ยรวมทั้ง 7 ข้อเป็น 74.98%

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

ตารางที่ 9 ค่าความแม่นยำในแต่ละข้อกรณีที่ใช้พจนานุกรม ในการจำแนกด้วยอัลกอริทึม K-NN

ข้อที่	เกรด A (%)	เกรด B (%)	เกรด C (%)	เกรด D (%)	เกรด F (%)	เฉลี่ย (%)
1	80.00	67.50	52.50	87.50	76.36	72.77
2	78.05	36.59	82.93	60.98	67.27	65.16
3	97.50	92.50	92.50	30.00	61.82	74.86
4	100.00	97.92	95.83	100.00	94.55	97.66
5	77.78	33.33	97.78	93.33	52.73	70.99
6	95.35	95.35	37.21	58.14	90.91	75.39
7	100.00	78.57	16.67	73.81	72.73	68.35
			Total			**75.03**

ตารางที่ 9 แสดงค่าความแม่นยำในแต่ละข้อกรณีที่ใช้พจนานุกรม ในการจำแนกด้วยอัลกอริทึม K-NN โดยในข้อที่ 1 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 72.77% ข้อที่ 2 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 65.16% ข้อที่ 3 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 74.86% ข้อที่ 4 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 97.66% ข้อที่ 5 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 70.99% ข้อที่ 6 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 75.39% ข้อที่ 7 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 68.35% และค่าความแม่นยำเฉลี่ยรวมทั้ง 7 ข้อเป็น 75.03%

ตารางที่ 10 ค่าความแม่นยำในแต่ละข้อในการจำแนกด้วยอัลกอริทึม K-NN ร่วมกับเทคนิค LSA

ข้อที่	เกรด A (%)	เกรด B (%)	เกรด C (%)	เกรด D (%)	เกรด F (%)	เฉลี่ย (%)
1	80.00	75.00	60.00	55.00	85.45	71.09
2	68.29	53.66	87.80	75.61	69.09	70.89
3	37.50	90.00	90.00	70.00	58.18	69.14
4	91.67	97.92	95.83	100.00	98.18	96.72
5	33.33	73.33	97.78	93.33	52.73	70.10
6	93.02	86.05	65.12	58.14	67.27	73.92
7	76.19	73.81	47.62	47.62	72.73	63.59
			Total			**73.64**

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

ตารางที่ 10 แสดงค่าความแม่นยำในแต่ละข้อในการจำแนกด้วยอัลกอริทึม K-NN ร่วมกับเทคนิค LSA โดยในข้อที่ 1 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 71.09% ข้อที่ 2 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 70.89% ข้อที่ 3 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 69.14% ข้อที่ 4 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 96.72% ข้อที่ 5 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 70.10% ข้อที่ 6 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 73.92% ข้อที่ 7 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 63.59% และค่าความแม่นยำเฉลี่ยรวมทั้ง 7 ข้อเป็น 73.64%

ตารางที่ 11 ค่าความแม่นยำในแต่ละข้อกรณีที่ใช้พจนานุกรม ในการจำแนกด้วยอัลกอริทึม K-NN ร่วมกับเทคนิค LSA

ข้อที่	เกรด A (%)	เกรด B (%)	เกรด C (%)	เกรด D (%)	เกรด F (%)	เฉลี่ย (%)
1	77.50	65.00	55.00	57.50	85.45	68.09
2	34.15	95.12	78.05	75.61	67.27	70.04
3	82.50	92.50	85.00	32.50	58.18	70.14
4	89.58	97.92	95.83	100.00	96.36	95.94
5	35.56	75.56	97.78	93.33	49.09	70.26
6	95.35	81.40	65.12	58.14	65.45	73.09
7	85.71	71.43	45.24	47.62	72.73	64.55
			Total			**73.16**

ตารางที่ 11 แสดงค่าความแม่นยำในแต่ละข้อกรณีที่ใช้พจนานุกรม ในการจำแนกด้วยอัลกอริทึม K-NN ร่วมกับเทคนิค LSA โดยในข้อที่ 1 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 68.09% ข้อที่ 2 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 70.04% ข้อที่ 3 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 70.14% ข้อที่ 4 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 95.94% ข้อที่ 5 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 70.26% ข้อที่ 6 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 73.09% ข้อที่ 7 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 64.55% และค่าความแม่นยำเฉลี่ยรวมทั้ง 7 ข้อเป็น 73.16%

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

ตารางที่ 12 ค่าความแม่นยำในแต่ละข้อในการจำแนกด้วยอัลกอริทึม ANN

ข้อที่	เกรด A (%)	เกรด B (%)	เกรด C (%)	เกรด D (%)	เกรด F (%)	เฉลี่ย (%)
1	82.50	65.00	57.50	80.00	81.82	73.36
2	97.56	97.56	82.93	63.41	63.64	81.02
3	97.50	92.50	85.00	32.50	61.82	73.86
4	97.92	97.92	97.92	100.00	98.18	98.39
5	75.56	71.11	97.78	93.33	85.45	84.65
6	95.35	88.37	48.84	58.14	94.55	77.05
7	100.00	71.43	73.81	71.43	80.00	79.33
			Total			**81.09**

ตารางที่ 12 แสดงค่าความแม่นยำในแต่ละข้อในการจำแนกด้วยอัลกอริทึม ANN โดยในข้อที่ 1 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 73.36% ข้อที่ 2 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 81.02% ข้อที่ 3 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 73.86% ข้อที่ 4 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 98.39% ข้อที่ 5 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 84.65% ข้อที่ 6 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 77.05% ข้อที่ 7 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 79.33% และค่าความแม่นยำเฉลี่ยรวมทั้ง 7 ข้อเป็น 81.09%

ตารางที่ 13 ค่าความแม่นยำในแต่ละข้อกรณีที่ใช้พจนานุกรม ในการจำแนกด้วยอัลกอริทึม ANN

ข้อที่	เกรด A (%)	เกรด B (%)	เกรด C (%)	เกรด D (%)	เกรด F (%)	เฉลี่ย (%)
1	82.50	65.00	57.50	75.00	85.45	73.09
2	97.56	97.56	80.49	68.29	65.45	81.87
3	95.00	90.00	85.00	30.00	60.00	72.00
4	100.00	100.00	97.92	100.00	98.18	99.22
5	77.78	80.00	97.78	93.33	83.64	86.51
6	95.35	93.02	53.49	58.14	94.55	78.91
7	100.00	73.81	78.57	71.43	78.18	80.40
			Total			**81.71**

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

ตารางที่ 13 แสดงค่าความแม่นยำในแต่ละข้อกรณีที่ใช้พจนานุกรม ในการจำแนกด้วยอัลกอริทึม ANN โดยในข้อที่ 1 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 73.09% ข้อที่ 2 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 81.87% ข้อที่ 3 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 72.00% ข้อที่ 4 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 99.22% ข้อที่ 5 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 86.51% ข้อที่ 6 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 78.91% ข้อที่ 7 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 80.40% และค่าความแม่นยำเฉลี่ยรวมทั้ง 7 ข้อเป็น 81.71%

ตารางที่ 14 ค่าความแม่นยำในแต่ละข้อในการจำแนกด้วยอัลกอริทึม ANN ร่วมกับเทคนิค LSA

ข้อที่	เกรด A (%)	เกรด B (%)	เกรด C (%)	เกรด D (%)	เกรด F (%)	เฉลี่ย (%)
1	95.00	62.50	55.00	45.00	83.64	68.23
2	100.00	85.37	85.37	41.46	63.64	75.17
3	97.50	95.00	85.00	27.50	56.36	72.27
4	93.75	100.00	93.75	100.00	96.36	96.77
5	77.78	82.22	97.78	93.33	78.18	85.86
6	93.02	81.40	65.12	58.14	67.27	72.99
7	100.00	80.95	76.19	76.19	92.73	85.21
			Total			**79.50**

ตารางที่ 14 แสดงค่าความแม่นยำในแต่ละข้อในการจำแนกด้วยอัลกอริทึม ANN ร่วมกับเทคนิค LSA โดยในข้อที่ 1 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 68.23% ข้อที่ 2 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 75.17% ข้อที่ 3 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 72.27% ข้อที่ 4 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 96.77% ข้อที่ 5 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 85.86% ข้อที่ 6 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 72.99% ข้อที่ 7 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 85.21% และค่าความแม่นยำเฉลี่ยรวมทั้ง 7 ข้อเป็น 79.50%

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

ตารางที่ 15 ค่าความแม่นยำในแต่ละข้อกรณีที่ใช้พจนานุกรม ในการจำแนกด้วยอัลกอริทึม ANN ร่วมกับเทคนิค LSI

ข้อที่	เกรด A (%)	เกรด B (%)	เกรด C (%)	เกรด D (%)	เกรด F (%)	เฉลี่ย (%)
1	95.00	67.50	60.00	45.00	83.64	70.23
2	100.00	80.49	82.93	48.78	67.27	75.89
3	95.00	87.50	85.00	35.00	54.55	71.41
4	97.92	97.92	93.75	100.00	94.55	96.83
5	77.78	77.78	95.56	93.33	83.64	85.62
6	93.02	83.72	62.79	58.14	60.00	71.53
7	100.00	83.33	66.67	73.81	92.73	83.31
			Total			**79.26**

ตารางที่ 15 แสดงค่าความแม่นยำในแต่ละข้อกรณีที่ใช้พจนานุกรม ในการจำแนกด้วยอัลกอริทึม ANN ร่วมกับเทคนิค LSA โดยในข้อที่ 1 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 70.23% ข้อที่ 2 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 75.89% ข้อที่ 3 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 71.41% ข้อที่ 4 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 96.83% ข้อที่ 5 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 85.62% ข้อที่ 6 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 71.53% ข้อที่ 7 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 83.31% และค่าความแม่นยำเฉลี่ยรวมทั้ง 7 ข้อเป็น 79.26%

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

ตารางที่ 16 ค่าความแม่นยำในแต่ละข้อในการจำแนกด้วยอัลกอริทึม Bayesian

ข้อที่	เกรด A (%)	เกรด B (%)	เกรด C (%)	เกรด D (%)	เกรด F (%)	เฉลี่ย (%)
1	92.50	57.50	55.00	92.50	76.36	74.77
2	100.00	51.22	80.49	29.27	69.09	66.01
3	97.50	30.00	65.00	70.00	61.82	64.86
4	95.83	95.83	95.83	100.00	96.36	96.77
5	24.44	84.44	97.78	93.33	58.18	71.64
6	93.02	51.16	62.79	58.14	67.27	66.48
7	100.00	80.95	71.43	45.24	72.73	74.07
			Total			**73.52**

ตารางที่ 16 แสดงค่าความแม่นยำในแต่ละข้อในการจำแนกด้วยอัลกอริทึม Bayesian โดยในข้อที่ 1 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 74.77% ข้อที่ 2 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 66.01% ข้อที่ 3 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 65.86% ข้อที่ 4 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 96.77% ข้อที่ 5 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 71.64% ข้อที่ 6 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 66.58% ข้อที่ 7 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 74.07% และค่าความแม่นยำเฉลี่ยรวมทั้ง 7 ข้อเป็น 73.67%

ตารางที่ 17 ค่าความแม่นยำในแต่ละข้อกรณีที่ใช้พจนานุกรม ในการจำแนกด้วยอัลกอริทึม Bayesian

ข้อที่	เกรด A (%)	เกรด B (%)	เกรด C (%)	เกรด D (%)	เกรด F (%)	เฉลี่ย (%)
1	92.50	57.50	55.00	92.50	76.36	74.77
2	100.00	51.22	80.49	29.27	69.09	66.01
3	97.50	25.00	75.00	70.00	61.82	65.86
4	95.83	95.83	95.83	100.00	96.36	96.77
5	24.44	84.44	97.78	93.33	58.18	71.64
6	93.02	53.49	62.79	58.14	65.45	66.58
7	100.00	80.95	71.43	45.24	72.73	74.07
			Total			**73.67**

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

ตารางที่ 17 แสดงค่าความแม่นยำในแต่ละข้อกรณีที่ใช้พจนานุกรม ในการจำแนกด้วยอัลกอริทึม Bayesian โดยในข้อที่ 1 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 74.77% ข้อที่ 2 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 66.01% ข้อที่ 3 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 64.86% ข้อที่ 4 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 96.77% ข้อที่ 5 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 71.64% ข้อที่ 6 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 66.48% ข้อที่ 7 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 74.07% และค่าความแม่นยำเฉลี่ยรวมทั้ง 7 ข้อเป็น 73.52%

ตารางที่ 18 ค่าความแม่นยำในแต่ละข้อในการจำแนกด้วยอัลกอริทึม Bayesian ร่วมกับเทคนิค LSA

ข้อที่	เกรด A (%)	เกรด B (%)	เกรด C (%)	เกรด D (%)	เกรด F (%)	เฉลี่ย (%)
1	80.00	62.50	55.00	52.50	85.45	67.09
2	100.00	2.44	80.49	63.41	69.09	63.09
3	2.50	92.50	92.50	70.00	61.82	63.86
4	100.00	97.92	95.83	100.00	94.55	97.66
5	24.44	84.44	97.78	93.33	58.18	71.64
6	95.35	74.42	60.47	58.14	67.27	71.13
7	100.00	61.90	83.33	42.86	72.73	72.16
			Total			**72.38**

ตารางที่ 18 แสดงค่าความแม่นยำในแต่ละข้อในการจำแนกด้วยอัลกอริทึม Bayesian ร่วมกับเทคนิค LSA โดยในข้อที่ 1 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 67.09% ข้อที่ 2 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 63.09% ข้อที่ 3 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 63.86% ข้อที่ 4 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 97.66% ข้อที่ 5 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 71.64% ข้อที่ 6 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 71.13% ข้อที่ 7 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 72.16% และค่าความแม่นยำเฉลี่ยรวมทั้ง 7 ข้อเป็น 72.38%

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

ตารางที่ 19 ค่าความแม่นยำในแต่ละข้อกรณีที่ใช้พจนานุกรม ในการจำแนกด้วยอัลกอริทึม Bayesian ร่วมกับเทคนิค LSA

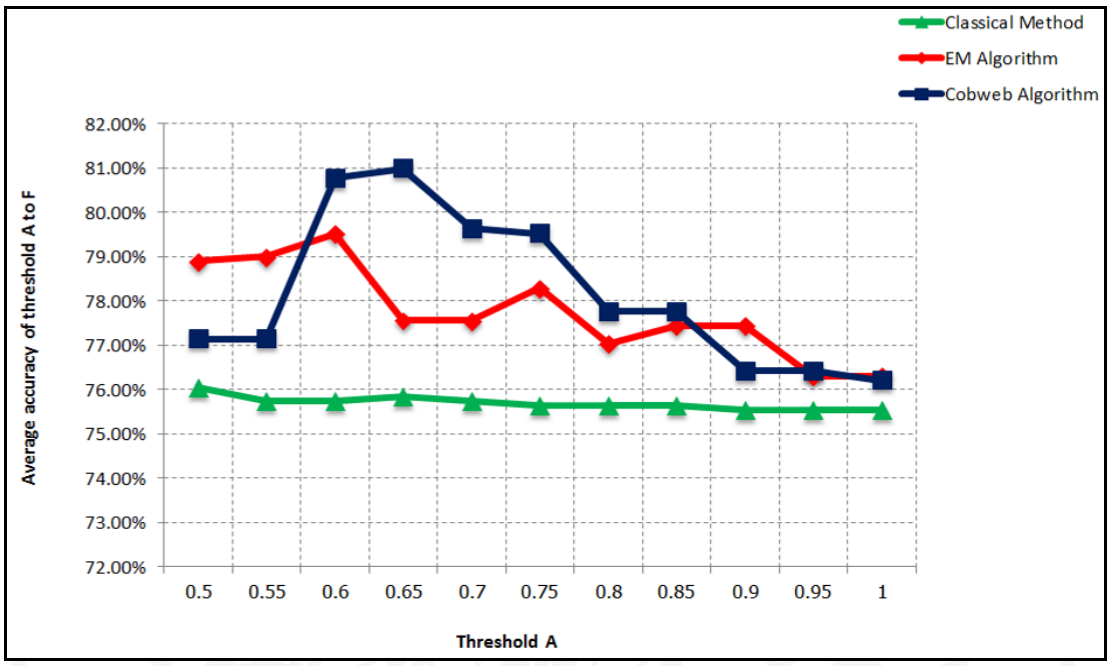
ข้อที่	เกรด A (%)	เกรด B (%)	เกรด C (%)	เกรด D (%)	เกรด F (%)	เฉลี่ย (%)
1	80.00	62.50	50.00	52.50	85.45	66.09
2	100.00	2.44	80.49	63.41	69.09	63.09
3	2.50	92.50	92.50	70.00	61.82	63.86
4	97.92	97.92	93.75	100.00	94.55	96.83
5	24.44	84.44	97.78	93.33	58.18	71.64
6	95.35	67.44	62.79	58.14	67.27	70.20
7	100.00	61.90	83.33	42.86	72.73	72.16
			Total			**71.98**

ตารางที่ 19 แสดงค่าความแม่นยำในแต่ละข้อกรณีที่ใช้พจนานุกรม ในการจำแนกด้วยอัลกอริทึม Bayesian ร่วมกับเทคนิค LSA โดยในข้อที่ 1 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 66.09% ข้อที่ 2 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 63.09% ข้อที่ 3 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 83.68% ข้อที่ 4 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 96.83% ข้อที่ 5 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 71.64% ข้อที่ 6 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 70.20% ข้อที่ 7 ให้ค่าความแม่นยำเฉลี่ยของทุกระดับเกรดเป็น 72.16% และค่าความแม่นยำเฉลี่ยรวมทั้ง 7 ข้อเป็น 74.81%

2. ผลการทดลองบนอัลกอริทึมการเรียนรู้แบบไม่มีผู้สอน

ผลการทดลองบนอัลกอริทึมการเรียนรู้แบบไม่มีผู้สอน จะแยกเป็น 4 แบบ คือ แบบปกติ แบบปกติที่ใช้พจนานุกรมแก้ปัญหาคำเหมือน แบบใช้เทคนิค LSA และแบบใช้เทคนิค LSA ร่วมกับนำพจนานุกรมแก้ปัญหาคำเหมือน บนวิธีการคิดเกรดแบบดั้งเดิม (Classical method) (วิธีการคิดเกรดแบบรายคำตอบ) และอัลกอริทึมการแบ่งกลุ่มทั้ง 2 เรียงตามช่วงค่า Threshold A ดังภาพที่ 8-11 และตารางที่ 20-23

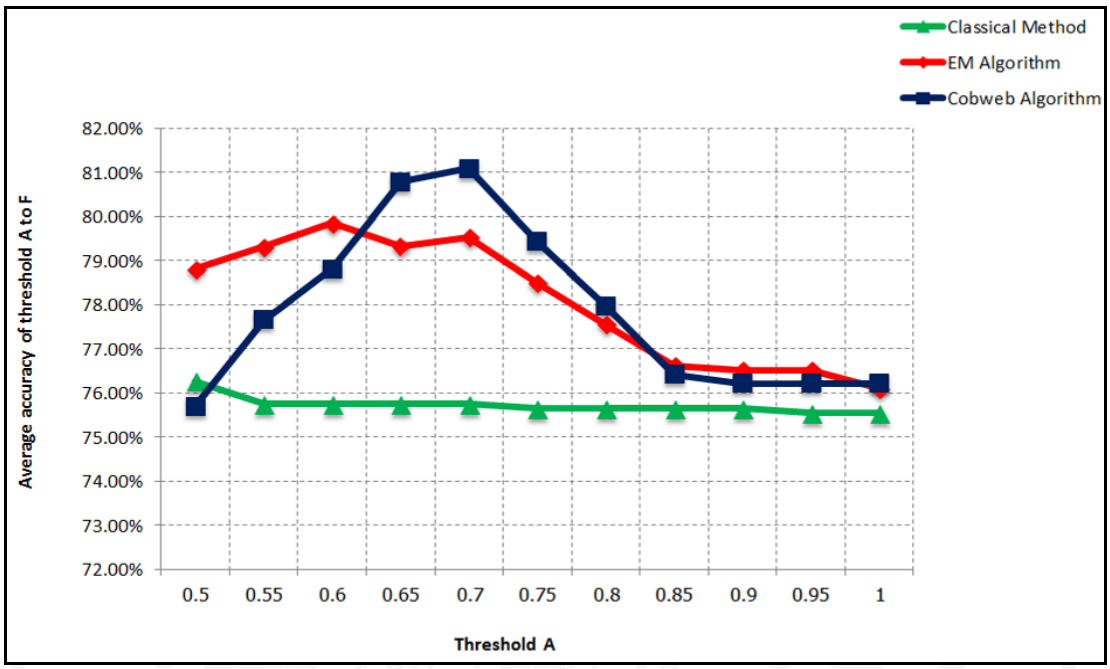
ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์



ภาพที่ 8 กราฟค่าความแม่นยำเฉลี่ยรวมข้อ 1-7 ของวิธีการคิดเกรดแบบดั้งเดิมและอัลกอริทึมการแบ่งกลุ่มทั้ง 2 เรียงตามช่วงค่า Threshold A แบบปกติ

จากภาพที่ 8 ค่าเกณฑ์ที่จะเลือกนำมาใช้ในวิธีการคิดเกรดแบบดั้งเดิม คือ 0.5 ในอัลกอริทึม EM คือ 0.6 และในอัลกอริทึม Cobweb คือ 0.65 ได้ค่าความแม่นยำเฉลี่ยรวมสูงสุดเป็น 76.06% 79.53% และ 80.99% ตามลำดับ

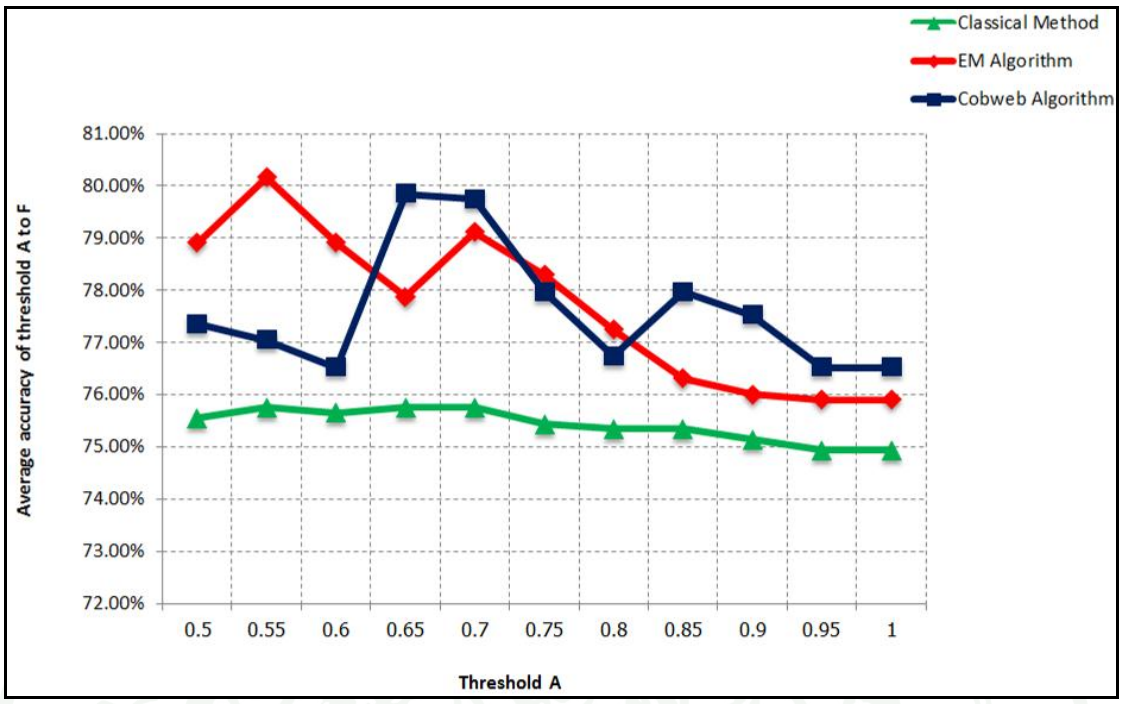
ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์



ภาพที่ 9 กราฟค่าความแม่นยำเฉลี่ยรวมข้อ 1-7 ของวิธีการคิดเกรดแบบดั้งเดิมและอัลกอริทึมการแบ่งกลุ่มทั้ง 2 เรียงตามช่วงค่า Threshold A แบบปกติร่วมกับพจนานุกรมแก้ปัญหาคำเหมือน

จากภาพที่ 9 ค่าเกณฑ์ที่จะเลือกนำมาใช้ในวิธีการคิดเกรดแบบดั้งเดิม คือ 0.5 ในอัลกอริทึม EM คือ 0.6 และในอัลกอริทึม Cobweb คือ 0.7 ได้ค่าความแม่นยำเฉลี่ยรวมสูงสุดเป็น 76.26% 79.84% และ 81.09% ตามลำดับ

