ระบบช่วยสรุปใจความสำคัญภาษาไทยอัตโนมัติ โดยใช้เทคนิคการจำแนกแบบไบนารี

AUTOMATIC THAI-TEXT SUMMARIZATION SYSTEM USING BINARY CLASSIFICATION

โดย

กาญจนิจ กิจกสิวัฒน์

นิชาภา ฉนอมสิงห์

อาจารย์ที่ปรึกษา

ผศ.ดร.พรฤดี เนติโสภากุล

นักวิจัยที่ปรึกษาร่วม

ดร.เทพชัย ทรัพย์นิธิ

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิทยาศาสตรบัณฑิต

สาขาวิชาเทคโนโลยีสารสนเทศ

คณะเทคโนโลยีสารสนเทศ

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ภาคเรียนที่ 2 ปีการศึกษา 2551

AUTOMATIC THAI-TEXT SUMMARIZATION SYSTEM

USING BINARY CLASSIFICATION

KANCHANIT KITKASIWAT

NICHAPA THANOMSING

A PROJECT SUBMITTED IN PARTIAL FULFILLMENT

OF THE REQUIREMENT FOR THE DEGREE OF

BACHELOR OF SCIENCE PROGRAM IN INFORMATION TECHNOLOGY

FACULTY OF INFORMATION TECHNOLOGY

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

2/2008

ใบรับรองปริญญานิพนธ์ ประจำปีการศึกษา 2551

คณะเทคโนโลยีสารสนเทศ

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง ระบบช่วยสรุปใจความสำคัญภาษาไทยอัตโนมัติ โดยใช้เทคนิคการจำแนกแบบไบนารี

AUTOMATIC THAI-TEXT SUMMARIZATION SYSTEM USING BINARY CLASSIFICATION

ผู้จัดทำ

1. นางสาว กาญจนิจ กิจกสิวัฒน์ รหัสนักศึกษา 48070095
2. นางสาว นิชาภา ถนอมสิงห์ รหัสนักศึกษา 48070134

.....อาจารย์ที่ปรึกษา

(ผศ.ดร.พรฤดี เนติโสภากุล)

หัวข้อวิทยานิพนธ์	ระบบช่วยสรุปใจความสำคัญภาษาไทยอัตโนมัติ โดยใช้เทคนิคการจำแนกแบบไบนารี
นักศึกษา	นางสาวกาญจนิจ กิจกสิวัฒน์ นางสาวนิชาภา ถนอมสิงห์
รหัสนักศึกษา	48070095 48070134
ปริญญา	วิทยาศาสตรบัณฑิต
สาขาวิชา	เทคโนโลยีสารสนเทศ
ปีการศึกษา	2551
อาจารย์ที่ปรึกษา	ผศ.ดร.พรฤดี เนติโสภากุล
นักวิจัยที่ปรึกษาร่วม	ดร.เทพชัย ทรัพย์นิธิ

บทคัดย่อ

โครงงานนี้มีจุดประสงค์เพื่อพัฒนาระบบสรุปใจความสำคัญอัตโนมัติภาษาไทยเพื่อช่วยให้ผู้ใช้ประหยัดเวลาในการอ่านบทความต่างๆ ภาษาไทย เทคนิคที่ใช้คือ การให้ระบบตัดคำฟุ่มเฟือยที่อยู่ในประโยคออก แต่เนื่องจากปัจจุบัน ไม่มีโครงงานใดที่ยืนยันว่าคำประเภทใดเป็นคำฟุ่มเฟือย จึงต้องสร้างคลังข้อมูลเพื่อวิเคราะห์หารูปแบบคำฟุ่มเฟือยโดยใช้โปรแกรมช่วยสร้างคลังข้อมูลและสร้างจากบทความด้าน “อาหารและสุขภาพ” ต่อจากนั้นนำคลังข้อมูลนี้ไปสร้างเป็นเทรนนิ่งเซต เพื่อให้ระบบเรียนรู้ว่าคำใดในประโยคคำใดเป็นคำฟุ่มเฟือยโดยใช้เทคนิคการจำแนกแบบไบนารี เปรียบเทียบประสิทธิภาพของโมเดลการเรียนรู้จากการจัดประเภทข้อมูลแบบต่างๆ ได้แก่ เนอีฟเบย์, เบย์เซียน เน็ตเวิร์ก, แม็กซิมัม เอนโทรปี, ซับพอท เวกเตอร์ แมชชีน นอกจากนี้ยังมีการพิจารณาคุณสมบัติของคำ ได้แก่ ชนิดของคำในประโยค คุณสมบัติของคำและคุณสมบัติต่างๆ ของคำข้างเคียง จากนั้นทดลองปรับเปลี่ยนการจัดประเภทและคุณสมบัติที่ใช้เพื่อเลือกโมเดลที่มีประสิทธิภาพสูงสุดมาประยุกต์ใช้พัฒนาระบบ

การทำงานของโครงงานแบ่งเป็นสามช่วงได้แก่ ช่วงแรกคือการรวบรวมคลังข้อมูลจากบทความภาษาไทยเพื่อสร้างเป็นเทรนนิ่งเซตแบบต่างๆ ช่วงที่สองเป็นการให้แมชชีนเลินนิ่ง เรียนรู้ข้อมูลเพื่อสร้างโมเดล สำหรับโมเดลที่ดีที่สุด คือ โมเดลของ เบย์เซียน เน็ตเวิร์กเป็นผลมาจากการทดสอบเปรียบเทียบประสิทธิภาพในการตัดคำกับเบสไลน์ พบว่า ค่าความถูกต้องของการตัดคำของโมเดลมีค่ามากกว่าเบสไลน์ โดยโมเดลและเบสไลน์ มีค่า F-measure ของการตัดคำเท่ากับ 46.6 และ 23.2 ตามลำดับ สำหรับช่วงสุดท้ายเป็นการนำโมเดลที่ได้ มาพัฒนาเป็นระบบสรุปใจความสำคัญอัตโนมัติ และทดสอบประสิทธิภาพการทำงานกับเอกสารต่างๆ

Thesis Title	Automatic Thai-Text Summarization System using Binary Classification
Student	Ms.Kanchanit Kitkasiwat Ms.Nichapa Thanomsing
Student ID.	48070095 48070134
Degree	Bachelor of Science
Programme	Information Technology
Academic Year	2008
Advisor	Asst.Prof. Dr.Ponrudee Netisopakul
Co-Advisor	Dr. Thepchai Supnithi, NECTEC, Thailand

ABSTRACT

The objective of this research is to develop the Automatic Thai-Text Summarization System which helps user to not waste the times when reading any articles. The specific technique is reducing the modifiers of the articles. Since the other researches can't recommend what type of word is a modifier now, so a corpus which classifies the forms of is necessary. The reachers use a program to generates the corpus of the modifiers from Food and Health's articles. Then, builts the training set from it to let the system learn which words in the sentences are modifiers by testing a Binary Classification experiment to find the comparison between each class of models' efficiency such as Naïve Bayes, Bayesian Network, Maximum Entropy, Support Vector Machines. In addition, there is a consideration of features, Part-Of-Speech, content word, and any function of adjacent words. This experiment will adjust the classifier and feature to find the most effective model.

The research is divided to three periods. The first period is to collect the corpus to make a training set . Then, the training set will be invited. The second period is to set the machine learning from the article to create the model which the best model is Bayesian network. This model is the result from the efficiency of reducing words and beseline. So, the accuracy value if reducing words in the model is more than in baseline which have F-measure of reducing words in the model and baseline are 46.6 and 23.2 respectively. The final period is to develop the Automatic Thai-Text Summarization System from the model and testing.

กิตติกรรมประกาศ

โครงงานฉบับนี้สำเร็จได้ด้วยความกรุณาจากอาจารย์ที่ปรึกษา ผศ.ดร.พรฤดี เนติโสภากุล และความกรุณาจาก ดร.เทพชัย ทรัพย์นิธิ (นักวิจัยหัวหน้ากลุ่มวิจัยเทคโนโลยีประมวลข้อความ หน่วยปฏิบัติการวิจัยวิทยาการมนุษยภาษา ศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์) ที่คอยให้ความช่วยเหลือ ให้คำชี้แนะ และช่วยแก้ปัญหา ตลอดจนให้ความรู้และประสบการณ์ที่ดีแก่พวกข้าพเจ้า

ขอขอบพระคุณ นายณัฐพล กฤษสุทธิกุล, นายพีรเชษฐ ปอแก้ว, นายธเนศ เรืองรจิตปกรณ์ นักวิจัยหน่วยปฏิบัติการวิจัยวิทยาการมนุษยภาษา ศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์ ที่คอยให้คำปรึกษา คำแนะนำ และชี้แนะแนวทางการทำงาน รวมทั้งแนวทางการพัฒนาระบบให้แก่พวกข้าพเจ้า จนในที่สุดทำให้โครงงานฉบับนี้สำเร็จลงได้

ขอขอบพระคุณ อาจารย์สุพัณณดา โชติพันธ์, ผศ.ธนิศา เครือไวศยวรรณ, รศ.ดร.อาริต ธรรมโน และอาจารย์วารุนี เครือคล้าย กรรมการสอบหัวข้อและโครงร่างโครงงานที่ได้กรุณาให้คำแนะนำตลอดจนข้อชี้แนะต่างๆ

ขอขอบคุณ เพื่อน ๆ ของพวกข้าพเจ้าที่ให้ความช่วยเหลือ ปรึกษา ช่วยแก้ปัญหาและเป็นกำลังใจที่ดีตลอดมา

สุดท้ายขอขอบคุณ ครอบครัว ของพวกข้าพเจ้าที่ให้คำแนะนำ และเป็นกำลังใจที่ดีตลอดมาเช่นกัน

สำหรับคุณงามความดีอันใดที่เกิดจากโครงงานฉบับนี้ พวกข้าพเจ้าขอมอบให้กับบิดามารดา ซึ่งเป็นที่รักและเคารพอย่างยิ่ง ตลอดจนครูอาจารย์ที่เคารพทุกท่านที่ได้ประสิทธิ์ประสาทวิชาความรู้และถ่ายทอดประสบการณ์ที่ดีให้แก่พวกข้าพเจ้า

กาญจนิจ กิจกสิวัฒน์
นิชาภา ถนอมสิงห์

สารบัญ

หน้า

สารบัญ (ต่อ)

หน้า

สารบัญตาราง

สารบัญตาราง (ต่อ)

ตารางที่ หน้า

สารบัญรูป

สารบัญรูป (ต่อ)

สารบัญรูป (ต่อ)

สารบัญรูป (ต่อ)

รูปที่ หน้า

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ในปัจจุบันมีข้อมูลข่าวสารมากมาย หากจะอ่านข้อมูลทั้งหมดต้องใช้เวลามาก จึงเกิดการวิจัยและพัฒนาระบบสรุปใจความสำคัญเพื่อลดรูปข้อความและข่าวสารต่างๆ ให้เหลือเฉพาะใจความที่สำคัญเท่านั้น

แต่งานวิจัยเกี่ยวกับระบบช่วยสรุปใจความสำคัญภาษาไทยยังอยู่ในช่วงเริ่มต้น เนื่องจากโครงสร้างประโยคมีความซับซ้อน การหาขอบเขตของประโยคในภาษาไทยทำได้ยาก คนไทยจึงขาดเครื่องมือที่จะช่วยลดทอนเวลาในการอ่าน ทำให้ขาดโอกาสที่จะรับรู้ข้อมูลข่าวสารต่างๆ ที่มีอยู่มากมายในแต่ละวัน ทั้งนี้เพราะบทความหรือข่าวสารต่างๆ นั้น นอกจากใจความสำคัญแล้วมักมีคำฟุ่มเฟือยรวมอยู่ด้วย เนื่องจากผู้เขียนต้องการให้เกิดความสละสลวยของภาษานั่นเอง

ดังนั้น หากมีเครื่องมือช่วยสรุปใจความสำคัญภาษาไทยอัตโนมัติ ที่ช่วยตัดคำเหล่านั้นออกไป จะทำให้ผู้ใช้ประหยัดเวลาในการอ่านลง ทำให้สามารถรับรู้ข้อมูลข่าวสารต่างๆ ได้มากขึ้นโดยใช้เวลาในการอ่านที่ไม่มากมายนัก

1.2 ความมุ่งหมายและวัตถุประสงค์ของโครงงานวิจัย

1. เพื่อพัฒนาเครื่องมือช่วยสร้างข้อมูลสำหรับใช้ในการสรุปใจความสำคัญภาษาไทยอัตโนมัติ
2. เพื่อสร้างคลังข้อมูล ใช้สำหรับเป็นแหล่งเรียนรู้สำหรับเครื่องมือ เพื่อเป็นฐานข้อมูลของระบบการสรุปใจความสำคัญภาษาไทยอัตโนมัติของบทความด้าน “อาหารและสุขภาพ”
3. เพื่อพัฒนาระบบช่วยสรุปใจความสำคัญภาษาไทยอัตโนมัติในรูปแบบโอเพนซอร์ส (Open Source) ที่ช่วยให้ผู้ใช้สามารถประหยัดเวลาในการอ่านเอกสารภาษาไทย และพัฒนาความสามารถในการสรุปใจความสำคัญของบทความ

1.3 สมมติฐานของการศึกษา

การสื่อสารด้วยภาษาไทย เพื่อให้ได้ใจความเดียวกันนั้น สามารถเป็นประโยคได้หลากหลายรูปแบบ นอกจากนี้ผู้เขียนมักต้องการให้บทความมีความสละสลวยทางภาษา จึงมักมีคำฟุ่มเฟือยรวมอยู่ด้วย

ดังนั้น หากมีเครื่องมือที่ช่วยสรุปใจความสำคัญ จะทำให้ผู้รับสารประหยัดเวลาในการประมวลความหมายลงได้ ในโครงงานเล่มนี้จะนำทฤษฎีแมชชีนเลินนิ่ง (Machine Learning) ที่มีคุณสมบัติในการเป็นตัวจำแนก (Classifier) แบบต่างๆ มาทำการเรียนรู้คำฟุ่มเฟือยในคลังข้อมูล

ด้าน “อาหารและสุขภาพ” ที่ได้สร้างขึ้นมา จากการเรียนรู้จะได้โมเดลและทำการเปรียบเทียบความถูกต้องในการทำงานของโมเดลแต่ละตัว เพื่อหาโมเดลที่มีความเหมาะสมกับการสรุปใจความสำคัญภาษาไทยมากที่สุด

1.4 ทฤษฎีหรือแนวคิดที่ใช้ในการวิจัย

เทคนิคที่จะนำเสนอในโครงงานนี้คือ การเรียนรู้อัตโนมัติเพื่อตัดคำฟุ่มเฟือย และนำโมเดลการเรียนรู้มาประยุกต์ใช้สำหรับพัฒนาระบบสรุปใจความอัตโนมัติภาษาไทย แต่อย่างไรก็ตามในปัจจุบัน ยังไม่มีงานวิจัยที่ยืนยันว่าคำประเภทใดบ้างที่เป็นคำที่ฟุ่มเฟือยในประโยค จึงจำเป็นอย่างยิ่งที่ต้องสร้างคลังข้อมูลเพื่อวิเคราะห์ว่ารูปแบบคำฟุ่มเฟือยเป็นอย่างไร และเมื่อได้คลังข้อมูลดังกล่าวแล้ว จะนำไปเป็นข้อมูลตัวอย่างเพื่อให้แมชชีนเลินนิ่งที่มีคุณสมบัติในการเป็นตัวจำแนกแบบต่างๆ เรียนรู้เพื่อสร้างโมเดลโดยใช้เทคนิคการจำแนกแบบไบนารี (Binary Classification) และเปรียบเทียบประสิทธิภาพในการตัดคำฟุ่มเฟือยออกจากบทความ ของโมเดลทั้งหมด และนำโมเดลที่ดีที่สุดมาพัฒนาเป็น “ระบบช่วยสรุปใจความสำคัญภาษาไทยอัตโนมัติ”

เทคนิคการจำแนกแบบไบนารี เป็นเทคนิคที่จะให้ระบบได้เรียนรู้ว่า ในประโยคคำใดบ้างที่ฟุ่มเฟือย

ตัวอย่างเช่น มีประโยคตั้งต้น และประโยคที่ตัดคำฟุ่มเฟือยแล้วดังนี้

“ประเทศไทยของเรากำลังเกิดวิกฤติเศรษฐกิจ เนื่องจากราคาข้าว และราคาน้ำมันที่พุ่งสูงขึ้น” (ประโยคตั้งต้น)

“ประเทศไทยของเรากำลังเกิดวิกฤติเศรษฐกิจ เนื่องจากราคาข้าว และราคาน้ำมันที่พุ่งสูงขึ้น” (ประโยคที่ตัดคำฟุ่มเฟือยแล้ว)

สิ่งที่ต้องการคือ สร้างโมเดลเพื่อแยกแยะว่า คำใดเป็นคำที่ฟุ่มเฟือยและคำใดที่ไม่ใช่ โดยอาจดูจากองค์ประกอบต่างๆ เช่น รูปผิวของคำ (Surface) หรือ คำที่อยู่รอบข้าง หรือปัจจัยอื่นๆที่คิดว่ามีผลต่อการพิจารณาคำฟุ่มเฟือย จากตัวอย่างข้างต้นคำที่ถูกขีดเส้นใต้เป็นคำที่ฟุ่มเฟือย ซึ่งในโครงงานนี้จะเปรียบเทียบโมเดลที่ได้จากตัวจำแนกแบบต่างๆ ได้แก่ 1. เนอีฟเบย์ (Naïve Bayes) 2. เบย์เซียน เน็ตเวิร์ก (Bayesian Network) 3. แม็กซิมัม เอนโทรปี (Maximum Entropy) 4. ซับพอท เวกเตอร์ แมชชีน (Support Vector Machine)

เนอีฟเบย์ เป็นการแยกประเภทของข้อมูลที่อาศัยหลักการของเงื่อนไขของความน่าจะเป็น (Conditional Probability) โดยพิจารณาว่าอินพุตเวกเตอร์ d จะอยู่ในคลาสใด โดยอาศัยสมการ

$$c^* = \arg\max_c P(c \mid d) \qquad (1.1)$$

และจากกฎของ Bayes

$$P(c \mid d) = \frac{P(c)P(d \mid c)}{P(d)} \tag{1.2}$$

จะได้

$$c * \arg\max_c \frac{P(cP(d \mid c))}{P(d)} \tag{1.3}$$

เบย์เซียน เน็ตเวิร์ก เป็นกราฟฟิกโมเดลทางด้านความน่าจะเป็นที่แทนเซตของตัวแปรและค่าความน่าจะเป็นที่เป็นอิสระต่อกัน โหนดต่างๆ สามารถแทนตัวแปรแต่ละประเภทได้ และไม่จำกัดการแทนค่าของตัวแปรสุ่มที่แทนลักษณะอื่นของเบย์เซียน เน็ตเวิร์ก ดังนั้นประสิทธิภาพของอัลกอลิทึมที่มีจะแสดงการอนุมานและการเรียนรู้ของเบย์เซียน ถ้าการกระจายแบบร่วมกันของค่าโหนดต่างๆ เป็นผลลัพธ์ของการกระจายขั้นพื้นฐานของแต่ละโหนดและโหนดพ่อแม่ของมัน จะได้ดังสมการ

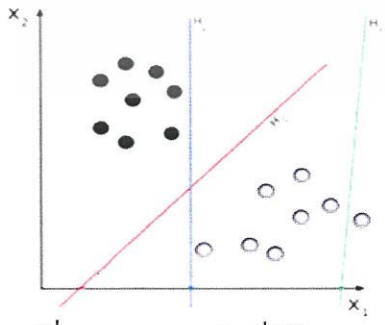
$$P(X_1,....,X_n) = \prod_{i=1}^{n} P(X_i \mid parents(X_i)) \tag{1.4}$$

แม็กซิมัม เอนโทรปี เป็นการแยกประเภทข้อมูลอีกแบบนึง ซึ่งบางครั้งให้ผลดีกว่าในเนอีฟเบย์ สำหรับการจำแนกข้อความ (Text Classification) ซึ่งสมการแม็กซิมัม เอนโทรปี เป็นดังนี้

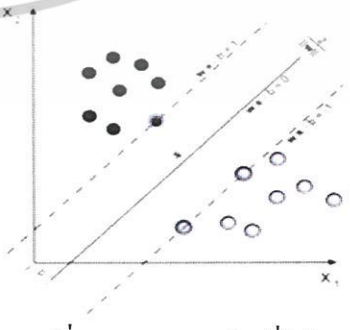
$$P_{ME}(c \mid d) := \frac{1}{Z(d)} \exp(\sum_i \lambda_{i,c} F_{i,c}(d,c) \tag{1.5}$$

โดยที่ $Z(d)$ คือฟังก์ชันที่นอลมอลไลซ์ (Normalization Function) และ $F_{i,c}$ คือฟีเจอร์คลาสฟังก์ชัน (Feature Class Function) และ λ คือค่าน้ำหนักของ $F_{i,c}$