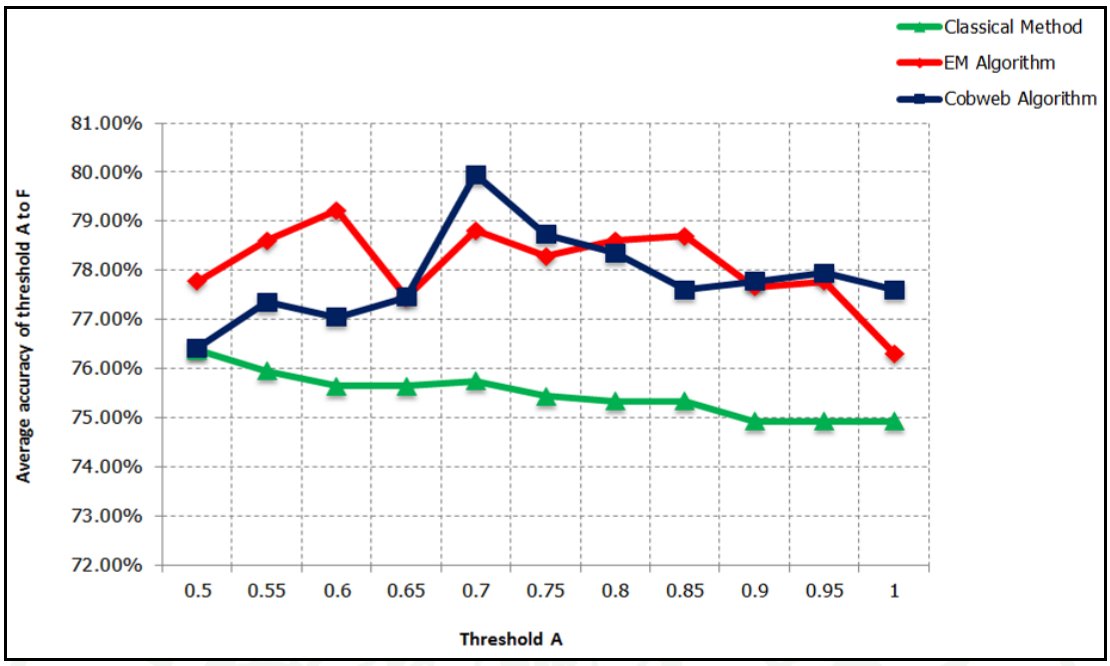
ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์



ภาพที่ 10 กราฟค่าความแม่นยำเฉลี่ยรวมข้อ 1-7 ของวิธีการคิดเกรดแบบดั้งเดิมและอัลกอริทึมการแบ่งกลุ่มทั้ง 2 เรียงตามช่วงค่า Threshold A แบบใช้เทคนิค LSA

จากภาพที่ 10 ค่าเกณฑ์ของเกรด A ที่จะเลือกมาใช้ในวิธีการคิดเกรดแบบดั้งเดิม คือ 0.55 ในอัลกอริทึม EM คือ 0.55 และในอัลกอริทึม Cobweb คือ 0.65 ได้ค่าความแม่นยำเฉลี่ยรวมสูงสุดเป็น 75.75% 80.16% และ 79.84% ตามลำดับ

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์



ภาพที่ 11 กราฟค่าความแม่นยำเฉลี่ยรวมข้อ 1-7 ของวิธีการคิดเกรดแบบดั้งเดิมและอัลกอริทึมการแบ่งกลุ่มทั้ง 2 เรียงตามช่วงค่า Threshold A แบบใช้เทคนิค LSA ร่วมกับพจนานุกรมแก้ปัญหาคำเหมือน

จากภาพที่ 11 ค่าเกณฑ์ที่จะเลือกนำมาใช้ในวิธีการคิดเกรดแบบดั้งเดิม คือ 0.5 ในอัลกอริทึม EM คือ 0.6 และในอัลกอริทึม Cobweb คือ 0.7 ได้ค่าความแม่นยำเฉลี่ยรวมสูงสุดเป็น 76.37% 79.22% และ 79.95% ตามลำดับ

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

ตารางที่ 20 ค่าความแม่นยำเฉลี่ยรวมข้อ 1-7 ของวิธีการคิดเกรดแบบดั้งเดิมและอัลกอริทึมการแบ่งกลุ่มทั้ง 2 เรียงตามช่วงค่า Threshold A แบบปกติ

Threshold A	ค่าความแม่นยำ (%)		
	วิธีการคิดเกรดแบบดั้งเดิม	อัลกอริทึม EM	อัลกอริทึม Cobweb
0.50	**76.06**	78.91	77.14
0.55	75.75	79.01	77.14
0.60	75.75	**79.53**	80.78
0.65	75.85	77.56	**80.99**
0.70	75.75	77.56	79.64
0.75	75.65	78.29	79.53
0.80	75.65	77.04	77.77
0.85	75.65	77.45	77.77
0.90	75.55	77.45	76.42
0.95	75.55	76.31	76.42
1.00	75.55	76.31	76.21

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

ตารางที่ 21 ค่าความแม่นยำเฉลี่ยรวมข้อ 1-7 ของวิธีการคิดเกรดแบบดั้งเดิมและอัลกอริทึมการแบ่งกลุ่มทั้ง 2 เรียงตามช่วงค่า Threshold A แบบปกติ ร่วมกับพจนานุกรมแก้ปัญหาคำเหมือน

Threshold A	**ค่าความแม่นยำ (%)**		
	วิธีการคิดเกรดแบบดั้งเดิม	**อัลกอริทึม EM**	**อัลกอริทึม Cobweb**
0.50	**76.26**	78.81	75.69
0.55	75.75	79.32	77.66
0.60	75.75	**79.84**	78.81
0.65	75.75	79.32	80.78
0.70	75.75	79.53	**81.09**
0.75	75.65	78.49	79.43
0.80	75.65	77.56	77.97
0.85	75.65	76.62	76.42
0.90	75.65	76.52	76.21
0.95	75.55	76.52	76.21
1.00	75.55	76.10	76.21

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

ตารางที่ 22 ค่าความแม่นยำเฉลี่ยรวมข้อ 1-7 ของวิธีการคิดเกรดแบบดั้งเดิมและอัลกอริทึมการแบ่งกลุ่มทั้ง 2 เรียงตามช่วงค่า Threshold A แบบใช้เทคนิค LSA

Threshold A	**ค่าความแม่นยำ (%)**		
	วิธีการคิดเกรดแบบดั้งเดิม	**อัลกอริทึม EM**	**อัลกอริทึม Cobweb**
0.50	75.55	78.91	77.35
0.55	**75.75**	**80.16**	77.04
0.60	75.65	78.91	76.52
0.65	75.75	77.87	**79.84**
0.70	75.75	79.12	79.74
0.75	75.44	78.29	77.97
0.80	75.34	77.25	76.73
0.85	75.34	76.31	77.97
0.90	75.14	76.00	77.52
0.95	74.93	75.90	76.52
1.00	74.93	75.90	76.52

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

ตารางที่ 23 ค่าความแม่นยำเฉลี่ยรวมข้อ 1-7 ของวิธีการคิดเกรดแบบดั้งเดิมและอัลกอริทึมการแบ่งกลุ่มทั้ง 2 เรียงตามช่วงค่า Threshold A แบบใช้เทคนิค LSA ร่วมกับพจนานุกรมแก้ปัญหาคำเหมือน

Threshold A	**ค่าความแม่นยำ (%)**		
	วิธีการคิดเกรดแบบดั้งเดิม	**อัลกอริทึม EM**	**อัลกอริทึม Cobweb**
0.50	**76.37**	77.77	76.42
0.55	75.95	78.60	77.35
0.60	75.65	**79.22**	77.04
0.65	75.65	77.45	77.45
0.70	75.75	78.81	**79.95**
0.75	75.44	78.29	78.73
0.80	75.34	78.60	78.35
0.85	75.34	78.70	77.60
0.90	74.93	77.66	77.77
0.95	74.93	77.77	77.95
1.00	74.93	76.31	77.60

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

ตารางที่ 24 ประสิทธิผลการจัดกลุ่มด้วยอัลกอริทึม EM

ชุดข้อมูล	จำนวนคำตอบ	มนุษย์	อัลกอริทึม EM							
			แบบปกติ		แบบปกติ + พจนานุกรม		เทคนิค LSA		เทคนิค LSA + พจนานุกรม	
		จำนวนกลุ่ม	จำนวนกลุ่ม	ค่าความแม่นยำ (%)	จำนวนกลุ่ม	ค่าความแม่นยำ (%)	จำนวนกลุ่ม	ค่าความแม่นยำ (%)	จำนวนกลุ่ม	ค่าความแม่นยำ (%)
D1	41	4(3/17/17/4/0)	3	23.59	5	21.54	3	23.90	2	22.93
D2	34	5(2/2/11/15/4)	2	9.03	2	10.00	2	22.35	3	27.22
D3	36	5(4/6/6/13/7)	1	9.03	4	28.11	3	26.11	3	11.35
D4	12	4 (1/2/4/0/5)	1	**48.33**	1	**48.33**	1	**46.67**	1	**48.33**
D5	36	5(14/9/2/2/9)	1	29.71	1	45.56	2	26.11	2	25.00
D6	42	5(5/5/19/11/2)	4	28.00	3	10.26	4	28.10	2	11.43
D7	39	5(1/11/10/14/3)	1	44.21	1	27.37	3	23.59	2	23.59

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

ตารางที่ 25 ประสิทธิผลการจัดกลุ่มด้วยอัลกอริทึม Cobweb

ชุดข้อมูล	จำนวนคำตอบ	มนุษย์	อัลกอริทึม Cobweb							
			แบบปกติ		แบบปกติ + พจนานุกรม		เทคนิค LSA		เทคนิค LSA + พจนานุกรม	
		จำนวนกลุ่ม	จำนวนกลุ่ม	ค่าความแม่นยำ (%)	จำนวนกลุ่ม	ค่าความแม่นยำ (%)	จำนวนกลุ่ม	ค่าความแม่นยำ (%)	จำนวนกลุ่ม	ค่าความแม่นยำ (%)
D1	41	4(3/17/17/4/0)	7	42.56	5	42.56	4	24.39	3	24.39
D2	34	5(2/2/11/15/4)	4	44.52	4	44.44	5	63.53	6	63.33
D3	36	5(4/6/6/13/7)	4	43.87	3	27.03	3	43.89	3	42.70
D4	12	4 (1/2/4/0/5)	8	**66.67**	8	**66.67**	11	66.67	12	**66.67**
D5	36	5(14/9/2/2/9)	1	8.00	2	12.78	9	**82.78**	4	60.00
D6	42	5(5/5/19/11/2)	2	28.00	2	27.69	3	28.10	2	28.57
D7	39	5(1/11/10/14/3)	4	27.37	4	27.37	4	44.10	2	12.82

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

ตารางที่ 24 และ 25 แสดงประสิทธิผลการจัดกลุ่มด้วยอัลกอริทึม EM และ Cobweb มีรายละเอียดต่างๆ คือ ชุดข้อมูลทั้งหมด 7 ข้อ (D1 – D7) จำนวนกลุ่มที่แบ่งโดยมนุษย์ เทียบกับจำนวนกลุ่มที่แบ่งได้ในอัลกอริทึม EM และ Cobweb ซึ่งแบ่งวิธีการทดลองเป็น 4 แบบ คือ แบบปกติ แบบปกติ + พจนานุกรม แบบใช้เทคนิค LSA และ แบบใช้เทคนิค LSA ร่วมกับ พจนานุกรม แสดงผลเป็นค่าความแม่นยำ

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

ตารางที่ 26 ค่าความแม่นยำและค่า MSE เฉลี่ยรวมในแต่ละอัลกอริทึมด้วยวิธีการคิดเกรดแบบต่างๆ

อัลกอริทึม	แบบปกติ		ใช้ พจนานุกรม		เทคนิค LSA		เทคนิค LSA + ใช้พจนานุกรม	
	ค่าความแม่นยำ (%)	ค่า MSE	ค่าความแม่นยำ (%)	ค่า MSE	ค่าความแม่นยำ (%)	ค่า MSE	ค่าความแม่นยำ (%)	ค่า MSE
K-NN	74.98	**2.45**	**75.03**	2.63	73.64	4.12	73.16	3.55
ANN	81.09	**1.22**	**81.71**	1.30	79.50	1.38	79.26	1.33
Bayesian	73.52	**2.94**	**73.67**	3.18	72.38	4.33	71.98	3.18
คิดเกรดแบบดั้งเดิม	76.06	2.68	76.26	2.54	75.75	2.77	**76.37**	**2.34**
EM	79.53	2.20	79.84	**1.65**	**80.16**	2.28	79.22	2.06
Cobweb	80.99	2.01	**81.09**	**1.97**	79.84	2.70	79.95	2.08

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

ตารางที่ 26 แสดงค่าความแม่นยำและค่า MSE เฉลี่ยรวมของเทคนิคต่างๆในแต่ละอัลกอริทึม โดยอัลกอริทึมที่ให้ค่าความแม่นยำเฉลี่ยรวมสูงสุดคืออัลกอริทึม ANN แบบปกติร่วมกับการใช้พจนานุกรมเพื่อแก้ปัญหาคำเหมือน ได้ค่าความแม่นยำเฉลี่ยรวมเป็น 81.71% และค่า MSE เป็น 1.22 ถัดมาคืออัลกอริทึม Cobweb ได้ค่าความแม่นยำเฉลี่ยรวมเป็น 81.09% และค่า MSE เป็น 1.97 ถัดมาคืออัลกอริทึม EM ได้ค่าความแม่นยำเฉลี่ยรวมเป็น 80.16% และค่า MSE เป็น 2.22 ถัดมาคือวิธีคิดเกรดแบบดั้งเดิม ได้ค่าความแม่นยำเฉลี่ยรวมเป็น 76.37% และค่า MSE เป็น 2.34 ถัดมาคืออัลกอริทึม K-NN ได้ค่าความแม่นยำเฉลี่ยรวมเป็น 75.03% และค่า MSE เป็น 2.63 และสุดท้ายคืออัลกอริทึม Bayesian ได้ค่าความแม่นยำเฉลี่ยรวมเป็น 73.67% และค่า MSE เป็น 3.18

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

วิจารณ์

จากผลการทดลองในงานวิจัยนี้ จะวิจารณ์ผลการทดลอง 2 ส่วน คือ วิจารณ์ผลการทดลองบนอัลกอริทึมการเรียนรู้แบบมีผู้สอนระบบ และวิจารณ์ผลการทดลองบนอัลกอริทึมการเรียนรู้แบบไม่มีผู้สอนระบบ มีรายละเอียดดังนี้

1. วิจารณ์ผลการทดลองบนอัลกอริทึมการเรียนรู้แบบมีผู้สอนระบบ

ผลการทดลองในส่วนแรกทดลองบนอัลกอริทึมการเรียนรู้แบบมีผู้สอนระบบ ซึ่งทดลองบนอัลกอริทึม K-NN ANN และ Bayesian ตามลำดับ จากตารางที่ 26 ซึ่งแสดงค่าความแม่นยำและค่า MSE เฉลี่ยรวมของเทคนิคต่างๆในแต่ละอัลกอริทึมพบว่า ตัววัดประสิทธิผลทั้ง 2 ได้ค่าที่ค่อนข้างแปรผกผันกัน กล่าวคือ ถ้าค่าความแม่นยำเฉลี่ยรวมมาก ค่า MSE ที่ได้จะมีค่าน้อย เช่น อัลกอริทึม ANN แบบปกติ ได้ค่าความแม่นยำเฉลี่ยรวมเป็น 81.09% ค่า MSE เฉลี่ยรวมเป็น 1.22 ส่วนในอัลกอริทึม K-NN แบบปกติ ได้ค่าความแม่นยำเฉลี่ยรวมเป็น 74.98% ค่า MSE เฉลี่ยรวมเป็น 2.45 จะเห็นได้ว่าค่าความแม่นยำเฉลี่ยรวมของอัลกอริทึม K-NN น้อยกว่าอัลกอริทึม ANN ซึ่งค่า MSE ที่ได้ในอัลกอริทึม ANN มีค่าน้อยกว่าอัลกอริทึม K-NN

ผลจากตารางที่ 26 เห็นได้ว่าการนำพจนานุกรมเข้ามาช่วยแก้ปัญหากรณีเกิดคำเหมือนช่วยทำให้ได้ค่าความแม่นยำเฉลี่ยรวมเพิ่มขึ้น 0.05% ในอัลกอริทึม K-NN 0.62% ในอัลกอริทึม ANN และ 0.15% ในอัลกอริทึม Bayesian ส่วนค่า MSE เพิ่มขึ้น 0.18 ในอัลกอริทึม K-NN 0.08 ในอัลกอริทึม ANN และ 0.23 ในอัลกอริทึม Bayesian เห็นได้ว่าการนำพจนานุกรมเข้ามาช่วยแก้ปัญหาคำเหมือนได้ค่าความแม่นยำเฉลี่ยรวมและค่า MSE ใกล้เคียงหรือไม่แตกต่าง เนื่องจากในชุดคำตอบมีการใช้คำเหมือนค่อนข้างน้อย ทำให้ยังไม่เห็นผลอย่างชัดเจน

ในกรณีที่ใช้เทคนิค LSA ซึ่งเป็นเทคนิคการลดขนาดของข้อมูลและหาความหมายที่ซ่อนอยู่ โดยการแปลงชุดข้อมูลในกระบวนการเริ่มต้น ผลที่ได้ปรากฏว่าได้ค่าความแม่นยำเฉลี่ยรวมลดลง 1.34% ในอัลกอริทึม K-NN 1.59% ในอัลกอริทึม ANN และ 1.14% ในอัลกอริทึม Bayesian ส่วนค่า MSE เพิ่มขึ้น 1.67 ในอัลกอริทึม K-NN 0.16 ในอัลกอริทึม ANN และ 1.39 ในอัลกอริทึม Bayesian และการนำพจนานุกรมเข้ามาช่วยกลับทำให้ค่าความแม่นยำลดลง 0.48% ในอัลกอริทึม K-NN 0.24% ในอัลกอริทึม ANN และ 0.4% ในอัลกอริทึม Bayesian ส่วนค่า MSE กลับลดลง 0.57 ในอัลกอริทึม K-NN 0.05 ในอัลกอริทึม ANN และ 1.15 ในอัลกอริทึม Bayesian แสดงให้เห็นว่า

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

การนำพจนานุกรมมาแก้ปัญหากรณีการเกิดคำเหมือนให้ผลในทางลบในเทคนิค LSA เนื่องจากในเทคนิค LSA มีจุดประสงค์เพื่อลดขนาดของข้อมูลที่มีขนาดใหญ่เพื่อหาความสัมพันธ์ของคำในคำตอบ ซึ่งใช้ได้ผลหรือเห็นผลได้ชัดเจนบนชุดข้อมูลที่มีขนาดใหญ่ (การใช้คำที่มีความแตกต่างกันมากๆ) แต่ไม่ค่อยเห็นผลกับข้อมูลที่มีขนาดเล็ก ซึ่งในการทดลองได้ใช้พจนานุกรมเข้ามาแก้ปัญหาคำเหมือน โดยคำเหมือนจะถูกจัดกลุ่มรวมกันทำให้ขนาดของข้อมูลเล็กลง จึงส่งผลต่อค่าความแม่นยำในด้านลบ

จากผลการทดลอง อัลกอริทึมการเรียนรู้แบบมีผู้สอนระบบที่ให้ค่าความแม่นยำเฉลี่ยรวมสูงสุดคือ อัลกอริทึม ANN ที่ใช้พจนานุกรมเข้ามาแก้ปัญหาคำเหมือน ด้วยค่าความแม่นยำเฉลี่ยรวมเป็น 81.71% รองมาคือ อัลกอริทึม K-NN ที่ใช้พจนานุกรมเข้ามาแก้ปัญหาคำเหมือน ด้วยค่าความแม่นยำเฉลี่ยรวมเป็น 75.03% และสุดท้ายคืออัลกอริทึม Bayesian ที่ใช้พจนานุกรมเข้ามาแก้ปัญหาคำเหมือน ด้วยค่าความแม่นยำเฉลี่ยรวมเป็น 73.67% แสดงให้เห็นว่าอัลกอริทึม ANN ซึ่งเป็นอัลกอริทึมที่ใช้สำหรับจดจำรูปแบบที่หลากหลาย เหมาะสมสำหรับการนำไปใช้พยากรณ์คะแนนในข้อมูลรูปแบบข้อความ หรือในระบบการตรวจข้อสอบอัตนัยแบบอัตโนมัติ

โดยทั่วไป ปัจจัยที่ส่งผลต่อค่าความแม่นยำในอัลกอริทึมการเรียนรู้แบบมีผู้สอนระบบ คือ ปริมาณของชุดข้อมูลสอนระบบต้องมีมากพอเพื่อใช้ในการสอนระบบให้เครื่องเกิดการเรียนรู้ที่หลากหลาย เพื่อให้ได้ค่าความแม่นยำในการตรวจที่สูง ซึ่งในทางปฏิบัติไม่เหมาะสม และไม่ยืดหยุ่นในการนำไปใช้จริง อีกทั้งไม่สามารถเตรียมได้ทุกระดับเกรด เช่น ระดับเกรด F ในงานวิจัยนี้พยายามหาวิธีแก้ปัญหาตรงจุดนี้ จึงเป็นที่มาของการทดลองบนอัลกอริทึมการเรียนรู้แบบไม่มีผู้สอนระบบ

2. วิจารณ์ผลการทดลองบนอัลกอริทึมการเรียนรู้แบบไม่มีผู้สอนระบบ

ในส่วนของการตรวจข้อสอบพบว่า คำตอบบางคำตอบมีการเขียนหรือสะกดคำผิด การเขียนคำตอบบางข้ออาจจะมีการใช้คำที่แตกต่างหรือมีความหลากหลายของการใช้คำกันมาก ซึ่งบางคำไม่พบในพจนานุกรมที่นำมาใช้ และการใช้คำที่นอกเหนือจากคำในเฉลย ทำให้การคำนวณคลาดเคลื่อนจากความเป็นจริงได้ เห็นได้ว่าการใช้ถุงของคำ (Bag of word) ขาดความสามารถในการค้นหาความหมายในคำตอบ แม้ว่าระดับความเหมือนจะกำหนดไว้สำหรับการให้เกรดที่ 0.8 ตามความคล้ายและความหมายของเนื้อหา แต่อาจจะส่งผลต่อตัวแทนในกลุ่มคำตอบได้ ดังนั้นในงานวิจัยนี้จึงทำการทดลองหาช่วงคะแนนที่เหมาะสมสำหรับการให้เกรดดังตารางที่ 20-23 แสดงค่า

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

ความแม่นยำเฉลี่ยในแต่ละช่วง Threshold A ของชุดข้อมูลทั้งหมด เห็นได้ว่าอัลกอริทึม Cobweb ได้ค่าความแม่นยำเฉลี่ยรวมสูงกว่าอัลกอริทึม EM และวิธีดังเดิมในการคิดเกรดคำตอบได้ค่าความแม่นยำเฉลี่ยรวมต่ำที่สุด และเกณฑ์ที่เหมาะสมของ Threshold A จะอยู่ในช่วง 0.5-0.7 เฉลี่ยคือช่วง 0.65

จากตารางที่ 24 และ 25 แสดงประสิทธิผลการจัดกลุ่มด้วย EM และ Cobweb แม้ว่าผู้ตรวจจะกำหนดจำนวนระดับเกรดไว้ 5 เกรด แต่ในความเป็นจริงคำตอบอาจจะมีไม่ครบทุกระดับเกรด เช่น ชุดข้อมูลที่ D4 มีคำตอบทุกระดับเกรดยกเว้นเกรด D นอกจากนี้บางชุดข้อมูลมีจำนวนคำตอบในแต่ละกลุ่มไม่เท่ากัน (ดูได้จากชุดข้อมูล D7) และจากผลการทดลองวัดประสิทธิผลการจัดกลุ่มด้วยอัลกอริทึม EM และ Cobweb พบว่าโดยภาพรวมอัลกอริทึม Cobweb สามารถจัดกลุ่มคำตอบได้ถูกต้องมากกว่าอัลกอริทึม EM อย่างเห็นได้ชัด ดังนั้นการจัดกลุ่มด้วยการสร้างต้นไม้การจำแนกในอัลกอริทึม Cobweb ได้ผลดีกว่า การจัดกลุ่มด้วยการประเมินค่าความคาดหวังมากสุดในอัลกอริทึม EM

จากตารางที่ 26 แสดงการเปรียบเทียบประสิทธิผลของวิธีการคิดเกรดคำตอบแบบต่างๆในแต่ละอัลกอริทึม เห็นได้ว่าผลลัพธ์ของอัลกอริทึมการเรียนรู้แบบไม่มีผู้สอนให้ค่าความแม่นยำเฉลี่ยรวมใกล้เคียงกับผลลัพธ์ของอัลกอริทึมการเรียนรู้แบบมีผู้สอน และตัววัดประสิทธิผลทั้ง 2 ได้ค่าที่ค่อนข้างแปรผกผันกันเช่นเดียวกับผลการทดลองบนอัลกอริทึมการเรียนรู้แบบมีผู้สอนระบบ

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

สรุปและข้อเสนอแนะ

สรุป

งานวิจัยนี้มีวัตถุประสงค์เพื่อนำเสนอวิธีการตรวจข้อสอบอัตนัยภาษาไทยแบบอัตโนมัติ โดยใช้ขั้นตอนวิธีเรียนรู้แบบมีข้อมูลสอนระบบด้วยอัลกอริทึม K-NN ANN และ Bayesian และหาวิธีแก้ปัญหาและหาข้อจำกัดต่างๆที่พบในอัลกอริทึมการเรียนรู้แบบมีข้อมูลสอนระบบ เช่น ปัญหาการเตรียมชุดข้อมูลสอนระบบ ซึ่งไม่สามารถเตรียมได้ครบทุกระดับเกรด เป็นต้น ด้วยขั้นตอนวิธีการเรียนรู้แบบไม่มีข้อมูลสอนระบบคือ อัลกอริทึม EM และ Cobweb ร่วมกับเทคนิค LSA และเทคนิคการขยายคำขอ (Query Expansion) ด้วยพจนานุกรมอังกฤษ-ไทย มาช่วยเพิ่มประสิทธิภาพในการให้คะแนนคำตอบแบบอัตโนมัติ และสร้างต้นแบบโปรแกรมการตรวจข้อสอบอัตนัยภาษาไทยแบบอัตโนมัติ โดยใช้ชุดข้อสอบวิชาการโปรแกรมเชิงวัตถุจำนวน 7 ข้อ มีชุดคำตอบของนักศึกษา 55 คน และชุดเฉลย 1 ชุดในการทดลองโดยความยาวคำตอบแบบสั้น โดยทำการจำลองการให้คะแนนคำตอบของอัลกอริทึมการเรียนรู้แบบไม่มีข้อมูลสอนระบบเพื่อแสดงผลการจัดกลุ่มของอัลกอริทึม EM และ Cobweb และให้คะแนนกลุ่มคำตอบ รวมถึงจำลองการให้คะแนนรายคำตอบในโปรแกรมการตรวจข้อสอบ โดยสามารถตอบคำถามและกดปุ่มคำนวณคะแนนคำตอบในแต่ละข้อได้ทันที

จากผลการทดลองสรุปได้ว่า วิธีการให้เกรดด้วยอัลกอริทึมการเรียนรู้แบบมีผู้สอนกับวิธีการให้เกรดด้วยอัลกอริทึมการเรียนรู้แบบไม่มีผู้สอนได้ค่าความแม่นยำและค่า MSE เฉลี่ยรวมใกล้เคียงกัน โดยวิธีการนำเทคนิคการขยายคำค้นเพื่อแก้ปัญหาคำเหมือน สามารถช่วยเพิ่มประสิทธิภาพได้ แต่ไม่ได้ผลในเทคนิค LSA เนื่องจากปัจจัยในเรื่องของขนาดของข้อมูลที่ลดลงจากการจัดกลุ่มคำเหมือนรวมกัน และชุดข้อมูลที่นำมาทดลองมีขนาดเล็กทำให้ไม่เห็นผลอย่างชัดเจน เพราะเทคนิค LSA จะเห็นผลได้อย่างชัดเจนในชุดข้อมูลที่มีขนาดใหญ่

ในการนำระบบไปใช้งานจริง อัลกอริทึมการเรียนรู้แบบไม่มีผู้สอนมีโอกาสนำไปใช้ในการตรวจข้อสอบจริงได้มากกว่า เนื่องจากข้อสอบมีการเปลี่ยนแปลงอยู่เสมอ ดังนั้นในการนำไปใช้จริงจึงไม่สามารถเตรียมชุดข้อมูลสอนระบบ และมีข้อจำกัดในการเตรียมชุดข้อมูลสอนระบบหลายอย่าง เช่น ไม่สามารถเตรียมข้อมูลสอนระบบได้ทุกระดับ ตัวอย่างกรณีที่ต้องการเตรียมชุดคำตอบของเกรด F ไม่สามารถเตรียมได้ เนื่องจากคำตอบที่ผิดมีคำตอบได้หลายรูปแบบและมีได้ไม่จำกัด เป็นต้น

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

ข้อเสนอแนะ

งานวิจัยในอนาคตมีประเด็นที่น่าสนใจ เช่น โครงสร้างในการเขียนภาษาไทย และลำดับของคำในคำตอบ จำเป็นต้องใช้เทคนิคทางด้านการประมวลผลภาษาธรรมชาติเข้าช่วย ซึ่งในภาษาไทยยังขาดฐานข้อมูลที่นำมาใช้ในการวิเคราะห์โครงสร้างของประโยคอยู่ นอกจากนี้ในระบบการตรวจข้อสอบจำเป็นต้องใช้ชุดเฉลยมากกว่า 1 ชุดในคำนวณคะแนนคำตอบ เพื่อให้ได้ค่าความแม่นยำมากขึ้นในวิธีการคิดเกรดด้วยอัลกอริทึมการเรียนรู้แบบไม่มีผู้สอน และทดลองกับอัลกอริทึมการแบ่งกลุ่มที่ใช้การเรียนรู้แบบ Unsupervised ตัวอื่นเพิ่มเติมเพื่อเพิ่มค่าความแม่นยำและให้ผลลัพธ์ใกล้เคียงกับมนุษย์ตรวจมากขึ้น รวมถึงการเพิ่มชุดข้อมูลเพื่อใช้ในการตรวจสอบความถูกต้องของคำตอบของผู้ตอบที่เขียนคำเพิ่มเข้ามานอกเหนือจากคำในเฉลย เป็นต้น

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

เอกสารและสิ่งอ้างอิง

จิราพร ยิ่งกว่าชาติ, บุญเสริม กิจศิริกุล และ ประสงค์ ปราณีตพลกรัง. 2549. การทำนายผลสำเร็จการศึกษาของนักศึกษาระดับอุดมศึกษาด้วยการเรียนรู้แบบเบย์และการทำเหมืองข้อมูล, *ใน* **รายงานการประชุมวิชาการผลงานวิจัยและนวัตกรรมสู่การพัฒนาที่ยั่งยืน**. มหาวิทยาลัยศรีปทุม, กรุงเทพฯ.

อุษานาฎ เอื้ออภิสิทธิ์วงศ์ 2552. **การพยากรณ์อากาศโดยใช้โมเดลเครือข่ายความเชื่อเบย์**. วิทยานิพนธ์ปริญญาโท, มหาวิทยาลัยขอนแก่น.

Alias-i. 2008. **SVD Tutorial**. Available Source: http://alias-i.com/lingpipe, October 1, 2009.

Amasaki, S., Y. Takagi, O. Mizuno and T. Kikuno. 2003. A Bayesian belief network for assessing the likelihood of the fault content, pp. 215-226. *In* **Proceedings of the 14th International Symposium on Software Reliability Engineering.** 17-20 Nov. 2003.

Andreyev, Y., Y. Attali and J. Burstein. 2005. **Automatic Essay Scoring System.** United States Patent No. 2005/0142,529.

Attali, Y. and J. Burstein. 2006. Automated Essay Scoring with E-Rater V.2. **Technology and Assessment Study Collaborative** 4 (3): 31.

Baoli, L., L. Qin and Y. Shiwen. 2004. An adaptive k-nearest neighbor text categorization strategy. **ACM Transactions on Asian Language Information Processing** 3: 215-226.

Bin, L., L. Jun, Y. Jian-Min and Z. Qiao-Ming. 2008. Automated Essay Scoring Using the K-NN Algorithm, pp. 735-738. *In* **Proceedings of the Computer Science and Software Engineering.** 12-14 Dec. 2008.

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

Botros, N.M., M. Siddiqi and M.Z. Deiri. 1993. Automatic speech recognition using hidden Markov models and artificial neural networks, pp. 1770-1775. *In* **Proceedings of the IEEE International Conference on Neural Networks.**

Burstein, J.C. and M. Chodorow. 1999. Automated essay scoring for nonnative English speakers**,** pp. 68-75. *In* **Proceedings of a Symposium on Computer Mediated Language Assessment and Evaluation in Natural Language Processing**. Pennsylvania, USA.

_____, C. Leacock and R. Swartz. 2001. Automated evaluation of essays and shot answers. *In* **Proceedings of the 5th CAA Conference.** Loughborough, UK.

_____, R.M. Kaplan, S. Wolff and C. Lu. 2000. **Automatic Essay Scoring System Using Content-Based Techniques**. United States Patent No. 6115683.

Chen, Y.Y., C.L. Liu, T.H. Chang and C.H. Lee. 2010. An Unsupervised Automated Essay-Scoring System. **Journal IEEE Intelligent System** 25: 61-67.

Cover, T. and P. Hart. 1967. Nearest neighbor pattern classification. **IEEE Transactions on Information Theory** 13: 21–27.

Das, A. and D. Kempe. 2008. Algorithms for subset selection in linear regression, pp. 45-54. *In* **Proceedings of the 40th annual ACM symposium on Theory of computing**. 17-20 May 2008. New York, USA.

Dellaert, F. 2002. **The Expectation Maximization Algorithm**. Available Source: http:// www.cc.gatech.edu/~dellaert/em-paper.pdf, December 11, 2009.

Dempster, A.P., N.M. Laird and D.B. Rubin. 1977. Maximum likelihood from incomplete data via the EM algorithm. **Journal of the Royal Statistical Society** 39 (1): 1–38.

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

Du, L., H. Jin, O. de Vel and N. Liu. 2008. A Latent Semantic Indexing and WordNet based Information Retrieval Model for Digital Forensics, pp. 70-75. *In* **Proceedings of the IEEE International Conference on Intelligence and Security Informatics.** 17-20 June 2008**.**

Fenton, N., M. Neil and D. Marquez. 2008. Using Bayesian networks to predict software defects and reliability, pp. 701-712. *In* **Proceedings of the Institution of Mechanical Engineers.**

Fisher, D.H. 1987. Knowledge Acquisition Via Incremental Conceptual Clustering. **Machine Learning** 2: 139-172.

Furnas, G.W., S. Deerwester, S.T. Dumais, T.K. Landauer, R.A. Harshman, L.A. Streeter and K.E. Lochbaum. 1988. Information retrieval using a singular value decomposition model of latent semantic structure, pp. 465-480. *In* **Proceedings of the 11th annual international ACM SIGIR conference on Research and development in information retrieval**. New York, USA.

Gauch, S. and J. Wang. 1997. A corpus analysis approach for automatic query expansion, pp. 278-284. *In* **Proceedings of the 6th international conference on Information and knowledge management.** New York, USA**.**

_____, _____, and S.M. Rachakonda. 1999. A corpus analysis approach for automatic query expansion and its extension to multiple databases. **ACM Transactions on Information Systems (TOIS)** 17: 250-269.

Gopi, E.S., N. Lakshmanan, T. Gokul, S. KumaraGanesh and P.R. Shah. 2006. Digital Image Forgery Detection using Artificial Neural Network and Auto Regrssive Coefficients, pp. 194-197. *In* **Proceedings of the Canadian Conference on Electrical and Computer Engineering.** 7-10 May 2006. Ottawa, Canada.

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

Gregory, M., R. Scata and E. Brown. 2006. Web document classification using machine learning clustering algorithms. **Journal of Computing Sciences in Colleges** 21: 276-277.

Kanejiya, D., A. Kumary and S. Prasad. 2003. Automatic evaluation of students' answers using syntactically enhanced LSA, pp. 53-60. *In* **Proceedings of the HLT-NAACL 03 workshop on Building educational applications using natural language processing**. Pennsylvania, USA.

Landauer, T.K., P.W. Foltz and D. Laham. 1998. Introduction to latent semantic analysis. **Discourse Processes** 25: 259-284.

Larkey, L.S. 1998. Automatic essay grading using text categorization techniques, pp. 90-95. *In* **Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval**. New York, USA.

Loraksa, C. and R. Peachavanish. 2007. Automatic Thai-Language Essay Scoring Using Neural Network and Latent Semantic Analysis, pp. 400-402. *In* **Proceedings of the First Asia International Conference on Modeling and Simulation**. 27-30 March 2007. Phuket, Thailand.

Maletic, J.I. and N. Valluri. 1999. Automatic Software Clustering via Latent Semantic Analysis, pp. 251 – 254. *In* **Proceedings of the 14th IEEE International Conference on Automated Software Engineering.** 12-15 October 1999. Washington, DC, USA.

Mirkin, B. 2001. Reinterpreting the Category Utility Function. **Kluwer Academic Publishers** 45: 219-228.

Nagata, R., J. Kakegawa and Y. Yabuta. 2009. A Topic-Independent Method for Automatically Scoring Essay Content Rivaling Topic-Dependent Methods, pp. 88-92. *In* **Proceedings of the IEEE International Conference on Advanced Learning Technologies**.

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

Nigam, K., A.K. Mccallum, S. Thrun and T. Mitchell. 2000. Text Classification from Labeled and Unlabeled Documents using EM. **Machine Learning - Special issue on information retrieval** 39: 103-134.

Page, E.B. 1966. The Imminence of Grading Essays by Computer. **Phi Delta Kappa International** 47(5): 238-243.

Palmer, J., R. Williams and H. Dreher. 2002. Automated Essay Grading System Applied to a First Year University Subject – How Can We do it Better?, pp. 1221-1229. *In* **Proceedings of the Informing Science and IT Education Conference.** 01 Jun 2002**.** Cork, Ireland**.**

Rudner, L.M. and T. Liang. 2002. Automated Essay Scoring Using Bayes' Theorem. **The Journal of Technology, Learning, and Assessment** 1 (2): 1-22.

Sadat, F., A. Maeda, M. Yoshikawa and S. Uemura. 2001. Query Expansion Techniques for the CLEF Bilingual Track, pp. 177-184. *In* **Proceedings of the Second Workshop of the Cross-Language Evaluation Forum on Evaluation of Cross-Language Information Retrieval Systems**. London, UK.

Sahoo, N., J. Callan, R. Krishnan, G. Duncan and R. Padman. 2006. Incremental hierarchical clustering of text documents, pp. 357-366. *In* **Proceedings of the 5th ACM international conference on Information and knowledge management**. New York, USA.

Sargur, S., J. Collins, R. Srihari, H. Srinivasan, S. Shetty, and J. Brutt-Griffler. 2008. Automatic scoring of short handwritten essays in reading comprehension tests. **Artificial Intelligence** 172: 300-324.

Semire, D. 2006. Automated essay scoring. **Turkish Online Journal of Distance Education-TOJDE** 7 (1): 49-62.

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

Shin, S.H., S.C. Lee, S.W. Kim, J. Lee and E.G. Im. 2010. Efficient shortest path finding of K-Nearest Neighbor objects in road network databases, pp. 1661-1665. *In* **Proceedings of the 2010 ACM Symposium on Applied Computing.** New York, USA**.**

Soucy, P. and G.W. Mineau. 2001**.** A simple KNN algorithm for text categorization, pp. 647-648. *In* **Proceedings of the IEEE International Conference on Data Mining.**

Stefan, W. 2009. A Bayesian network approach to assess and predict software quality using activity-based quality models, pp. 1230-1241. *In* **Proceedings of the 5th International Conference on Predictor Models in Software Engineering.** New York, USA**.**

Thomas, W., S. Instvan, D. Carlos and L. Michael. 2009. Exploring compact reinforcement-learning representations with linear regression, pp. 591-598. *In* **Proceedings of the 25th Conference Annual Conference on Uncertainty in Artificial Intelligence.** New York, USA**.**

Tuomo, K. and E. Sutinen. 2003. Automatic assessment of the content of essays based on course materials, pp. 126-130. *In* **Proceedings of the 2th Information Technology: Research and Education.** 28 June - 1 July 2004.

_____, N. Myller, J. Timonen and E. Sutinen. 2005. Automatic essay grading with probabilistic latent semantic analysis, pp. 29-36. *In* **Proceedings of the 2th workshop on Building Educational Applications Using NLP.**

Valenti, F., F. Neri and A. Cucchiarelli. 2003. An overview of current research on automated essay grading. **Journal of Information Technology Education** 2: 319-330.

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

Xiaojun, W. and M. Leeser. 2007. K-means Clustering for Multispectral Images Using Floating-Point Divide, pp. 151-162. *In* **Proceedings of the 15th Annual IEEE Symposium on Foundations of Computer Science.** 23-25 April 2007.

Xiujuan, C., S. Shan, X. Chen and W. Gao. 2006. Local Linear Regression for Pose Invariant Face Recognition. **IEEE Transactions on Image Processing** 16: 1716-1725.

Xu, J. and W.B. Croft. 1996. Query Expansion Using Local and Global Document Analysis, pp. 4-11. *In* **Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval**. New York, USA.

Yanjun, L., C. Luo and S.M. Chung. 2008. Text Clustering with Feature Selection by Using Statistical Data. **IEEE Transactions on Knowledge and Data Engineering** 20: 641-652.

Yin, Z., M. Shokouhi and N. Craswell. 2009. Query Expansion Using External Evidence, pp. 362-374. *In* **Proceedings of the 31th European Conference on IR Research on Advances in Information Retrieval.**

Yuri, A.I., B. Bruce and P. Alex. 2001. Expectation Maximization for Weakly Labeled Data, pp. 218-225. *In* **Proceedings of the 18th International Conference on Machine Learning**. San Francisco, USA.

Zhang, X., Huang H. Huang and K. Zhang. 2009. K-NN Text Categorization Algorithm Based on Semantic Centre, pp. 249-252. *In* **Proceedings of the Information Technology and Computer Science.** 25-26 July 2009. Beijing, China.

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

ภาคผนวก

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

ภาคผนวก ก

รายละเอียดผลการทดลอง

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

ภาคผนวก ก

รายละเอียดผลการทดลอง

ในงานวิจัยนี้ รายละเอียดผลการทดลองจะแสดงค่าความแม่นยำในแต่ละข้อ และสรุปค่าความแม่นยำรวม โดยแยกแสดงรายละเอียดในแต่ละ Threshold เริ่มจาก A ถึง F ของแต่ละวิธีการให้เกรด ซึ่งแสดงรายละเอียดเฉพาะการใช้พจนานุกรมเข้ามาแก้ปัญหาคำเหมือนเท่านั้นดังตารางผนวกที่ ก1 ถึง ก15

ตารางผนวกที่ ก1 ค่าความแม่นยำของแต่ละช่วง Threshold A ของข้อที่ 1-7 ในวิธีการให้เกรดแบบดั้งเดิม

Threshold A	**ค่าความแม่นยำ (%)**						
	ข้อที่ 1	**ข้อที่ 2**	**ข้อที่ 3**	**ข้อที่ 4**	**ข้อที่ 5**	**ข้อที่ 6**	**ข้อที่ 7**
0.50	92.86	96.43	92.86	98.21	75.00	91.07	100.00
0.55	92.86	96.43	92.86	98.21	75.00	91.07	100.00
0.60	92.86	96.43	92.86	98.21	75.00	91.07	100.00
0.65	92.86	96.43	92.86	98.21	75.00	91.07	100.00
0.70	92.86	96.43	92.86	98.21	75.00	91.07	100.00
0.75	92.86	96.43	92.86	98.21	75.00	91.07	100.00
0.80	92.86	96.43	92.86	98.21	75.00	91.07	100.00
0.85	92.86	96.43	92.86	98.21	75.00	91.07	100.00
0.90	92.86	96.43	92.86	98.21	75.00	89.29	100.00
0.95	94.64	96.43	92.86	98.21	75.00	89.29	100.00
1.00	94.64	96.43	92.86	98.21	75.00	89.29	100.00
ค่าความแม่นยำรวม (%)	93.18	96.43	92.86	98.21	75.00	90.58	100.00

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

ตารางผนวกที่ ก2 ค่าความแม่นยำของแต่ละช่วง Threshold B ของข้อที่ 1-7 ในวิธีการให้เกรดแบบดั้งเดิม

Threshold B	**ค่าความแม่นยำ (%)**						
	ข้อที่ 1	**ข้อที่ 2**	**ข้อที่ 3**	**ข้อที่ 4**	**ข้อที่ 5**	**ข้อที่ 6**	**ข้อที่ 7**
0.40	64.29	91.07	89.29	94.64	80.36	87.50	80.00
0.45	60.71	91.07	87.50	94.64	80.36	89.29	80.00
0.50	62.50	91.07	87.50	94.64	80.36	89.29	80.00
0.55	66.07	91.07	87.50	94.64	80.36	89.29	80.00
0.60	66.07	91.07	87.50	94.64	80.36	89.29	80.00
0.65	66.07	91.07	87.50	94.64	80.36	89.29	80.00
0.70	66.07	91.07	87.50	94.64	80.36	89.29	80.00
0.75	66.07	91.07	87.50	94.64	80.36	89.29	80.00
0.80	66.07	91.07	87.50	94.64	80.36	89.29	80.00
0.85	64.29	91.07	87.50	94.64	80.36	87.50	80.00
0.90	64.29	91.07	87.50	94.64	80.36	87.50	80.00
ค่าความแม่นยำรวม (%)	64.77	91.07	87.66	94.64	80.36	88.80	80.00

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

ตารางผนวกที่ ก3 ค่าความแม่นยำของแต่ละช่วง Threshold C ของข้อที่ 1-7 ในวิธีการให้เกรดแบบดั้งเดิม

Threshold C	**ค่าความแม่นยำ (%)**						
	ข้อที่ 1	**ข้อที่ 2**	**ข้อที่ 3**	**ข้อที่ 4**	**ข้อที่ 5**	**ข้อที่ 6**	**ข้อที่ 7**
0.30	64.29	76.79	85.71	89.29	94.64	66.07	78.18
0.35	66.07	76.79	85.71	89.29	94.64	62.50	78.18
0.40	66.07	78.57	85.71	91.07	94.64	62.50	81.82
0.45	66.07	78.57	87.50	91.07	94.64	64.29	81.82
0.50	67.86	78.57	87.50	91.07	94.64	64.29	81.82
0.55	67.86	78.57	87.50	91.07	94.64	64.29	81.82
0.60	67.86	78.57	87.50	91.07	94.64	64.29	81.82
0.65	67.86	78.57	87.50	91.07	94.64	64.29	81.82
0.70	67.86	78.57	87.50	91.07	94.64	64.29	81.82
0.75	67.86	78.57	87.50	91.07	94.64	64.29	81.82
0.80	67.86	78.57	87.50	91.07	94.64	64.29	81.82
ค่าความแม่นยำรวม (%)	67.05	78.25	87.01	90.75	94.64	64.13	81.16

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

ตารางผนวกที่ ก4 ค่าความแม่นยำของแต่ละช่วง Threshold D ของข้อที่ 1-7 ในวิธีการให้เกรดแบบดั้งเดิม

Threshold D	**ค่าความแม่นยำ (%)**						
	ข้อที่ 1	**ข้อที่ 2**	**ข้อที่ 3**	**ข้อที่ 4**	**ข้อที่ 5**	**ข้อที่ 6**	**ข้อที่ 7**
0.20	85.71	67.86	58.93	92.86	91.07	55.36	63.64
0.25	83.93	66.07	69.64	96.43	92.86	58.93	67.27
0.30	83.93	64.29	69.64	96.43	92.86	64.29	69.09
0.35	82.14	64.29	69.64	96.43	92.86	64.29	69.09
0.40	78.57	66.07	69.64	98.21	92.86	64.29	72.73
0.45	78.57	66.07	71.43	98.21	92.86	66.07	72.73
0.50	83.93	66.07	71.43	98.21	92.86	66.07	72.73
0.55	87.50	66.07	71.43	98.21	92.86	66.07	72.73
0.60	87.50	66.07	71.43	98.21	92.86	66.07	72.73
0.65	87.50	66.07	71.43	98.21	92.86	66.07	72.73
0.70	87.50	66.07	71.43	98.21	92.86	66.07	72.73
ค่าความแม่นยำรวม (%)	84.25	65.91	69.64	97.24	92.70	63.96	70.75

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

ตารางผนวกที่ ก5 ค่าความแม่นยำของแต่ละช่วง Threshold F ของข้อที่ 1-7 ในวิธีการให้เกรดแบบดั้งเดิม