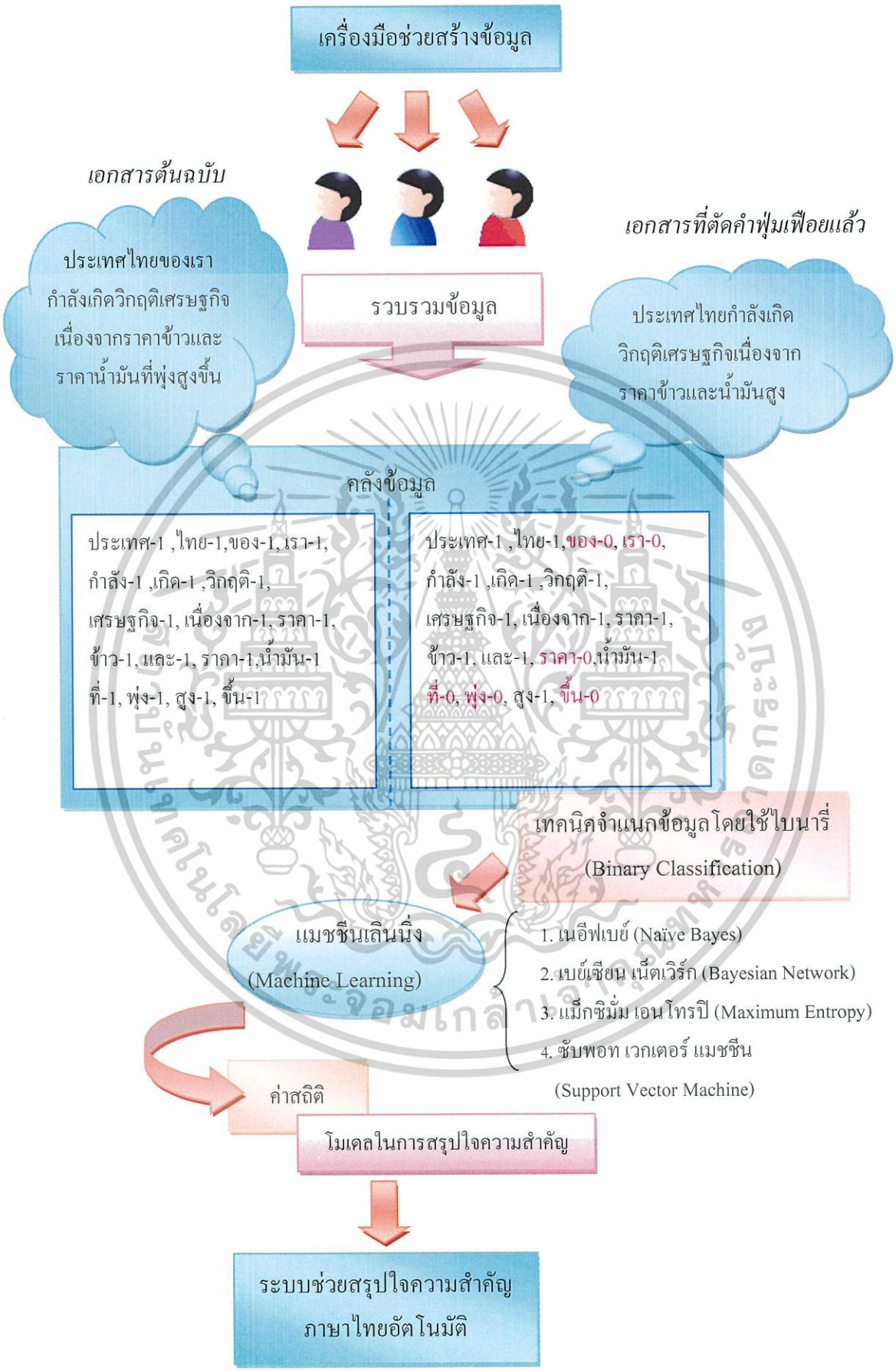
ซับพอทเวกเตอร์แมชชีน เป็นกระบวนการจำแนกโดยการหาระนาบเกิน (Hyperplane) เพื่อแบ่งข้อมูล 2 ชุดออกจากกัน โดยที่ให้ขนาดของขอบมีมากที่สุด



รูปที่ 1.1 ระนาบเกินที่ได้จากการตัวจำแนกแบบอื่นๆ



รูปที่ 1.2 ระนาบเกินที่ได้จากซับพอทเวกเตอร์แมชชีน

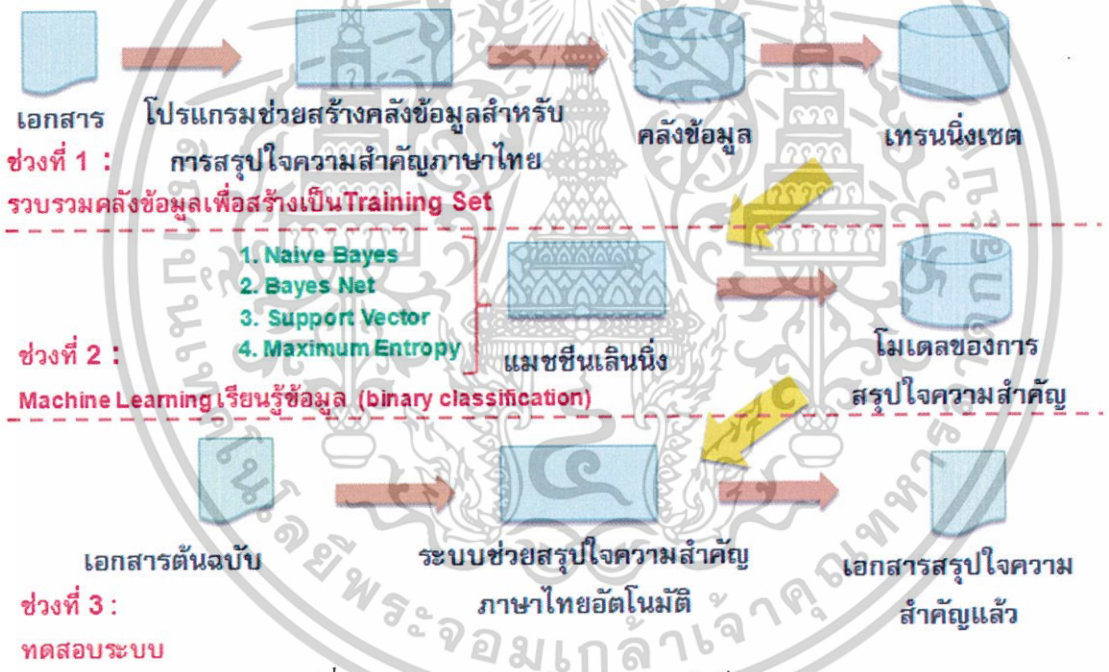


รูปที่ 1.3 ภาพรวมของระบบ

1.5 ขอบเขตและระเบียบวิจัย

ในโครงงานนี้ได้นำเสนอการพัฒนาระบบช่วยสร้างข้อมูลสำหรับใช้ในการสรุปใจความสำคัญภาษาไทยอัตโนมัติ โดยใช้ภาษาจาวา (JAVA) เป็นเครื่องมือในการรวบรวมข้อมูลเพื่อสร้างเป็นคลังข้อมูลของบทความด้าน “อาหารและสุขภาพ” และทำการสร้างโมเดลจากทฤษฎีการจัดประเภทข้อมูลที่แตกต่างกัน ซึ่งจะใช้เครื่องมือชื่อเวก้า (WEKA Tool) ที่สามารถสร้างโมเดลจากการเรียนรู้ของแมชชีนเลินนิ่งที่มีคุณสมบัติในการเป็นตัวจำแนกแบบต่างๆ จากนั้นจะนำโมเดลแบบต่างๆนั้นมาวัดและเปรียบเทียบประสิทธิภาพในการทำงาน สุดท้ายจะมีการเลือกโมเดลที่ดีที่สุดมาพัฒนาเป็นระบบช่วยสรุปใจความสำคัญภาษาไทยอัตโนมัติต่อไป

1.6 ขั้นตอนของการศึกษา



รูปที่ 1.4 ภาพรวมของโครงงาน แบ่งเป็น 3 ช่วง

ช่วงที่ 1 ช่วงการรวบรวมคลังข้อมูลเพื่อสร้างเป็นเทรนนิ่งเซต (Training Set)

เริ่มจากรวบรวมเอกสารบทความภาษาไทยในหัวข้อที่สนใจ นำไปสร้างคลังข้อมูลโดยใช้โปรแกรมช่วยสร้างคลังข้อมูล ซึ่งจะมีการวิเคราะห์ว่าคำใดเป็นคำฟุ่มเฟือยในประโยคต่างๆ ต่อจากนั้นนำคลังข้อมูลดังกล่าวไปสร้างเป็นเทรนนิ่งเซตแบบต่างๆ ซึ่งมีทั้งหมด 22 แบบ

ช่วงที่ 2 แมชชีนเลินนิ่ง เรียนรู้ข้อมูลเพื่อสร้างโมเดล

ใช้แมชชีนเลินนิ่งที่มีคุณสมบัติในการเป็นตัวจำแนกทั้ง 4 ตัวมาเรียนรู้เทรนนิ่งเซตแบบ

ต่างๆ และสร้างเป็นโมเดล จากนั้นทำการเปรียบเทียบหาโมเดลที่มีความถูกต้องในการตัดคำสูงสุด และนำโมเดลนั้นมาทดสอบประสิทธิภาพในการตัดคำเทียบกับเบสไลน์(Baseline) โดยเป็นโมเดลที่เกิดขึ้นจากการพิจารณาค่าความถี่ที่มากกว่าของการตัดและไม่ตัดของคำในเทรนนิ่งเซต และเลือกโมเดลที่ดีที่สุดนั้นเป็น “โมเดลของการสรุปใจความสำคัญ”

ช่วงที่ 3 การทดสอบระบบ

นำการโมเดลที่ได้ มาพัฒนาเป็นระบบสรุปใจความสำคัญอัตโนมัติ และทดสอบประสิทธิภาพการทำงานของระบบกับเอกสารต่างๆ

1.7 แผนกิจกรรมที่จะดำเนินการ

ตารางที่ 1.1 เวลาและแผนของกิจกรรมที่จะทำ ระยะเวลาในการปฏิบัติงาน 10 เดือน ตั้งแต่ มิถุนายน 2551 ถึง มีนาคม 2552

กิจกรรมในการดำเนินการ	เดือน									
	ปี 2551							ปี 2552		
	มิ.ย.	ก.ค.	ส.ค.	ก.ย.	ต.ค.	พ.ย.	ธ.ค.	ม.ค.	ก.พ.	มี.ค.
1.ศึกษาวิธีการตัดคำฟุ่มเฟือยออกจากบทความภาษาไทย พร้อมทั้งวิเคราะห์กรณีต่างๆของการตัดคำที่เกิดขึ้น	←	→								
2.ศึกษาเทคนิคและอัลกอริธึม ต่างๆที่ใช้ในการสรุปใจความสำคัญ พร้อมทั้งหาข้อดี ข้อเสียของแต่ละวิธี	←	→								
3.พัฒนาเครื่องมือช่วยสร้างข้อมูล - ออกแบบหน้าจอของระบบ - ออกแบบโครงสร้างข้อมูล - คิด Function ต่างๆ - เชื่อมต่อโครงสร้างต่างๆจนใช้งานได้จริง และมีความง่ายในการใช้		←	→							
4.กระจายเครื่องมือช่วยสร้างข้อมูลสู่ผู้ใช้ เพื่อให้ผู้ใช้ช่วยสร้างข้อมูล และรวบรวมข้อมูลทั้งหมดที่ได้เป็นคลังข้อมูล			←	→						
5.ออกแบบระบบช่วยสรุปใจความภาษาไทยอัตโนมัติ			←	→						
6.ออกแบบโครงสร้างข้อมูลที่ใช้ในระบบช่วยสรุปใจความภาษาไทย				←	→					

ตารางที่ 1.1 (ต่อ) เวลาและแผนของกิจกรรมที่จะทำ ระยะเวลาในการปฏิบัติงาน 10 เดือน ตั้งแต่ มิถุนายน 2551 ถึง มีนาคม 2552

กิจกรรมในการดำเนินการ	เดือน									
	ปี 2551							ปี 2552		
	มิ.ย.	ก.ค.	ส.ค.	ก.ย.	ต.ค.	พ.ย.	ธ.ค.	ม.ค.	ก.พ.	มี.ค.
7.พัฒนาระบบช่วยสรุปใจความภาษาไทยอัตโนมัติ					◄	———	►			
8.ทดสอบระบบเปรียบเทียบกับการทำงานกับอัลกอริธึมอื่นๆ และปรับปรุงระบบให้มีประสิทธิภาพที่ดีขึ้น							◄——►			
9. ทดสอบการใช้งานจริง									◄——►	
10.รายงานและสรุปผล										◄——►

บทที่ 2
ทฤษฎีพื้นฐานที่ใช้ในการวิจัย

ในหัวข้อนี้จะกล่าวถึงทฤษฎีพื้นฐานต่างๆ ที่เกี่ยวข้องในการวิจัยของโครงงานนี้ ทั้งแนวทางของการสรุปใจความสำคัญ ลักษณะและประเภทการสรุปใจความสำคัญ และงานวิจัยที่มีมาก่อนหน้านี้ รวมถึงคุณลักษณะของแมชชีนเลินนิ่ง (Machine Learning) ที่มีคุณสมบัติในการเป็นตัวจำแนก (Classifier) แบบต่างๆ ที่จะนำมาใช้ในการวิจัยด้วย

2.1 ลักษณะของการสรุปใจความสำคัญ

สามารถแบ่งลักษณะของการสรุปใจความสำคัญ ได้ดังต่อไปนี้

2.1.1 ยึดตามค่าความถี่เป็นหลักสำคัญ (Surface-Level หรือ Frequency-base) (Aone,C., Gorlinshy, J. and Others, 1999 and Inderjeet Mani and Mark T. Maybury,1998) วิธีการนี้ จะดูความถี่และตำแหน่งของคำในเอกสารเป็นหลัก

2.1.2 ยึดตามฐานความรู้ (Entity-Level หรือ Knowledge-base) (Aone,C., Gorlinshy, J. and Others, 1999 and Inderjeet Mani and Mark T. Maybury,1998) โครงสร้างหลักของเอกสารที่ได้ จะขึ้นอยู่กับการตีความจากฐานความรู้ที่มี โดยใช้เทคนิคของ การแยกแยะ (Classification) จะเน้นทางด้านใดด้านหนึ่งโดยเฉพาะ อาจมีการสร้างเป็นโครงรูป (Template) ตามประเภทของเอกสาร

2.1.3 ยึดตามความสอดคล้องและความเข้ากันได้ (Discourse-Level หรือ Discourse-base) (Aone,C., Gorlinshy, J. and Others, 1999 and Inderjeet Mani and Mark T. Maybury,1998) วิธีการนี้จะดูความเข้ากันได้และความสอดคล้องกันของประโยค โดยยึดตามโครงสร้างประโยคของภาษานั้นๆ ใช้แก้ปัญหาประโยคที่ได้จาก Frequency-base ที่ไม่สอดคล้องกัน

นักวิจัยจะนำแนวทางทั้ง 3 มาประยุกต์ใช้ในวิธีการสรุปใจความสำคัญ ซึ่งมี 2 วิธี คือ วิธีดึงจากต้นฉบับ (Extract) และ วิธีจัดทำเป็นบทคัดย่อ (Abstract) โดยการสรุปทั้ง 2 ประเภท จะมีลักษณะสำคัญที่แตกต่างกัน

2.2 ประเภทของการสรุปใจความสำคัญ

2.2.1 การสรุปใจความสำคัญแบบดึงจากต้นฉบับ จะแยกคำสำคัญ แยกประโยค และเรียงประโยคเป็นใจความสำคัญ

2.2.2 การสรุปใจความสำคัญแบบจัดทำเป็นบทคัดย่อ เริ่มด้วยการนำเอกสารต้นฉบับไปแปลเพื่อให้ระบบทราบความหมายที่จะนำไปเป็นตัวแทนของเอกสาร จากนั้นตัวแทนของเอกสารจะถูกนำไปใช้แยกข้อมูลที่เป็นประเด็นสำคัญ ซึ่งรวมถึงหัวเรื่องของเอกสารด้วย ข้อมูลเหล่านั้นจะถูกกลั่นกรองและกำจัดข้อมูลที่ซ้ำๆ กันออกไป สุดท้ายจะถูกนำเสนอในรูปแบบใหม่

2.3 ตัวอย่างงานวิจัยที่ผ่านมา

1. งานวิจัยของ Luhn (Luhn,1958), Edmundson(Edmundson,1969) และงานวิจัยของ Rush, Salvador, Zamora (Rush, Salvador and Zamora,1971) ใช้วิธีการนับความถี่ของคำสำคัญ, ระยะห่างระหว่างคำสำคัญในประโยค ซึ่งเป็นแนวทางคิดแบบยึดตามค่าความถี่เป็นหลักสำคัญ
2. KPC (Kupiec, Pederson, and Chen,1995) และ TOPIC (Reimer and Hahn,1988) เป็นแนวคิดแบบยึดตามฐานความรู้
3. งานวิจัยของ Brandow, Mitze, และ Rau (Brandow, Mitze, and Rau ,1995) ใช้วิธีการทดัชนี (indexing) เป็นเป็นแนวคิดแบบยึดตามความสอดคล้องและความเข้ากันได้

งานวิจัยตัวอย่างที่กล่าวมาทั้งหมดนั้นล้วนแล้วแต่เป็นระบบสรุปใจความสำคัญที่ไม่ใช่ภาษาไทย

งานวิจัยเกี่ยวกับการสรุปใจความสำคัญภาษาไทยนั้น ที่ผ่านมามีน้อยมาก เนื่องจากภาษาไทยเป็นภาษาที่มีโครงสร้างซับซ้อนและมีความกำกวม ปัจจุบันยังไม่มีการพัฒนาระบบสรุปใจความสำคัญภาษาไทยอัตโนมัติมาให้ใช้โดยไม่เสียค่าใช้จ่าย หากมีการพัฒนาจริงก็จะเป็นประโยชน์แก่ผู้ใช้อย่างมาก งานวิจัยเกี่ยวกับการสรุปใจความสำคัญภาษาไทยก่อนหน้า เช่น

1. ใช้โครงสร้างปริจเฉทและโครงสร้างลำดับชั้นที่ใช้ในการจำแนกกลุ่มของวัตถุหรือสิ่งที่เรากำลังสนใจ (Ontology) เฉพาะทาง (T. Sukvaree , J. Charoensuk , M. Wattanamethanont)
2. ใช้คำสำคัญบอกความสำคัญทางปริจเฉท (Thana Sukvaree, Asanee Kawtrakul and Jean Caelen)
3. ใช้เทคนิคการคำนวณน้ำหนักของคำสำคัญในเอกสาร (Chuleerat Jaruskulchai and Canasai Kruengkrai)

2.4 เนอีฟเบย์ (Naïve Bayes)

เนอีฟเบย์เป็นโมเดลการจัดกลุ่มที่ใช้หลักความน่าจะเป็น อยู่บนพื้นฐานของข้อพิสูจน์ทางคณิตศาสตร์ของเบย์ (Bayes' Theorem) การจัดกลุ่มได้มาจากการใช้เมธอดที่มีความสัมพันธ์กันอย่างง่ายในการจัดกลุ่มของข้อมูล ซึ่งอาจมีการเกิดของเหตุการณ์ต่างๆที่ใช้ในการจัดกลุ่มมากกว่า

1 ชนิด และเมื่อนำมาประยุกต์ใช้กับข้อพิสูจน์ทางคณิตศาสตร์ของเบย์จะทำให้เกิดการคำนวณที่ซับซ้อน เนื่องจากการขึ้นต่อกันของการเกิดของเหตุการณ์ ดังนั้น โมเดลเนอีฟเบย์จึงตั้งสมมติฐานให้แต่ละเหตุการณ์ที่ใช้ในการจัดกลุ่มนั้นเป็นอิสระต่อกัน (Independence) ซึ่งเป็นที่มาของคำว่านีฟ

โมเดลเนอีฟเบย์มีการนำมาใช้งานเป็นจำนวนมากเพราะมีลักษณะไม่ซับซ้อนและผลลัพธ์จากการคำนวนที่ได้ออกมามีประสิทธิภาพ อย่างไรก็ตาม เรามักพบว่าประสิทธิภาพลดลง เมื่อใช้หาค่าการแจกแจงภายหลัง (Posterior distribution) หลายคนให้ความสนใจกับความจริงที่ว่า เนอีฟเบย์มีแนวโน้มที่จะสร้างการแจกแจง 2 ค่า ด้วยค่ารวมเข้าใกล้ 0 และ เข้าใกล้ 1 อย่างไม่มีหลักเกณฑ์ การคำนวนเหล่านี้จะดีถ้าเนอีฟเบย์สามารถทำงานถูกต้องทุกครั้ง แต่มันก็ไม่สามารถทำได้ มันมีแนวโน้มที่ไม่สามารถวัดค่าประมาณที่เป็นไป

2.4.1 ทฤษฎีพื้นฐานของเนอีฟเบย์

สมมติให้ S เป็นพื้นที่ตัวอย่างที่ถูกแบ่งเป็นพื้นที่เล็กๆ Xi โดยที่ i เป็นตัวเลขอยู่ในSet [1, n] กำหนดให้เหตุการณ์ Y เกิดขึ้น โดยที่ Y เป็นข้อมูลตัวอย่างใน S เรารู้เพียงแค่ว่าข้อมูลตัวอย่างจาก S เป็นหนึ่งในข้อมูลใน Y เราไม่รู้ว่าส่วนใดประกอบด้วยข้อมูลใดบ้าง

ดังนั้น ความน่าจะเป็นของข้อมูลที่ปรากฏใน Xi สามารถหาได้จากสมการต่อไปนี้

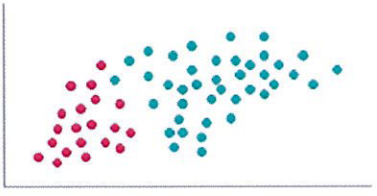
$$P[X_i \mid Y] = \frac{P[Y \mid X_i].P[X_i]}{\sum_j P[Y \mid X_j].P[X_j]} \qquad (2.1)$$

2.4.2 เทคนิคการแยกประเภทของเนอีฟเบย์

เทคนิคของการแยกประเภทของเนอีฟเบย์อยู่บนพื้นฐานของแนวคิดเบย์เซียน (Bayesian) ที่เป็นที่แพร่หลายและเหมาะกับขนาดของข้อมูลที่ป้อนเข้าไปสูง อย่างไรก็ตามในการทำงานอย่างง่ายเนอีฟเบย์มักจะให้ผลดีกว่าวิธีในการแยกประเภทแบบอื่น

นี่คือเงื่อนไขของความน่าจะเป็นของ Xi เมื่อรู้ Y มีค่าเท่ากับค่าของผลทางด้านขวา มันสามารถถูกแทนค่าได้อย่างง่ายดายจากการคำนวนทางสถิติอย่างง่าย โดยใช้สมการความน่าจะเป็น

$$P [A \& B] = P [A|B]*P [B]) \qquad (2.2)$$



รูปที่ 2.1 ข้อมูลสามารถถูกแยกประเภทเป็นสีเขียวและสีแดง

ในการพิสูจน์แนวคิดการแยกประเภทของเนอีฟเบย์ เมื่อพิจารณาตัวอย่างข้างบน ข้อมูลสามารถถูกแยกประเภทเป็นสีเขียวและสีแดง งานของมันคือจัดกลุ่มเมื่อข้อมูลเข้ามา ดูว่า ระดับคลาสใดที่มันควรจะอยู่ โดยดูจากพื้นฐานของข้อมูลที่มีอยู่แล้ว