Threshold F	**ค่าความแม่นยำ (%)**						
	ข้อที่ 1	**ข้อที่ 2**	**ข้อที่ 3**	**ข้อที่ 4**	**ข้อที่ 5**	**ข้อที่ 6**	**ข้อที่ 7**
0.10	41.07	41.07	60.71	91.07	50.00	33.93	47.27
0.15	41.07	39.29	48.21	87.50	48.21	25.00	43.64
0.20	39.29	39.29	48.21	85.71	48.21	19.64	38.18
0.25	37.50	39.29	46.43	85.71	48.21	17.86	38.18
0.30	35.71	37.50	46.43	87.50	48.21	17.86	34.55
0.35	32.14	37.50	44.64	87.50	48.21	16.07	34.55
0.40	26.79	37.50	44.64	87.50	48.21	16.07	34.55
0.45	23.21	37.50	44.64	87.50	48.21	16.07	34.55
0.50	23.21	37.50	44.64	87.50	48.21	16.07	34.55
0.55	23.21	37.50	44.64	87.50	48.21	16.07	34.55
0.60	23.21	37.50	44.64	87.50	48.21	16.07	34.55
ค่าความแม่นยำรวม (%)	31.49	38.31	47.08	87.50	48.37	19.16	37.19

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

ตารางผนวกที่ ก6 ค่าความแม่นยำของแต่ละช่วง Threshold A ของข้อที่ 1-7 ในวิธีการแบ่งกลุ่มด้วยอัลกอริทึม EM

Threshold A	**ค่าความแม่นยำ (%)**						
	ข้อที่ 1	**ข้อที่ 2**	**ข้อที่ 3**	**ข้อที่ 4**	**ข้อที่ 5**	**ข้อที่ 6**	**ข้อที่ 7**
0.50	80.00	96.36	83.64	96.36	74.55	34.55	98.18
0.55	80.00	96.36	83.64	96.36	74.55	90.91	98.18
0.60	80.00	96.36	92.73	96.36	74.55	90.91	98.18
0.65	80.00	96.36	92.73	96.36	74.55	90.91	98.18
0.70	80.00	96.36	92.73	98.18	74.55	90.91	98.18
0.75	80.00	96.36	92.73	98.18	74.55	90.91	98.18
0.80	94.55	96.36	92.73	98.18	74.55	90.91	98.18
0.85	94.55	96.36	92.73	98.18	74.55	90.91	98.18
0.90	94.55	96.36	92.73	98.18	74.55	90.91	98.18
0.95	94.55	96.36	92.73	98.18	74.55	90.91	98.18
1.00	94.55	96.36	92.73	98.18	74.55	90.91	98.18
ค่าความแม่นยำรวม (%)	86.61	96.36	91.08	97.52	74.55	85.79	98.18

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

ตารางผนวกที่ ก7 ค่าความแม่นยำของแต่ละช่วง Threshold B ของข้อที่ 1-7 ในวิธีการแบ่งกลุ่มด้วยอัลกอริทึม EM

Threshold B	**ค่าความแม่นยำ (%)**						
	ข้อที่ 1	**ข้อที่ 2**	**ข้อที่ 3**	**ข้อที่ 4**	**ข้อที่ 5**	**ข้อที่ 6**	**ข้อที่ 7**
0.40	60.00	92.73	89.09	96.36	81.82	90.91	47.27
0.45	67.27	92.73	89.09	96.36	81.82	41.82	47.27
0.50	67.27	92.73	87.27	96.36	81.82	41.82	80.00
0.55	67.27	92.73	87.27	96.36	81.82	90.91	80.00
0.60	67.27	92.73	89.09	90.91	81.82	90.91	80.00
0.65	67.27	92.73	89.09	90.91	81.82	90.91	80.00
0.70	63.64	92.73	89.09	96.36	81.82	90.91	80.00
0.75	63.64	92.73	89.09	96.36	81.82	90.91	80.00
0.80	67.27	92.73	89.09	96.36	81.82	90.91	80.00
0.85	67.27	92.73	89.09	96.36	81.82	90.91	80.00
0.90	67.27	92.73	89.09	96.36	81.82	90.91	80.00
ค่าความแม่นยำรวม (%)	65.95	92.73	88.76	95.37	81.82	81.98	74.05

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

ตารางผนวกที่ ก8 ค่าความแม่นยำของแต่ละช่วง Threshold C ของข้อที่ 1-7 ในวิธีการแบ่งกลุ่มด้วยอัลกอริทึม EM

Threshold C	**ค่าความแม่นยำ (%)**						
	ข้อที่ 1	**ข้อที่ 2**	**ข้อที่ 3**	**ข้อที่ 4**	**ข้อที่ 5**	**ข้อที่ 6**	**ข้อที่ 7**
0.30	69.09	52.73	89.09	92.73	47.27	65.45	72.73
0.35	65.45	80.00	89.09	92.73	96.36	65.45	81.82
0.40	65.45	80.00	89.09	92.73	96.36	65.45	60.00
0.45	69.09	80.00	89.09	92.73	96.36	63.64	60.00
0.50	69.09	80.00	83.64	92.73	96.36	63.64	81.82
0.55	69.09	80.00	83.64	92.73	96.36	65.45	81.82
0.60	69.09	80.00	89.09	90.91	96.36	65.45	81.82
0.65	69.09	80.00	89.09	90.91	96.36	65.45	81.82
0.70	65.45	80.00	89.09	92.73	96.36	65.45	81.82
0.75	65.45	80.00	89.09	92.73	96.36	65.45	81.82
0.80	69.09	80.00	89.09	92.73	96.36	65.45	81.82
ค่าความแม่นยำรวม (%)	67.77	77.52	88.10	92.40	91.90	65.12	77.03

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

ตารางผนวกที่ ก9 ค่าความแม่นยำของแต่ละช่วง Threshold D ของข้อที่ 1-7 ในวิธีการแบ่งกลุ่มด้วยอัลกอริทึม EM

Threshold D	**ค่าความแม่นยำ (%)**						
	ข้อที่ 1	**ข้อที่ 2**	**ข้อที่ 3**	**ข้อที่ 4**	**ข้อที่ 5**	**ข้อที่ 6**	**ข้อที่ 7**
0.20	89.09	67.27	67.27	98.18	94.55	65.45	72.73
0.25	89.09	65.45	72.73	100.00	45.45	67.27	60.00
0.30	89.09	65.45	72.73	100.00	45.45	67.27	60.00
0.35	52.73	67.27	72.73	100.00	94.55	67.27	72.73
0.40	52.73	67.27	72.73	100.00	94.55	67.27	65.45
0.45	89.09	67.27	72.73	100.00	94.55	36.36	65.45
0.50	89.09	67.27	56.36	100.00	94.55	36.36	72.73
0.55	89.09	67.27	56.36	100.00	94.55	67.27	72.73
0.60	89.09	67.27	72.73	94.55	94.55	67.27	72.73
0.65	89.09	67.27	72.73	94.55	94.55	67.27	72.73
0.70	67.27	67.27	72.73	100.00	94.55	67.27	72.73
ค่าความแม่นยำรวม (%)	80.50	66.94	69.26	98.84	85.62	61.48	69.09

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

ตารางผนวกที่ ก10 ค่าความแม่นยำของแต่ละช่วง Threshold F ของข้อที่ 1-7 ในวิธีการแบ่งกลุ่มด้วยอัลกอริทึม EM

Threshold F	**ค่าความแม่นยำ (%)**						
	ข้อที่ 1	**ข้อที่ 2**	**ข้อที่ 3**	**ข้อที่ 4**	**ข้อที่ 5**	**ข้อที่ 6**	**ข้อที่ 7**
0.10	85.45	85.45	80.00	90.91	81.82	80.00	89.09
0.15	85.45	85.45	60.00	89.09	81.82	78.18	89.09
0.20	85.45	85.45	60.00	89.09	81.82	78.18	89.09
0.25	85.45	36.36	60.00	89.09	47.27	78.18	76.36
0.30	85.45	36.36	60.00	89.09	47.27	78.18	76.36
0.35	41.82	36.36	60.00	89.09	47.27	78.18	76.36
0.40	41.82	36.36	60.00	89.09	47.27	78.18	32.73
0.45	41.82	36.36	60.00	89.09	47.27	14.55	32.73
0.50	41.82	36.36	43.64	89.09	47.27	14.55	32.73
0.55	41.82	36.36	43.64	89.09	47.27	14.55	32.73
0.60	41.82	36.36	43.64	87.27	47.27	14.55	32.73
ค่าความแม่นยำรวม (%)	61.65	49.75	57.36	89.09	56.69	55.21	60.00

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

ตารางผนวกที่ ก11 ค่าความแม่นยำของแต่ละช่วง Threshold A ของข้อที่ 1-7 ในวิธีการแบ่งกลุ่มด้วยอัลกอริทึม Cobweb

Threshold A	**ค่าความแม่นยำ (%)**						
	ข้อที่ 1	**ข้อที่ 2**	**ข้อที่ 3**	**ข้อที่ 4**	**ข้อที่ 5**	**ข้อที่ 6**	**ข้อที่ 7**
0.50	40.00	96.36	92.73	87.27	74.55	36.36	40.00
0.55	41.82	96.36	92.73	98.18	74.55	90.91	98.18
0.60	96.36	96.36	92.73	98.18	74.55	90.91	98.18
0.65	96.36	96.36	92.73	98.18	74.55	90.91	98.18
0.70	96.36	96.36	92.73	98.18	74.55	90.91	98.18
0.75	96.36	96.36	92.73	98.18	74.55	90.91	98.18
0.80	96.36	96.36	92.73	98.18	74.55	90.91	98.18
0.85	96.36	96.36	92.73	98.18	74.55	90.91	98.18
0.90	96.36	96.36	92.73	98.18	74.55	90.91	98.18
0.95	96.36	96.36	92.73	98.18	74.55	90.91	98.18
1.00	96.36	96.36	92.73	98.18	74.55	90.91	98.18
ค่าความแม่นยำรวม (%)	86.28	96.36	92.73	97.19	74.55	85.95	92.89

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

ตารางผนวกที่ ก12 ค่าความแม่นยำของแต่ละช่วง Threshold B ของข้อที่ 1-7 ในวิธีการแบ่งกลุ่มด้วยอัลกอริทึม Cobweb

Threshold B	**ค่าความแม่นยำ (%)**						
	ข้อที่ 1	**ข้อที่ 2**	**ข้อที่ 3**	**ข้อที่ 4**	**ข้อที่ 5**	**ข้อที่ 6**	**ข้อที่ 7**
0.40	67.27	92.73	41.82	96.36	81.82	90.91	80.00
0.45	65.45	92.73	89.09	85.45	81.82	40.00	54.55
0.50	54.55	92.73	89.09	85.45	81.82	40.00	54.55
0.55	56.36	92.73	89.09	96.36	81.82	90.91	80.00
0.60	67.27	92.73	89.09	96.36	81.82	90.91	80.00
0.65	67.27	92.73	89.09	96.36	81.82	90.91	80.00
0.70	67.27	92.73	89.09	96.36	81.82	90.91	80.00
0.75	67.27	92.73	89.09	96.36	81.82	90.91	80.00
0.80	67.27	92.73	89.09	96.36	81.82	90.91	80.00
0.85	67.27	92.73	89.09	96.36	81.82	90.91	80.00
0.90	67.27	92.73	89.09	96.36	81.82	90.91	80.00
ค่าความแม่นยำรวม (%)	64.96	92.73	84.79	94.38	81.82	81.65	75.37

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

ตารางผนวกที่ ก13 ค่าความแม่นยำของแต่ละช่วง Threshold C ของข้อที่ 1-7 ในวิธีการแบ่งกลุ่มด้วยอัลกอริทึม Cobweb

Threshold C	**ค่าความแม่นยำ (%)**						
	ข้อที่ 1	**ข้อที่ 2**	**ข้อที่ 3**	**ข้อที่ 4**	**ข้อที่ 5**	**ข้อที่ 6**	**ข้อที่ 7**
0.30	67.27	54.55	85.45	90.91	40.00	65.45	81.82
0.35	67.27	80.00	45.45	90.91	96.36	65.45	81.82
0.40	69.09	80.00	49.09	92.73	96.36	65.45	81.82
0.45	70.91	80.00	89.09	89.09	96.36	61.82	56.36
0.50	67.27	80.00	89.09	89.09	96.36	61.82	56.36
0.55	65.45	80.00	89.09	92.73	96.36	65.45	81.82
0.60	69.09	80.00	89.09	92.73	96.36	65.45	81.82
0.65	69.09	80.00	89.09	92.73	96.36	65.45	81.82
0.70	69.09	80.00	89.09	92.73	96.36	65.45	81.82
0.75	69.09	80.00	89.09	92.73	96.36	65.45	81.82
0.80	69.09	80.00	89.09	92.73	96.36	65.45	81.82
ค่าความแม่นยำรวม (%)	68.43	77.69	81.16	91.74	91.24	64.79	77.19

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

ตารางผนวกที่ ก14 ค่าความแม่นยำของแต่ละช่วง Threshold D ของข้อที่ 1-7 ในวิธีการแบ่งกลุ่มด้วยอัลกอริทึม Cobweb

Threshold D	**ค่าความแม่นยำ (%)**						
	ข้อที่ 1	**ข้อที่ 2**	**ข้อที่ 3**	**ข้อที่ 4**	**ข้อที่ 5**	**ข้อที่ 6**	**ข้อที่ 7**
0.20	89.09	67.27	70.91	100.00	94.55	67.27	72.73
0.25	89.09	67.27	72.73	100.00	38.18	67.27	72.73
0.30	87.27	67.27	65.45	98.18	38.18	67.27	72.73
0.35	87.27	67.27	54.55	98.18	94.55	67.27	72.73
0.40	89.09	67.27	61.82	100.00	94.55	67.27	72.73
0.45	87.27	67.27	72.73	85.45	94.55	34.55	50.91
0.50	32.73	67.27	72.73	85.45	94.55	34.55	50.91
0.55	34.55	67.27	72.73	100.00	94.55	67.27	72.73
0.60	89.09	67.27	72.73	100.00	94.55	67.27	72.73
0.65	89.09	67.27	72.73	100.00	94.55	67.27	72.73
0.70	89.09	67.27	72.73	100.00	94.55	67.27	72.73
ค่าความแม่นยำรวม (%)	78.51	67.27	69.26	97.02	84.30	61.32	68.76

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

ตารางผนวกที่ ก15 ค่าความแม่นยำของแต่ละช่วง Threshold F ของข้อที่ 1-7 ในวิธีการแบ่งกลุ่มด้วยอัลกอริทึม Cobweb

Threshold F	**ค่าความแม่นยำ (%)**						
	ข้อที่ 1	**ข้อที่ 2**	**ข้อที่ 3**	**ข้อที่ 4**	**ข้อที่ 5**	**ข้อที่ 6**	**ข้อที่ 7**
0.10	87.27	80.00	81.82	89.09	78.18	80.00	87.27
0.15	87.27	80.00	80.00	89.09	78.18	80.00	87.27
0.20	87.27	80.00	80.00	89.09	78.18	80.00	87.27
0.25	87.27	36.36	80.00	89.09	47.27	80.00	87.27
0.30	85.45	36.36	72.73	90.91	47.27	80.00	87.27
0.35	85.45	36.36	43.64	90.91	47.27	80.00	87.27
0.40	85.45	36.36	43.64	90.91	47.27	80.00	87.27
0.45	83.64	36.36	43.64	87.27	47.27	14.55	32.73
0.50	21.82	36.36	43.64	87.27	47.27	14.55	32.73
0.55	21.82	36.36	43.64	87.27	47.27	14.55	32.73
0.60	21.82	36.36	43.64	87.27	47.27	14.55	32.73
ค่าความแม่นยำรวม (%)	68.59	48.26	59.67	88.92	55.70	56.20	67.44

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

ภาคผนวก ข

ผลงานตีพิมพ์

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

ภาคผนวก ข

ผลงานตีพิมพ์

ผลงานตีพิมพ์ในงานวิจัยนี้ ตีพิมพ์ในประเทศจำนวน 2 บทความ และตีพิมพ์ต่างประเทศ จำนวน 1 บทความ ดังนี้

ผลงานตีพิมพ์ในประเทศ

ชื่อผลงาน “Automated Thai-Language Essay Scoring using K-NN”

การประชุมทางวิชาการ ครั้งที่ 48 มหาวิทยาลัยเกษตรศาสตร์ วันที่ 3 – 5 กุมภาพันธ์ พ.ศ. 2553 ณ อาคารศูนย์การเรียนรวม 3 มหาวิทยาลัยเกษตรศาสตร์ บางเขน

หมายเหตุ เนื่องจากการทดลองในตัวเล่มวิทยานิพนธ์นี้ ได้มีการเปลี่ยนแปลงการกำหนดค่า K ในอัลกอริทึม K-NN ใหม่ โดยกำหนดค่า K = 3 (เดิมกำหนดค่า K = 1 และกำหนดเกณฑ์ในการให้เกรดจากค่าสถิติในชุดข้อมูลฝึก) แบ่งชุดข้อมูลฝึกในแต่ละระดับเกรดไม่เกิน 3 คำตอบ และทดลองโดยการเพิ่มการใช้พจนานุกรมมาแก้ปัญหาคำเหมือน และเพิ่มเทคนิค LSA เพื่อใช้ในการวิเคราะห์ผลเพิ่มเติม

ชื่อผลงาน “Automated Thai-Language Essay Scoring”

งานสัมมนาวิชาการนานาชาติ ICSEC & NCSEC 2010 The 1st International and the 14th National Computer Science and Engineering Conference วันที่ 17 – 19 พฤศจิกายน พ.ศ. 2553 ณ โรงแรมเซ็นทาราดวงตะวัน อ.เมือง จ.เชียงใหม่

หมายเหตุ เนื่องจากการทดลองในตัวเล่มวิทยานิพนธ์นี้ ได้มีการเปลี่ยนแปลงวิธีการทดลอง ดังนี้

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

ในอัลกอริทึม K-NN กำหนดค่า K = 3 (เดิมกำหนดค่า K = 1 และกำหนดเกณฑ์ในการให้เกรดจากค่าสถิติในชุดข้อมูลฝึก) แบ่งชุดข้อมูลฝึกในแต่ละระดับเกรดไม่เกิน 3 คำตอบ และทดลองโดยการเพิ่มการใช้พจนานุกรมมาแก้ปัญหาคำเหมือน และเพิ่มเทคนิค LSA เพื่อใช้ในการวิเคราะห์ผลเพิ่มเติม

ในอัลกอริทึม ANN และ BBN ได้เพิ่มการทดลองแบบปกติ แบบปกติร่วมกับพจนานุกรม และแบบการใช้เทคนิค LSA นอกจากนี้ได้เปลี่ยนวิธีการแปลงข้อมูลใหม่ จากเดิมทำการแทนที่คำเหมือนในเวกเตอร์เดิม เปลี่ยนไปเป็นการจับคำเหมือนรวมเป็นคอลัมน์เดียว เนื่องจากทำให้ขนาดของเมทริกลดลง และสะดวกต่อการนำไปใช้ในการคำนวณมากกว่า และเปลี่ยนชื่อ BBN เป็น Bayesian เนื่องจากโครงสร้างเป็นแบบ Naïve Bay

ผลงานตีพิมพ์ต่างประเทศ

ชื่อผลงาน “Automated Thai-Language Essay Scoring with Unsupervised Learning Algorithm”

งานสัมมนาวิชาการ The 6th International Conference on Knowledge, Information and Creativity Support Systems (KICSS 2011) วันที่ 22-24 ตุลาคม พ.ศ. 2554 ณ อาคาร AMSS Siyuan กรุงปักกิ่ง ประเทศจีน

หมายเหตุ เนื่องจากการทดลองในตัวเล่มวิทยานิพนธ์นี้ ได้มีการเปลี่ยนแปลงวิธีการทดลอง โดยทำการเปลี่ยนวิธีการแปลงข้อมูลใหม่ จากเดิมทำการแทนที่คำเหมือนในเวกเตอร์เดิม เปลี่ยนไปเป็นการจับคำเหมือนรวมเป็นคอลัมน์เดียว เนื่องจากทำให้ขนาดของเมทริกลดลง และสะดวกต่อการนำไปใช้ในการคำนวณมากกว่า

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

การตรวจข้อสอบอัตนัยภาษาไทยแบบอัตโนมัติโดยใช้ K-NN

Automated Thai-Language Essay Scoring using K-NN

สมมาตร อังคเศรณีกุล[1] และ ชุลีรัตน์ จรัสกุลชัย[1]

Sommart Aungkaseraneekul[1] and Chuleerat Jaruskulchai[1]

บทคัดย่อ

อัลกอริทึมทางด้านปัญญาประดิษฐ์ถูกนำมาใช้ในการสร้างระบบการตรวจข้อสอบอัตนัยแบบอัตโนมัติหลากหลายอัลกอริทึม เช่น อัลกอริทึม K-Nearest Neighbor (K-NN), Neural Network, Latent Semantic, และทฤษฎีของ Bays เป็นต้น ในงานวิจัยนี้เลือกอัลกอริทึม K-NN มาจำแนกคำตอบโดยเทียบกับผลการตรวจคำตอบโดยผู้สอน ชุดคำตอบของนักเรียนแต่ละคนและผลเฉลยจะถูกแทนเป็นโมเดลเวกเตอร์ (Vector Space Model) โดยลบคำหยุด (Stop word) ออกก่อน เพื่อลดขนาดของเวกเตอร์ลง ระบบน้ำหนัก ความถี่ของคำ (Term frequency) และความถี่ของเอกสารผกผัน (Inversed document frequency) นำมาใช้เพื่อเพิ่มความสำคัญของคำในชุดเอกสาร การคำนวณค่าความเหมือนระหว่างคำตอบนักเรียนกับผลเฉลยด้วยค่า cosine การเปรียบเทียบประสิทธิผลของอัลกอริทึมในการจำแนกโดยการแบ่งกลุ่มคะแนนออกเป็น 5 กลุ่ม และทำการทดลองกับข้อสอบวิชาการโปรแกรมเชิงวัตถุจำนวน 5 ข้อ และ 55 ชุดคำตอบของนักเรียน และค่าความแม่นยำโดยรวมอยู่ระหว่าง 76.73% – 92.73%

ABSTRACT

Many algorithms in the artificial intelligence field have been applied for automatic essay scoring system, such as K-Nearest Neighbor (K-NN), Neural Network, Latent Semantic, and Bays' Theory. In this research, K-NN algorithm is applied to classifying the students' answer sets with teacher's answer. Student's answers and teacher's answer are converted into Vector Space Model, stop words are removed to reduce the vector's size. To increase word significance, term frequency and inversed document frequency are applied. Similarity between student's answer set and teacher's answer set is computed using cosine similarity. The efficiency of algorithm is evaluated by classify the students' answer into 5 categories. Five questions and 55 students answer from object-orient programming with written in Thai are used to evaluation and accuracy value between 76.73% to 92.73%

Keywords: Automated Thai-Language Essay Scoring; K-NN algorithm; vector space model

S.Aungkaseraneekul: g5164145@ku.ac.th

1 ภาควิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยเกษตรศาสตร์

Department of Science, Faculty of Science, Kasetsart University.