เมื่อจำนวนข้อมูลสีเขียวมีจำนวนเป็นสองเท่าของสีแดง มันเป็นเหตุผลที่ทำให้เชื่อว่า กรณีใหม่ที่ยังไม่ถูกสังเกต เป็นสองเท่าเมื่อการหาสมาชิกของสีเขียวมากกว่าสีแดง ในการวิเคราะห์ เบย์เซียนจะรู้ในความน่าจะเป็นที่มาก่อน (Prior probability) โดยอาศัยพื้นฐานของประสบการณ์ก่อนหน้านี้ ในเหตุการณ์นี้เปอร์เซ็นต์ของข้อมูลสีเขียวและสีแดง และมักจะใช้ในการคาดคะเนผลลัพธ์ก่อนที่มันจะเกิดขึ้นจริง

ดังนั้น เราจะเขียนได้ว่า

$$\text{Prior probability for Green } \alpha \frac{NumberOfGreenObjects}{TotalNumberOfObjects}$$

$$\text{Prior probability for Red } \alpha \frac{NumberOf\ \text{Re}dObject}{TotalNumberOfObjects} \qquad (2.3)$$

รูปที่ 2.2 แสดงที่รวมจำนวนจุดโดยไม่คำนึงถึงประเภทของสี

การกำหนดค่าความน่าจะเป็นที่สำคัญกว่า เราพร้อมที่จะแยกประเภทข้อมูลใหม่ (White circle) เมื่อข้อมูลถูกรวมเป็นกลุ่มเรียบร้อยแล้ว มันจะสมมติฐานว่า ยิ่งข้อมูลสีเขียว (หรือสีแดง) ในบริเวณใกล้กับ X ข้อมูลใหม่ก็ยิ่งถูกจัดอยู่ในกลุ่มของสีที่มากกว่า การวัดความเป็นไปได้ เราวาดวงกลมรอบ X ที่รวมจำนวนจุดโดยไม่คำนึงถึงประเภทของสี เราจะคำนวนจำนวนของจุดในวงกลมว่าเป็นของสีใด เราใช้สูตรต่อไปนี้

$$\text{Likelihood of X given GREEN } \alpha \frac{NumberOfGreenInTheVicinityOfX}{TotalNumberOfGreenCases}$$

$$\text{Likelihood of X given RED } \alpha \frac{NumberOfREDinTheVicinityOfX}{TotalNumberOfREDCases} \qquad (2.4)$$

จากตัวอย่างข้างบน แสดงว่า ความเป็นไปได้ของ X ให้สีเขียว น้อยกว่า ความเป็นไปได้ของ X ที่ให้สีแดง โดยวงกลมล้อมรอบข้อมูลสีเขียว 1 ตัว และสีแดง 3 ตัว ดังนั้น จะได้ว่า

Probability of X given GREEN α 1/40, Probability of X given RED α 3/20 อย่างไรก็ตาม ความน่าจะเป็นที่สำคัญกว่า ของ X จะเป็นสีเขียว (ให้สีเขียวเป็นสองเท่าของสีแดง) ความเป็นไปได้จะเป็นในทางกลับกัน คือ สมาชิกของคลาส ของ X คือสีแดง (ให้สีแดงมีค่าใกล้แกน X มากกว่าสีเขียว) ในการวิเคราะห์ของเบย์เซียน การแยกประเภทครั้งสุดท้ายจะทำโดยการรวมทั้งสองข้อมูลเข้าด้วยกัน เพื่อสร้างการแจกแจงภายหลัง ใช้กฎของเบย์

Posterior probability of X being GREEN α

Prior probability of GREEN x Likelihood of X given GREEN = $\frac{4}{6} \times \frac{1}{40} = \frac{1}{60}$

Posterior probability of X being RED α

Prior probability of RED x Likelihood of X given RED = $\frac{2}{6} \times \frac{3}{20} = \frac{1}{20}$ (2.5)

สุดท้ายนี้ เราจัดประเภทของ X ว่าเป็นสีแดงโดย สมาชิกของคลาส มันมีค่าความน่าจะเป็นหลัง (Posterior probability) มากที่สุด ตัวอย่างข้างบนไม่เป็นนอลมัลไลซ์ (Normalize) อย่างไรก็ตาม มันไม่มีผลกระทบต่อผลจากการแยกประเภท เมื่อการนอลมัลไลซิ่ง (Normalizing) ที่เกิดขึ้นมีลักษณะเหมือนกัน

2.4.3 ตัวอย่างการแก้ปัญหา

กำหนดให้ P (H) คือความน่าจะเป็นที่จะเกิดเหตุการณ์ H

P (H|E) คือความน่าจะเป็นที่จะเกิดเหตุการณ์ H เมื่อเกิดเหตุการณ์ E จากตัวแปรที่กำหนด

แนวคิดของข้อพิสูจน์ทางคณิตศาสตร์ของเบย์ นั้นสามารถทำนายเหตุการณ์ที่พิจารณาได้จากการเกิดของเหตุการณ์ต่างๆ ได้ดังสมการ

$$P(H|E) = [P(E|H) \times P(H)]/P(E) \quad (2.6)$$

การทำนายว่าฝนจะตกเมื่อมีเหตุการณ์มีเมฆดำ กำหนดให้ H คือเหตุการณ์ที่ฝนตก และ E คือเหตุการณ์มีเมฆดำ แล้วจะสามารถทำนายสภาพจราจรได้ดังสมการต่อไปนี้

$$P(\text{ฝนตก} \mid \text{เมฆดำ}) = [P(\text{เมฆดำ} \mid \text{ฝนตก}) \times P(\text{ฝนตก})]/P(\text{เมฆดำ}) \quad (2.7)$$

กำหนดให้

P (เมฆดำ | ฝนตก) คือความน่าจะเป็นที่มีเมฆดำเมื่อฝนตก ซึ่งในกรณีนี้จะพิจารณาการเกิดเมฆดำเมื่อมีเหตุการณ์ฝนตกเท่านั้น

P (ฝนตก) คือความน่าจะเป็นที่ฝนจะตก ความน่าจะเป็นนี้สามารถเก็บรวบรวมโดยใช้หลักการณ์ทางสถิติ เช่น การบันทึกวันที่มีฝนตกภายใน 1 ปี

P (เมฆดำ) คือความน่าจะเป็นที่มีเมฆดำ เช่นเดียวกันความน่าจะเป็นนี้สามารถเก็บรวบรวมโดยใช้หลักการณ์ทางสถิติ แต่อย่างไรก็ตามการเกิดของเหตุการณ์ต่างๆที่ใช้ในการจัดกลุ่มซึ่งไม่ใช่เหตุการณ์หลักที่เราพิจารณามักจะไม่ถูกบันทึกไว้ และในบางกรณีเหตุการณ์เหล่านี้ยากต่อการบันทึก เช่น P (เมฆดำ) เป็นต้น

จากตัวอย่างที่กล่าวมานั้นสามารถทำนายเหตุการณ์โดยสังเกตุการเกิดของเหตุการณ์บางอย่าง ซึ่งเหตุการณ์ที่นำมาใช้ในการทำนายนั้นต้องสอดคล้องกับเหตุการที่จะทำนาย เช่น ถ้าหากต้องการทำนายการตกของฝน จะไม่ใช้การเกิดแผ่นดินไหวมาพิจารณา เพราะการเกิดแผ่นดินไหวไม่ได้สอดคล้องกับการเกิดฝนตก

สามารถแสดงการคำนวณการจัดกลุ่มของเหตุการณ์ที่มีการเกิดของเหตุการณ์ต่างๆที่ใช้ในการจัดกลุ่มมากกว่า 1 ชนิด ได้ดังสมการต่อไปนี้

$$P(H|E1,E2,...,En) = [P(E1,E2,.....,En|H) \times P(H)]/P(E1,E2,.....,En) \quad (2.8)$$

เมื่อกำหนดให้เหตุการณ์ E1, E2,.....,En คือเหตุการณ์ n เหตุการณ์ที่ใช้ในการจัดกลุ่ม และจากสมมติฐานที่กำหนดให้แต่ละเหตุการณ์ต่างๆที่ใช้ในการจัดกลุ่มเป็นอิสระต่อกันแล้วนั้น จะสามารถแสดงการคำนวณ โดยใช้ข้อพิสูจน์ทางคณิตศาสตร์ของเบย์ ได้ดังสมการต่อไปนี้

$$P(E1,E2,..,En|H) = [P(E1|H) \times P(E2|H) \times ... \times P(En|H) \times P(H)]/P(E1,H) \times P(E2|H) \times \ldots \times P(En|H) \quad (2.9)$$

2.4.4 วิธีการสร้างโมเดลของเนอีฟเบย์

ตารางที่ 2.1 แสดงข้อมูล14 ตัวอย่างที่ประกอบด้วย 5 คุณลักษณะ

ตัวอย่างที่	พยากรณ์	อุณหภูมิ	ความชุ่มชื้น	มีลม	เล่น
1	มีฝนตก	เย็น	ปกติ	ไม่จริง	ใช่
2	มีฝนตก	เย็น	ปกติ	จริง	ไม่ใช่
3	มีเมฆมาก	เย็น	ปกติ	จริง	ใช่
4	มีแดดจัด	เย็น	ปกติ	ไม่จริง	ใช่
5	มีแดดจัด	ร้อน	สูง	ไม่จริง	ไม่ใช่
6	มีแดดจัด	ร้อน	สูง	จริง	ไม่ใช่
7	มีเมฆมาก	ร้อน	สูง	ไม่จริง	ใช่
8	มีเมฆมาก	ร้อน	ปกติ	ไม่จริง	ใช่
9	มีฝนตก	อบอุ่น	สูง	ไม่จริง	ใช่

ตารางที่ 2.1 (ต่อ) แสดงข้อมูล14 ตัวอย่างที่ประกอบด้วย 5 คุณลักษณะ

ตัวอย่างที่	พยากรณ์	อุณหภูมิ	ความชุ่มชื้น	มีลม	เล่น
1	มีฝนตก	เย็น	ปกติ	ไม่จริง	ใช่
2	มีฝนตก	เย็น	ปกติ	จริง	ไม่ใช่
3	มีเมฆมาก	เย็น	ปกติ	จริง	ใช่
4	มีแดดจัด	เย็น	ปกติ	ไม่จริง	ใช่
5	มีแดดจัด	ร้อน	สูง	ไม่จริง	ไม่ใช่
6	มีแดดจัด	ร้อน	สูง	จริง	ไม่ใช่
7	มีเมฆมาก	ร้อน	สูง	ไม่จริง	ใช่
8	มีเมฆมาก	ร้อน	ปกติ	ไม่จริง	ใช่
9	มีฝนตก	อบอุ่น	สูง	ไม่จริง	ใช่
10	มีแดดจัด	อบอุ่น	สูง	ไม่จริง	ไม่ใช่
11	มีแดดจัด	อบอุ่น	ปกติ	ไม่จริง	ใช่
12	มีแดดจัด	อบอุ่น	ปกติ	จริง	ใช่
13	มีเมฆมาก	อบอุ่น	สูง	จริง	ใช่
14	มีฝนตก	อบอุ่น	สูง	จริง	ไม่ใช่

จากตาราง 2.1 สามารถสร้าง โมเดลเนอีฟเบย์ ได้ดังต่อไปนี้

ตารางที่ 2.2 โมเดลของเนอีฟเบย์ที่สร้างจาก 14 ข้อมูลตัวอย่าง

พยากรณ์	ใช่	ไม่	อุณหภูมิ	ใช่	ไม่	ความชื้น	ใช่	ไม่	มีลม	ใช่	ไม่ใช่	เล่น	ใช่	ไม่
มีแดดจัด	2	3	ร้อน	2	2	สูง	3	4	ไม่	6	2		9	5
มีเมฆมาก	4	0	อบอุ่น	4	2	ปกติ	6	1	จริง	3	3			
มีฝนตก	3	2	เย็น	3	1									
มีแดดจัด	2/9	3/5	ร้อน	2/9	2/5	สูง	3/9	4/5	ไม่	6/9	2/5		9/14	5/14
มีเมฆมาก	4/9	0	อบอุ่น	4/9	2/5	ปกติ	6/9	1/2	จริง	3/9	3/5			
มีฝนตก	3/9	2/5	เย็น	3/9	1/5									

เมื่อได้โมเดลมาแล้วนั้นจะสามารถทำนายเหตุการณ์ “เล่น” บนพื้นฐานของการเกิดเหตุการณ์ทั้ง 4 ยกตัวอย่างเช่น เมื่อสังเกตุพบว่า พยากรณ์=มีฝนตก, อุณหภูมิ=เย็น, ความชุ่มชื้น= สูง และ มีลม=ไม่จริง จะสามารถคำนวณหาความน่าจะเป็นที่จะเป็นจริงของเหตุการณ์ “เล่น” (P (เล่น=ใช่|E)) ได้ดังสมการต่อไปนี้

P (เล่น=ใช่|E) =P (พยากรณ์=มีฝนตก|ใช่) x P (อุณหภูมิ=เย็น|ใช่) x P (ความชุ่มชื้น=สูง|ใช่) x P (มีลม=จริง|ใช่) x P (ใช่) x (1/P (E)) (2.10)

แทนค่าตามสมการจะได้

P (เล่น=ใช่|E) = (2/9) x (3/9) x (3/9) x (3/9) x (9/14) x (1/P(E)) (2.11)

และคำนวณหาความน่าจะเป็น P (เล่น=ไม่ใช่|E) ได้ดังสมการต่อไปนี้

P (เล่น=ไม่ใช่|E) = P (พยากรณ์=มีฝนตก|ไม่ใช่) x P (อุณหภูมิ=เย็น|ไม่ใช่) x P (ความชุ่มชื้น=สูง|ไม่ใช่) x P (มีลม=จริง|ไม่ใช่) x P (ไม่ใช่) x (1/P (E)) (2.12)

แทนค่าตามสมการจะได้

P (Play=ไม่ใช่|E) = (3/5) x (1/5) x (4/5) x (3/5) x (5/14) x (1/P (E)) (2.13)

ในขั้นตอนการจัดกลุ่มจัดกลุ่มว่าในสถานการณ์ที่ยกตัวอย่างมานั้นจะพิจารณาว่าค่า P(เล่น=ไม่ใช่|E) และ P(เล่น=ใช่|E) ค่าใดมีค่ามากที่สุดก็จะกำหนดให้เหตุการณ์นั้นอยู่ในกลุ่มนั้น ดังนั้นเมื่อพิจารณาถึงสมการที่ (2.30) และ (2.32) จะสามารถไม่พิจารณาถึงค่า P(E) เนื่องจากเป็นค่าคงที่ ยังผลให้สามารถคำนวณค่าความคล้ายคลึง (Likelihood) ได้ดังต่อไปนี้

Likelihood (ใช่) = (2/9) x (3/9) x (3/9) x (3/9) x (9/14) = 0.0053 (2.14)

Likelihood (ไม่ใช่) = (3/5) x (1/5) x (4/5) x (3/5) x (5/14) = 0.0206 (2.15)

จากค่าความคล้ายคลึงที่แสดงมาในขั้นต้นนั้นเราสามารถสรุปได้ว่าเมื่อเกิดเหตุการณ์ พยากรณ์=มีฝนตก, อุณหภูมิ=เย็น, ความชุ่มชื้น= สูง และ มีลม=ไม่จริง จะจัดกลุ่ม เล่น=ไม่ใช่

เนอีฟเบย์เป็น โมเดลที่ชัดเจนและง่ายต่อการจัดการการเรียนรู้การจัดประเภทของข้อมูล ปัญหาในการทดสอบมักจะพบว่าการทำงานมีประสิทธิภาพน้อยกว่าวิธีอื่น เช่น แม็กซิมัม เอนโทรปี ซับพอท เวกเตอร์ แมชชีน แต่บางครั้งการทำงานของมันมีประสิทธิภาพดีกว่าเทคนิคอื่น

วิธีนี้จะมีการทำงานดีที่สุด เมื่อค่าของคุณลักษณะ แต่ละตัวเป็นอิสระต่อกัน สำหรับปัญหาที่คุณลักษณะ มีความสัมพันธ์ที่ซับซ้อนมาก มักจะทำงานได้ไม่ดีนัก

2.5 เบย์เซียน เน็ตเวิร์ก (Bayesian Network)

เบย์เซียน เน็ตเวิร์ก คือ กราฟฟิกโมเดลทางด้านความน่าจะเป็นที่แทนเซตของตัวแปลและค่าความน่าจะเป็นที่เป็นอิสระต่อกัน เช่น เบย์เซียน เน็ตเวิร์กจะแทนความสัมพันธ์ที่เป็นไปได้ระหว่างโรคภัยไข้เจ็บกับอาการของโรค ถ้ากำหนดให้อาการของโรค เป็นสิ่งที่เน็ตเวิร์ก (Network) สามารถใช้เพื่อคำนวนหาความน่าจะเป็นของโรคต่างๆ ที่สามารถมองเห็นได้

โดยปกติแล้วเบย์เซียน เน็ตเวิร์กเป็นกราฟที่ไม่เป็นวงกลมโดยตรงโดยโหนดจะแทนตัวแปรต่างๆ และขอบที่ไม่ปรากฏ (Missing edge) จะแทนเงื่อนไขที่เป็นอิสระต่อกันระหว่างตัวแปร โหนดต่างๆ สามารถแทนตัวแปรแต่ละประเภทได้ ไม่ว่าจะเป็น ตัวแปรในการวัดค่า ตัวแปรแฝง (Latent variable) และ ข้อสันนิษฐานต่างๆ จะไม่จำกัดการแทนค่าของตัวแปรสุ่มที่แทนลักษณะอื่นของเบย์เซียน เน็ตเวิร์ก ประสิทธิภาพของอัลกอลิทึมที่มีจะแสดงการอ้างอิงและการเรียนรู้ของเบย์เซียน เน็ตเวิร์ก ดังนั้น โมเดลเบย์เซียน เน็ตเวิร์ก ที่มีความต่อเนื่องของตัวแปรเรียกว่า ไดนามิก เบย์เซียน เน็ตเวิร์ก (Dynamic Bayesian Networks) สรุปได้ว่า เบย์เซียน เน็ตเวิร์กสามารถแทนและแก้ปัญหาการตัดสินใจภายใต้เงื่อนไขที่ไม่แน่นอน (Influence Diagram)

2.5.1 ความหมายและหลักการทำงาน

ถ้ามีขอบ (Direct Edge) หรือลูกศร ลากจากโหนด A ไปยังโหนด B โหนด A จะถูกเรียกว่าเป็น พ่อแม่ของโหนด B และ B ถือเป็นลูกของโหนด A เซตของโหนดพ่อแม่ของโหนด X_i ใดๆ แสดงด้วย พ่อแม่ของ (X_i) ส่วนกราฟเส้นตรงแบบไม่เป็นวงกลมเป็น เบย์เซียน เน็ตเวิร์กที่สัมพันธ์กับเซตของตัวแปร ถ้าการกระจายแบบร่วมกันของค่าโหนดต่างๆ ถูกเขียนว่าเป็นผลลัพธ์ของการกระจายขั้นพื้นฐานของแต่ละโหนดและโหนดพ่อแม่ของมัน จะได้ดังสมการ

$$P(X_1,...,X_n) = \prod_{i=1}^{n} P(X_i \mid parents(X_i)) \qquad (2.16)$$

ถ้าโหนด X_i ไม่มีโหนดพ่อแม่ การกระจายความน่าจะเป็นขั้นพื้นฐานจะเป็นแบบไม่มีเงื่อนไข มิฉะนั้นแล้วมันคือเงื่อนไข ถ้าค่าของโหนดถูกเก็บไว้แล้วโหนดนี้จะถือว่าเป็นโหนดที่ถูกพิสูจน์แล้ว (Evidence Node)

2.5.2 ความเป็นอิสระต่อกันและการแยกส่วน (D-Separation)

กราฟแสดงความเป็นอิสระต่อกันของตัวแปรต่างๆ เงื่อนไขที่เป็นอิสระต่อกันสามารถถูกสร้างโดยคุณสมบัติของการแยกส่วน (D-Separation) ถ้าเซต 2 เซตของโหนดที่ D-Separation โหนด X จากโหนดอื่นทุกๆ ตัวถูกสร้างโดย X ของ Markov blanket

ส่วน p โดยไม่เป็นส่วนโดยตรง เรียกว่า การแยกส่วน โดยเซตของโหนด Z ถ้าตรงตามเงื่อนไขต่อไปนี้อย่างน้อย 1 เงื่อนไข

1. p ประกอบด้วย ความสัมพันธ์ i -> m -> j ทำให้โหนด m ตรงกลางนั้นอยู่ใน Z
2. p ประกอบด้วย ความสัมพันธ์ i <- m <- j ทำให้โหนด m ตรงกลางนั้นอยู่ใน Z
3. p ประกอบด้วย fork i <- m <- j ทำให้โหนด m ตรงกลางนั้นอยู่ใน Z
4. p ประกอบด้วย inverted fork หรือ collider i -> m <- j ทำให้โหนด m ตรงกลางไม่ได้อยู่ใน Z และไม่ได้สืบทอด m มาจาก Z

เซตของ Z เรียกว่า การแยกส่วน X จาก y ในกราฟเส้นตรงที่ไม่เป็นวงกลมของ G ถ้าทุกๆ ทางจาก x ไปหา y ในกราฟ G เป็นการแยกส่วน โดย Z ค่า d ใน การแยกส่วน จะเป็นแบบโดยตรงเมื่อคุณสมบัติของสามโหนดเชื่อมต่อทางนั้นขึ้นอยู่กับการเชื่อมต่อของลูกศรในวงนั้นๆ

โหนด 2 โหนดเป็นอิสระต่อกัน (ไม่มีเงื่อนไข) ถ้าโหนดทั้ง 2 โหนดไม่มีการสืบทอดจากบรรพบุรุษ เมื่อมันมีค่าเท่ากันจึงบอกได้ว่าระยะทางระหว่างโหนดเหล่านี้ประกอบไปด้วยอย่างน้อยโหนด 1 โหนดที่มีความขัดแย้งกัน ซึ่งมีค่าเท่ากับการกล่าวได้ว่าโหนดทั้ง 2 โหนดเป็นการแยกส่วนโดยเซตว่าง

เบย์เซียน เน็ตเวิร์ก จากความหมายข้างต้น สามารถบอกได้ว่าใช้อธิบายความไม่ขึ้นต่อกันอย่างมีเงื่อนไขระหว่างตัวแปร ทำให้เราสามารถใช้ความรู้ก่อนหน้าเกี่ยวกับความไม่ขึ้นต่อกันระหว่างตัวแปร ร่วมกับตัวอย่าง หรือที่เรียกกันว่า เบย์เซียน เน็ตเวิร์ก

2.5.3 เงื่อนไขที่ไม่ขึ้นต่อกัน (Conditional Independence)

เมื่อให้ X เป็นเงื่อนไขที่เป็นอิสระไม่ขึ้นต่อกันของ Y ที่ถูกกำหนดโดย Z ถ้าการควบคุมความน่าจะเป็นของการกระจาย X นั้นเป็นอิสระต่อกันกับค่าของ Y ที่ถูกกำหนดโดย ค่าของ Z จะเป็นดังสมการ

$$(\forall x_i, y_j, z_k) P(X = x_i | Y = y_j, Z = z_k) = P(X = x_i | Z = z_k) \quad (2.17)$$

หรือ

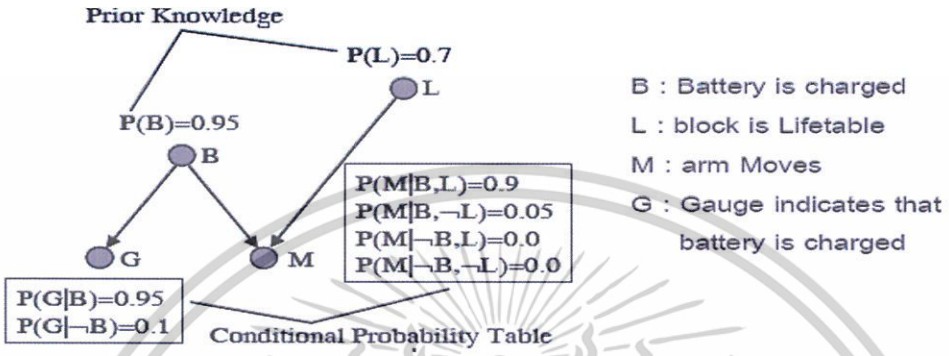
$$P(X \mid Y_i Z) = P(X \mid Z)_b \quad (2.18)$$

ตัวอย่าง

ฟ้าผ่าเป็นเงื่อนไขที่เป็นอิสระกับการเกิดฝนตกซึ่งถูกกำหนดโดยแสง จะได้สมการ ดังนี้

$$P(\text{Thunder} \mid \text{Rain}, \text{Lightning}) = P(\text{Thunder} \mid \text{Lightning}) \quad (2.19)$$

เบย์เซียน เน็ตเวิร์ก แสดงเซตของความไม่ขึ้นต่อกันของตัวแปร แสดงโดยกราฟเส้นตรงที่เป็นไม่เป็นวงกลม หรือ (DAG; Directed Acyclic Graph) โดยแต่ละโหนดจะไม่ขึ้นต่อกันอย่างมีเงื่อนไข กับโหนดอื่นๆ เมื่อรู้โหนดพ่อแม่โดยตรง (Immediate Predecessors)



รูปที่ 2.3 การแสดงการกระจายความน่าจะเป็นร่วม (Join Probability Distribution)

ตัวอย่างเช่น P (Battery, Liftable, Gauge, Move)

$$P(y_1,\ldots,y_n) = \prod_{i=1}^{n} P(y_i \mid parents(y_i)) \qquad (2.20)$$

โดยที่ Parents(Y_i) คือโหนดพ่อแม่โดยตรงของโหนด Y_i การกระจายความน่าจะเป็นร่วมนิยามโดยกราฟและ P(y_i| Parents(Y_i))

จากตัวอย่าง

$$P(G,M,B,L) = P(G \mid B,M,L)\, P(M \mid B,L)\, P(B \mid L)\, P(L) = P(G \mid B)\, P(M \mid B,L)\, P(B)\, P(L) \qquad (2.21)$$

2.5.4 รูปแบบการอนุมานในเบย์เซียน เน็ตเวิร์ก

1. การอนุมานจากเหตุ (Causal Reasoning) เช่น ต้องการคำนวณ P (M|L) คือ หาความน่าจะเป็นที่แขนจะเคลื่อนได้เมื่อกล่องยกได้ (กล่องยกได้เป็นสาเหตุหนึ่งของการที่แขนจะเคลื่อนที่ได้) กระจาย P (M|L) ให้อยู่ในรูปของผลรวมของความน่าจะเป็นร่วมระหว่าง M กับโหนดพ่อแม่อื่นนอกจาก L (M กับ B)

$$P(M|L) = P(M|B,L)\, P(M,-B) \qquad (2.22)$$

จัดรูปให้ M ขึ้นกับโหนดพ่อแม่ (B, L) โดยใช้ chain rule

$$P(M \mid L) = P(M \mid B,L) + P(M \mid \neg B,L)P(\neg B \mid L)$$
$$P(M \mid L) = P(M,B,L) + P(M \mid \neg B,L)P(\neg B)$$
$$= 0.855 \qquad (2.23)$$

2. การอนุมานจากผล (Diagnosis Reasoning) เช่น ต้องการคำนวณ P (-L|-M) ความน่าจะเป็นที่กล่องยกไม่ได้เมื่อรู้ว่าแขนไม่ได้เคลื่อน นั่นคือการใช้ผลเพื่อหาสาเหตุ

$$P(\neg L \mid \neg M) = \frac{P(\neg M \mid L)P(\neg L)}{P(\neg M)} \tag{2.24}$$

คำนวน P(-M|-L) ได้ 0.9525

$$P(\neg L \mid \neg M) = \frac{0.9525 \times 0.3}{P(\neg M)} = \frac{0.28575}{P(\neg M)} \tag{2.25}$$

จะได้

$$P(\neg L \mid \neg M) + P(L \mid \neg M) = 1 \Rightarrow P(\neg L \mid \neg M) = 0.7379 \tag{2.26}$$

3. การทำเหตุผลที่เป็นเหตุเป็นผลกันเหตุผลที่ใช้ในการพิจารณา (Explaining way) เช่น เมื่อรู้ –M (แขนไม่เคลื่อน) สามารถคำนวณ –L ความน่าจะเป็นที่กล่องไม่สามารถยกได้ แต่ถ้าเรารู้ –B (แบตเตอรี่ไม่ได้ชาร์จ) แล้ว –L ควรจะมีความน่าจะเป็นลดลง
แต่ในกรณีนี้ xplain $\neg M$, making $\neg L$ less certain

$$P(\neg L \mid \neg B, \neg M) = \frac{P(\neg M, \neg B \mid \neg L)P(\neg L)}{P(\neg B, \neg M)} \text{ (Bayes' rule)}$$

$$= \frac{P(\neg M \mid \neg B, \neg L)P(\neg B \mid \neg L)P(\neg L)}{P(\neg B, \neg M)}$$

$$= \frac{P(\neg M \mid \neg B, \neg L)P(\neg B)P(\neg L)}{P(\neg B, \neg M)} \tag{2.27}$$

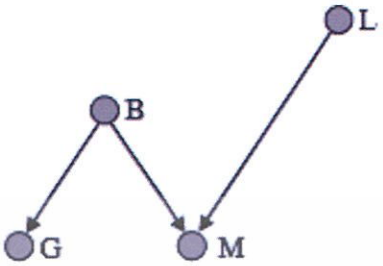
หลังคำนวณ $P(\neg B, \neg M)$ เราได้ $P(\neg L \mid \neg B, \neg M) = 0.30$ (2.28)

การเรียนรู้เบย์เซียน เน็ตเวิร์ก คือการหาเน็ตเวิร์ก (โครงสร้าง และ ตารางเงื่อนไขความน่าจะเป็น) ที่สอดคล้องกับตัวอย่างมากที่สุด ปัญหาการเรียนรู้แบ่งเป็น

1. โครงสร้างที่ไม่รู้จัก
2. โครงสร้างที่รู้จัก แบ่งเป็น ข้อมูลที่ไม่สูญหาย และ ข้อมูลที่สูญหาย

กรณีของโครงสร้างที่ไม่รู้จักและข้อมูลที่ไม่สูญหาย เป็นกรณีที่ง่ายที่สุด ที่สามารถทำการเรียนรู้ได้ในลักษณะเดียวกับการแยกประเภทของเบย์เซียน เน็ตเวิร์ก โครงสร้างการเรียนรู้ด้วยค่าข้อมูลที่ไม่สูญหาย จะใช้การคำนวน CPT สำหรับแต่ละโหนด โดยต้องใช้ตัวอย่างจำนวนมากเพื่อ

ให้ค่าทางสถิติถูกต้อง ดังตัวอย่าง



รูปที่ 2.4 โครงสร้าง CPT

$$\hat{P}(V_i = v_i | \text{Parents}(V_i) = p_i) = \frac{\text{จำนวนตัวอย่างที่มี } V_i = v_i}{\text{จำนวนตัวอย่างที่มี Parents}(V_i) = p_i} \quad (2.29)$$

$$P(B=\text{True}) = (54+1+7+27+3+2)/100 = 0.94 \quad (2.30)$$

$$P(L=\text{True}) = (54+7+3+4)/100 = 0.68 \quad (2.31)$$

$$P(M|B,-L) = \text{อัตราส่วนที่ } M=\text{True} \text{ เมื่อ } B=\text{True}, L=\text{False} = 1/(1+27+2) = 0.03 \quad (2.32)$$

ตารางที่ 2.3 การคำนวณ CPT สำหรับโหนด G ได้ในทำนองเดียวกัน

G	M	B	L	No. of instance
True	True	True	True	54
True	True	True	False	1
True	False	True	True	7
True	False	True	False	21
False	True	True	True	3
False	False	True	False	2
False	False	False	True	4
False	False	False	False	2
				100

ตารางที่ 2.4 ตัวอย่างของข้อมูลต่อไปนี้ โดยที่ * แทนค่าที่หายไป

G	M	B	L	No. of instance
True	True	True	True	54
True	True	True	False	1
*	*	True	True	7
True	False	True	False	21
False	True	*	True	3
False	False	True	False	2
False	False	False	True	4
False	False	False	False	2

เมื่อพิจารณากรณีของ 3 ตัวอย่างที่มีค่า G = False, M = True, L = True ในกรณีที่เราไม่รู้ค่าของ B แต่อาจคำนวณ P(B|-G,M,L) หรือ P(-B|-G,M,L) ได้ ถ้าเรารู้ CPT แต่ขณะนี้ยังไม่รู้ จากนั้นจะแทนที่ตัวอย่างทั้งสามด้วยตัวอย่างมีน้ำหนัก (weight example) 2 ตัว

ตัวแรก คือ ตัวอย่างที่ B = True มีน้ำหนัก P (B|-G, M, L)

ตัวที่สอง คือ ตัวอย่างที่ B = False มีน้ำหนัก P (-B|-G, M, L)

ในทำนองเดียวกัน กรณีของ 7 ตัวอย่างที่มีค่า B = True, L = True และ G, M ไม่รู้ค่านั้น สามารถแทนที่ตัวอย่างทั้งเจ็ดด้วยตัวอย่างมีน้ำหนัก 4 ตัว ดังนี้

ตัวอย่าง G = True, M = True มีน้ำหนัก P (G, M|B, L)

ตัวอย่าง G = True, M = False มีน้ำหนัก P (G,-M|B, L)

ตัวอย่าง G = False, M = True มีน้ำหนัก P (-G, M|B, L)

ตัวอย่าง G = False, M = False มีน้ำหนัก P (-G,-M|B, L)

สามารถค่าความน่าจะเป็นเหล่านี้ได้ถ้ารู้ CPT และ โครงสร้างของเบย์เซียน เน็ตเวิร์ก จากนั้นจะใช้ตัวอย่างที่มีน้ำหนักเหล่านี้ ร่วมกับตัวอย่างที่เหลือเพื่อคำนวณ CPT ได้ ตัวอย่างที่ไม่รู้ค่าถูกแทนด้วยตัวอย่างที่มีน้ำหนัก

2.6 แม็กซิมัม เอนโทรปี (Maximum Entropy)

เป็นเทคนิคการคำนวนหาค่าความน่าจะเป็นที่ใช้อย่างกว้างขวางในงานประเภทภาษาธรรมชาติ (Natural Language) เช่น โมเดลของภาษาธรรมชาติ, สำนวนหรือประโยค และการแบ่งบทความเป็นส่วนๆ จุดเด่นสำคัญของแม็กซิมัม เอนโทรปี คือ ไม่อาศัยความรู้จากภายนอก แต่ให้ความสำคัญกับชุดข้อมูล (Data Set) ที่มีอยู่

แนวความคิดที่ก่อให้เกิดแม็กซิมัม เอนโทรปี คือการอ้างอิงถึง โมเดลที่มีรูปแบบ (Uniform Model) ที่ตอบสนองเงื่อนไขต่างๆที่มี เช่น เมื่อมองไปที่การจัดกลุ่มบทความแบบ 4 ทาง ซึ่งค่าเฉลี่ย 40% ของบทความคือคำ กำหนดให้คำว่า “อาจารย์” (Professor) จัดอยู่ในคลาส “คณะ” (Faculty Class) และเมื่อใดที่มีคำว่า “อาจารย์” ปรากฏขึ้น จะบอกได้ว่ามีโอกาสเพียงแค่ 40% ที่จะอยู่ในเอกสารคณะ และมีโอกาส 20% สำหรับคลาสอื่นอีก 3 คลาส ถ้าในบทความไม่มีคำว่า“อาจารย์” คาดว่าจะแบ่งให้แต่ละคลาสที่มีรูปแบบ คลาสละ 25% โมเดลนี้จะคล้ายกับเงื่อนไขที่รู้กันดี การคำนวนในตัวอย่างเป็นโมเดลที่ง่าย แต่เมื่อมีหลายๆเงื่อนไขมาให้อธิบาย จะทำให้ยากมากขึ้น

2.6.1 เงื่อนไขและรูปแบบ

ในโมเดลนี้เราใช้ข้อมูลสำหรับการเรียนรู้ในการจัดกลุ่มเงื่อนไข เพื่อใช้เป็นเงื่อนไขในการจัดแบ่งกลุ่ม แต่ละเงื่อนไขจะส่งคุณสมบัติของข้อมูลสำหรับการเรียนรู้ ที่ควรถูกแสดงไปยังระบบการจัดกลุ่มให้เรียนรู้ เราให้ค่าฟังค์ชันจริงของเอกสารและคลาสเป็นรูปแบบ $f_i(d,c)$ โมเดลนี้ให้เราจำกัดการกระจายของโมเดลเพื่อให้มีค่าคาดหวังเท่ากัน สำหรับรูปแบบนี้จะเห็นในข้อมูลสำหรับการเรียนรู้ D ดังนั้นเรากำหนดเงื่อนไขที่ระบบกระจายเรียนรู้เงื่อนไข P(c|d) ควรมีคุณสมบัติดังนี้

$$\frac{1}{|D|}\sum_{d\in D} f_i(d,c,(d)) = \sum_d P(d)\sum_c P(c\mid d) f_i(d,c) \tag{2.33}$$

ในทางปฏิบัติ ระบบกระจายเอกสาร P(d) เป็นค่าที่ไม่รู้จัก และไม่ให้ความสนใจในการสร้างโมเดลนั้น ดังนั้น ใช้ข้อมูลสำหรับการเรียนรู้(Training data) ไม่ใช้ระดับของคลาสเหมือนกับการประมาณไปยังระบบกระจายเอกสาร ให้ทำตามเงื่อนไขต่อไปนี้

$$\frac{1}{|D|}\sum_{d\in D} f_i(d,c,(d)) = \frac{1}{|D|}\sum_{d\in D}\sum_c P(c\mid d)P(c\mid d) f_i(d,c) \tag{2.34}$$

ดังนั้นเมื่อใช้แม็กซิมัม เอนโทรปี ขั้นแรกคือ สร้างเซตของรูปแบบของฟังค์ชันที่จะเป็นประโยชน์ในการจัดกลุ่ม หลังจากนั้นแต่ละรูปแบบ ทำการวัดค่าคาดหวังผ่านทางข้อมูลสำหรับการเรียนรู้ และใช้มันเป็นเงื่อนไขในโมเดลการกระจาย

เมื่อเงื่อนไขได้ถูกกำหนดมาแล้ว มันรับประกันว่าระบบกระจายที่มีอยู่มีแม็กซิมัม เอนโทรปี ยิ่งกว่านั้น ระบบกระจายจะถูกแสดงออกมาในรูปแบบเอกโพเนนเชี่ยล (Exponential form) ดังนี้

$$P(c \mid d) = \frac{1}{|z(d)|} \exp(\sum_i \lambda_i f_i(d \mid c) \tag{2.35}$$

แต่ละ $fi(d,c)$ เป็นรูปแบบ λ_i เป็นค่าตัวแปรที่ถูกประมาณไว้และ Z(d) เป็นปัจจัยในการนอลมัลไลซิ่ง normalizing เพื่อให้ค่าความน่าจะเป็นที่เหมาะสม

$$Z(d) = \sum \exp(\sum_i \lambda_i f_i(d \mid c) \tag{2.36}$$

เมื่อตัวแปรถูกประมาณค่าจากระดับข้อมูลสำหรับการเรียนรู้ วิธีในการแก้ปัญหาแม็กซิมัม เอนโทรปี คือ การแก้ปัญหาจากการปัญหาความเป็นไปได้ที่จะเกิดอย่างมากที่สุดสองทางของโมเดลของรูปแบบเอกโพเนนเชี่ยล

2.6.2 แม็กซิมัม เอนโทรปี (Maximum Entropy) สำหรับการจัดกลุ่มของบทความ

ในการแปลงแม็กซิมัม เอนโทรปี ไปเป็นโดเมน จำเป็นต้องเลือกเซตของรูปแบบเพื่อใช้ในการตั้งเงื่อนไข การจัดประเภทของบทความด้วยแม็กซิมัม เอนโทรปี ใช้การนับคำต่างๆ ส่วนประกอบของแต่ละคลาสของคำ ยกตัวอย่างได้ดังนี้

$$f_{w,c'}(d,c) = \begin{cases} 0 & \text{if } c \neq c' \\ \frac{N(d,w)}{N(d)} & \text{Otherwise,} \end{cases} \tag{2.37}$$

เมื่อ N(d,w) เป็นจำนวนครั้งของคำ w ที่มีอยู่ในเอกสาร d, N(d) เป็นจำนวนคำใน d

ด้วยการแทนที่นี้ ถ้าคำปรากฏบ่อยในคลาสหนึ่ง จะคาดได้ว่าน้ำหนักของคำของคลาสนั้นจะมีค่าสูงกว่าคำที่อยู่ในคลาสอื่นๆ ในรูปแบบงานของภาษาธรรมชาติ โดยใช้แม็กซิมัม เอนโทรปี เป็นรูปแบบไบนารี่ (binary) ในการจัดกลุ่มของคำ คาดว่ารูปแบบของจำนวนครั้งที่คำหนึ่งๆ จะปรับปรุงการจัดประเภทของคำ เช่น การใช้เนอีฟเบย์ ในการนับจำนวนในการทำงานให้ผลดีกว่าการที่ไม่ใช้ จำไว้ว่า ใช้การนับระดับเป็นรูปแบบแทนการนับทั่วๆไป เลือกค่าเริ่มต้นของการแทนที่สำหรับประสิทธิภาพของการคำนวน โดยให้มันแสดงการทำซ้ำของแต่ละรูปแบบที่ใกล้เคียง

2.7 ซับพอท เวกเตอร์ แมชชีน (Support Vector Machine)

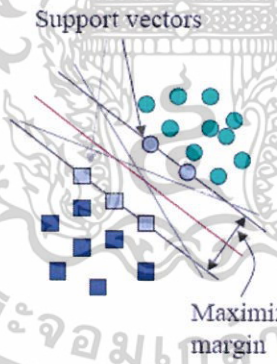
ซับพอท เวกเตอร์ แมชชีน เป็นโมเดลที่มีความเกี่ยวข้องกับ เครือข่ายประสาทเทียม การทำงานของซับพอท เวกเตอร์ แมชชีนจะทำการค้นหาข้อมูลและเปรียบเทียบข้อมูลที่มีรูปแบบข้อมูลที่ซับซ้อน แล้วทำการสร้างเซตของข้อมูล โดยการเปรียบเทียบ หารูปแบบความสัมพันธ์ของข้อมูล จัดแบ่งประเภท ตำแหน่ง และกลุ่มข้อมูล รวมทั้งกำหนดทิศทางความสัมพันธ์ของข้อมูลให้มีความเหมาะสม

ลักษณะของข้อมูลที่ใช้ในจัดประเภท จะเป็นข้อมูลที่มีการถ่ายทอดกันเป็นลำดับชั้น ข้อมูลที่มีความต่อเนื่องกัน ข้อมูลหลากหลายชนิดที่มีการถ่ายทอดกันเป็นลำดับชั้น ข้อมูลเอกสารต่างๆ

2.7.1 แนวความคิดการทำงานของซับพอท เวกเตอร์ แมชชีน

ให้ทฤษฎีทางคณิตศาสตร์ เพื่อกำหนดจุดประสงค์ในการให้เครื่องเรียนรู้ จากนั้น กำหนดฟังก์ชั่นหลัก (Kernel Function) เพื่อใช้ในการแบ่งแยกข้อมูลที่ไม่สามารถจัดกลุ่มด้วยการทำงานแบบเส้นตรงให้สามารถจัดกลุ่มข้อมูลให้มีผลลัพธ์ที่มีประสิทธิภาพมากขึ้น