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

คำนำ

การเรียนการสอนในปัจจุบัน จะเห็นได้ว่าในการประเมินการเรียนรู้ของนักเรียน โดยใช้ข้อสอบแบบเขียนตอบหรืออัตนัย ถูกนำมาใช้อย่างกว้างขว้าง เพื่อเปิดโอกาสให้นักเรียนแสดงทักษะทางความคิด, ความรู้ รวมถึงลำดับทางความคิดของนักเรียน ยกตัวอย่างเช่น การสังเคราะห์และการวิเคราะห์ ซึ่งใช้วัดความสามารถในการทดสอบได้ดีกว่าการใช้คำถามแบบมีตัวเลือกตอบ (Multiple-choice) เนื่องจากคำถามแบบมีตัวเลือกตอบนั้น มีข้อบกพร่องในการวัดในเรื่องของการสร้างทัศนคติการแก้ปัญหาของมนุษย์ แต่อย่างไรก็ตามเมื่อจำนวนข้อสอบมีปริมาณมากขึ้น ครูผู้ตรวจจะพบกับความยุ่งยากในการตรวจข้อสอบของนักเรียน เนื่องจากมีปัจจัยหลายปัจจัยมาเกี่ยวข้อง เช่น ปัจจัยในเรื่องของเวลา เพราะใช้เวลานานในการตรวจ, ความเที่ยงตรงในการตรวจ เพราะมีอารมณ์และความรู้สึกต่างๆมาเกี่ยวข้อง อาจทำให้การประเมินเกิดความผิดพลาดขึ้นได้

จากปัญหาเหล่านี้ จึงได้มีการคิดสร้างระบบการตรวจข้อสอบอัตนัยแบบอัตโนมัติขึ้น โดยใช้คอมพิวเตอร์และหลักการด้านปัญญาประดิษฐ์ (Artificial Intelligence) ต่างๆมาเป็นเครื่องมือหนึ่งในการสนับสนุนการตรวจข้อสอบอัตนัย เนื่องจากคอมพิวเตอร์ทำงานได้รวดเร็ว, ประหยัดเวลาในการตรวจของผู้ตรวจได้มาก, ลดปัญหาความไม่เที่ยงตรงในการตรวจ, ลดปัญหาความคงที่หรือความแน่นอนในการตรวจ และไม่มีเรื่องของอารมณ์, ความเมื่อยล้าที่เกิดขึ้นในมนุษย์มาเกี่ยวข้องด้วย

ในการสร้างระบบการตรวจข้อสอบอัตนัยแบบอัตโนมัตินั้น ปัจจุบันได้มีหลายๆประเทศสร้างและพัฒนาเป็นจำนวนมาก และมีการจดสิทธิบัตรแล้ว เช่น ประเทศอเมริกา เป็นต้น ซึ่งทฤษฎีและวิธีการสร้างแตกต่างกันออกไป ยกตัวอย่างเช่น ระบบการให้คะแนนข้อสอบอัตนัยแบบอัตโนมัติโดยใช้ Neural Network และ Latent Semantic [1], ระบบการให้คะแนนข้อสอบอัตนัยแบบอัตโนมัติโดยใช้ทฤษฎีของ Bays [2], ระบบการให้คะแนนข้อสอบอัตนัยแบบอัตโนมัติโดยใช้อัลกอริทึม K-NN [3], และระบบการให้คะแนนข้อเขียนอัตนัยแบบอัตโนมัติโดยใช้ e-rater [4][5] ซึ่งเป็นแอปพลิเคชั่นหนึ่งในการให้คะแนนข้อสอบแบบอัตนัย เป็นต้น โดยที่แต่ละภาษาก็จะมีกระบวนการทำงานของระบบแตกต่างกันออกไปขึ้นกับลักษณะของภาษานั้นๆ ในงานวิจัยนี้จะใช้ข้อสอบอัตนัยภาษาไทยมาทำการทดลอง โดยประยุกต์ใช้วิธีการและหลักการของประเทศต่างๆที่ได้ทำการทดลองไว้แล้ว ซึ่งสิ่งที่แตกต่างจากภาษาอื่น ยกตัวอย่างเช่น ภาษาไทยมีลักษณะการเขียน คือเขียนคำแต่ละคำติดกันหมด และไม่มีเครื่องหมายที่บ่งบอกว่าจบประโยคเหมือนในภาษาอังกฤษที่มีเครื่องหมายจุด (.) เพื่อบ่งบอกว่าจบประโยคแล้ว และแยกคำแต่ละคำด้วยช่องว่าง เป็นต้น ทำให้ภาษาไทยมีความยากในเรื่องของการแทนข้อมูล เพื่อให้อยู่ในรูปแบบที่สามารถนำไปประมวลผลได้

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

อุปกรณ์และวิธีการ

อุปกรณ์

1. เครื่องคอมพิวเตอร์ส่วนบุคคล หน่วยประมวลผล AMD Althlon(tm) 64 X2 Dual Core Processor 3800+ 2.00 GHz, 3.00 GB of RAM Physical Address Extension
2. โปรแกรม Eclipse Version 3.3.2 และโปรแกรม Microsoft Office 2007
3. โปรแกรมตัดคำภาษาไทย [11]
4. ระบบปฏิบัติการ Microsoft Windows XP Professional

วิธีการ

1. ข้อมูลที่ใช้ในการทดลอง

ในการทดลองสร้างระบบการตรวจให้คะแนนข้อสอบอัตนัยนั้น จะใช้ข้อสอบวิชา "การโปรแกรมเชิงวัตถุด้วยภาษาจาวา" ในระดับปริญญาตรี มหาวิทยาลัยเกษตรศาสตร์ บางเขน ภาควิชาวิทยาการคอมพิวเตอร์ โดยคัดเลือกมา 5 ข้อ จากทั้งหมด 17 ข้อ แต่ละข้อมีคำตอบของนักเรียนทั้งหมด 55 คน ซึ่งทำการบันทึกใหม่และจัดเก็บอยู่ในรูปแบบไฟล์ข้อความ (ไฟล์ Text) สถิติลักษณะคำตอบของนักเรียนแสดงดังตารางที่ 1

ตารางที่ 1 สถิติของข้อมูล

ข้อที่	จำนวนคำสูงสุดในคำตอบนักเรียน	จำนวนคำต่ำสุดในคำตอบนักเรียน	จำนวนคำเฉลี่ยในคำตอบนักเรียน	จำนวนคำในเฉลย
1	70	2	22.2041	38
2	78	1	31.8810	43
3	85	2	23.9756	41
4	68	10	33.0833	33
5	78	6	30.9744	16

จากตารางที่ 1 จำนวนคำสูงสุด, ต่ำสุด, และจำนวนคำเฉลี่ยในคำตอบนักเรียนหาได้จากการนับจำนวนคำแต่ละคำของนักเรียนแต่ละคนที่ได้จากการตัดคำด้วยโปรแกรมตัดคำภาษาไทย [11] แล้วหาค่าสูงสุด, ค่าต่ำสุด, และค่าเฉลี่ยตามลำดับ

2. การแปลงรูปแบบข้อมูลในรูปเวกเตอร์

ชุดคำตอบของนักเรียนทำการบันทึกใหม่และจัดเก็บในรูปของอิเล็กทรอนิกส์ไฟล์ คำตอบแต่ละข้อของนักเรียนจัดเก็บในรูปเวคเตอร์ของคำ การเขียนคำตอบแบบอัตนัยมีการเว้นวรรคตอบตามระบบเขียนหนังสือทั่ว ๆ ไป จึงมีความจำเป็นในการแยกคำจากสายอักขระของคำ เพื่อให้สามารถจัดในรูปของเวคเตอร์ทางคณิตศาสตร์ได้ คำที่มีการใช้ซ้ำในข้อสอบแต่ละข้อแทนด้วยความถี่ของคำ (Term Frequency) ใช้ระบบถ่วงน้ำหนักเพื่อให้

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

ความสำคัญกับคำด้วยความถี่ของเอกสารผกผัน (Inverted Document Frequency) สมการที่ 1 คือรปแบบของการแทนชุดคำตอบและคำเฉลย

$$d_j = (w_{1j}, w_{2j}, w_{3j}, \ldots, w_{mj}) \qquad (1)$$

โดยที่ d_j คือ คำตอบของนักเรียนคนที่ j

w_{ij} คือ ค่าน้ำหนักของแต่ละคำในคำตอบของนักเรียนคนที่ j ดังสมการที่ 2

$$w_{ij} = \text{TF-IDF}(i,j) = \text{TF}(i,j) * \log(N/\text{DF}(i)) \qquad (2)$$

โดยที่ TF(i,j) คือ ค่าความถี่ของคำที่ i ในคำตอบของนักเรียนคนที่ j

N คือ จำนวนนักเรียนทั้งหมดในชุดข้อมูล

DF(i) คือ จำนวนคำตอบของนักเรียนที่ปรากฏคำที่ i อยู่

มีขั้นตอนในการแปลงรูปแบบข้อมูลดังนี้

1) แปลงข้อมูลโดยการบันทึกใหม่เพื่อจัดในรูปอิเล็กทรอนิกส์ไฟล์ (ดูตัวอย่างข้อมูลดิบในตารางที่ 1)
2) จากนั้นนำไฟล์ที่ได้จากการบันทึก ไปตัดคำด้วยโปรแกรมตัดคำภาษาไทยจะได้ไฟล์ข้อความอีกไฟล์กลับมา
3) เมื่อตัดคำแล้ว ขั้นต่อไปจะต้องหาวิธีลดขนาดของเวกเตอร์หรือลบคำที่ไม่จำเป็นออก เนื่องจากขนาดของเวกเตอร์มีผลต่อการคิดเกรดและเวลาในการคำนวณของเครื่อง (ดูได้จากตารางที่ 5 และ 6) ในการทดลองจะทำการลบคำหยุด (Stop word) เช่น “และ”, “หรือ” เป็นต้น
4) ทำการนับความถี่ของคำที่เหลือจากการลบคำหยุดออกแล้ว (ตารางที่ 2)

ตารางที่ 1 ตัวอย่างคำตอบของนักเรียน 3 คน

ข้อที่ 1
นักเรียนคนที่ 1: 1. การสืบทอด (inheritance) (abstract) 2. polymorphism
นักเรียนคนที่ 2: สร้าง class ขึ้น โดยสามารถอ้างถึง class นั้นได้โดยไม่ต้องสร้างใหม่
นักเรียนคนที่ 3: ผู้ใช้สามารถสร้างตัวแปรประเภท object ขึ้นมาเองได้ โดย object ที่สร้างขึ้นมามีคุณสมบัติของ object อย่างชัดเจน เช่นมี Attribute method สามารถสืบทอดกัน

ตารางที่ 2 การแทนคำตอบในรูปความถี่ของคำ

นักเรียนคนที่	สืบทอด	inheritance	abstract	class	polymorphism	object	ตัวแปร
1	1	1	1	0	1	0	0
2	0	0	0	2	0	0	0
3	1	0	0	0	0	2	1

5) นำระบบน้ำหนักเข้ามาแปลงค่าความถี่ของคำใหม่ โดยใช้สมการ TF-IDF ดังสมการที่ 2

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

3. . ขั้นตอนการจำแนกด้วย K-Nearest Neighbor

เมื่อได้ข้อมูลที่พร้อมที่จะนำไปประมวลผลแล้ว ต่อไปจะทำการเขียนโปรแกรมเพื่อประมวลผลข้อมูลเพื่อคิดเกรด ในที่นี้เลือกเขียนโปรแกรมด้วยภาษาจาวา เนื่องจากมีฟังก์ชันต่างๆที่มีการพัฒนามากมายให้เลือกใช้ โดยจะทำการเขียนโปรแกรมเพื่ออ่านไฟล์ Text แล้วทำการกำหนดคำในแต่ละข้อ (รวมคำในไฟล์ผลเฉลยในข้อนั้นๆด้วย) ซึ่งคำที่ได้จะไม่ซ้ำกัน และทำการนับความถี่ของคำแต่ละคำแทนในรูปแบบของค่า TF-IDF

การคำนวณค่าความเหมือนในอัลกอริทึม K-NN

การวัดค่าความเหมือนในอัลกอริทึม K-NN มีหลายสมการที่ใช้วัด ยกตัวอย่างเช่น Euclidean distance, Cosine เป็นต้น ในการทดลองจะใช้การวัดแบบ Cosine โดยที่หลักการของอัลกอริทึม K-NN คือ เป็นเทคนิคที่เหมาะกับปัญหาแบบการจำแนก (classification) ซึ่งเทคนิคนี้จะแตกต่างจากเทคนิคอื่นตรงที่ไม่ใช้ข้อมูลฝึกหัด (training data) เพื่อมาสร้างโมเดล แต่จะใช้ชุดข้อมูลนั้นมาเป็นโมเดลเลย จึงเป็นการเรียนรู้แบบ lazy learning ซึ่งในการทดลองนี้ได้ทำการเลือก K-NN มาใช้เนื่องจากชุดข้อมูลที่ใช้มีไม่เพียงพอต่อการสร้างโมเดลการจำแนก ส่วนในการใช้งาน K-NN algorithm นั้นต้องระบุค่าตัวเลขจำนวนเต็มบวกให้กับตัวแปร K ด้วย ซึ่งค่านี้จะเป็นตัวบอกถึงลำดับความคล้ายกันมากที่สุด K ลำดับแรก เช่น 3-NN หมายถึง จะค้นหาลำดับแรกที่มีค่าความคล้ายกันมากที่สุด 3 ลำดับ แล้วทำการลงคะแนนหรือโหวตว่าข้อมูลที่ใช้ทำนายอยู่กลุ่มไหนซึ่งควรกำหนดเป็นเลขคี่ เพื่อหลีกเลี่ยงปัญหาในกรณีที่ผลโหวตเท่ากัน

ในการคำนวณค่าความเหมือนในการทดลองนี้ จะทำการวัดค่าความเหมือนของคำตอบนักเรียนแต่ละคนเทียบกับผลเฉลยในข้อนั้นๆ ถ้าได้ค่าความเหมือนมาก โอกาสที่จะได้คะแนนมากก็สูงขึ้นตาม ซึ่งในการคำนวณวัดค่าความเหมือนบนพื้นฐานข้ออัลกอริทึม K-NN นั้น จะแตกต่างจากวิธีการทั่วไป คือ จะทำการกำหนดค่า K เป็น 1 เท่านั้น เนื่องจากมีชุดเฉลยเพียงชุดเดียว ซึ่งสมการ Cosine ดังสมการที่ 3

$$S(d_i, d_j) = \cos(\alpha) = \frac{\sum_{k=1}^{m} w_{ki} \cdot w_{kj}}{\sqrt{\sum_{k=1}^{m} w_{ki}^2} \cdot \sqrt{\sum_{k=1}^{m} w_{kj}^2}} \qquad (3)$$

4. การจำแนกกลุ่มคะแนนและค่าความแม่นยำ (Accuracy)

ในการจำแนกกลุ่มตามเกณฑ์การให้คะแนน โดยจัดออกเป็น 5 กลุ่ม คือ A, B, C, D, และ F โดยที่เกรด A คำตอบของนักเรียนจะมีความเหมือนใกล้เคียงกับคำเฉลยมากที่สุดดังนั้นผลการคำนวณค่าความเหมือนระหว่างคำตอบนักเรียนกับผลเฉลยจะเรียงลำดับจากค่าความเหมือนสูงสุดไปค่าความเหมือนต่ำสุด การเปรียบเทียบประสิทธิภาพของอัลกอริทึม โดยคำนวณจากค่าความเหมือนเฉลี่ยระหว่างคำตอบของนักเรียนกับผลเฉลยในแต่ละเกรด ซึ่งจะกำหนดเกรดดังนี้คือ: g_1, g_2, …, g_c, โดยจะเกี่ยวเนื่องกับช่วงค่า threshold คือ L_1, L_2, …, L_{C+1} โดยที่ C คือจำนวนของเกรด [10] ได้ผลลัพธ์ดังตารางที่ 3

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

ตารางที่ 3 แสดงช่วงค่า threshold ของแต่ละเกรด

ข้อที่	L1	L2	L3	L4	L5	L6
1	0	0.042	0.147	0.149	0.234	1
2	0	0.028	0.077	0.083	0.237	1
3	0	0.041	0.074	0.083	0.197	1
4	0	0.019	0.215	0.235	0.401	1
5	0	0.024	0.074	0.091	0.134	1

หลังจากได้ช่วงค่า threshold ของแต่ละเกรดมาแล้ว จะนำคำตอบของนักเรียนที่ต้องการให้เกรดมาวัดค่าความเหมือนกับผลเฉลย แล้วดูว่าตกอยู่ในช่วงค่า threshold ของเกรดใดก็ให้เกรดนั้น เมื่อได้เกรดออกมาแล้วจะทำการเปรียบเทียบกับเกรดผลเฉลยที่ได้จากผู้ตรวจว่าตรงกันหรือไม่ จากนั้นคำนวณค่าความแม่นยำ (Accuracy) โดยจะนับจำนวนคำตอบที่ผลเฉลยให้เกรดตรงกับเกรดที่ระบบให้ แล้วทำการคิดเป็นอัตราร้อยละ ซึ่งสูตรที่ใช้ในการคิด ดังสมการที่ 4

$$Acc = \frac{TP + TN}{TP + FP + FN + TN} \qquad (4)$$

โดยที่ TP คือ จำนวนคำตอบที่ระบบให้ผลเป็นจริงและผลเฉลยให้ผลเป็นจริงด้วย (True Positive)

TN คือ จำนวนคำตอบที่ระบบให้ผลเป็นเท็จและผลเฉลยให้ผลเป็นเท็จด้วย (True Negative)

FP คือ จำนวนคำตอบที่ระบบให้ผลเป็นจริงแต่ผลเฉลยให้ผลเป็นเท็จ (False Positive)

FN คือ จำนวนคำตอบที่ระบบให้ผลเป็นเท็จแต่ผลเฉลยให้ผลเป็นจริง (False Negative)

ตารางที่ 4 ตัวอย่างผลลัพธ์ที่ได้จากการประมวลผลสำหรับเกรด A

ข้อที่ 1	ผลลัพธ์ที่ได้จากระบบ		
ผลลัพธ์ที่ได้จากเฉลย		จำนวนคำตอบที่ได้เกรด A	จำนวนคำตอบที่ไม่ได้เกรด A
	จำนวนคำตอบที่ได้เกรด A	7 (TP)	2 (FN)
	จำนวนคำตอบที่ไม่ได้เกรด A	5 (FP)	41 (TN)
ค่า Accuracy(%) = 87.27			

ผลการทดลอง

ในการทดลอง จะทำการวัดประสิทธิภาพด้วยการวัดค่าความแม่นยำ (Accuracy) เพื่อตรวจสอบว่าผลการทดลองที่ได้สอดคล้องกับเฉลยที่ผู้ตรวจให้เกรดมากน้อยเพียงใด (เกรดที่ได้จากระบบกับเกรดที่ได้จากเฉลย

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

ตรงกัน) ซึ่งสรุปชุดข้อมูลที่นำมาทดลองทั้งหมด 5 ข้อ โดยแยกเป็น 2 กรณี คือ กรณีที่ไม่ตัดคำหยุดและกรณีที่ตัดคำหยุด ดังตารางที่ 5 และ 6 ตามลำดับ

ตารางที่ 5 แสดงค่าความแม่นยำในแต่ละข้อกรณีที่ไม่ตัดคำหยุด

ข้อที่	เกรด A (%)	เกรด B (%)	เกรด C (%)	เกรด D (%)	เกรด F (%)	เฉลี่ย (%)
1	87.27	65.46	81.82	80.00	69.09	76.73
2	98.18	74.55	72.73	69.09	83.64	79.64
3	90.91	76.36	76.36	74.55	76.36	78.91
4	98.18	94.55	96.36	85.46	89.09	92.73
5	72.73	67.27	92.73	72.73	80.00	77.092

ตารางที่ 6 แสดงค่าความแม่นยำในแต่ละข้อกรณีที่ตัดคำหยุด

ข้อที่	เกรด A (%)	เกรด B (%)	เกรด C (%)	เกรด D (%)	เกรด F (%)	เฉลี่ย (%)
1	87.27	65.46	81.82	80.00	69.09	76.73
2	98.18	74.55	70.91	69.09	85.46	79.64
3	90.91	76.36	76.36	74.55	76.36	78.91
4	98.18	94.55	96.36	85.46	89.09	92.73
5	72.73	67.27	87.27	90.91	80.00	79.64

วิจารณ์

จากตารางที่ 5 และ 6 ในข้อที่ 5 จะเห็นได้ว่าค่าความแม่นยำในกรณีที่ตัดคำหยุดออก มากกว่ากรณีที่ไม่ตัดคำหยุด แสดงให้เห็นว่าการลดขนาดของเวกเตอร์โดยการลบคำที่ไม่สำคัญออกหรือคำหยุดมีผลต่อค่าความแม่นยำของระบบ และจากตารางทั้ง 2 ในข้อที่ 1 จะเห็นว่าค่าความแม่นยำที่ได้มีค่าต่ำที่สุด เนื่องจากข้อสอบในข้อนี้เป็นลักษณะวัดความจำ และจำเป็นที่จะต้องพิจารณาลำดับของคำด้วย ซึ่งกระบวนการในการคิดเกรดของระบบนี้ จะพิจารณาลักษณะข้อมูลหรือคำตอบแยกเป็นคำๆโดยไม่พิจารณาลำดับของ, วลี, หรือประโยค ทำให้การตรวจให้เกรดเกิดความผิดพลาดขึ้นได้

สรุปผลการทดลองเบื้องต้น

จากผลการทดลอง ได้ค่าความแม่นยำโดยรวมอยู่ระหว่าง 76.73% – 92.73% สรุปได้ว่า การลดขนาดของเวกเตอร์ส่งผลต่อค่าความแม่นยำโดยรวมขอระบบ โดยจะเห็นได้จากในข้อที่ 5 ในตารางที่ 5 ได้ค่าความแม่นยำโดยเฉลี่ยต่ำกว่าในตารางที่ 6 และในข้อที่ 1 ค่าความแม่นยำเฉลี่ยทั้งตารางที่ 5 และ 6 ต่ำที่สุด เนื่องจากการพิจารณาลำดับของคำ, วลี, หรือโครงสร้างประโยค ส่งผลต่อค่าความแม่นยำ ยกตัวอย่างเช่น ต้องการคำตอบคือ int 4 byte, double 8 byte ถ้าคำตอบของนักเรียนเป็น int 8 byte, double 4 byte เมื่อนำไปประมวลผลใน

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

ระบบ ผลที่ได้คือระบบจะให้เกรด A เนื่องจากมีคำที่ตรงกับเฉลยครบทุกคำ แต่ลำดับของคำไม่ตรงกัน ทำให้ผลที่ได้ผิดพลาด ด้วยเหตุนี้ในการทดลองต่อไปเราจะลองนำอัลกอริทึมต่างๆที่สามารถวิเคราะห์ลำดับของคำ, วิเคราะห์ประโยคในระดับโครงสร้าง หรือปรับปรุงวิธีการตัดคำ เช่น ตัดเป็นวลี หรือประโยค เพื่อปรับปรุงค่าความแม่นยำต่อไป เช่นเพิ่มอัลกอริทึม Latent Semantic ที่ใช้ในการพิจารณาความหมายของคำที่ซ่อนอยู่ เป็นต้น

เอกสารอ้างอิง

[1] Chanunya Loraksa and Ratchata Peachavanish. Automatic Thai-Language Essay Scoring Using Neural Network and Latent Semantic Analysis. IEEE. 2007.

[2] Lawrence M. Rudner and Tahung Liang. Automated Essay Scoring Using Bayes' Theorem. ACM. 2002.

[3] Li Bin Lu Jun Yao Jian-Min and Zhu Qiao-Ming. Automated Essay Scoring Using the K-NN Algorithm. IEEE. 2008.

[4] Jill Burstein and Martin Chodorow. Automated Essay Scoring for Nonnative English Speakers.

[5] Jill Burstein, Claudia Leacock and Richard Swartz. Automated Evaluation of Essays and Shot Answers. 2001.

[6] Greg Hamerly and Charles Elkan. Alternatives to the k-means algorithm that find better clusterings. ACM. 2002.

[7] จิราพร ยิ่งกว่าชาติ,บุญเสริม กิจศิริกุลและประสงค์ ปราณีตพลกรัง. "การทำนายผลสำเร็จการศึกษาของนักศึกษาระดับอุดมศึกษาด้วยการเรียนรู้แบบเบย์และการทำเหมืองข้อมูล." JCSSE2007. 2007.

[8] Dharmendra Kanejiya and Arun Kumary and Surendra Prasad. Automatic Evaluation of Students' Answers using Syntactically Enhanced LSA. ACM

[9] Leah S. Larkey. Automatic Essay Grading Using Text Categorization Techniques. ACM. 1998

[10] Tuomo Kakkonen, Niko Myller, Jari Timonen, and Erkki Sutinen. Automatic Essay Grading with Probabilistic latent Semantic Analysis. ACM. 2005

[11] นายฆนาศัย กรึงไกร, นางสาวชุลีรัตน์ จรัสกุลชัย, และนายคาซามา จุนอิชิ, แบบจำลองผสมแบบแยกแยะสำหรับการแบ่งคำไทย. 2009

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

การตรวจข้อสอบอัตนัยภาษาไทยแบบอัตโนมัติ

Automated Thai-Language Essay Scoring

สมมาตร อังคเศรณีกุล และ ชุลีรัตน์ จรัสกุลชัย
ภาควิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยเกษตรศาสตร์
50 ถนนพหลโยธิน แขวงลาดยาว เขตจตุจักร กรุงเทพฯ 10900 E-mail: g5164145@ku.ac.th, fscichj@ku.ac.th

บทคัดย่อ

การวิจัยด้านการเรียนรู้ของเครื่องจักรถูกนำประยุกต์ใช้ในชีวิตประจำวันมากขึ้น รวมทั้งการตรวจข้อสอบอัตนัยแบบอัตโนมัติ แนวคิดเบื้องต้นในการตรวจข้อสอบอัตโนมัติคล้ายกับการจัดกลุ่มเอกสาร หรือการจำแนกเอกสารตามหมวดหมู่ของคำตอบ งานวิจัยในอดีตที่ผ่านมาเป็นการตรวจข้อสอบประเภทที่มีชุดคำแบบคำถามปลายเปิด และขนาดของคำตอบมีความยาวเกินกว่าหนึ่งหน้ากระดาษ

บทความนี้นำเสนอการตรวจข้อสอบอัตนัยภาษาไทยแบบอัตโนมัติที่ใช้ชุดคำตอบแบบปลายปิด และความยาวของคำตอบประมาณ 3 ถึง 5 บรรทัด โดยใช้อัลกอริทึม K-Nearest Neighbor (K-NN) Artificial Neural Network (ANN) Bayesian Belief Network (BBN) และเทคนิค Latent Semantic Indexing (LSI) มาคิดเกรดข้อสอบเพื่อทำการเปรียบเทียบประสิทธิผลในแต่ละอัลกอริทึม และนำพจนานุกรมไทย-อังกฤษมาใช้แก้ปัญหาการใช้คำที่มีความหมายเดียวกัน ในการทดสอบประสิทธิผลทดลองกับข้อสอบวิชาการโปรแกรมเชิงวัตถุจำนวน 7 ข้อ มีชุดคำตอบของนักศึกษา 55 คน และชุดเฉลย 1 ชุด ผลการตรวจข้อสอบด้วยการใช้อัลกอริทึม K-NN ร่วมกับเทคนิค LSI และพจนานุกรมไทย-อังกฤษ ให้ค่าความแม่นยำเฉลี่ยรวมสูงสุดเป็น 81.13%

คำสำคัญ: ตรวจข้อสอบ, ภาษาไทย, อัตโนมัติ, K-Nearest Neighbor, Latent Semantic Analysis, Artificial Neural Network, Bayesian Belief Network, Vector Space Model

Abstract

Machine learning researches, including the automatically essay grading, have been applied in the daily life. The concept of automatically essay grading is similar to the documents clustering or classification according to the set of answers. In the past researches, the examination scoring is based on open questions and the length of answer is more than a page.