2.7.2 องค์ประกอบของการทำงานแบบซับพอทเวกเตอร์แมชชีน



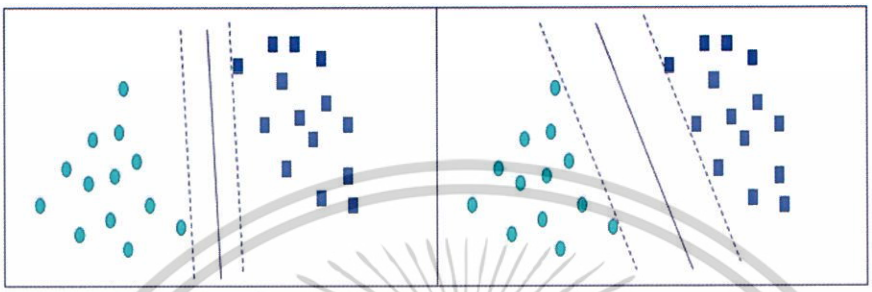
รูปที่ 2.5 แสดงองค์ประกอบของของซับพอท เวกเตอร์ แมชชีน

จากรูปที่ 2.5	Vector	คือ เซตของลักษณะข้อมูลต่างๆ ที่จะจัดแบ่งประเภท
	Support vector	คือ เวกเตอร์ที่จำกัดความกว้างของขอบเขต
	Margin	คือ ระยะห่างระหว่าง support vector ของข้อมูลทั้งสองประเภท

2.7.3 การใช้ฟังก์ชั่นหลัก (Kernel Function) แบบเส้นตรง (Linear Function)

สมมติว่าต้องการที่จะแบ่งกลุ่มข้อมูล โดยข้อมูลเหล่านั้นมีตัวแปรเป้าหมายที่แน่นอน ซึ่งแบ่งออกเป็นสองประเภท จึงกำหนดตัวแปรที่ต้องการจะหา 2 ตัวแปรด้วย ค่าที่ต่อเนื่อง ถ้าวาง

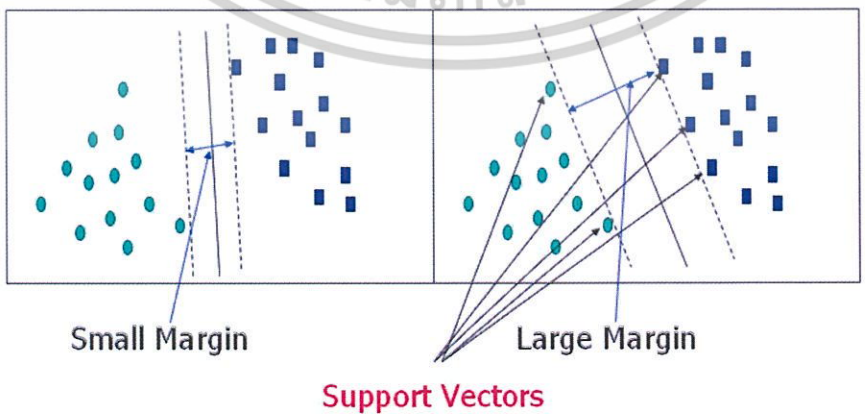
แผนจุดของข้อมูลโดยใช้ค่าของหนึ่งในค่าที่ต้องการทำนายอยู่บนแกน X และอีกค่าอยู่บนแกน y ผลลัพธ์ที่ได้ควรสิ้นสุดด้วยภาพที่แสดงดังข้างล่างนี้ วัตถุประเภทหนึ่งของตัวแปรถูกแทนโดยสี่เหลี่ยม อีกประเภทถูกแทนโดยวงกลม



รูปที่ 2.6 แสดงถึงการใช้ฟังก์ชั่นหลัก (Kernel Function) แบบเส้นตรง (Linear Function) ในซับพอท เวกเตอร์ แมชชีน

จากรูปที่ 2.6 สถานการณ์ที่เซตของข้อมูลประเภทหนึ่งอยู่ในฝั่งซ้ายล่างและอีกกลุ่มของข้อมูลเป็นอีกประเภทที่อยู่บนฝั่งขวาด้านบน ปัญหาด้านบนได้ถูกแบ่งแยกเรียบร้อยแล้ว การวิเคราะห์ ซับพอท เวกเตอร์ แมชชีนจะพยายามที่จะหาเส้นแบ่งคลาสระนาบเกิน (Hyperplane) 1 มิติที่สามารถแบ่งแยกข้อมูลโดยอิงตามพื้นฐานของเป้าหมายของประเภทข้อมูล มันมีตัวเลขที่มีค่าไม่สิ้นสุด ของเส้นที่เป็นไปได้ เส้นร่วม (Candidate Line) 2 เส้นแสดงอยู่ด้านบน คำถามคือ เส้นไหนที่มีค่าดีกว่ากันและ จะทำอย่างไรในการสร้างเส้นที่ดีที่สุด

เส้นประที่วาดเป็นเส้นขนานกันเป็นสัญลักษณ์ในการแบ่งระยะทางระหว่างเส้นสำหรับการแบ่งแยกกับเวกเตอร์ที่ใกล้กับเส้นที่สุด ระยะทางระหว่างเส้นที่เป็นเส้นประทั้งสองเส้นเรียกว่าขอบเขตสำหรับเวกเตอร์ ที่จำกัดความกว้างของขอบเขตคือซับพอทเวกเตอร์ตามที่แสดงดังรูปนี้



รูปที่ 2.7 แสดงขอบเขตระหว่างซับพอตเวกเตอร์ที่แคบที่สุดและกว้างที่สุด

การวิเคราะห์ซับพอท เวกเตอร์ แมชชีน พบเส้นที่กำหนดตำแหน่งที่ค่าขอบเขตระหว่างซับพอตเวกเตอร์ที่แคบสุดและกว้างที่สุด

สามารถแบ่งแยกเส้นแบ่งคลาส โดยใช้สมการเส้นตรง

$$ax + by = c \tag{2.38}$$

เวกเตอร์นำเข้าที่ใช้ในการหาค่าผลลัพธ์ มีได้ 2 ค่า คือ

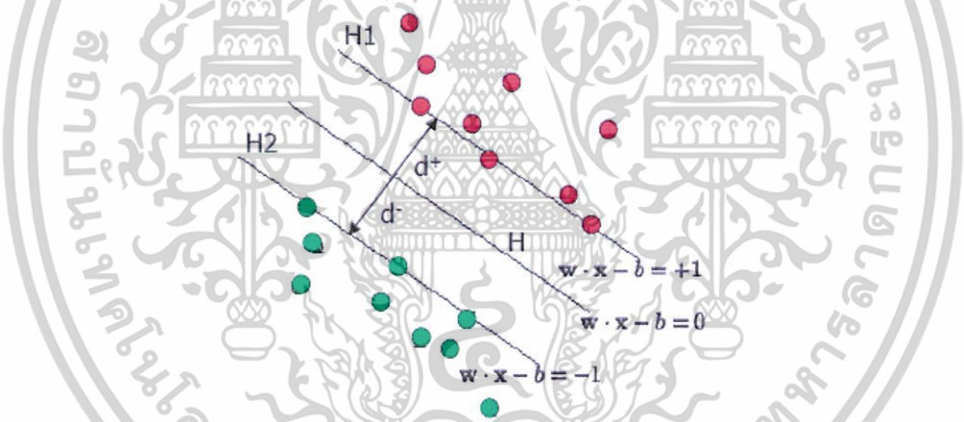
$$W_0 T_X + b_0 = 1 \tag{2.39}$$

$$W_0 T_X + b_0 = -1 \tag{2.40}$$

ทำการสร้างเส้นแบ่งคลาส H ให้มีค่าต่างๆ ดังนี้

$$X_i \cdot W + b \geq +1 \quad \text{เมื่อ } y_i = +1$$

$$X_i \cdot W + b \leq -1 \quad \text{เมื่อ } y_i = -1 \tag{2.41}$$



รูปที่ 2.8 แสดงระนาบที่เกิดจากการใช้สมการเส้นตรง

จากรูป H1 และ H2 เป็นระนาบ (plane) ที่เป็นซับพอทเวกเตอร์ (support vector)

$$H1: X_i \cdot W + b = +1 \tag{2.42}$$

$$H2: X_i \cdot W + b = -1 \tag{2.43}$$

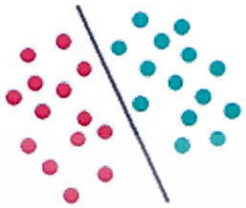
d+ คือ ระยะทางที่สั้นที่สุดที่จะใกล้กับค่าที่เป็นค่าบวก

d- คือ ระยะทางที่ใกล้ที่สุดที่จะใกล้กับค่าที่เป็นลบ

Margin ของการแบ่งเส้นของ Hyperplane คือ $d^+ + d^-$

การเลื่อนซับพอทเวกเตอร์ ไม่มีผลใดๆ ต่อการตัดสินใจแบ่งขอบเขตของข้อมูล ซับพอทเวกเตอร์ แมชชีน จะพยายามทำให้ค่าขอบเขตที่อยู่รอบๆเส้นเส้นแบ่งคลาส มีค่าสูงสุดเท่าที่จะเป็นไปได้

2.7.4 การใช้ฟังก์ชั่น แบบไม่เป็นเส้นตรง



รูปที่ 2.9 การแบ่งข้อมูลโดยใช้เส้นตรง

รูปที่ 2.10 การแบ่งข้อมูลโดยไม่ใช้เส้นตรง

จากรูปที่ 2.9 เป็นตัวอย่างการแบ่งประเภทของข้อมูลด้วยเส้นตรง การแบ่งแยกประเภทต่างๆ ของข้อมูลนั้น โครงสร้างจะมีความซับซ้อนมากกว่าและจำเป็นต้องใช้วิธีการแบ่งประเภทข้อมูลที่ดีที่สุดเพื่อให้ผลลัพธ์จากการจัดประเภทของข้อมูลมีความถูกต้องมากที่สุด

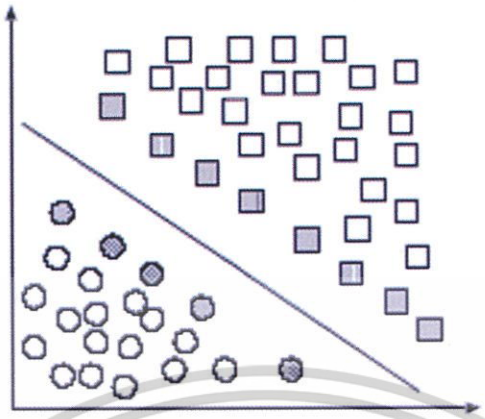
จากรูปที่ 2.10 เมื่อเปรียบเทียบกับรูปที่ 2.9 แสดงให้เห็นชัดเจนว่ามีการแบ่งแยกระหว่างข้อมูลสีแดงและสีเขียวอย่างชัดเจน โดยการใช้เส้นโค้ง การจัดประเภทของงานขึ้นอยู่กับการวาดภาพเส้นแบ่งกลุ่มที่จะแบ่งแยกความแตกต่างระหว่างข้อมูลทั้งสองประเภทให้อยู่ในคลาสที่มีความแตกต่างกันอย่างที่ รู้จักกันในชื่อ การจัดกลุ่มเส้นแบ่งคลาส ซับพอท เวกเตอร์ แมชชีนเป็นการจัดการงานให้มีความเหมาะสม

2.7.5 ฟังก์ชั่น ชนิดอื่นๆ

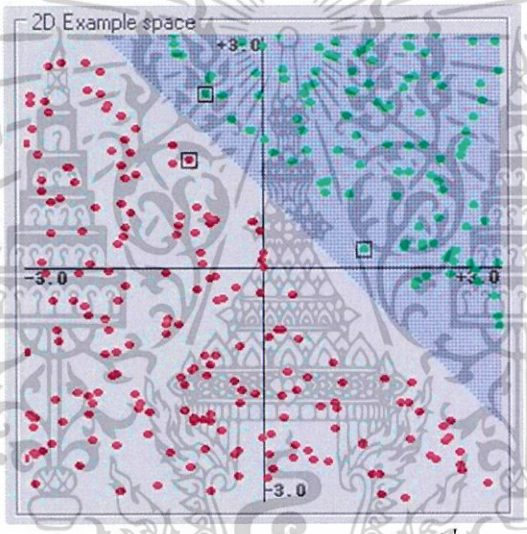
มีฟังก์ชั่นหลัก (Kernel Function) หลายประเภทที่สามารถใช้ในโมเดลซับพอท เวกเตอร์ แมชชีน ตัวอย่างมีดังนี้

$$\phi = \begin{cases} x_i * x_j \\ (\gamma x_i x_j + coefficient) \\ \exp(-\gamma \mid x_i - x_l \mid^2) \\ \tanh(\gamma x_i x_l Z + coefficient) \end{cases}$$

2.7.5.1 RBF เป็นสมการเกี่ยวกับกราฟพาราโบลา ซึ่งนิยมใช้มากชนิดนึง ในการสร้างฟังก์ชั่นหลักในซับพอท เวกเตอร์ แมชชีนที่เป็นที่นิยมเพราะข้อจำกัดและขอบเขตการตอบสนองผ่านเขตทั้งหมดของแกน X

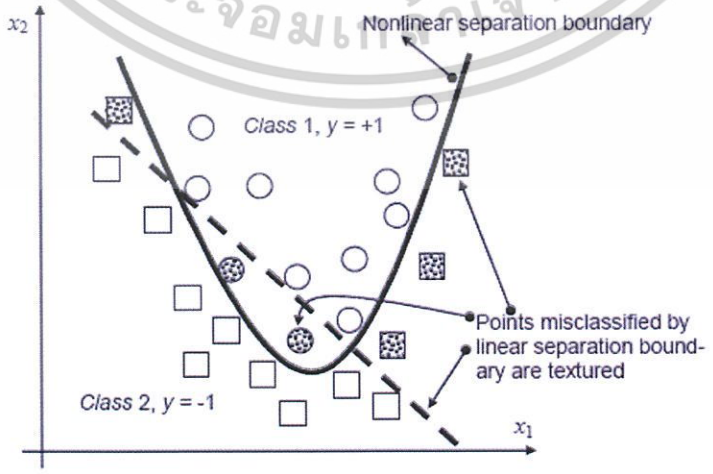


รูปที่ 2.11 แสดงกราฟตัวอย่างการจำแนกข้อมูลของฟังก์ชั่น RBF ในมุมมอง 1 มิติ

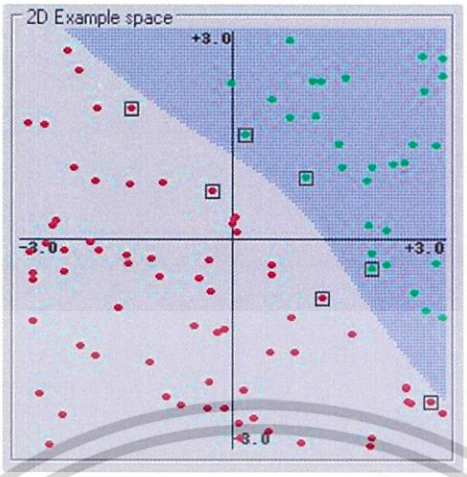


รูปที่ 2.12 แสดงกราฟตัวอย่างการจำแนกข้อมูลของฟังก์ชั่น RBF ในมุมมอง 2 มิติ

2.7.5.2 Polynomial: (gamma*u'*v + coef0)^degree

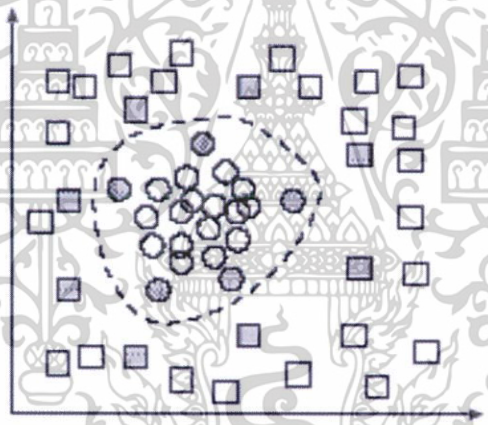


รูปที่ 2.13 แสดงกราฟตัวอย่างการจำแนกข้อมูลของฟังก์ชั่น Polynomial ในมุมมอง 1 มิติ

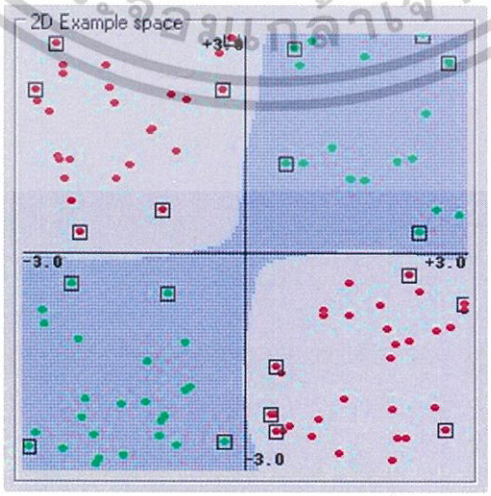


รูปที่ 2.14 แสดงกราฟตัวอย่างการจำแนกข้อมูลของฟังก์ชั่น Polynomial ในมุมมอง 2 มิติ

2.7.5.3 Radial basis: exp (-gamma*|u-v|^2)

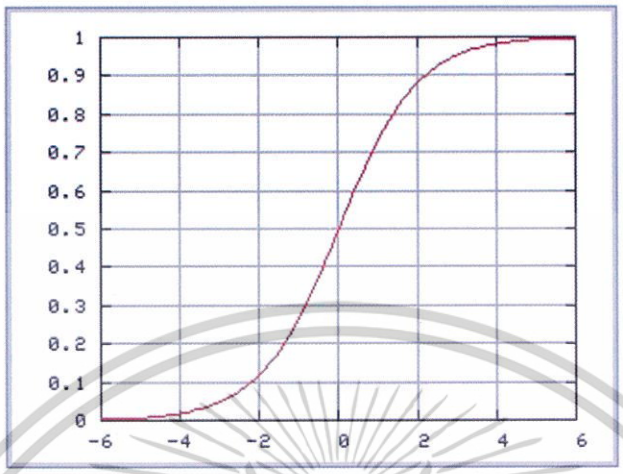


รูปที่ 2.15 แสดงกราฟตัวอย่างการจำแนกข้อมูลของฟังก์ชั่น Radial basis ในมุมมอง 1 มิติ



รูปที่ 2.16 แสดงกราฟตัวอย่างการจำแนกข้อมูลของฟังก์ชั่น Radial basis ในมุมมอง 2 มิติ

2.7.3.4 Sigmoid (feed-forward neural network): tanh(gamma*u'*v + coef0)



รูปที่ 2.17 แสดงกราฟของฟังก์ชั่น Sigmoid

2.7 สรุปข้อดี-ข้อเสียของแต่ละแมชชีนเลินนิ่ง

ตารางที่ 2.5 แสดงข้อดีและข้อเสียของตัวจำแนกทุกตัวที่จะนำมาใช้ในการวิจัย

1. เนอีฟเบย์ (Naïve Bayes)	**ข้อดี**
	• ทำได้ง่ายและมีค่อนข้างประสิทธิภาพ • บ่อยครั้งเป็นโมเดลทางเลือกที่มีประโยชน์อย่างมาก • ทำงานได้ดีในการจัดกลุ่ม ของเซตตัวอย่างมีจำนวนมากซึ่งบ่งบอกได้จาก ลักษณะของเซตตัวอย่าง (Feature) • คุณสมบัติ (Attribute) ของข้อมูลตัวอย่างไม่ขึ้นต่อกัน
	ข้อเสีย
	• ไม่สามารถตอบสนองการทำงานเยอะๆ อย่างต่อเนื่องได้ • ในบางครั้ง ไม่สามารถจำแนกความแตกต่างให้ตรงกับคลาส ที่มีอยู่ได้ - ถ้ามีคุณลักษณะน้อยจะทำให้มีผลต่อคลาส เพราะจะทำให้ความถูกต้องนั้นลดลง - เพราะเนอีฟเบย์มีการให้ค่าน้ำหนักกับทุกๆ คุณลักษณะ • ค่าความน่าจะเป็นก่อนหน้าสามารถให้ผลลัพธ์ได้ 2 ค่าทำให้ผลลัพธ์ไม่มีความแน่นอน

ตารางที่ 2.5 (ต่อ) แสดงข้อดีและข้อเสียของตัวจำแนกทุกตัวที่จะนำมาใช้ในการวิจัย

	ข้อดี
2. เบย์เซียนเน็ตเวิร์ก (Bayesian Network)	• เหมาะสำหรับการอนุมานที่มีข้อมูลเริ่มต้นไม่แน่นอน ผลลัพธ์ที่ได้ไม่มีความกำกวม • การเปลี่ยนแปลงในโมเดลไม่ส่งผลกระทบต่อประสิทธิภาพการทำงานของระบบ • การบำรุงรักษาและการปรับปรุงโมเดลทำได้ง่าย • สามารถแก้ปัญหางานที่มีการจัดประเภทข้อมูลและปัญหาที่เสื่อมถอยได้ (Regression Problem) • โครงสร้างของทฤษฎีใช้ในการจัดการข้อมูลที่สูญหาย (Missing Data) • สามารถใช้ได้กับตัวแปรที่มีความหลากหลายได้
	ข้อเสีย
	• ประสิทธิภาพและการขยายในส่วนของข้อมูลที่มีมาก่อนหน้าจะใช้การอ้างอิงของเบย์เซียน ความน่าเชื่อถือของผลลัพธ์ขึ้นอยู่กับความน่าเชื่อถือของความรู้ที่ป้อนเข้าไป • การคำนวนมีความยุ่งยากและต้องใช้เวลานาน และใช้ค่าใช้จ่ายสูง • ไม่เหมาะกับระบบที่มีขนาดใหญ่
	ข้อดี
3. แม็กซิมัมเอนโทรปี (Maximum Entropy)	• สามารถกำหนดข้อมูลสำคัญๆที่น่าจะเป็นไปได้ได้ - สามารถกำหนดคุณลักษณะ ที่ซับซ้อนได้ • สามารถใช้คุณลักษณะที่แตกต่างกันหลายๆคุณลักษณะ ในการถ่วงค่าน้ำหนักได้ • มีการรวมขอบข่ายงาน(Framework) สำหรับคุณลักษณะ ในการเลือกและการจำแนก - คุณลักษณะจำนวนมากช่วยเพิ่มความสามารถในการจัดการกับกระบวนการเรียนรู้ • ประกอบด้วย แหล่งความรู้ที่เป็นสัมพันธ์กัน สามารถเพิ่มความรู้ของข้อมูลได้