This article presents the automatically Thai-language essay scoring of closed questions and the length of answer is about 3-5 lines. There are four score-grading methods to compare the effectiveness, K-Nearest Neighbor (K-NN), Artificial Neural Network (ANN), Bayesian Belief Network (BBN) and Latent Semantic Indexing (LSI). Thai-English dictionaries are also used to solve the synonym problem. The effectiveness of the algorithms is tested with seven questions object-oriented programming test, together with answers from fifty-five students and an answer solution. The result of four methods comparison showed that the method returning the most total average accuracy value is applying K-NN together with LSI and Thai-English dictionaries, with 81.13% as the returned value.

Keywords: Automated, Thai-Language, Essay Scoring, K-Nearest Neighbor, Latent Semantic Analysis, Artificial Neural Network, Bayesian Belief Network, Vector Space Model

1. คำนำ

การเรียนการสอนในปัจจุบัน จะเห็นได้ว่าการประเมินการเรียนรู้ของนักเรียน โดยใช้ข้อสอบแบบเขียนตอบหรืออัตนัย ถูกนำมาใช้อย่างกว้างขว้าง เพื่อเปิดโอกาสให้นักเรียนแสดงทักษะทางความคิด ความจำ ความรู้ รวมถึงลำดับความคิด การสังเคราะห์และการวิเคราะห์ เป็นต้น ซึ่งใช้ประเมินการเรียนรู้ได้ดีกว่าการใช้คำถามแบบตัวเลือกตอบ (Multiple-choice) เพราะคำถามแบบมีตัวเลือกตอบสามารถเดาได้ทำให้ไม่สามารถประเมินการเรียนรู้ได้อย่างแท้จริง อย่างไรก็ตามเมื่อจำนวนข้อสอบมีปริมาณมากขึ้น ครูผู้ตรวจจะพบกับความยุ่งยากในการตรวจ เนื่องจากมีปัจจัยหลายปัจจัย เช่น ปัจจัยในเรื่องของเวลาเพราะใช้เวลานานในการตรวจ ความเที่ยงตรงในการตรวจเพราะมีอารมณ์และความรู้สึกต่างๆเข้ามาเกี่ยวข้อง

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

ในต่างประเทศจึงมีการคิดริเริ่มสร้างระบบการตรวจข้อสอบอัตนัยแบบอัตโนมัติด้วยคอมพิวเตอร์ขึ้น โดยนำเทคนิควิธีการต่างๆ เช่น การประมวลผลภาษาธรรมชาติ (NLP) ทฤษฎีทางด้านปัญญาประดิษฐ์ (Artificial Intelligence) และหลักสถิติ เป็นต้น มาเป็นเครื่องมือในการสนับสนุนการตรวจข้อสอบอัตนัย เนื่องจากในปัจจุบันคอมพิวเตอร์ประมวลผลได้รวดเร็วกว่าแต่ก่อนมาก ทำให้ประหยัดเวลาในการตรวจของผู้ตรวจได้มาก ลดปัญหาความไม่เที่ยงตรงในการตรวจ และไม่มีเรื่องของอารมณ์ ความเมื่อยล้าที่เกิดขึ้นในมนุษย์มาเกี่ยวข้อง

แนวคิดเบื้องต้นในการตรวจข้อสอบจะคล้ายกับงานทางด้านการจัดกลุ่มเอกสาร หรือจำแนกหมวดหมู่ของเอกสาร ปัจจุบันมีหลายประเทศสร้างและพัฒนาเป็นจำนวนมาก ซึ่งทฤษฎีและหลักการแตกต่างกันไป ยกตัวอย่างเช่น ระบบการให้คะแนนข้อสอบแบบอัตโนมัติโดยใช้เทคนิค LSA [7,13] ระบบการให้คะแนนข้อสอบอัตนัยแบบอัตโนมัติโดยใช้ทฤษฎีของ Bays [2] ระบบการให้คะแนนข้อสอบอัตนัยแบบอัตโนมัติโดยใช้อัลกอริทึม K-NN [3] เป็นต้น ส่วนในประเทศไทยพบเพียงงานวิจัยเดียวที่ทำวิจัยเรื่องนี้ โดยนำอัลกอริทึม Neural Network มาใช้ในการเรียนรู้จากชุดข้อมูลฝึกเพื่อใช้ในการตรวจข้อสอบ และเพิ่มประสิทธิผลด้วยเทคนิค LSA [1] ซึ่งถือได้ว่ามีการทำวิจัยในเรื่องนี้น้อยมากเมื่อเทียบกับต่างประเทศ

บทความนี้นำเสนออัลกอริทึมต่างๆที่ใช้ในการตรวจข้อสอบอัตนัยแบบอัตโนมัติในต่างประเทศที่ส่วนมากทดลองกับชุดข้อสอบอัตนัยภาษาอังกฤษ มาใช้กับชุดข้อสอบอัตนัยภาษาไทย เพื่อทำการเปรียบเทียบประสิทธิผลที่ได้ในแต่ละอัลกอริทึม โดยสิ่งที่แตกต่างจากงานในต่างประเทศคือ ลักษณะของชุดข้อมูลที่นำมาใช้ในการเรียนรู้เพื่อคิดเกรดคำตอบจะใช้ชุดเฉลยร่วมกับชุดคำตอบที่มีการจำแนกผลการตรวจโดยมนุษย์แล้วมาใช้ในการคิดเกรดคำตอบ ซึ่งในต่างประเทศจะใช้เพียงชุดคำตอบที่มีการจำแนกผลการตรวจไว้แล้วเท่านั้นมาคิดเกรดคำตอบ ในบทความนี้จะเพิ่มค่าความแม่นยำในการตรวจด้วยการใช้พจนานุกรมไทย-อังกฤษมาแก้ปัญหากรณีการใช้คำที่มีความหมายเหมือนกัน

2. งานวิจัยที่เกี่ยวข้อง

ในต่างประเทศได้มีการคิดสร้างระบบการตรวจข้อสอบอัตนัยแบบอัตโนมัติหลายประเทศ เช่น ประเทศอเมริกาได้มีการจดสิทธิบัตรแล้ว [4,5] เป็นต้น มีงานวิจัยที่ทำการประเมินประสิทธิภาพของระบบด้วยค่าความสัมพันธ์(Correlation) ซึ่งเป็นการเปรียบเทียบระหว่างผลลัพธ์ที่ได้จากคอมพิวเตอร์ตรวจกับมนุษย์ผู้เชี่ยวชาญตรวจได้ค่าความสัมพันธ์ค่อนข้างสูง [6] แสดงให้เห็นว่าผลลัพธ์ที่ได้จากคอมพิวเตอร์ตรวจนั้นสอดคล้องกับผลลัพธ์ที่ได้จากมนุษย์ผู้เชี่ยวชาญตรวจค่อนข้างมาก ทำให้มีโอกาสที่จะสร้างระบบการตรวจข้อสอบอัตนัยที่สามารถนำไปใช้จริงได้

ในต่างประเทศได้มีการทำวิจัย ศึกษาและทดลองแก้ปัญหาเหล่านี้ ตัวอย่างงานวิจัยในต่างประเทศเช่น การตรวจข้อสอบอัตนัยโดยใช้ K-NN [3] เป็นการนำอัลกอริทึม K-NN มาใช้ในการคิดคะแนนคำตอบ โดยดูจากค่าความคล้ายระหว่างคำตอบกับชุดข้อมูลฝึกฝน งานวิจัยในประเทศไทยมีการทำวิจัยเรื่องการตรวจข้อสอบอัตนัยภาษาไทยโดยใช้ อัลกอริทึม Neural Network และ LSA [1] โดยอัลกอริทึม Neural Network นำมาใช้ในการเรียนรู้ชุดข้อมูลฝึก และทำนายเกรดของคำตอบ มีการทดลองระหว่างการนำเทคนิค LSA ซึ่งเป็นเทคนิคการแปลงและลดขนาดข้อมูลมาช่วยทำให้ได้ค่าความแม่นยำเพิ่มมากขึ้น และมีการเพิ่มส่วนของ Part of Speech tag (POS tag) [7] มาใช้ในการวิเคราะห์โครงสร้างของเอกสาร เพื่อใช้แก้ปัญหาเรื่องของการพิจารณาลำดับและโครงสร้างของคำในเทคนิค LSA ทำให้ได้ประสิทธิภาพที่สูงขึ้น และอัลกอริทึม Bayesian Networks [2] ถูกนำมาใช้ในการทำนายคะแนนโดยดูจากความน่าจะเป็นของคำต่างๆ ในแต่ละประเภทของระดับคะแนน เพื่อใช้ทำนายคะแนนของคำตอบ ได้ค่าความแม่นยำโดยรวม 80% เป็นต้น

3. อัลกอริทึมและเทคนิคที่นำมาใช้สำหรับคิดเกรด

3.1 อัลกอริทึม K-Nearest Neighbor (K-NN)

K-NN เป็นอัลกอริทึมที่ใช้สำหรับจำแนกข้อมูลโดยอาศัยชุดข้อมูลฝึก ตัวอย่างการนำ K-NN มาประยุกต์ใช้ เช่น ใช้ในการจำแนกข้อความในเอกสาร เป็นกระบวนแยกประเภทหรือจัดหมวดหมู่ของเอกสาร [8] เป็นต้น วิธีการของอัลกอริทึมนี้จะคำนวณค่าความคล้ายระหว่างข้อมูลที่ต้องการทำนายกับชุดข้อมูลฝึก โดยจะเรียงจากค่าความคล้ายมากสุดไปน้อยสุด จากนั้นทำนายผลลัพธ์จากการโหวตให้คะแนนใน K ลำดับ ตอบผลโหวตมากที่สุด โดยค่า K ควรกำหนดให้เป็นเลขคี่เพื่อหลีกเลี่ยงกรณีที่ผลโหวตคะแนนเสียงเท่ากัน เรียกวิธีการนี้ว่า การลงคะแนนเสียงข้างมาก (Majority vote) สูตรที่ใช้วัดความคล้ายในอัลกอริทึม K-NN มีหลายสูตร ตัวอย่างสูตรที่นำมาใช้ในบทความนี้ คือ Cosine ดังสมการ (1)

$$S(d_i,d_j)=\cos(\alpha)=\frac{\sum_{k=1}^{m} w_{ki}\cdot w_{kj}}{\sqrt{\sum_{k=1}^{m} w_{ki}^2}\cdot\sqrt{\sum_{k=1}^{m} w_{kj}^2}} \quad (1)$$

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

โดยที่ $S(d_i,d_j)$ คือ ค่าความคล้ายระหว่าง 2 เอกสาร

w_{ki} และ w_{kj} คือ ค่าความถี่ของคำระหว่าง 2 เอกสาร

3.2 เทคนิค Latent Semantic Indexing (LSI)

อัลกอริทึมการค้นหาข้อมูลแบบ LSI ประยุกต์จาก Latent Semantic Analysis (LSA) เป็นเทคนิคทางด้าน Natural Language Processing (NLP) ที่ใช้ในการแปลงข้อมูลเข้า เพื่อลดขนาดของข้อมูล และสามารถหาความหมายที่ซ่อนอยู่ในข้อมูลด้วย สิ่งที่สำคัญของอัลกอริทึมนี้คือ SVD (Singular Value Decomposition) ในพีชคณิตเชิงเส้น โดยจะเริ่มด้วยเมทริกของ Term-Document และแตกย่อยเป็น 3 ส่วน คือ U, S, และ V ดังสมการ (2) ซึ่งเป็นการวิเคราะห์ความสัมพันธ์ระหว่างชุดเอกสารแต่ละชุดหรือเวกเตอร์ของเอกสาร (ส่วนมากจะแทนในรูปของ TF-IDF) กับคำต่างๆไปยังพื้นที่เวกเตอร์ (Vector Space) ที่มีมิติน้อยกว่า และจะทำการวัดค่าความความคล้ายระหว่างเอกสารกับคำค้น (Query) สูตรที่ใช้วัดค่าความคล้ายเช่น Cosine ซึ่งผลลัพธ์จะได้ชุดของแนวคิดที่สัมพันธ์กับเอกสารแต่ละเอกสารและคำแต่ละคำตามลำดับค่าความคล้ายจากมากไปน้อย ซึ่ง LSI มีการนำมาประยุกต์ใช้ในงานวิจัยต่างๆ เช่น ใช้ในการจัดกลุ่มซอฟแวร์ [9] เป็นต้น

$$A = USV^T \quad (2)$$

โดยที่ A คือ เมทริกของ Term-Document

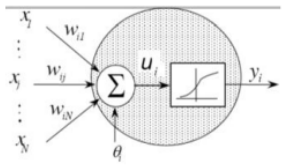
U คือ เมทริกแถวตั้งทั้งหมดได้จาก Eigenvector ของเมทริก AA^T

S คือ เมทริกแบบ Diagonal ขนาด r*r เมื่อ r เป็น Rank ของ A

V คือ เมทริกแถวตั้งทั้งหมดได้จาก Eigenvector ของเมทริก A^TA

3.3 อัลกอริทึม Artificial Neural Network (ANN)

Artificial Neural Network เป็นเครือข่ายประสาทเทียม เป็นแบบจำลองมาจากการทำงานของระบบประสาทส่วนกลางที่มีโครงสร้างเป็นลักษณะของเครือข่ายที่เชื่อมโยงกันระหว่างหน่วย ซึ่งสามารถที่จะรับรู้ข้อมูลและปรับตัวเข้ากับสถานการณ์หรือสิ่งแวดล้อมที่เผชิญอยู่ และเป็นเครื่องมือที่มีประสิทธิภาพในการประยุกต์ใช้งานต่างๆ ที่เกี่ยวข้องกับการคำนวณและการจดจำ เช่นการบีบอัดข้อมูล การกรองสัญญาณ และการทำนายเหตุการณ์ เป็นต้น ตัวอย่างโครงสร้างวงจร ดังรูปที่ 1



รูปที่ 1 โครงสร้าง Neural Network

โดยที่ $x = [x_1,x_2, ..., x_N]^T$ คือ ข้อมูลเข้า

$w = [w_1,w_2, ..., w_N]^T$ คือ น้ำหนักความสำคัญที่ให้กับข้อมูลเข้าแต่ละตัว

ANN ประกอบด้วยชุดของบัฟหรือโหนด ซึ่งเป็นได้ทั้งโหนดนำเข้า (Input nodes) และโหนดผลลัพธ์ (Output nodes) หรือโหนดระหว่างกลางเรียกว่า โหนดซ่อน (Hidden node) มีการเชื่อมต่อระหว่างโหนด โดยจะต้องมีการกำหนดค่าน้ำหนัก (Weight) ในเส้นเชื่อมทุกเส้น เมื่อเริ่มทำงาน จะทำการกำหนดค่าให้กับโหนดนำเข้าซึ่งได้จากข้อมูลที่จะนำมาทดลอง จากนั้นโหนดนำเข้าจะส่งค่าไปตามเส้นเชื่อม โดยค่าที่ส่งออกไปจะถูกคูณกับค่านำหนักของเส้นเชื่อม โหนดถัดไปจะรับค่าที่ได้จากผลรวมจากโหนดต่างๆ แล้วคำนวณเป็นผลลัพธ์ ซึ่งโดยทั่วไปฟังก์ชันที่นิยมใช้ในการคำนวณผลลัพธ์คือ ฟังก์ชันซิกมอยด์ (Sigmoid function) สูตรดังสมการ (3)

$$g(x) = 1/(1 + \exp\{-\beta x\}) \quad (3)$$

โดยที่ β คือ ค่าความชัน

จากนั้นจะส่งค่าต่อไปยังชั้นถัดไปเรื่อยๆ จนถึงโหนดผลลัพธ์ จะได้ผลลัพธ์ออกมา และทำการตรวจผลลัพธ์ที่ได้กับเฉลยดูว่าได้ผลลัพธ์เป็นที่น่าพอใจหรือไม่ ถ้าไม่จะทำการปรับค่าน้ำหนักและเริ่มกระบวนการใหม่ ทำซ้ำเช่นนี้จนกว่าจะได้ค่าน้ำหนักที่เหมาะสม ตัวอย่างในการนำ ANN มาประยุกต์ใช้ในงานด้านต่างๆ เช่น ใช้ในการตรวจจับการปลอมแปลงภาพ [10] เป็นต้น

3.4 อัลกอริทึม Bayesian Belief Network (BBN)

อัลกอริทึม BBN เป็นโมเดลทางสถิติ มีพื้นฐานมาจากทฤษฎีของ Bay's เป็นทฤษฎีที่เกี่ยวข้องกับความน่าจะเป็นของเหตุการณ์ต่างๆ ที่เกิดขึ้นจริงและเราสนใจซึ่งมีความไม่แน่นอน เพื่อนำมาใช้ในการวิเคราะห์หรือทำนายเหตุการณ์ที่อาจจะเกิดขึ้นในอนาคต สูตรที่นำมาใช้ในการคำนวณดังสมการ (4)

$$P(c|x) = P(x|c) * P(c) / P(x) \quad (4)$$

โดยที่ P(c) คือ ความน่าจะเป็นที่จะเกิดเหตุการณ์ c

P(x) คือ ความน่าจะเป็นที่จะเกิดเหตุการณ์ x

P(x|c) คือ ความน่าจะเป็นที่เกิดเหตุการณ์ x เมื่อ c เกิดขึ้นแล้ว

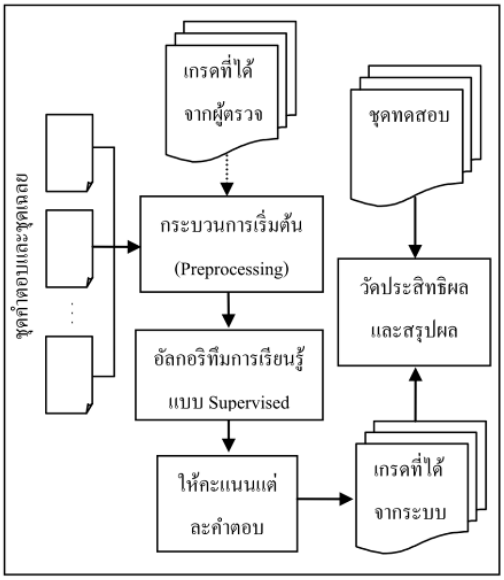
P(c|x) คือ ความน่าจะเป็นที่เกิดเหตุการณ์ c เมื่อ x เกิดขึ้นแล้ว

ด้วยเหตุนี้โมเดล BNN ถูกนำมาประยุกต์ใช้อย่างกว้างขวาง เช่น ใช้พยากรณ์อากาศ [11] เป็นต้น

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

4 การทดลอง

ขั้นตอนในการสร้างระบบการตรวจข้อสอบอัตนัยภาษาไทยแบบอัตโนมัติ แสดงรูปที่ 2



รูปที่ 2: ขั้นตอนการสร้างระบบการตรวจข้อสอบอัตนัยภาษาไทยแบบอัตโนมัติ

4.1 ข้อมูลที่ใช้ในการทดลอง

ข้อมูลที่ใช้ในการทดลองจะใช้ข้อสอบวิชา "การโปรแกรมเชิงวัตถุด้วยภาษาจาวา" มีชุดเฉลย 1 ชุดและทำการตรวจให้เกรดชุดคำตอบทั้งหมดจากผู้ตรวจ โดยแบ่งกลุ่มเป็น 5 กลุ่ม คือ A, B, C, D และ F โดยที่เกรด A เป็นกลุ่มที่มีคะแนนความถูกต้องมากสุด คัดเลือกมา 7 ข้อ จาก 17 ข้อ และจากคำตอบนักเรียนทั้งหมด 55 คน โดยทำการแบ่งลักษณะการตั้งคำถามตามหลัก Wh-questions ในไวยากรณ์ภาษาอังกฤษสามารถจำแนกได้เป็น 7 ประเภท บางประเภทจะให้คำตอบแบบสั้นๆ เช่น Who (ใคร) คำตอบเป็นชื่อบุคคล Where (ที่ไหน) คำตอบเป็นสถานที่ When (เมื่อไร) คำตอบเป็น วัน และเวลา และ Which (สิ่งไหน อันไหน) คำตอบเป็นลักษณะให้เลือกอย่างใดอย่างหนึ่ง เป็นต้น บางประเภทจะให้คำตอบความยาว 2-3 บรรทัด เช่น What (อะไร) คำตอบเป็นสิ่งของ เวลา ความคิด ฯลฯ Why (ทำไม) คำตอบเป็นเหตุผล และ How (อย่างไร) คำตอบเป็นวิถีทาง ในบทความนี้สนใจการตรวจข้อสอบที่ประเภทคำถามที่ต้องตอบเป็นวิธีการ เช่น อะไร อย่างไร และอันไหน เป็นต้น

ในการทดลอง จะทำการบันทึกชุดคำตอบและชุดเฉลยใหม่โดยจัดเก็บอยู่ในรูปแบบไฟล์ข้อความ (ไฟล์ Text) ค่าสถิติและประเภทคำถามของข้อมูล แสดงดังตารางที่ 1

ตารางที่ 1 ค่าสถิติและประเภทคำถามของข้อมูล

ข้อที่	จำนวนคำสูงสุด	จำนวนคำต่ำสุด	จำนวนคำเฉลี่ย	จำนวนคำในเฉลย	ประเภทคำถาม
1	70	2	22.20	38	What
2	78	1	31.88	43	What
3	85	2	23.98	41	How
4	68	10	33.08	33	How
5	78	6	30.97	16	What
6	70	1	14.04	35	Which
7	89	1	28.55	53	How

4.2 การแปลงรูปแบบข้อมูลในรูปเวกเตอร์

การแปลงรูปแบบข้อมูลให้อยู่ให้ในรูปแบบที่สามารถทำการประมวลผลได้ จะใช้โปรแกรมตัดคำภาษาไทย [12] เพื่อแยกเป็นคำก่อน จากนั้นทำการแทนแต่ละคำในคำตอบและเฉลยในรูปแบบเวกเตอร์ความถี่ของคำและถ่วงน้ำหนักเพื่อให้ความสำคัญของคำด้วย TF-IDF ดังสมการ (5) และ (6) และเก็บอยู่ในรูปเมทริก โดยจะทำการตัดคำที่เป็นคำหยุดออกก่อน เช่นคำว่า 'และ', 'หรือ' ซึ่งไม่มีความสำคัญต่อการคิดเกรด

$$d_j = (w_{1j}, w_{2j}, w_{3j}, \ldots, w_{mj}) \quad (5)$$

โดยที่ d_j คือ คำตอบของนักเรียนคนที่ j

w_{ij} คือ ค่าน้ำหนักของคำที่ i ในคำตอบของนักเรียนคนที่ j ดังสมการ (6)

$$w_{ij} = \text{TF-IDF}(i,j) = \text{TF}(i,j) \log(N / \text{DF}(i)) \quad (6)$$

โดยที่ TF(i,j) คือ ค่าความถี่ของคำที่ i ในคำตอบของนักเรียนคนที่ j

N คือ จำนวนนักเรียนทั้งหมดในชุดข้อมูล

DF(i) คือ จำนวนคำตอบของนักเรียนที่ปรากฏคำที่ i อยู่

4.3 การจำแนกด้วยอัลกอริทึม K-NN

การจำแนกด้วยอัลกอริทึม K-NN จะทำการวัดค่าความคล้ายของคำตอบนักเรียนแต่ละคนเทียบกับผลเฉลยในข้อนั้นๆ จำนวน 1 ชุด ถ้าได้ค่าความคล้ายมากจะได้คะแนนมากขึ้นตาม ซึ่งในการคำนวณวัดค่าความคล้ายบนพื้นฐานของอัลกอริทึม K-NN นั้น จะแตกต่างจากวิธีการทั่วไป คือ จะทำการกำหนดค่า K เป็น 1 เท่านั้น เนื่องจากมีชุดเฉลยเพียงชุดเดียว

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

ดังนั้นในการคิดเกรดจะกำหนดเกรดดังนี้ $g_1, g_2, \ldots, g_C$ ซึ่งจะเกี่ยวข้องกับช่วงค่า Threshold ในแต่ละช่วงเกรด คือ $L_1, L_2, \ldots, L_{C+1}$ โดยที่ C คือจำนวนของเกรด [13] สามารถหาช่วงค่า Threshold ได้จากชุดคำตอบที่ผู้ตรวจให้เกรดไว้แล้ว โดยนำค่าเฉลี่ยที่ได้จากการวัดค่าความคล้ายด้วย Cosine ในแต่ละกลุ่มมากำหนดเป็นช่วง ในการทดลองนี้กำหนด C เป็น 5 ตัวอย่างช่วงค่า Threshold ในแต่ละช่วงดังตารางที่ 2

ตารางที่ 2 ตัวอย่างช่วงค่า Threshold ของแต่ละเกรดในการจำแนกด้วยอัลกอริทึม K-NN

ข้อที่	L_1	L_2	L_3	L_4	L_5	L_6
1	0	0.039	0.061	0.081	0.310	1
2	0	0.010	0.025	0.099	0.677	1

หลังจากได้ช่วงค่า Threshold ของแต่ละเกรดมาแล้ว จะนำคำตอบที่ต้องการให้เกรดมาวัดความคล้ายกับผลเฉลย แล้วให้เกรดตามค่าความคล้ายที่อยู่ในช่วง Threshold ของช่วงเกรดนั้น

4.4 การจำแนกด้วยเทคนิค LSI

การทดลองชุดนี้เป็นการใช้เทคนิค LSI เพื่อใช้คุณสมบัติของ LSI ในการค้นหาคำที่ความหมายเหมือนกัน เพื่อช่วยเพิ่มความเหมือนโดยนำเทคนิค LSI ผสมกับการจำแนกด้วยอัลกอริทึม K-NN โดยจะนำชุดคำตอบที่ถูกแปลงเป็นค่า TF-IDF ในรูปแบบของเมทริกแล้ว มาทำการแตกเป็นเมทริกย่อย 3 เมทริกดังสมการ (2) ซึ่งเรียกว่าพื้นที่เชิงความหมาย (Semantic space) จากนั้นจะใช้วิธีการเดียวกันกับการจำแนกด้วยอัลกอริทึม K-NN ในการกำหนดช่วงค่า Threshold และคิดเกรด

4.5 การจำแนกด้วยอัลกอริทึม ANN และ BBN

การจำแนกด้วยอัลกอริทึม ANN และ BBN จะใช้ชุดคำตอบที่ให้เกรดในแต่ละรูปแบบ (A B C D และ F) ของชุดคำตอบทั้งหมด 55 ชุด โดยแบ่งตามรูปแบบเกรดรูปแบบละ 3 ชุด รวมเป็น 15 ชุด เสมือนเป็นการแทนคำเฉลยจากผู้ตรวจในแต่ละรูปแบบเกรด เพื่อสร้างโมเดลที่ใช้สำหรับทำนายเกรด ชุดคำตอบที่เหลือใช้เป็นชุดทดสอบสำหรับวัดประสิทธิผลของโมเดลที่ได้ ซึ่งในการทดลองอัลกอริทึม ANN จะใช้การเรียนรู้แบบ Back-propagation โดยใช้จำนวนโหนดนำเข้าเป็นโหนดความถี่ของคำแต่ละคำ ซึ่งโจทย์ทั้ง 7 ข้อมีจำนวนโหนดนำเข้าแตกต่างกันดังนี้ 161 285 217 124 243 173 และ 199 ตามลำดับ (เรียงลำดับจากข้อ 1 ถึง 7) และโหนดผลลัพธ์มีหนึ่งโหนด ผลลัพธ์เป็นเกรด A B C D และ F

4.6 การใช้พจนานุกรมเข้ามาช่วยแก้ปัญหากรณีเกิดคำเหมือน

เพื่อเป็นการแก้ปัญหากรณีใช้คำที่ต่างกันในคำตอบ แต่มีความหมายเหมือนกันในคำเฉลย จะใช้พจนานุกรมไทย-อังกฤษของ NECTEC มาแก้ปัญหา โดยจะนำคำในคำตอบแต่ละคำไปตรวจสอบในพจนานุกรม เพื่อนำคำเหมือนมาตรวจสอบว่าคำเหมือนนั้นมีในชุดเฉลยหรือไม่ ถ้ามีจะเปลี่ยนคำเดิมเป็นคำเหมือนนั้นโดยที่ขนาดของเวกเตอร์ยังคงเดิม ส่งผลให้การวัดค่าความคล้ายระหว่างคำตอบกับเฉลยมีค่ามากขึ้นทำให้ได้ความถูกต้องเพิ่มขึ้น ซึ่งจะมีผลกับการจำแนกด้วยอัลกอริทึม K-NN และเทคนิค LSI เท่านั้น เนื่องจากอัลกอริทึมทั้ง 2 นี้ต้องใช้ชุดเฉลยในการตรวจให้เกรดเป็นหลัก ส่วนอีก 2 อัลกอริทึมคือ ANN และ BBN ไม่จำเป็นต้องใช้เพราะไม่ใช่เป็นการหาความคล้ายกับชุดเฉลย แต่เป็นการทำนายผลโดยอาศัยการเรียนรู้จากชุดข้อมูลฝึกหรือชุดคำตอบที่มีการให้เกรดโดยผู้ตรวจไว้แล้ว

5 การวัดประสิทธิผล

ในการวัดประสิทธิผล จะคำนวณค่าความแม่นยำ (Accuracy) โดยจะนับจำนวนคำตอบที่ผลเฉลยให้เกรดตรงกับเกรดที่ระบบให้ จากนั้นทำการคิดเป็นอัตราร้อยละของแต่ละเกรดนั้นแล้วหาค่าเฉลี่ยรวม ซึ่งสูตรที่ใช้ในการคิดดังสมการ (7)

$$Acc = \frac{TP + TN}{TP + FP + FN + TN} \quad (7)$$

โดยที่ TP คือ จำนวนคำตอบที่ระบบให้ผลเป็นจริงและผลเฉลยให้ผลเป็นจริงด้วย (True Positive)

TN คือ จำนวนคำตอบที่ระบบให้ผลเป็นเท็จและผลเฉลยให้ผลเป็นเท็จด้วย (True Negative)

FP คือ จำนวนคำตอบที่ระบบให้ผลเป็นจริงแต่ผลเฉลยให้ผลเป็นเท็จ (False Positive)

FN คือ จำนวนคำตอบที่ระบบให้ผลเป็นเท็จแต่ผลเฉลยให้ผลเป็นจริง (False Negative)

6 ผลการทดลอง

ในการทดลอง ทั้ง 4 วิธี จะทำการวัดประสิทธิผลด้วยการวัดค่าความแม่นยำ (Accuracy) เพื่อตรวจสอบว่าผลการทดลองที่ได้สอดคล้องกับผลเฉลยที่ผู้ตรวจให้เกรดมากน้อยเพียงใด (เกรดที่ได้จากระบบกับเกรดที่ได้จากเฉลยตรงกัน) ผลลัพธ์ดังตารางที่ 4

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

ตารางที่ 4 ค่าความแม่นยำเฉลี่ยรวมทั้ง 4 วิธี

วิธีการคิดเกรด	ค่าความแม่นยำเฉลี่ยรวม
อัลกอริทึม K-NN	80.68%
อัลกอริทึม K-NN + พจนานุกรม	80.99%
อัลกอริทึม K-NN + เทคนิค LSI	80.57%
อัลกอริทึม K-NN + เทคนิค LSI + พจนานุกรม	81.30%
อัลกอริทึม ANN	80.10%
อัลกอริทึม BBN	74.85%

7. สรุปผลการทดลองและงานในอนาคต

บทความนี้นำเสนอวิธีการคิดเกรดในการตรวจข้อสอบอัตนัยภาษาไทยแบบอัตโนมัติด้วยอัลกอริทึม K-NN ANN BBN และเทคนิค LSI เพื่อเปรียบเทียบประสิทธิผลในแต่ละอัลกอริทึม ใช้ข้อสอบวิชาการโปรแกรมเชิงวัตถุจำนวน 7 ข้อ มีชุดคำตอบของนักศึกษา 55 คน และชุดเฉลย 1 ชุด จากผลการทดลองพบว่าวิธีการคิดเกรดด้วยอัลกอริทึม K-NN ร่วมกับเทคนิค LSI และพจนานุกรมไทย-อังกฤษ ได้ค่าความแม่นยำเฉลี่ยรวมสูงสุดเป็น 81.30% รองมาคือวิธีการคิดเกรดด้วยอัลกอริทึม K-NN ร่วมกับพจนานุกรมไทย-อังกฤษ ได้ค่าเป็น 80.99% สรุปได้ว่าการนำพจนานุกรมเข้ามาช่วยแก้ปัญหากรณีเกิดคำเหมือน ส่งผลให้ค่าความแม่นยำเพิ่มมากขึ้น 0.31% ในอัลกอริทึม K-NN และ 0.73% ในอัลกอริทึม K-NN ร่วมกับเทคนิค LSI เนื่องจากเป็นการขยายคำในคำตอบที่มีความหมายเหมือนกันในคำเฉลยแต่เขียนไม่เหมือนกัน ทำให้ได้ค่าความถูกต้องเพิ่มขึ้น

การนำเทคนิค LSI มาคิดเกรด มีแนวโน้มได้ค่าความแม่นยำเฉลี่ยรวมดีกว่าการใช้อัลกอริทึม K-NN เพราะเทคนิค LSI ช่วยในการหาความหมายที่ซ่อนอยู่ซึ่งอัลกอริทึม K-NN ไม่มี ส่วนวิธีการคิดเกรดด้วยอัลกอริทึม ANN และ BBN ได้ค่าความแม่นยำเฉลี่ยรวมน้อยกว่าวิธีอื่นๆ เนื่องจากทั้ง 2 อัลกอริทึมเป็นอัลกอริทึมการเรียนรู้ที่ต้องการจำนวนชุดข้อมูลฝึกและความหลากหลายของลักษณะของข้อมูลค่อนข้างมากเพื่อให้เกิดประสิทธิภาพในการเรียนรู้ ทำให้ไม่เหมาะสมที่จะนำมาใช้ในการคิดเกรดในลักษณะของคำตอบแบบสั้นที่มีความหลากหลายของคำไม่มาก และจำนวนชุดฝึกน้อย และพบข้อบกพร่องในการเตรียมชุดข้อมูลฝึกในกรณีที่เป็นเกรดรูปแบบ F (คำตอบไม่ถูกต้อง) ไม่สามารถเตรียมได้ครบ เพราะคำตอบเป็นได้หลากหลายรูปแบบทำให้ได้ผลลัพธ์ที่ไม่ถูกต้อง

ในงานวิจัยต่อไป จะทำการทดลองเพิ่มส่วนของ Parser เพื่อใช้ในการวิเคราะห์โครงสร้างประโยคภาษาไทย ลำดับของคำ และทำการทดลองตรวจให้คะแนนคำตอบโดยใช้ชุดเฉลยเพียงชุดเดียวในการให้เกรด เนื่องจากในความเป็นจริงผู้ตรวจที่เป็นมนุษย์จะไม่มีชุดข้อมูลที่ให้เกรดในแต่ละรูปแบบของคำตอบอยู่ก่อน โดยจะใช้อัลกอริทึมการเรียนรู้แบบ Unsupervised ซึ่งเป็นการเรียนรู้แบบไม่มีชุดข้อมูลฝึกฝน เพื่อทำการแบ่งกลุ่มคำตอบแล้วหาวิธีการกำหนดช่วงเพื่อใช้ในการตัดเกรด

เอกสารอ้างอิง

[1] Chanunya Loraksa and Ratchata Peachavanish, “Automatic Thai-Language Essay Scoring Using Neural Network and Latent Semantic Analysis”, IEEE, 2007.

[2] Lawrence M. Rudner and Tahung Liang, “Automated Essay Scoring Using Bayes’ Theorem”, ACM, 2002.

[3] Li Bin Lu Jun Yao Jian-Min and Zhu Qiao-Ming, “Automated Essay Scoring Using the K-NN Algorithm”, IEEE, 2008.

[4] Jill C. Burstein, Randy Mark Kaplan, Susanne Wolff, and Chi Lu, “AUTOMATIC ESSAY SCORING SYSTEM USING CONTENT-BASED TECHNIQUES”, United States Patent, 2000.

[5] Yvacheslav Andreyev, Yigal Attali, and Jill Burstein, “AUTOMATIC ESSAY SCORING SYSTEM”, United States Patent, 2005.

[6] Thomas K Landauer, Peter W. Foltz, and Darrell Laham, “An Introduction to Latent Semantic Analysis”, Discourse Processes, pp. 259-284, 1998.

[7] Dharmendra Kanejiya and Arun Kumary and Surendra Prasad, “Automatic Evaluation of Students’ Answers using Syntactically Enhanced LSA”, ACM.

[8] Pascal Soucy, Guy W.Mineau, “A Simple KNN Algorithm for Text Categorization”, IEEE, 2001.

[9] Jonathan I. Maletic and Naveen Valluri, “Automatic Software Clustering via Latent Semantic Analysis”, IEEE, 1999.

[10] E.S.Gopi, N.Lakshmanan, T.Gokul, S.KumaraGanesh, and Prerak.R.Shah, “DIGITAL IMAGE FORGERY DETECTION USING ARTIFICIAL NEURAL NETWORK AND AUTO REGRESSIVE COEFFICIENTS, IEEE, 2006

[11] อุษานาฎ เอื้ออภิสิทธิ์วงศ์ และ ศาสตรา วงศ์ธนวสุ, “การพยากรณ์อากาศโดยใช้ตัวแบบเครือข่ายความเชื่อเบย์”, JCSSE, 2009.

[12] ฆนาศัย กรึงไกร, ชุลีรัตน์ จรัสกุลชัย, และคาซามา จุนอิชิ, “แบบจำลองผสมแบบแยกแยะสำหรับการแบ่งคำไทย, 2009.

[13] Tuomo Kakkonen, Niko Myller, Jari Timonen, and Erkki Sutinen, “Automatic Essay Grading with Probabilistic latent Semantic Analysis”, ACM, 2005.

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

Automated Thai-Language Essay Scoring with Unsupervised Learning Algorithm

Sommart Aungkaseraneekul[†] **Chuleerat Jaruskulchai**[‡]
[†]Department of Computer Science, Faculty of Science,
Kasetsart University, Bangkok, Thailand
[‡]Department of Computer Science, Faculty of Science,
Kasetsart University, Bangkok, Thailand
g5164145@ku.ac.th, fscichj@ku.ac.th

Abstract

The concept of automatic essay grading is comparable to the documents clustering or classification according to a set of essays. However, the preparation of the training data need to include every category of answer sets which impractical. The training data usually includes the appropriate essays and it might be available only for some types of data. Additionally, the number of clusters may not be known in advantage. Thus, EM and Cobweb are used to discover automatic the number of clusters for each question. The similarity between each cluster and solution is used to grade the student's answers. Experiment is tested on short descriptive answers of object-oriented programming with total of fifty five students' answers. The result of this experiment revealed EM to return a superior average accuracy value with 80.57%.

Keywords: Automated, Thai-Language, Essay, Scoring, EM Algorithm, Cobweb Algorithm, Unsupervised Learning

1 Introduction

While essay writing on the computer is an even more essential part of the educational process, methodology of automated essay scoring is comparable to that of document categorization and classification. It classifies essays into two sets, one of which is analogous to the solution and another set is to be dissimilar. The students' answers virtually diverge from the solution, with a multitude of levels of similarity; therefore the applications of merely, two clusters are not sufficient. The scoring method requires large training data to create a model of score prediction to assess the levels of similarity to the solution. For example, E-rater system [1] applies multiple a linear regressions model to predict scores. This paper represents an essay with word features, grammars, styles and specific vocabulary usage. A number of machine learning algorithms have been applied for automatic grading. For examples, supervised algorithms are a neural network [2], a Bayesian network [3], and the K-NN [4] algorithms which use training data to create a model for scoring. Text-categorization such as Bayesian independence is applied to distinguish 'good' and 'bad' essays, and then linear regression is used to assign score of essays [5]. To identify patterns or relationships of terms and concepts contained in an unstructured text, LSA technique is deployed [6]. LSA applied Singular Value Decomposition (SVD) to discover terms with tend to have similar meanings. However, simple LSA has some drawbacks. Probabilistic Latent Semantic Analysis (PLSA) technique [7] and syntactic information which is a word order given by the part-of-speech tag [8] has been proposed to automated essay grading system.

Open-ended essays are used assessed student's understanding or ability to express student's opinion. This type of essay does not required solutions and may not required training data. Thus, the scoring is based on the similarity of essays, an unsupervised learning and voting is reported in [9]. The number of unique words and sharing terms between essays is assigned as an initial score. The final score is obtained from the voting algorithms.

This paper presents the automatic Thai language essay scoring with single solution and no training data. An unsupervised learning such as Expectation Maximization (EM) and Cobweb algorithms are investigated number of cluster. Furthermore, the study of effect of similarity to assign essay score is investigated.

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

2 Related Research

Automatic essay system has been implemented and use as a tool for human for many years. Several issues have been studied. The first issue is a comparative study grading between human and computer grading [10][11][12]. The current research has been proved that the computer grading is correlated significantly with human rater.

The essay writing can be classified into two types of essay. The first type of essay writing is used to test writing ability of students. This type of essay which called open-ended questions does not need human answers. Thus, natural language processing and some intelligent systems are the main approaches to score these essays. Finally, the close-ended questions, it is required short answers or training data to grade.

Several researches of automated essay scoring system have reported with diverse of techniques. E-Rater system [1] from ETS, Princeton University has been used for scoring the Graduate Management Admission Test®Analytical Writing Assessment (GMAT®AWA) since 1999. This first version was trained on a sample of essays written on the same topic that had been scored by human readers and composed of two applications, e-rater engine for scoring and Critique for diagnostic feedback. In 2005, Burstein et al extended the E-Rater system by using information extracted from previous version by standardizing some features with regard to essay length. The three most importance to improve modification are feature section, model building and score assignment algorithm. Feature selection is based on statistic usage and national language processing technique. The E-rater evaluated its effectiveness by benchmark with human graded essays.

Another view of essay scoring, it can be viewed as text classification. Bayesian independence and K-Nearest Neighbor (K-NN) algorithms are applied to automatic essay grading [5]. The Bayesian independence classifier is distinguish essay into two classes. Then linear regression is applied to assign score. A neural network [2][6][13] and K-NN algorithm [4] are used to divide the essays into number of categories. The most importance of automatic essay scoring is features selection and transformation essays into vectors.

To identify patterns or relationships of terms and concepts contained in an unstructured text, Latent Semantic Analysis (LSA) technique is deployed [7][8][14]. LSA applied Singular Value Decomposition (SVD) to discover terms with tend to have similar meanings. LSA has been extended by adding syntactic information [8]. There are various ways to define the syntactic information such as a full parse tree, a shallow parse, POS, and tag sequence etc. This paper applied 45 tags from Penn tree-bank.

This paper evaluated their results using answers from the essay of computer science questions. The results from the SELSA approaches confirm the effectiveness of the essay scoring for text better than LSA. The effectiveness of the SELSA approaches depends on the syntactic structure.

For LSA approach, there are many drawbacks such as the probability distribution. The approximation matrix may contain negative entries after the dimension reduction, choosing the number of dimension. In order to solving these problems, the Probabilistic Latent Semantic Analysis (PLSA) technique has been proposed to an automated essay grading system [7]. This technique uses the Expectation Maximization (EM) algorithm to build the model with maximum likelihood probability of the learning task. They performed an experiment using three essay sets collected from courses on education, marketing and software engineering. PLSA was provided equal or better results when the number of latent variable is increased.

In supervised approaches, a training data is required to build the training model which is costing and infeasible in some environment. In practice, the essays can classify into two classes. However, there exists partial corrected answer which is not in both classes. Yen-Yu Chen et al [9] proposed an unsupervised-learning approach based on a voting algorithm. At the initial scoring step, all essays are assigned score based on the number of unique terms of the essay. Then the iterative step, the score of an essay is voted by the other essays. In this step, the score is recalculated based on the similarity measure is based on the number of sharing term between each essay and the previous score. The algorithm of scoring is an iteratively until the score in not change in some number.

The first paper of Thai essay scoring [2] reported in 2007. This work explored the human

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

grading and machine grading. A Backpropagation Neural Network (BNN) and LSA are purposed. 40 high school's essays are used to train a model. Features section based on the word segmentation and IDF weight. However, there is no information about characteristic of input data. It is proved that the BNN with LSA give the best performance.

3 Algorithms Used to Categorize Essays

Unsupervised learning algorithm was applied to cluster essays by the similarity. The essays with similar words were possible to be in the same cluster. In this experiment two algorithms were selected; expectation maximization (EM) and Cobweb algorithms, to automatically cluster without the initial amount of cluster. They can solve the problem about the received contingent cluster number such as in the case that all examinees give the correct essays in the same question resulted as only a cluster. Detail of such algorithms is displayed below:

3.1 EM Algorithm

The Expectation Maximization (EM) algorithm was first introduced by Dempster et al [15] as an approach to estimate unknown parameters. Typically, EM algorithm performs in two steps. First, the log likelihood is computed with respect to the current parameter estimates. Then maximize the expectation computed in the first step. This experiment applied the EM algorithm in the Weka program [10]. The definition of the number of cluster was performed by means of cross validation testing with the steps outlined below:
Step 1: the number of clusters is set to 1
Step 2: the training set is split randomly into 10 folds.
Step 3: EM is performed 10 times using the 10 folds the usual cross validation way.
Step 4: the log likelihood is averaged over all 10 results.
Step 5: if log likelihood has increased the number of clusters is increased by 1 and the program continues at step 2.

Upon the definition of the essay cluster, the initial essay from each cluster was randomized. The subsequent step is to cluster the left essays with the application of the EM algorithm in two steps:

- E-Step

Computation of the membership probability of x in each cluster: with equations (1) to (3)

$$Pr(c|x) = \frac{W_c^j \times Pr^j(x|c)}{Pr^j(x)} \quad (1)$$

$$Pr^j(x|c) = \frac{1}{\sqrt{(2\pi)^d \times |\Sigma^c|}} \times exp\left(\frac{(x-m^c)^T \times (x-m^c)}{2 \times \Sigma^c}\right) \quad (2)$$

$$Pr^j(x) = \sum_{c=1}^{k} W_c^j \times Pr(x|c) \quad (3)$$

where $Pr(c|x)$ is the probability of cluster c in essay x, $Pr_j(x|c)$ is the probability of essay x in cluster c at iteration j, $Pr_j(x)$ is the probability of essay x at iteration j, W_c^j is the coefficients that represent the fraction of the database represented by the corresponding cluster, x is parameters of essays in each record of all essay, m is the average value of the parameter in the essay, d is the dimension of Gaussian probability distribution, and Σ_c is the variance value of cluster c

- M-step
Update of the mixture model parameters and verification of the condition to stop the process as equation (4) to (6)

$$W_c^{j+1} = \sum_{x \in D} Pr(c|x) \quad (4)$$

$$m^{j+1} = \frac{\sum_{x \in D} x \times Pr(c|x)}{\sum_{x \in D} Pr(x|c)} \quad (5)$$

$$\Sigma^j = \frac{\sum_{x \in D} Pr(c|x) \times (x - m^{j+1})}{\sum_{x \in D} Pr(c|x)} \times \frac{(x - m^{j+1})^T}{\sum_{x \in D} Pr(c|x)} \quad (6)$$

Verification of the condition,
if $|E_j - E_{j+1}| \leq \epsilon$, stop Else set $j = j + 1$ and repetition of the identical process. E_j

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

is the log likelihood of the mixture model at iteration j as equation (7)

$$E^j = \sum_{x \in D} log \left(\sum_{c=1}^{k} W^j \times Pr^j(x|c) \right) \tag{7}$$

For example, EM algorithm was used to classify the text from labeled and unlabeled documents [16], with Expectation Maximization for Weakly Labeled Data [17].

3.2 Cobweb Algorithm

COBWEB algorithm constructs a classification tree incrementally by stepwise insertion of the objects into the classification tree. Upon insertion of an object into the classification tree, the COBWEB algorithm traverses the tree top-down initialized from the root node. At each node, the COBWEB algorithm considers four possible operations are inserting, creating, merging, and splitting node and subsequently selects the one that yields the most superior CU function value [18][19] as equation (8) and (9)

$$CU(C) = \frac{1}{n} \times \sum_{i} \Delta(C, A_i) \tag{8}$$

$$\Delta(C|A_i) = \sum_{k=1}^{n} P(C_k) \times \sum_{j} P(A_i = V_{ij}|C_k)^2 - \sum_{j} P(A_i = V_{ij})^2 \tag{9}$$

where CU(C) is the category utility function in cluster C, C_k denotes cluster k, A_i refers to the frequency of unique words in an essay at position i, n is the number of clusters.

For example, Cobweb algorithm was used in Conceptual Clustering Categorical Data with Uncertainty [20], Incremental hierarchical clustering of text documents [21], and Web document classification with the application of machine learning clustering algorithms [22].