ตารางที่ 2.5 (ต่อ) แสดงข้อดีและข้อเสียของตัวจำแนกทุกตัวที่จะนำมาใช้ในการวิจัย

	ข้อเสีย
3. แม็กซิมัม เอนโทรปี (Maximum Entropy)	• ใช้ได้เฉพาะคุณลักษณะแบบไบนารี (Binary feature) เท่านั้น - มีประสิทธิภาพต่ำในบางสถานการณ์ที่ต้องการการแยกแยะมากกว่า 2 คลาส • ใช้เวลาในการคำนวนแต่ละครั้งนานมาก เพราะค่าคาดหวังในแต่ละครั้ง เกิดจากการทำซ้ำคือนอมัลไลซิ่ง (Normalizing) ตลอดเวลา ทำให้ เปลืองเวลาในการนำข้อมูลเข้าและออก (I/O) ขณะทำงาน • ขาดความแน่นอนในการจัดการกับปัญหา
4.ซับพอท เวกเตอร์ แมชชีน (Support Vector Machine)	**ข้อดี**
	• มีเมธอด (Method) จำนวนมาก ที่มีประสิทธิภาพสูงในการทำงาน • ซับพอท เวกเตอร์ แมชชีนมักจะมีประสิทธิภาพดีที่สุดในการทำงาน เมื่อเทียบกับโมเดลอื่นๆ • ฟังก์ชันหลัก (Kernels Function) ของซับพอท เวกเตอร์ แมชชีน มีความแน่นอนและมีประสิทธิภาพในการจับคู่ข้อมูลเพื่อที่จะแสดงผล • ซับพอท เวกเตอร์ แมชชีนมีระนาบเกินที่ดี ทำให้การจำแนกข้อมูลมีประสิทธิภาพสูง • สามารถจัดกลุ่มประเภทของข้อมูลให้หลายประเภท
	ข้อเสีย
	• ใช้กฎการตัดสินใจแบบไบนารี่ (Binary Decision Rule) - สามารถสร้างเส้นแบ่งคลาสระนาบเกินได้แต่ในข้อมูลที่ไม่เคยมีมาก่อนอาจจะทำให้เกิดการผิดพลาดในการจับคู่ ข้อมูลได้ - สามารถสร้างความน่าจะเป็นที่จะแบ่งคลาสเป็นรูปแบบต่างๆได้ แต่ในบางครั้งมันก็ทำงานได้ไม่ดีพอ • ขนาดของเวกเตอร์ (Vector) จะโตตามขนาดของเซตของข้อมูล (Data Set) • การทำเซตข้อมูลที่ใช้ในการเรียนรู้ (Training Set Data)ใช้เวลานาน

บทที่ 3
การรวบรวมคลังข้อมูลและสร้างเทรนนิ่งเซต ที่จะใช้ในการทดลอง รวมถึงเครื่องมือที่ใช้

3.1 การสร้างคลังข้อมูล (Corpus)

การเก็บรวบรวมข้อมูลเพื่อนำมาสร้างคลังข้อมูล (Corpus) เป็นขั้นตอนหนึ่งในการสร้างเทรนนิ่งเซต (Training Set) เพื่อนำไปให้ใช้แมชชีนเลินนิ่ง (Machine Learning) ที่มีคุณสมบัติในการเป็นตัวจำแนก (Classifier) แบบต่างๆ ทำการเรียนรู้

3.1.1 รวบรวมบทความในหัวข้อที่สนใจ

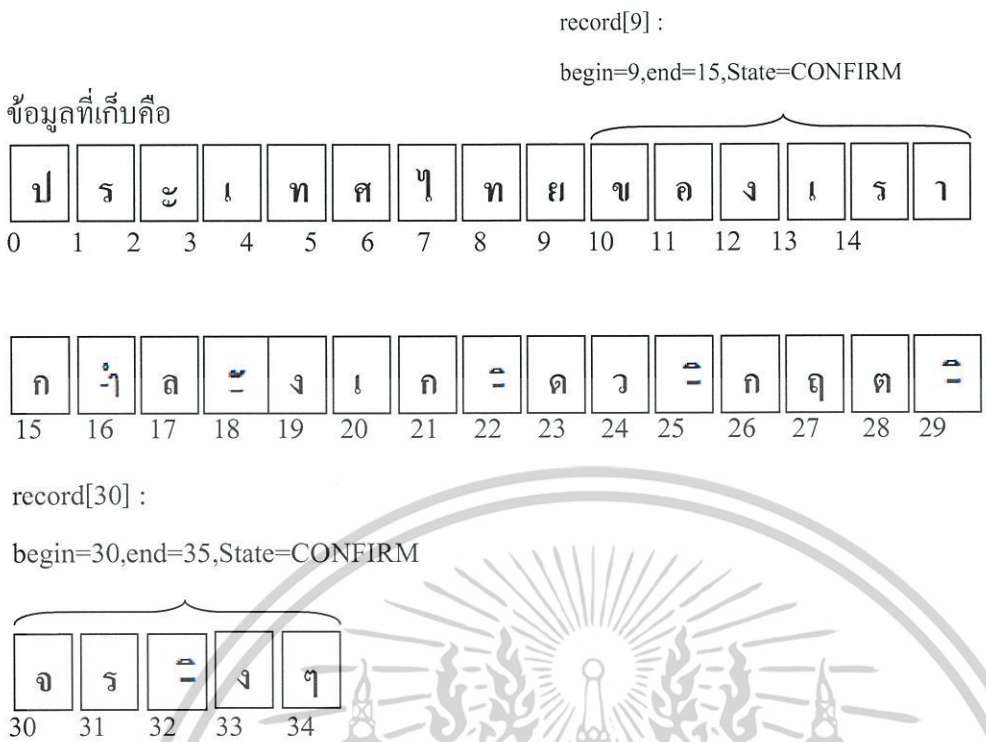
ในโครงงานนี้จะเลือกบทความในหัวข้อที่เกี่ยวกับ "อาหารและสุขภาพ" แหล่งที่มาของบทความเหล่านี้มาจากหนังสือนิตยสารสุขภาพต่างๆ และบทความจากทางเวปไซต์

3.1.2 ตัดคำฟุ่มเฟือยออกจากแต่ละบทความและรวบรวมเป็นคลังข้อมูล

เมื่อได้บทความในหัวข้อที่สนใจแล้ว จะนำบทความดังกล่าวมาตัดเอาคำฟุ่มเฟือยหรือคำที่ไม่เป็นสาระสำคัญของประโยคออก โดยการใช้ "โปรแกรมช่วยสร้างคลังข้อมูลสำหรับการสรุปใจความสำคัญภาษาไทย" (Summarized Corpus Construction Tool) ที่ได้พัฒนาขึ้นภายใต้โครงงานนี้ ซึ่งจะกล่าวถึงรายละเอียดของโปรแกรมนี้ในหัวข้ออื่นถัดไป

การทำงานของโปรแกรมอย่างคร่าวๆ โปรแกรมจะเก็บตำแหน่งของคำที่เราได้กระทำการใดๆกับคำนั้น โดยจะบอกสถานะว่าได้ทำการตัดคำนั้นออกจากบทความเรียบร้อยแล้ว (Confirm) หรือขีดเส้นใต้ไว้เพื่อแสดงว่าอาจจะทำการตัดคำนี้ออกไปในอนาคต (Pending) หรือ ทำการเปลี่ยนคำที่ฟุ่มเฟือยให้เป็นคำที่มีความกระชับขึ้น (Replace with) ถ้าเป็นสถานะ Confirm และ Pending โปรแกรมจะทำการเก็บตำแหน่งเริ่มต้นและตำแหน่งสุดท้ายของคำนั้นไว้แล้วเขียนกำกับว่าคำนั้นถูกกระทำด้วยสถานะใด สำหรับการ Replace with นั้นโปรแกรมจะเก็บตำแหน่งเริ่มต้นและสิ้นสุดของคำนั้น รวมทั้งบอกไว้ว่าคำที่แทนที่คำนั้นคือคำว่าอะไร ดังตัวอย่างนี้

ประเทศไทยของเรากำลังเกิดวิกฤติจริงๆ



รูปที่ 3.1 แสดงการเก็บข้อมูลลงในเรคคอร์ดของข้อมูลตัวอย่าง

และเมื่อทำการตัดคำฟุ่มเฟือยออกจากบทความแล้ว ก็ทำการบันทึกและรวบรวมเป็นคลังข้อมูลสำหรับระบบ

3.1.3 ลักษณะของคลังข้อมูลที่เหมาะสมกับการทำทดลอง

การเก็บคลังข้อมูล จำเป็นต้องใช้จำนวนข้อมูลที่มีปริมาณที่เหมาะสม ยิ่งข้อมูลมีจำนวนมากเท่าไหร่ ยิ่งส่งผลดีต่อการเรียนรู้ของแมชชีนเลินนิ่ง เพราะจะทำให้การเรียนรู้มีประสิทธิภาพมากยิ่งขึ้น อีกทั้งประเภทของข้อมูลก็มีผลต่อการเรียนรู้เช่นกัน เพราะยิ่งบทความเป็นบทความประเภทเดียวกัน จะพบคำที่ซ้ำกันหลายคำในแต่ละบทความ การตัดคำจะมีการตัดคำที่ไม่ต้องการซ้ำๆ กันและมีลักษณะที่ใกล้เคียงกัน ทำให้ผลการเก็บข้อมูลการตัดมีความสอดคล้องกัน ส่งผลดีต่อการเรียนรู้ของแมชชีนเลินนิ่งทำให้สามารถเรียนรู้ได้โมเดลที่มีความถูกต้องและแม่นยำมากยิ่งขึ้น

3.2 เทรนนิ่งเซต

เทรนนิ่งเซต คือ ชุดของข้อมูลที่ใช้ในการเรียนรู้ของแมชชีนเลินนิ่ง โดยจะทำการเรียนรู้ว่าคำใดบ้างเป็นคำที่ฟุ่มเฟือยของบทความ ผลลัพธ์ที่ได้จากการเรียนรู้คือ โมเดลที่สามารถนำไปใช้ในการออกแบบระบบสรุปใจความภาษาไทยอัตโนมัติได้

3.2.1 รูปแบบของเทรนนิ่งเซตที่จะใช้

การเรียนรู้เทรนนิ่งเซต นั้นต้องใช้เครื่องมือเพื่อช่วยให้เกิดการเรียนรู้ของตัวจำแนก โดยเครื่องมือที่เลือกมาใช้ในโครงงานนี้ คือเวก้า (WEKA Tool) โดยในเวก้านั้นมีตัวจำแนก ให้เลือกใช้ทำงานครอบคลุมกับที่จะใช้ในการทดลอง ทำให้ประหยัดเวลาในการเตรียมเทรนนิ่งเซต

ดังนั้น เมื่อเลือกเวก้า เป็นเครื่องมือที่จะใช้ ก็ต้องสร้างเทรนนิ่งเซต ให้มีรูปแบบที่เวก้า จะรองรับด้วย โดยรูปแบบของข้อมูลที่สามารถใช้งานในเวก้า ได้ต้องอยู่ในรูปแบบ ASCII อาจจะเป็น ARFF, CSV, C45

ในกรณีที่แฟ้มข้อมูลอยู่ในเครือข่ายอินเตอร์เน็ต ผู้ใช้สามารถเรียกใช้ได้โดยอาศัยการเรียกผ่าน URL หรืออาจจะใช้ข้อมูลที่อยู่ในฐานข้อมูลที่เชื่อมโยงผ่าน JDBC ซึ่งในการทดลองของโครงงานนี้ จะเลือกสร้างเทรนนิ่งเซตให้อยู่ในรูปแบบ “ARFF”

3.2.1.1 รูปแบบข้อมูล ARFF

ARFF ย่อมาจาก Attribute-Relation File Format เป็นการเก็บข้อมูลโดยใช้ ASCII มีรูปแบบการเก็บข้อมูลภายใน ดังนี้

	“ไฟล์.ARFF”
@relation *name*	*เป็นบรรทัดที่บอกชื่อข้อมูลเชิงสัมพันธ์*
@attribute *attribute-name type*	เป็นบรรทัดที่บอกชื่อลักษณะประจำ (attribute) และชนิด Numeric หรือ real หมายถึง ลักษณะประจำเก็บเป็นตัวเลข $\{V_1, V_2, \ldots, V_n\}$ หมายถึง ลักษณะประจำเก็บค่าไม่ต่อเนื่องที่กำหนดไว้ให้
@data	เป็นบรรทัดที่บอกถึงแถวที่ตามมาจะเป็นข้อมูล แถวละหนึ่งเรคคอร์ดเรียงตามลักษณะประจำที่บอกไว้ข้างต้น คั่นด้วยคอมม่า (,)

3.2.1.2 ตัวอย่างไฟล์ข้อมูล ARFF

```
@relation weather

@attribute outlook {sunny, overcast, rainy}
@attribute temperature real
@attribute humidity real
@attribute windy {TRUE, FALSE}
```

```
@attribute play {yes, no}

@data
sunny,85,85,FALSE,no
sunny,80,90,TRUE,no
overcast,83,86,FALSE,yes
rainy,70,96,FALSE,yes
rainy,68,80,FALSE,yes
rainy,65,70,TRUE,no
overcast,64,65,TRUE,yes
sunny,72,95,FALSE,no
sunny,69,70,FALSE,yes
rainy,75,80,FALSE,yes
sunny,75,70,TRUE,yes
overcast,72,90,TRUE,yes
overcast,81,75,FALSE,yes
rainy,71,91,TRUE,no
```

จากตัวอย่างไฟล์ข้อมูล ARFF มีชื่อข้อมูลเชิงสัมพันธ์ว่า "Weather" มีลักษณะประจำ (Attribute) 5 ตัว คือ outlook, temperature, humidity, windy และ play และมีข้อมูล (Data) จำนวน 14 เรคคอร์ด

3.3 การสร้างเทรนนิ่งเซตเพื่อใช้ในการทดลอง

เมื่อรวบรวมคลังข้อมูลและเลือกรูปแบบของเทรนนิ่งเซตที่จะใช้ในการทดลองได้แล้ว ขั้นต่อไปจะทำการออกแบบเทรนนิ่งเซตที่จะใช้ในการทดลอง โดยจะต้องออกแบบว่าจะใช้คุณสมบัติ (Feature) ใดบ้างเพื่อกำหนดเป็น ลักษณะประจำ (Attribute) ใน เทรนนิ่งเซต

3.3.1 คุณสมบัติ ต่างๆ ที่จะใช้ในเทรนนิ่งเซต

คุณสมบัติ เป็นรูปแบบเงื่อนไขที่เรากำหนดให้แก่คำในบทความ และให้ค่าความสัมพันธ์ของคำและค่าน้ำหนักของคำนั้นๆ เพื่อให้แมชชีนเลินนิ่ง ได้ทำการเรียนรู้และสร้างเป็นความรู้ในการที่จะตัดหรือไม่ตัดคำๆ นั้น ประเภทของคุณสมบัติ ที่ใช้มีด้วยกัน 4 ประเภท คือ

3.3.1.1 หมายเลขประจำ (ID) ของคำแต่ละคำ

เนื่องจากในเทรนนิ่งเซตไม่สามารถนำคำแต่ละคำเข้าไปเรียนรู้ได้โดยตรง จึงต้องทำการแทนคำแต่ละคำด้วยตัวเลข ซึ่งจะเรียกว่า "หมายเลขประจำ (ID) ของคำแต่ละคำ"

ตารางที่ 3.1 แสดงตัวอย่าง การแทนคำด้วยหมายเลขประจำ (ID)

หมายเลขประจำ (ID)	คำ	หมายเลขประจำ (ID)	คำ
1	สับปะรด	11	ทำ
2	เป็น	12	อาหาร
3	ผลไม้	13	ทั้ง
4	อม	14	คาว
5	หวาน	15	และ
6	เปรี้ยว	16	ได้
7	ที่	17	อร่อย
8	สามารถ	18	หลาย
9	นำ	19	ชนิด
10	ไป	20	ให้

3.3.1.2 ชนิดของคำในประโยค (POS; Part-Of-Speech)

ภาษาไทยไม่เหมือนภาษาอังกฤษหรือภาษาอื่นๆ ในยุโรปคือไม่มีเครื่องหมายแบ่งประโยคที่ชัดเจนหรือแม้กระทั่งช่องว่างที่ท้ายประโยค แต่ไม่ใช่ทุกช่องว่างในย่อหน้าที่จะเป็นเครื่องหมายของการจบประโยค มันยังสามารถใช้สำหรับจุดหมายอื่นได้อีกด้วย เช่น การใช้ระหว่างวลี หรืออนุประโยค ภายในประโยค, ระหว่างประโยคที่อยู่ในกลุ่มประโยค, ก่อนและหลังตัวเลข ฯลฯ มีบางงานได้ขยายอัลกอริทึมสำหรับกำกับหน้าที่คำ (Part-Of-Speech Tagging หรือ POS Tagging) ในเงื่อนไขความเป็นได้เพื่อแยกแยะประโยค หน้าที่นี้สามารถมองได้ว่าเป็นปัญหาการแยกกลุ่ม สามารถจำกัดความช่องว่างตามหน้าที่ของมันได้สองชนิดคือเป็นจุดหยุดของประโยค (Sentence-Break) และไม่เป็นจุดหยุดของประโยค (Non-Sentence-Break)

ตารางข้างล่างแสดงตัวอย่างของหน้าที่ของคำในคลังข้อความในออร์คิด (Orchid Corpus) ซึ่งเป็นคลังข้อความภาษาไทยที่ใช้กันแพร่หลาย

ตารางที่ 3.2 ตารางชนิดของคำในประโยค (POS; Part-Of-Speech)

ลำดับ	POS	คำอธิบาย	Feature	ตัวอย่างคำ
1	NPRP	Proper noun	Content	โคโรน่า, พระอาทิตย์
2	NCNM	Cardinal number	Content	หนึ่ง, สอง, สิบ, 1, 2, 10
3	NONM	Ordinal number	Content	ที่หนึ่ง, ที่สอง, ที่สาม
4	NLBL	Label number	Content	1, 2, 3, ก, ข, a, b
5	NCMN	Common noun	Content	หนังสือ, อาหาร, น้ำ, อากาศ
6	NTTL	Title noun	Content	ดร., พลเอก, นาย
7	PPRS	Personal pronoun	Content	คุณ, เขา, ฉัน
8	PDMN	Demonstrative pronoun	Function	นั่น, ที่นี่, ที่นั่น, ที่โน่น
9	PNTR	Interrogative pronoun	Function	ใคร, อะไร, อย่างไร
10	PREL	Relative pronoun	Function	ที่, ซึ่ง, อัน, ผู้
11	VACT	Active verb	Content	ทำงาน, เดิน, กิน
12	VSTA	Stative verb	Content	เห็น, รู้, คือ
13	VATT	Attributive verb	Content	อ้วน, ดี,สวย
14	XVBM	Pre-verb auxiliary, before negator "ไม่"	Function	เกิด, เกือบ, กำลัง
15	XVAM	Pre-verb auxiliary, after negator "ไม่"	Function	ค่อย, น่า, ได้
16	XVMM	Pre-verb, before or after negator "ไม่"	Function	ควร, เคย, ต้อง
17	XVBB	Pre-verb auxiliary, in imperative mood	Function	กรุณา, อย่า, ห้าม, เชิญ
18	XVAE	Post-verb auxiliary	Function	ไป, มา, ขึ้น
19	DDAN	Definite determiner, after noun without classifier in between	Function	นี่, นั่น, โน่น, ทั้งหมด
20	DDAC	Definite determiner, allowing classifier in between	Function	นี่, นั้น, โน้น, นู้น
21	DDBQ	Definite determiner, between noun and classifier or preceding quantitative expression	Function	ทั้ง, อีก, เพียง

ตารางที่ 3.2 (ต่อ) ตารางชนิดของคำในประโยค (POS; Part-Of-Speech)

ลำดับ	POS	คำอธิบาย	Feature	ตัวอย่างคำ
22	DDAQ	Definite determiner, following noun; allowing classifier in between	Function	พอดี, ถ้วน
23	DIAC	Indefinite determiner, following noun; allowing classifier in between	Function	ไหน, อื่น, ต่างๆ
24	DIBQ	Indefinite determiner, between noun and classifier or preceding quantitative expression	Function	บาง, ประมาณ, เกือบ
25	DIAQ	Indefinite determiner, following quantitative expression	Function	กว่า, เศษ
26	DCNM	Determiner, cardinal number expression	Content	หนึ่งคน, เสือ 2 ตัว
27	DONM	Determiner, ordinal number expression	Content	ที่หนึ่ง, ที่สอง, ที่สุดท้าย
28	ADVN	Adverb with normal form	Content	เก่ง, เร็ว, ช้า, สม่ำเสมอ
29	ADVI	Adverb with iterative form	Content	เร็วๆ, เสมอๆ, ช้าๆ
30	ADVP	Adverb with prefixed form	Function	โดยเร็ว
31	ADVS	Sentential adverb	Function	โดยปกติ, ธรรมดา
32	CNIT	Unit classifier	Content	ตัว, คน, เล่ม
33	CLTV	Collective classifier	Content	คู่, กลุ่ม, ฝูง, เล่ม, ด้าน, ทาง
34	CMTR	Measurement classifier	Content	กิโลกรัม, แก้ว, ชั่วโมง
35	CFQC	Frequency classifier	Content	ครั้ง, เที่ยว
36	CVBL	Verbal classifier	Content	ม้วน, มัด
37	JCRG	Coordinating conjunction	Function	และ, หรือ, แต่
38	JCMP	Comparative conjunction	Content	กว่า, เหมือน, เท่ากับ
39	JSBR	Subordinating conjunction	Function	เพราะว่า, เนื่องจาก, ที่, แม้ว่า
40	RPRE	Preposition	Content	จาก, ละ, ของ, ใต้, บน
41	INT	Interjection	Function	โอ๊ย, โอ้, เออ, เอ๋, อ๋อ
42	FIXN	Nominal prefix	Function	การทำงาน, ความสนุกสนาน

ตารางที่ 3.2 (ต่อ) ตารางชนิดของคำในประโยค (POS; Part-Of-Speech)

ลำดับ	POS	คำอธิบาย	Feature	ตัวอย่างคำ
43	FIXV	Adverbial prefix	Function	อย่างเร็ว
44	EAFF	Ending for affirmative sentence	Function	จ๊ะ, ค่ะ, ครับ, นะ, น่า, เถอะ
45	EITT	Ending for interrogative sentence	Function	หรือ, เหรอ, ไหม, มั้ย
46	NEG	Negator	Content	ไม่, มิได้, ไม่ได้, มิ
47	PUNC	Punctuation	Function	( ), *, .., :, “ ”

3.3.1.3 ลักษณะของคำ (Feature of word)

ลักษณะของคำแบ่งออกเป็น 2 ประเภท คือ Content Feature และ Function Feature

1) Content Feature

เป็นลักษณะของคำในภาษาไทย ที่มีความหมายในตัวเอง หรือเป็นคำที่อ้างอิงไปยังคำข้างหน้า ลักษณะของคำที่เป็น Content Feature ได้แก่

- คำนาม (Noun)
- สรรพนาม (Pronoun)
- คำกริยา (Verb)
- คำวิเศษณ์ (Adverb)
- คำคุณศัพท์ (Adjective)

2) Function Feature

เป็นลักษณะของคำในภาษาไทยที่ทำหน้าที่ทางไวยากรณ์ ไม่ว่าจะทำหน้าที่ในการเชื่อมคำกับคำ เชื่อมประโยคกับประโยค หรือเชื่อมคำกับประโยค คำบอกตำแหน่งต่างๆ ลักษณะของคำที่เป็น Function Feature ได้แก่

- คำสันธาน (Conjunction)
- คำบุพบท (Preposition)
- คำอุทาน (Interjection)

3.3.2 การแปลงคลังข้อมูล ให้เป็น ไฟล์ ARFF

การสร้างเทรนนิ่งเซต ซึ่งอยู่ในรูปแบบของไฟล์ ARFF โดยการแปลงข้อมูลจากคลังข้อมูล มีกระบวณการในการทำดังต่อไปนี้

3.3.2.1 สร้างหมายเลขประจำ (ID) ให้คำแต่ละคำ โดยมีขั้นตอนดังนี้

1) ใช้ SWATH (Smart Word Analysis for Thai) เครื่องมือที่พัฒนาโดย หน่วยปฏิบัติการวิจัยวิทยาการมนุษยภาษา (HLT; Human Language Technology Laboratory) ศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติ (NECTEC; National Electronics and Computer Technology Center) มาตัดคำ ในประโยค จะได้ผลลัพธ์เป็นไฟล์ .txt ที่มีการตัดคำแล้ว

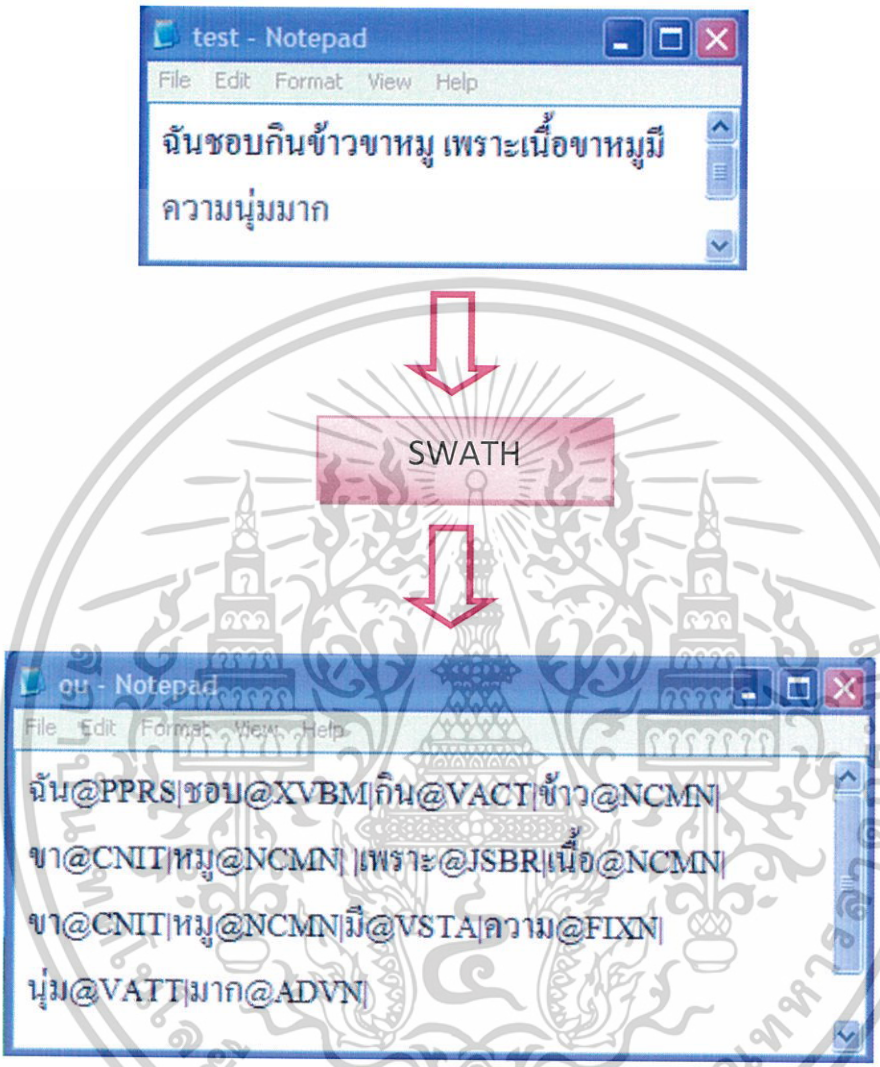
รูปที่ 3.2 แสดงการทำงานของโปรแกรม SWATH เพื่อตัดคำ

2) นำคำแต่ละคำที่ถูกตัดแล้ว มากำหนดหมายเลขประจำ ดังตัวอย่างที่แสดงในตารางที่ 3.1 เพื่อใช้อ้างอิงในการสร้างเทรนนิ่งเซต

3.3.2.2 กำหนดชนิดของคำในประโยค (Part-Of-Speech: POS) และลักษณะของคำ (Feature of word) ให้คำแต่ละคำ

โดยใช้ SWATH (Smart Word Analysis for Thai) เครื่องมือที่พัฒนาโดยหน่วยปฏิบัติการวิจัยวิทยาการมนุษยภาษา (HLT; Human Language Technology Laboratory) ศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติ (NECTEC; National Electronics and Computer Technology Center) โดยมีขั้นตอนดังนี้

1) ให้ SWATH บอกชนิดของคำออกมาให้



รูปที่ 3.3 แสดงการทำงานของโปรแกรม SWATH เพื่อบอกชนิดของคำ

2) นำชนิดของคำในประโยคและลักษณะของคำว่าคำเป็นชนิด Content หรือ Function โดยพิจารณาตามเกณฑ์ที่ระบุไว้ในตาราง 3.2 ของแต่ละคำที่ได้ไปสร้างเทรนนิ่งเซต

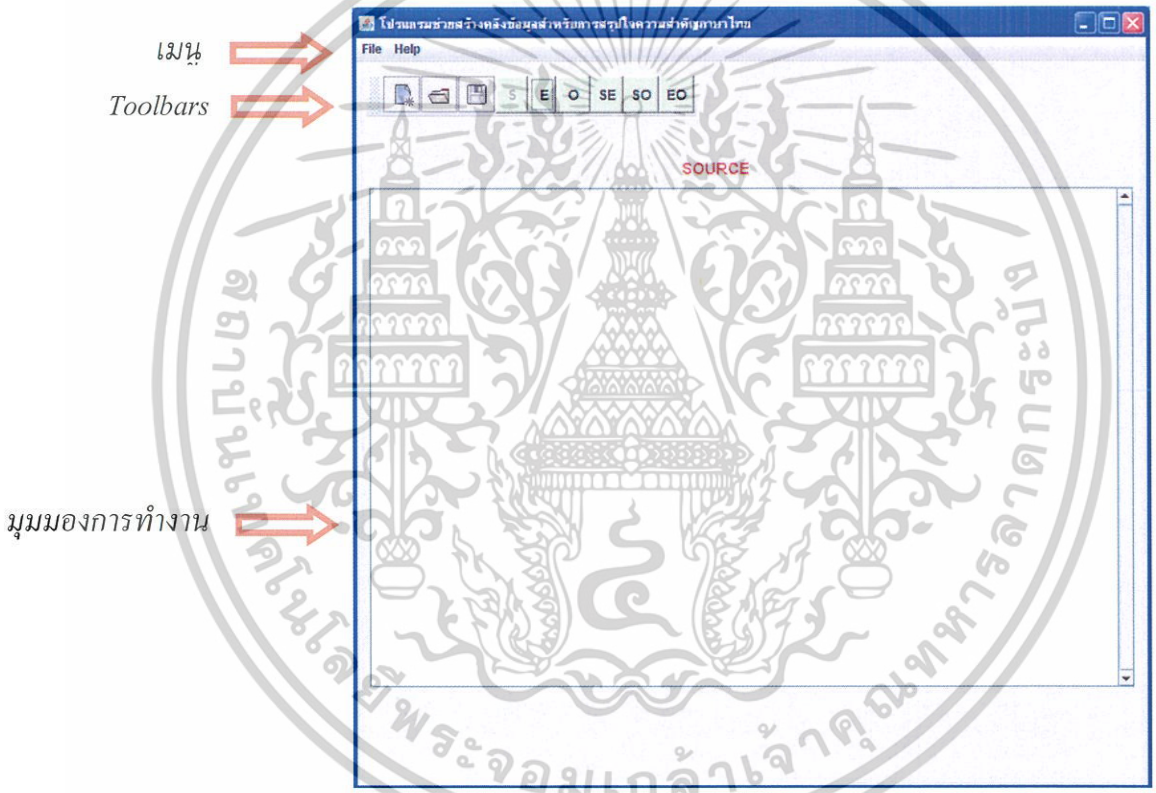
3.4 เครื่องมือที่ใช้

จากกระบวนการการรวบรวมคลังข้อมูล และสร้างเทรนนิ่งเซต ที่กล่าวมาในหัวข้อข้างต้น มีการใช้เครื่องมือต่างๆ ดังนี้

3.4.1 โปรแกรมช่วยสร้างคลังข้อมูลสำหรับการสรุปใจความสำคัญภาษาไทย (Summarized Corpus Construction Tool)

เป็นโปรแกรมที่พัฒนาขึ้นภายใต้โครงงานนี้ จุดประสงค์เพื่อใช้ในการสร้างคลังข้อมูลสำหรับการสรุปใจความสำคัญภาษาไทย โดยจะทำการเก็บตำแหน่งและสถานะของคำที่ผู้ใช้ทำการตัดออกจากบทความ และนำข้อมูลเหล่านั้น ใช้อ้างอิงเพื่อสร้างเป็นเทรนนิ่งเซตต่อไป ซึ่งเป็นมีรูปร่างหน้าจอ (Graphic User Interface) ที่ใช้งานง่าย มีหลายมุมมองการทำงาน

3.4.1.1 หน้าต่างหลักของโปรแกรม

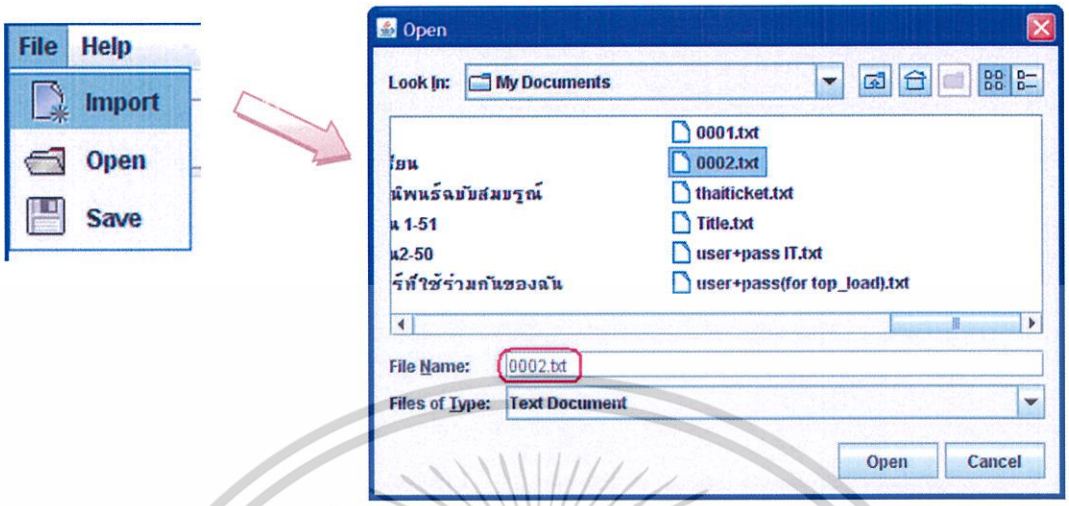


รูปที่ 3.4 แสดงหน้าตาหลักของโปรแกรมช่วยสร้างคลังข้อมูลสำหรับการสรุปใจความสำคัญภาษาไทย

3.4.1.2 เมนูของโปรแกรม

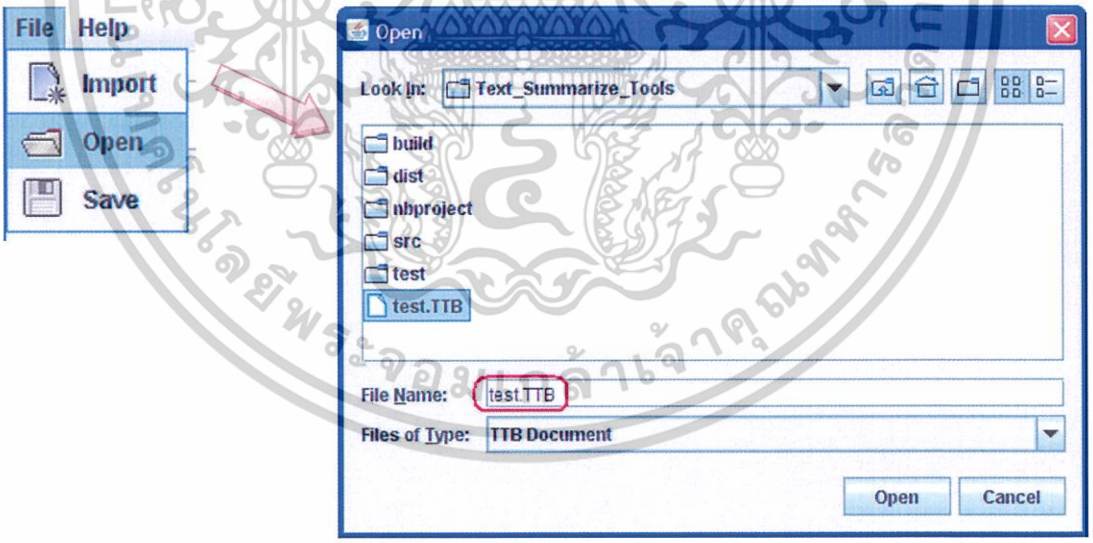
1. **เมนูไฟล์ (File Menu)** เป็นเมนูที่ใช้จัดการเกี่ยวกับไฟล์

   1.1 Import เป็นเมนูที่ใช้เปิดไฟล์เข้ามาเป็นประโยคตั้งต้น (ไฟล์ที่เปิดต้องเป็น .txt File และมี Encoding UTF-8 เท่านั้น)



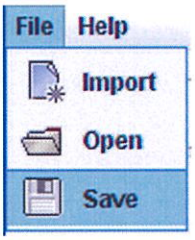
รูปที่ 3.5 แสดงการใช้เมนู Import ของโปรแกรมช่วยสร้างคลังข้อมูลสำหรับการสรุปใจความสำคัญภาษาไทย

1.2 Open เป็นเมนูที่ใช้เปิดไฟล์ที่ทำงานยังไม่เสร็จสิ้นเพื่อนำมาทำงานต่อ (ไฟล์ต้องเป็นไฟล์ที่ผู้ใช้ทำการ Save ไว้ และเป็นไฟล์ .TTB เท่านั้น)



รูปที่ 3.6 แสดงการใช้เมนู Open ของโปรแกรมช่วยสร้างคลังข้อมูลสำหรับการสรุปใจความสำคัญภาษาไทย

1.3 Save เป็นเมนูที่ใช้บันทึกข้อมูลลงไฟล์เมื่อผู้ใช้ยังไม่เสร็จสิ้นการตัดคำ หรือเพิ่มคำต่างๆ แต่ต้องการหยุดการทำงาน เพื่อ Open File กลับมาทำงานต่อได้



รูปที่ 3.7 แสดงการใช้เมนู save ของโปรแกรมช่วยสร้างคลังข้อมูลสำหรับการสรุปใจความสำคัญภาษาไทย

2. **เมนู Help (Help Men)** เป็นเมนูที่ใช้เป็นตัวช่วยสำหรับผู้ใช้ อธิบายถึงวิธีกาใช้งานต่างๆของโปรแกรม

3.4.1.3 แถบเครื่องมือ (Toolbars) ของโปรแกรม

แถบเครื่องมือ (Toolbars) เป็นเครื่องมือที่สามารถช่วยให้ผู้ใช้ทำให้ใช้งานได้ง่ายและสะดวกมากขึ้น

รูปที่ 3.8 แสดงแถบเครื่องมือ (Toolbars) ของโปรแกรมช่วยสร้างคลังข้อมูลสำหรับการสรุปใจความสำคัญภาษาไทย

1. **เครื่องมือจัดการไฟล์**

1.1 เครื่องมือ Import (Import Tool)  เป็นเครื่องมือที่ใช้เปิดไฟล์เข้ามาเป็นประโยคตั้งต้น (ไฟล์ที่เปิดต้องเป็น .txt File และมี Encoding UTF-8 เท่านั้น)

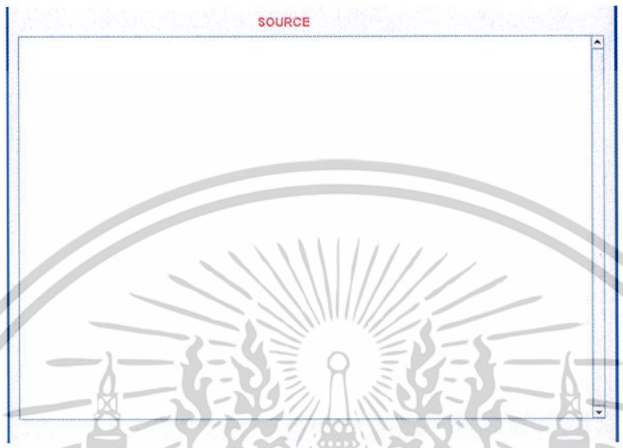
1.2 เครื่องมือ Open (Open Tool)  เป็นเครื่องมือที่ใช้เปิดไฟล์ที่ทำงานยังไม่เสร็จสิ้นเพื่อนำมาทำงานต่อ

1.3 เครื่องมือ Save (Save Tool)  เป็นเครื่องมือที่ใช้บันทึกข้อมูลลงไฟล์เมื่อผู้ใช้ยังไม่เสร็จสิ้นการตัดคำ หรือเพิ่มคำต่างๆ แต่ต้องการหยุดการทำงาน เพื่อเปิดไฟล์กลับมาทำงานต่อได้

2. เครื่องมือมุมมองการทำงาน

2.1 Source's View 

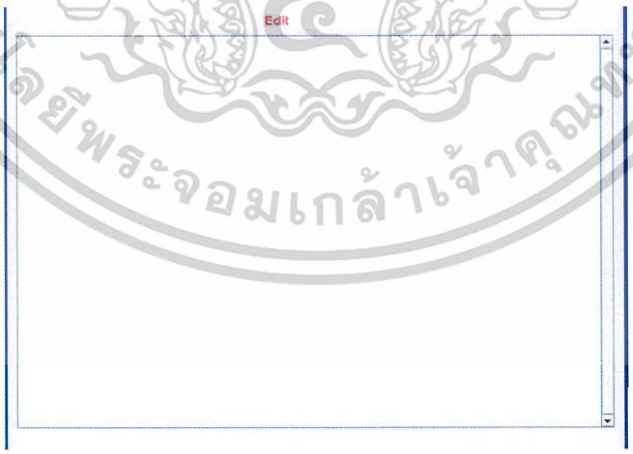
เป็นครื่องมือที่ทำให้มุมมองการทำงานถูกเปลี่ยนเป็นมุมมองต้นฉบับของเอกสาร



รูปที่ 3.9 แสดงมุมมองต้นฉบับของเอกสารของโปรแกรมช่วยสร้างคลังข้อมูลสำหรับการสรุปใจความสำคัญภาษาไทย

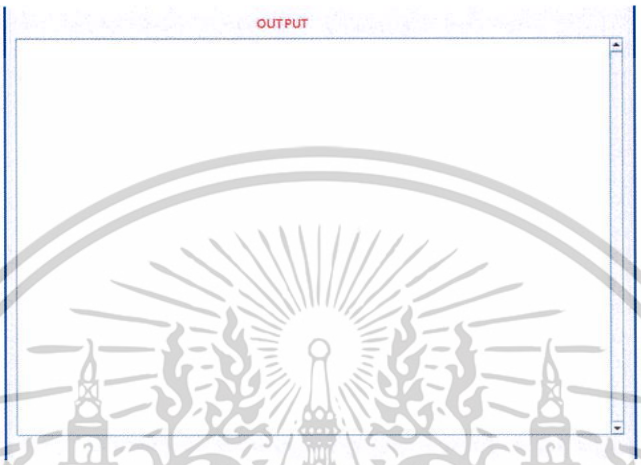
2.2 Edit's View 

เป็นเครื่องมือที่ทำให้มุมมองการทำงานถูกเปลี่ยนเป็นมุมมองที่ผู้ใช้สามารถตัดคำและเปลี่ยนคำได้



รูปที่ 3.10 แสดงมุมมองที่สามารถตัดคำได้ในเอกสารได้ของโปรแกรมช่วยสร้างคลังข้อมูลสำหรับการสรุปใจความสำคัญภาษาไทย