4 Experiment

In Figure 1. Shows the processes of automatic Thai language essay scoring system. Details are in the following paragraphs.

4.1 Data Set

Experimental data used in this paper were collected from the examination of an object-oriented programming course. The data set was manually typed and consists of seven questions with each question composed of 55 students' essay. Most of the questions required a short descriptive answer. The score range was defined as 'A' thru 'F', with 'A' as the best answer or close to the solutions. A sample question is "How does Java language support Dynamic Binding or Late Binding?" to which the appropriate respond should have been a short explanation such as "It defines the value of variable at the stage of execution."

4.2 Preprocessing

The preprocessing step is a process to transform textual information into a word vector. Since Thai language differs from English as it does not employ spacing between the individual items of vocabulary; therefore a Thai word segmentation program [23] is needed. In previous work [24], it has been studied Thai stop-word should be removed. To solve the rare terms, TF-IDF is utilized. Since, each Thai essay may contain transliterate word or the synonym problems, these problems are solved by a Thai-English dictionary mapping. Finally, LSA technique was also applied to replace vectors with semantic space. Thus, this work applied all best features which have been done in our previous work.

4.3 Unsupervised Learning Algorithms

Unsupervised learning algorithm is applied for cluster the similar essays into a same group. Thus essays in the same cluster will be graded in the same score. Since number of cluster will not know in advance. The algorithm EM and Cobweb clustering were applied and detail as follows:

4.3.1 Expectation Maximization (EM algorithm)

The EM algorithm is both capable to discover number of cluster and solving the missing values. Experiment with WEKA, a number of parameters need to supplied to the system. Several experiments have been explored to find the best parameters. To automatic identify number of cluster, numClusters is set to -1. Number of iteration is less than 100 (maxIterations = 100),

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

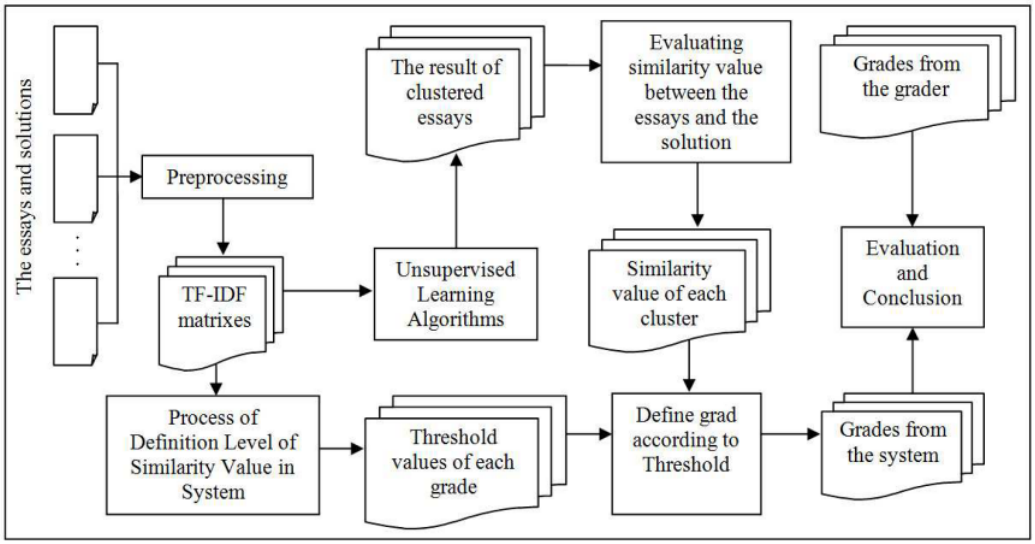


Figure 1. The processes of the automatic Thai language essay scoring system

a minimum acceptable Standard Deviation value of 0.001 (minStdDev = 0.001), and a vector pattern of word frequency in the essay set to 119 times (seed = 119).

When all clusters have been covered, a vector of cluster is the average of TF-IDF of the essays in the same cluster. This vector will be used to find the similarity of the cluster and a solution. Detail of scoring algorithm is shown in Figure 2

4.3.2 Cobweb Algorithm

Clustering by Cobweb algorithm using WEKA, number of parameters is given such as Standard Deviation for accepting a model in this experiment this parameter is 1.0. The threshold value of category utility function that is used for node cutting is 0.0028 (cutoff=0.0028) with the frequency in vector pattern of word frequency sampling set to 42 (seed=42).

4.4 Essay Scores Calculation

When all essays have been grouped according to the clustering algorithms, the scoring is performed. The score criteria in this work are based on the similarity of a cluster and a solution. If the cluster and the solution have the some content the similarity measure is close to 1. According to the grading system, the grader may grade the essay based on the corrected content or reduce a score when content is inappropriate. Additionally, the grading system grade the highest score as an excellent (A), score is lower than 50 is graded as fail (F). Thus, similarity is measured by Cosine equation as given in equation (10)

$$S(c_i, s_j) = \frac{\sum_{k=1}^{m} w_{ki} \times w_{kj}}{\sqrt{\sum_{k=1}^{m} w_{ki}^2} \times \sqrt{\sum_{k=1}^{m} w_{kj}^2}} \quad (10)$$

where $Sim(c_i, s_j)$ is similarity value between cluster i and solution j, w_{ki} and w_{kj} are frequency of word k among cluster i and solution j.

The algorithm for scoring essay is shown in Figure 2

4.5 Finding the similarity range for scoring

Since, the scoring is based on the similarity measure a cluster and a solution. There may have some factors affecting the similarity such as the transformation word frequency vectors. Furthermore, if there is an essay which may contain words out of the content, does this affects the similarity values.

When number of unique words increase, the vector size increases as well. If these unique words are not related to content of essay, the score of these essays should be reduced. Figure 3 showed the effect on number of unique words. If number of unique increased (words are not used in the corrected content) the similarity value is drop. However, the vector size increases due to

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

```
Algorithm for scoring essay.
Scoring function
  Given K is a vector of a corrected answer
  S = {G1,...., GN}  a set of essays which is grouping by
      their similarity and centroid of cluster is average of
      all essays in a cluster.
  sim(K, Gi) using eq. 10 //Computer similarity between
      each group of cluster and answer key (K) using
      cosine similarity
  ck ∈ {0.7, 0.6, 0.5, 0.4 } // ck  is a score criteria by
      dividing similarity into 5-point scale (0.7, 0.6, 0.5,
      0.4). This score criteria may varied for experiment
  for all group
      regrouping which the closed similarity into 5 group
  endfor
  for all essay ei
     for all groups j
        if ei ∈ Gj
          assign ci to  ei
     endfor
  endfor
```

Figure 2. The algorithm for scoring essay

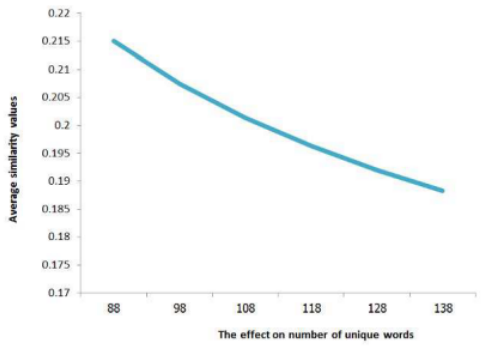


Figure 3. Relationship of average similarity values and the effect on number of unique word

the unique words, the corrected essay and a solution do not effect from this unique words. Thus, to solve the problem of synonyms, the synonyms words must group into one attribute.

Finding the similarity rage for scoring, the essay gets highest score should have similarity closed to 1. Based on this information, the score is divided into five points ('A', 'B', 'C', 'D' and 'F'). Each interval of each range is in between 0.1 and the score of 'A' will starts from 0.8. If similarity less than 0.5, score is assign to 'F'.

5 Effectiveness Evaluation

An accuracy measure is used to evaluation the effectiveness of this experiment. This measure is to compare the results with manual grading. The equation for accuracy was given in (11)

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \quad (11)$$

where TP is the number of essays with both 'true' result from the system and 'true' result from the solution. (True Positive), TN is the number of essays with both 'false' result from the system and 'false' result from the solution. (True Negative), FP denotes the number of essays with 'true' result from the system and 'false' result from the solution. (False Positive), and FN refers to the number of essays with 'false' result from the system and 'true' result from the solution (False Negative).

6 Result and Discussion

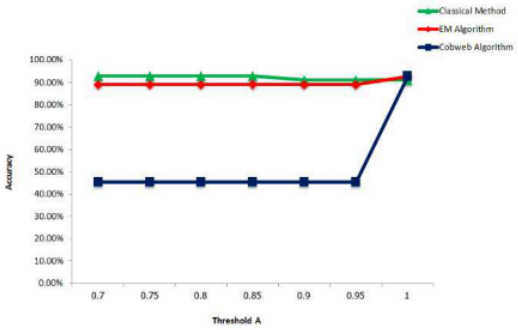


Figure 4. Accuracy values of classical method and two algorithms ordered by the determined ranks of Threshold A

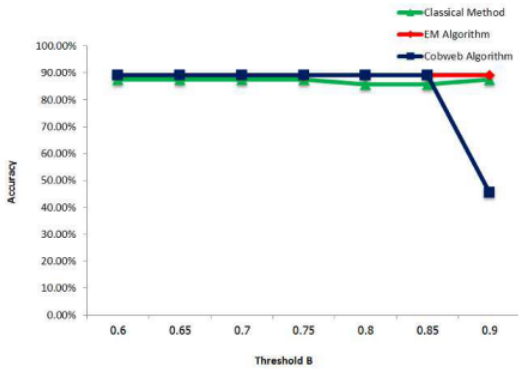


Figure 5. Accuracy values of classical method and two algorithms ordered by the determined ranks of Threshold B

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

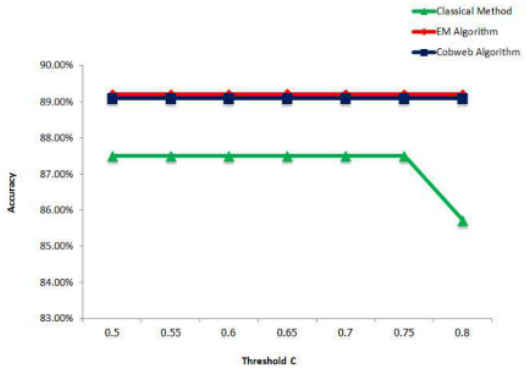


Figure 6. Accuracy values of classical method and two algorithms ordered by the determined ranks of Threshold C

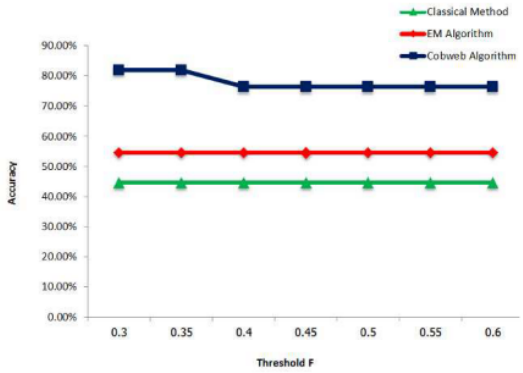


Figure 8. Accuracy values of classical method and two algorithms ordered by the determined ranks of Threshold F

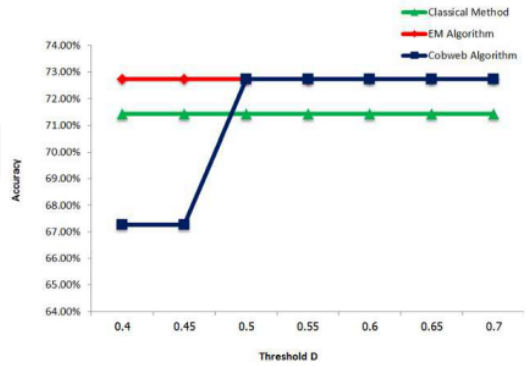


Figure 7. Accuracy values of classical method and two algorithms ordered by the determined ranks of Threshold D

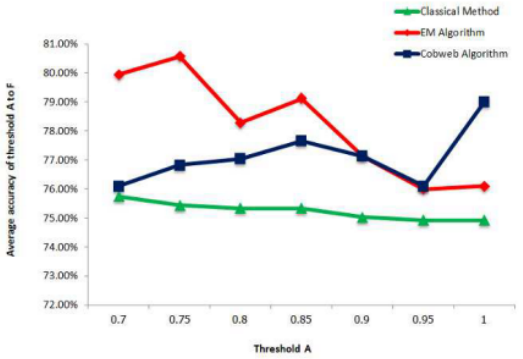


Figure 9. Average accuracy values of classical method and two algorithms ordered by the determined ranks

The effectiveness of the clustering algorithms presented in Table 1. Accuracy of each algorithm is compared with group of human grading. Although human grading has set up a fix point for each grade, a set of essays may not contain all score level. For instance, there are all level grades except level 'D' in data set D4. Additionally, some data set, the number of essays in each cluster may be unequal (see data set D1). Examining the context of data set, it is found that some data sets need more than one solution because writing an essay may use different vocabularies. Furthermore, some essays use the same keywords which are the same as solution but semantic of context is different. Thus it is grouped into the same cluster. This means that a bag of word lack of ability to explore the semantic in the context.

Although, the level of similarity for grading has fixed to 0.8 according to similar content and meaning, this may affect when using the centroid of cluster. Thus, the second experiment is to explore the appropriate level for grading. From Figure 4 to Figure 8 shows similarity level of each point and their accuracy. Figure 9 demonstrates the mean average of score's 'A' of all data sets. This can be concluded that the threshold for scoring A of Cobweb is higher than EM.

The classical method gives the best result when threshold is 0.7, EM algorithm is 0.75, and the Cobweb algorithm is 1.0 with maximum total average accuracy values of 75.75%, 80.57%, and 79.01%, respectively.

Table 2 shows comparison of all grading methods. The K-NN approach is our previous

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

Table 1. The effectiveness of the clustering algorithms

Data set	# Essay	Human	EM		Cobweb	
			# Cluster	Accuracy	# Cluster	Accuracy
D1	41	4(3/17/17/4/0)	4	3.09%	7	5.29%
D2	34	5(2/2/11/15/4)	1	42.73%	5	6.66%
D3	36	5(4/6/6/13/7)	2	43.75%	3	41.05%
D4	12	4 (1/2/4/0/5)	1	41.67%	8	27.78%
D5	36	5(14/9/2/2/9)	1	41.59%	1	38.89%
D6	42	5(5/5/19/11/2)	6	48.41%	2	47.39%
D7	39	5(1/11/10/14/3)	1	39.12%	4	3.62%

Table 2. The total average accuracy of classical method, EM, Cobweb, and K-NN algorithm

Method	Average Accuracy
Classical method	75.75%
EM Algorithm	80.57%
Cobweb Algorithm	79.01%
K-NN Algorithm	81.30%

studies [24] which require 5 training data sets with respected to score rages. The results of the supervised and unsupervised algorithms are not significant. However, the unsupervised algorithms do not requited training data set and the procedure to rater is the same as human rater.

7 Conclusion and Future Work

This article presents the automatic Thai language essay scoring with a single answer and tested with a short descriptive essay. The EM and Cobweb algorithm is applied to group the similar essay. Several research problems have been addressed. For example, the similarity level may effect from the number of unique words, cluster representation and synonyms. Scoring is based on the similarity of a cluster and a solution.

The results of the experiment showed that it is not significant between supervised and unsupervised learning. Additionally, the unsupervised is more practical when the assessment of student is to evaluation the student writing ability and the ability to recall student's knowledge.

For further research, there are a number of issues to investigate such as Thai written structure and word-ordered. Moreover, such a system needs to test by the alternative unsupervised learning methods to increase accuracy. Furthermore, a set of data is needed to verify the accuracy of essays to written words added in addition to the solution.

Acknowledgment

This work was supported by budget for overseas academic conference from the faculty of science, Kasetsart University and the graduate school, Kasetsart University.

References

[1] Y. Attali and J. Burstein. Automated essay scoring with E-Rater. *J. Technology, Learning and Assessment*, 2006.

[2] Chanunya Loraksa and Ratchata Peachavanish. Automatic Thai-language essay scoring using Neural Network and Latent Semantic Analysis. *IEEE*, 2007.

[3] Lawrence M. Rudner and Tahung Liang. Automated essay scoring using Bayes' theorem. *ACM*, 2002.

[4] Li Bin, Lu Jun, Yao Jian-Min, and Zhu Qiao-Ming. Automated essay scoring using the K-NN algorithm. *IEEE*, 2008.

[5] Leah S. Larkey. Automatic essay grading using text categorization techniques. *ACM*, 1998.

[6] Thomas K Landauer, Peter W. Foltz, and Darrell Laham. An introduction to Latent Semantic Analysis. *Discourse Processes*, 1998.

[7] Tuomo Kakkonen, Niko Myller, Jari Timonen, and Erkki Sutinen. Automatic essay

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

grading with probabilistic Latent Semantic Analysis. *ACM*, 2005.

[8] Dharmendra Kanejiya, Arun Kumary, and Surendra Prasad. Automatic evaluation of students' answers using syntactically enhanced LSA. *ACM*.

[9] Yen-Yu Chen, Chien-Liang Liu, Chia-Hoang Lee, and Tao-Hsing Chang. An unsupervised automated essay-scoring system. *IEEE*, 2010.

[10] Salvatore Valenti, Francesca Neri, and Alessandro Cucchiarelli. An overview of current research on automated essay grading. *Material published*, 2003.

[11] John Palmer, Robert Williams, and Heinz Dreher. Automated essay grading system applied to a first year university subject how can we do it better? *Material published*, 2002.

[12] Semire Dikli. Automated essay scoring. *Turkish Online Journal of Distance Education-TOJDE*, 2006.

[13] Sargur Srihari, Jim Collins, Rohini Srihari, Harish Srinivasan, Shravya Shetty, and Janina Brutt-Griffler. Automatic scoring of short handwritten essays in reading comprehension tests. *Artif. Intell*, 2008.

[14] Tuomo Kakkonen and Erkki Sutinen. Automatic assessment of the content of essays based on course materials. *IEEE*, 2003.

[15] A.P.Dempster, N.M.Laird, and D.B.Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society*, 1977.

[16] Kamal Nigam, Andrew Kachites Mccallum, Sebastian Thrun, and Tom Mitchell. Text classification from labeled and unlabeled documents using EM. *ACM*, 2000.

[17] Yuri Ivanov, Bruce Blumberg, and Alex Pentland. Expectation Maximization for weakly labeled data. *ACM*, 2001.

[18] Douglas H. Fisher. Knowledge acquisition via incremental conceptual clustering. *ACM*, 1987.

[19] Boris Mirkin. Reinterpreting the Category Utility function. *ACM*, 2001.

[20] Yuni Xia and Bowei Xi. Conceptual clustering categorical data with uncertainty. *IEEE*, 2007.

[21] Nachiketa Sahoo, Jamie Callan, Ramayya Krishnan, George Duncan, and Rema Padman. Incremental hierarchical clustering of text documents. *ACM*, 2006.

[22] Marlon Gregory, Roberto Scata, and Eric Brown. Web document classification using machine learning clustering algorithms. *ACM*, 2006.

[23] Canasai Kruengkrai1 and Chuleerat Jaruskulchai Jun'ichi Kazama. A discriminative hybrid model for Thai word segmentation. *National Software Contest Thailand*, 2009.

[24] Sommart Aungkaseraneekul and Chuleerat Jaruskulchai. Automated Thai-language essay scoring. *National Computer Science and Engineering Conference*, 2010.

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

ภาคผนวก ค

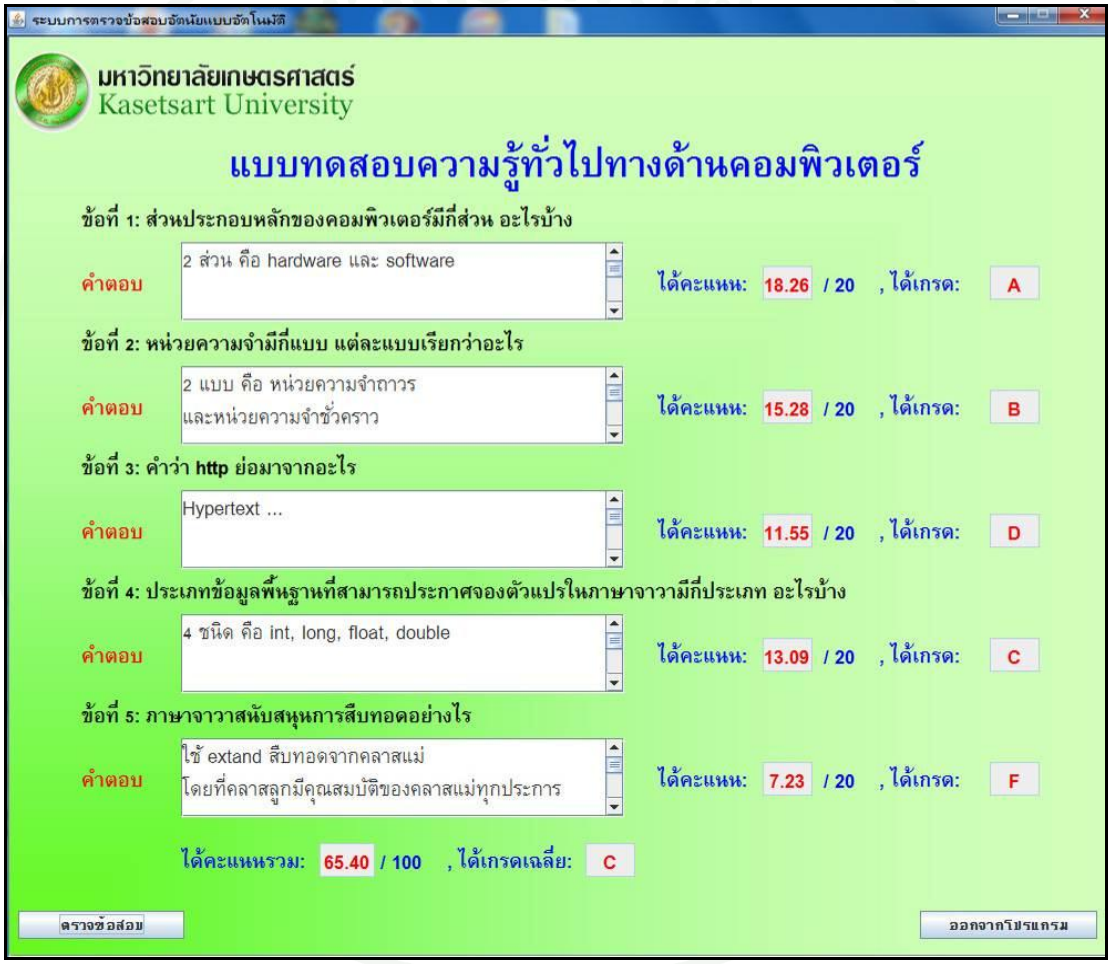
ตัวอย่างโปรแกรมตรวจข้อสอบอัตนัยภาษาไทย

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

ภาคผนวก ค

ต้นแบบโปรแกรมการตรวจข้อสอบอัตนัยภาษาไทยแบบอัตโนมัติในงานวิจัยนี้

ต้นแบบโปรแกรมการตรวจข้อสอบอัตนัยภาษาไทยแบบอัตโนมัติในงานวิจัยนี้ ได้ส่งประกวดโครงการ IT One Innovation Awards 2011 (http://www.itone.co.th/innovationawards/) ซึ่งได้รับรางวัลรองชนะเลิศอันดับ 1



ภาพผนวกที่ ค1 โปรแกรมตรวจข้อสอบอัตนัยภาษาไทยด้วยวิธี Classical method

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

ภาพผนวกที่ ค2 โปรแกรมตรวจข้อสอบอัตนัยภาษาไทยด้วยอัลกอริทึม EM

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

ประวัติการศึกษา และการทำงาน

ชื่อ – นามสกุล	สมมาตร อังคเศรณีกุล
วัน เดือน ปี ที่เกิด	7 พฤศจิกายน พ.ศ. 2528
สถานที่เกิด	กรุงเทพมหานคร
ประวัติการศึกษา	วท.บ. (วิทยาการคอมพิวเตอร์) ระดับปริญญาตรี มหาวิทยาลัยศรีนครินทรวิโรฒ ประสานมิตร
ประวัติการทำงาน	ผู้ช่วยนักวิจัยชั่วคราวที่ Biotec ณ ห้องปฏิบัติการ Bioinformatics สำนักงานพัฒนาวิทยาศาสตร์และเทคโนโลยีแห่งชาติ (สวทช.)
ทุนการศึกษาที่ได้รับ	ได้รับทุนนำเสนอผลงานวิชาการในประเทศและต่างประเทศ จากบัณฑิตวิทยาลัยและภาควิชาวิทยาการคอมพิวเตอร์ มหาวิทยาลัยเกษตรศาสตร์ (พ.ศ. 2553-2554)

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์