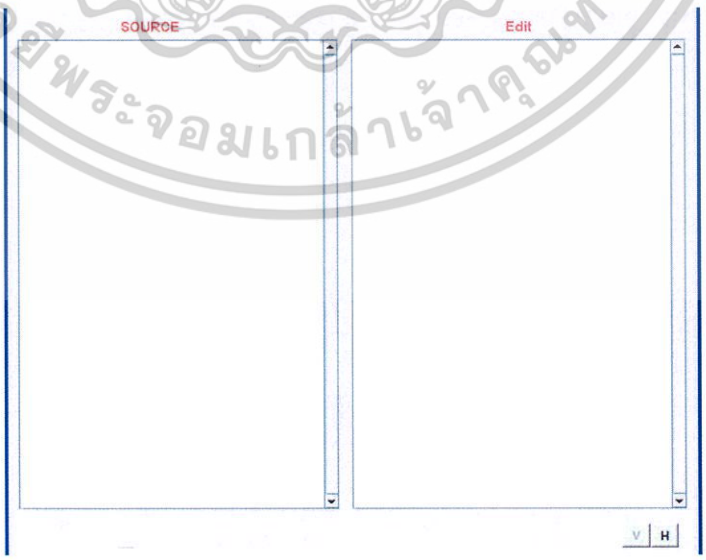
2.3 Output's View  เป็นเครื่องมือที่ทำให้มุมมองการทำงานถูกเปลี่ยนเป็นมุมมองที่แสดงผลลัพธ์ ของประโยคตั้งต้นที่ถูกตัดแล้ว (ผลลัพธ์ที่ได้มาจากกระบวนการต่างๆที่เกิดขึ้นใน Edit's View)



รูปที่ 3.11 แสดงมุมมองแสดงผลลัพธ์ของโปรแกรมช่วยสร้างคลังข้อมูลสำหรับการสรุปใจความสำคัญภาษาไทย

2.4 SE's View  เป็นเครื่องมือที่แสดงมุมมอง Source 's View และ Edit 's View ควบคู่กันโดยมีให้เลือกใช้สองรูปแบบคือ

1. แนวตั้ง (Vertical)



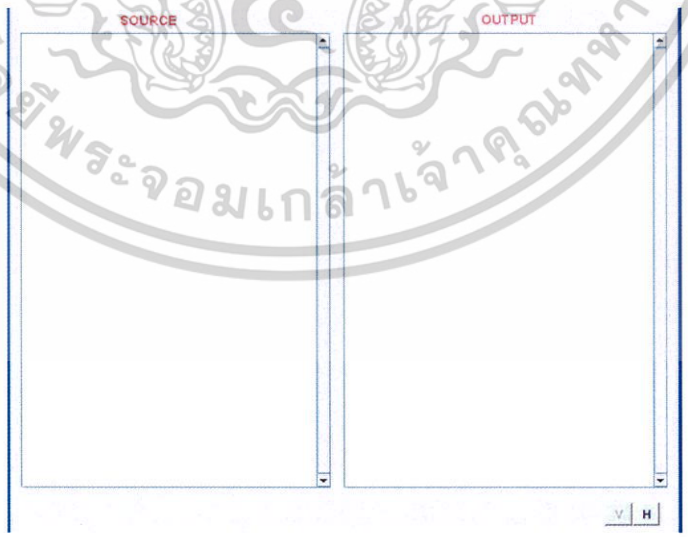
รูปที่ 3.12 แสดงมุมมอง Source's View และ Edit's View ของโปรแกรมช่วยสร้างคลังข้อมูลสำหรับการสรุปใจความสำคัญภาษาไทยในแนวตั้ง

2. แนวนอน (Horizontal)

รูปที่ 3.13 แสดงมุมมอง Source's View และ Edit's View ของโปรแกรมช่วยสร้างคลังข้อมูลสำหรับการสรุปใจความสำคัญภาษาไทยในแนวนอน

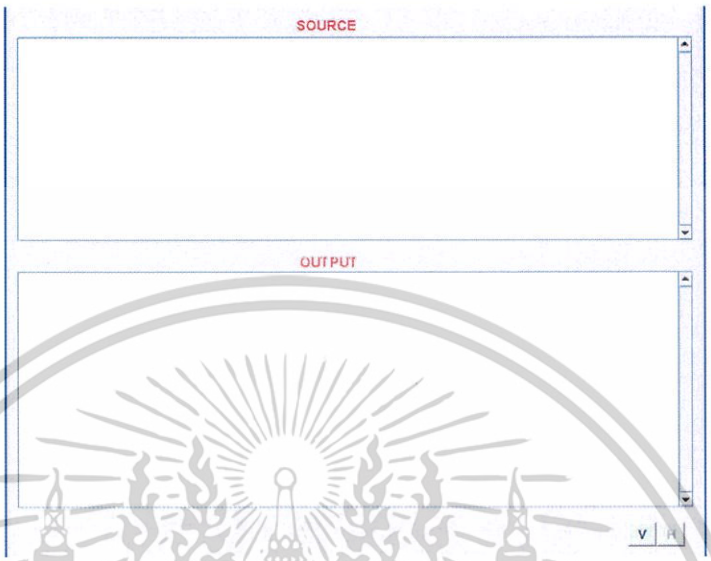
2.5 SO's View  เป็นเครื่องมือที่แสดงมุมมอง Source's View และ Output's View ควบคู่กันมีให้เลือกใช้สองรูปแบบคือ

1. แนวตั้ง (Vertical)



รูปที่ 3.14 แสดงมุมมอง Source's View และ Output's View ของโปรแกรมช่วยสร้างคลังข้อมูลสำหรับการสรุปใจความสำคัญภาษาไทยในแนวตั้ง

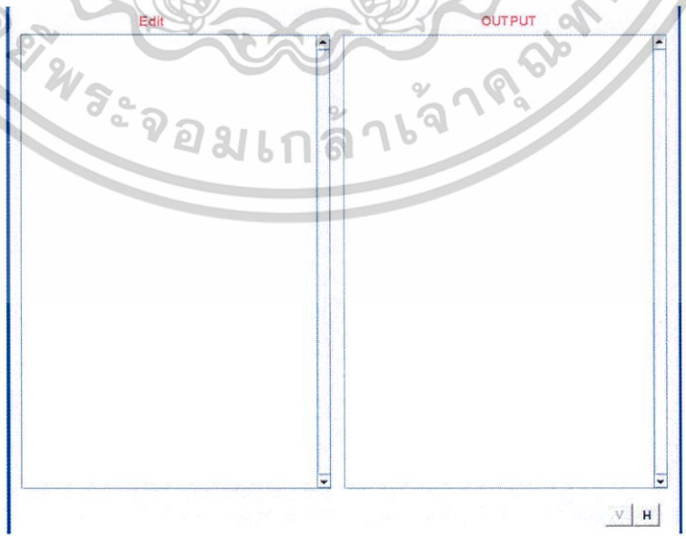
2. แนวนอน (Horizontal)



รูปที่ 3.15 แสดงมุมมอง Source's View และ Output's View ของโปรแกรมช่วยสร้างคลังข้อมูลสำหรับการสรุปใจความสำคัญภาษาไทยในแนวนอน

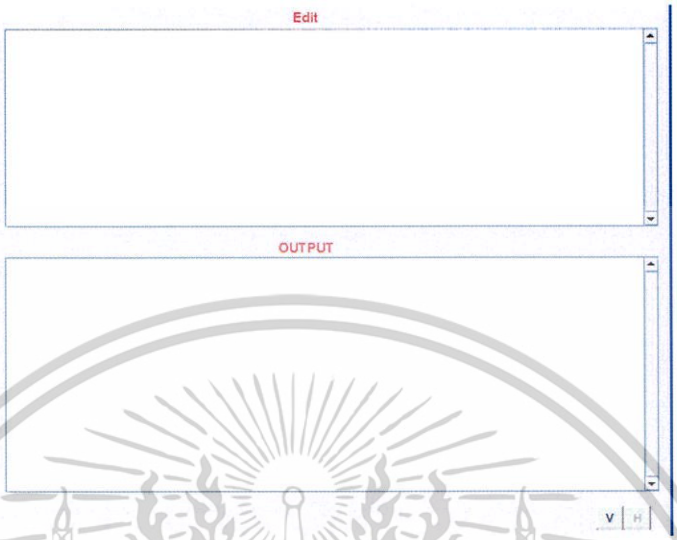
2.6 EO's View  เป็นเครื่องมือที่แสดงมุมมอง Edit's View และ Output's View ควบคู่กันมีให้เลือกใช้สองรูปแบบคือ

1. แนวตั้ง (Vertical)



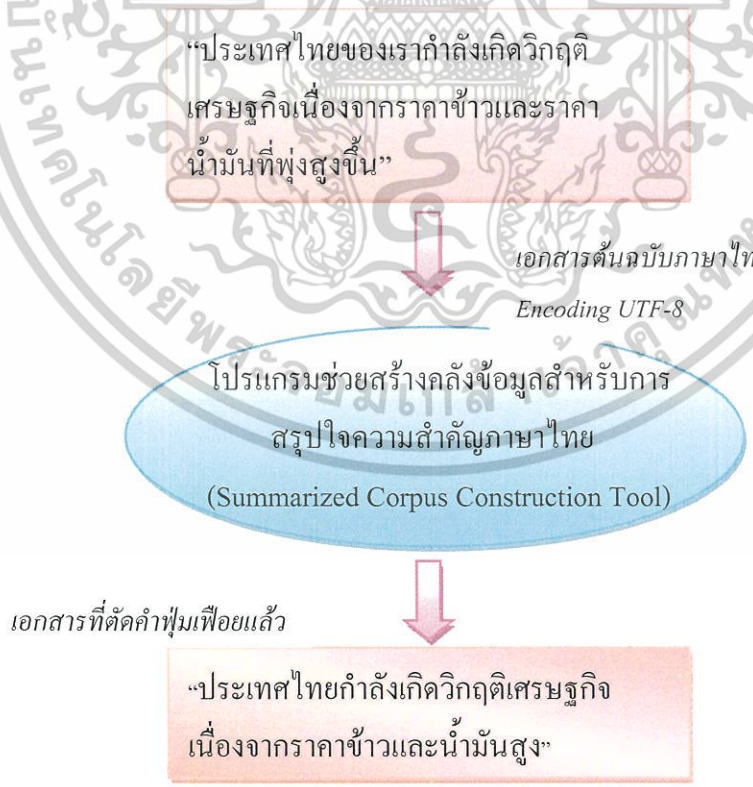
รูปที่ 3.16 แสดงมุมมอง Edit's View และ Output's View ของโปรแกรมช่วยสร้างคลังข้อมูลสำหรับการสรุปใจความสำคัญภาษาไทยในแนวตั้

2. แนวนอน (Horizontal)



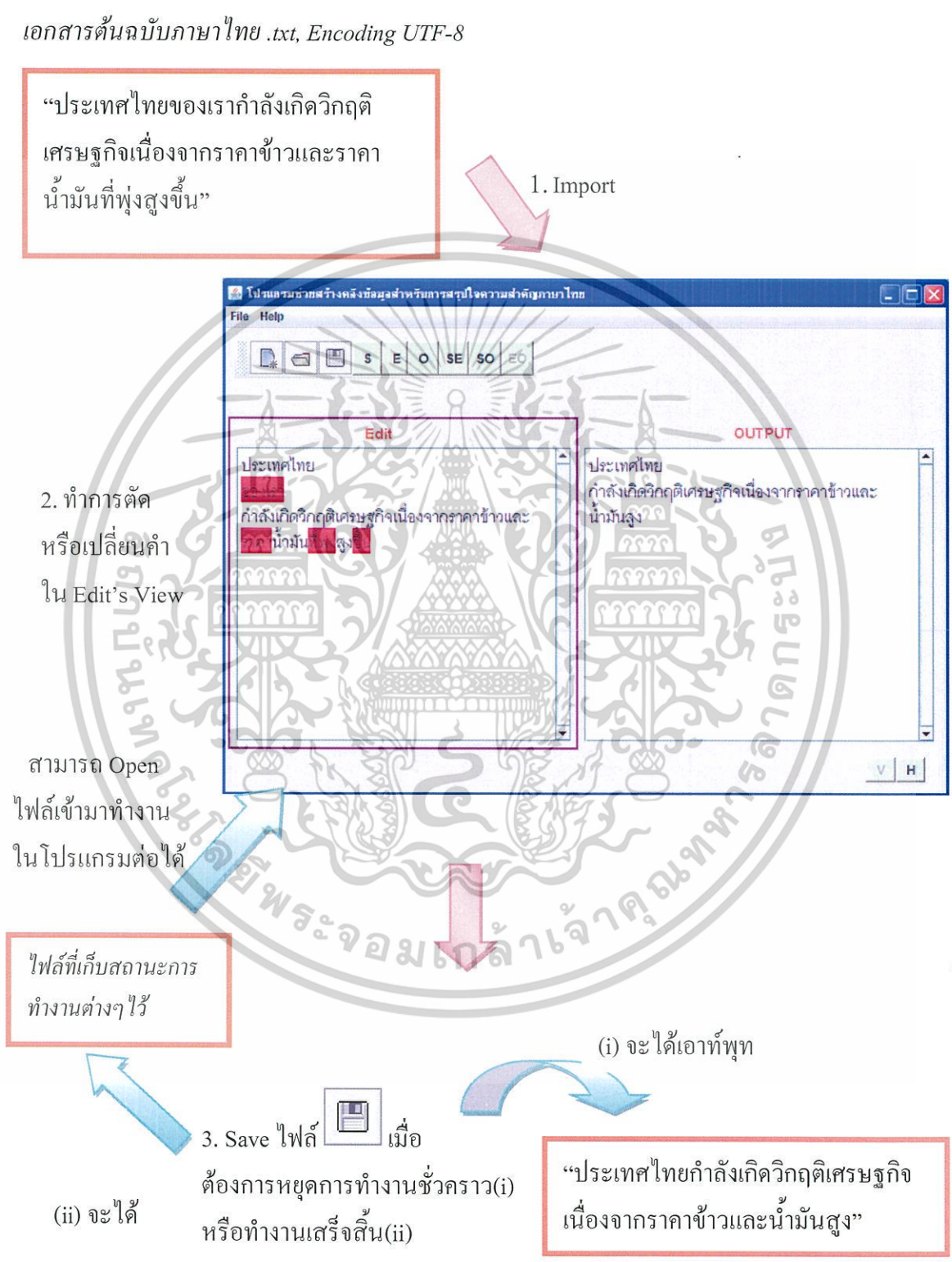
รูปที่ 3.17 แสดงมุมมอง Edit's View และ Output's View ของโปรแกรมช่วยสร้างคลังข้อมูลสำหรับการสรุปใจความสำคัญภาษาไทยในแนวนอน

3.4.1.3 ภาพรวมการทำงานของโปรแกรม



รูปที่ 3.18 แสดงภาพรวมการทำงานของโปรแกรมช่วยสร้างคลังข้อมูลสำหรับการสรุปใจความสำคัญภาษาไทย

3.4.1.4 กระบวนการทำงานของโปรแกรม

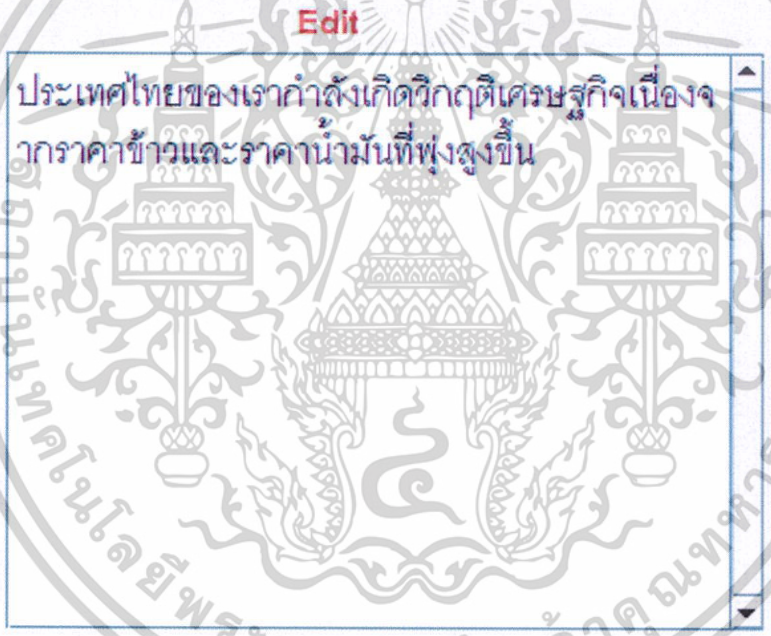


รูปที่ 3.19 แสดงกระบวณการทำงานของโปรแกรมช่วยสร้างคลังข้อมูลสำหรับการสรุปใจความสำคัญภาษาไทย

จากรูปที่ 3.19 สามารถอธิบายได้ดังนี้

1. Import ไฟล์เอกสารภาษาไทยนามสกุล .txt ,Encoding UTF-8 เข้ามาในโปรแกรม
2. ทำการตัด หรือเปลี่ยนคำของเอกสารในมุมมองการทำงาน Edit's View
3. เมื่อเสร็จสิ้นการทำงาน ทำการ Save File จะได้ข้อมูลที่มีการตัดและเปลี่ยนแปลงคำต่างๆเพื่อรวบรวมเป็นคลังข้อมูลที่ใช้ในการสรุปภาษาไทย แต่หากระหว่างการทำงานนั้นยังไม่เสร็จสิ้น แต่อยากหยุดการทำงาน ก็สามารถ Save File เพื่อเปิดกลับมาทำงานต่อได้

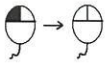
3.4.1.3 การทำงานใน Edit's View



รูปที่ 3.20 แสดงข้อมูลในมุมมอง Edit's View ของโปรแกรมช่วยสร้างคลังข้อมูลสำหรับการสรุปใจความสำคัญภาษาไทย

ใน Edit's View ผู้ใช้สามารถตัดคำได้ตามต้องการ โดยมีวิธีการดังนี้...

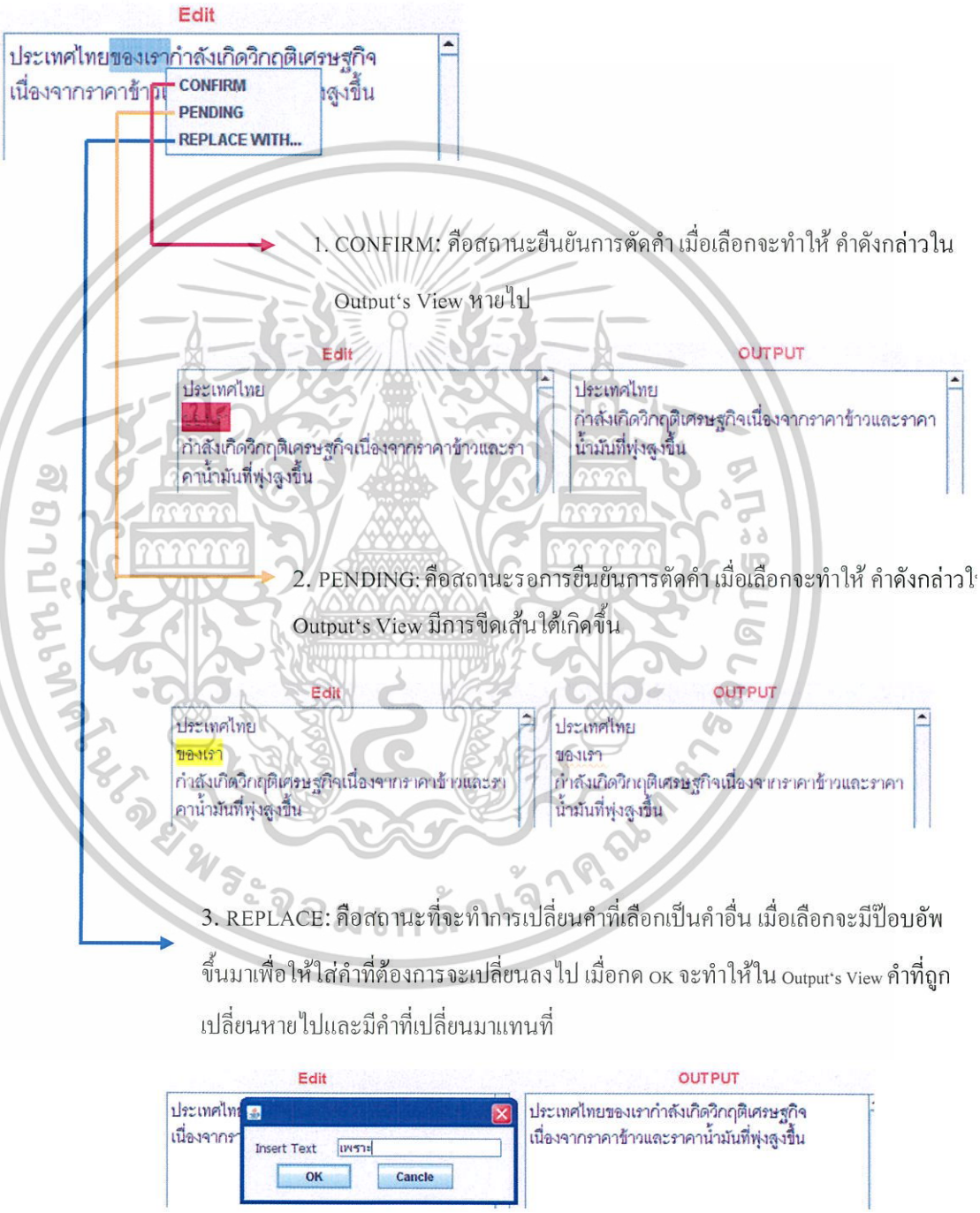
1) วิธีการเลือกตำแหน่งที่จะตัด คลิกเมาส์ลากทับตำแหน่งที่ต้องการตัด โดยจะต้องลากเมาส์จากซ้ายไปขวาเท่านั้น



ประเทศไทยของเรากำลังเกิดวิกฤติเศรษฐกิจ
เนื่องจากราคาข้าวและราคาน้ำมันที่พุ่งสูงขึ้น

2) วิธีการเลือกสถานะของตำแหน่งที่เลือกไว้

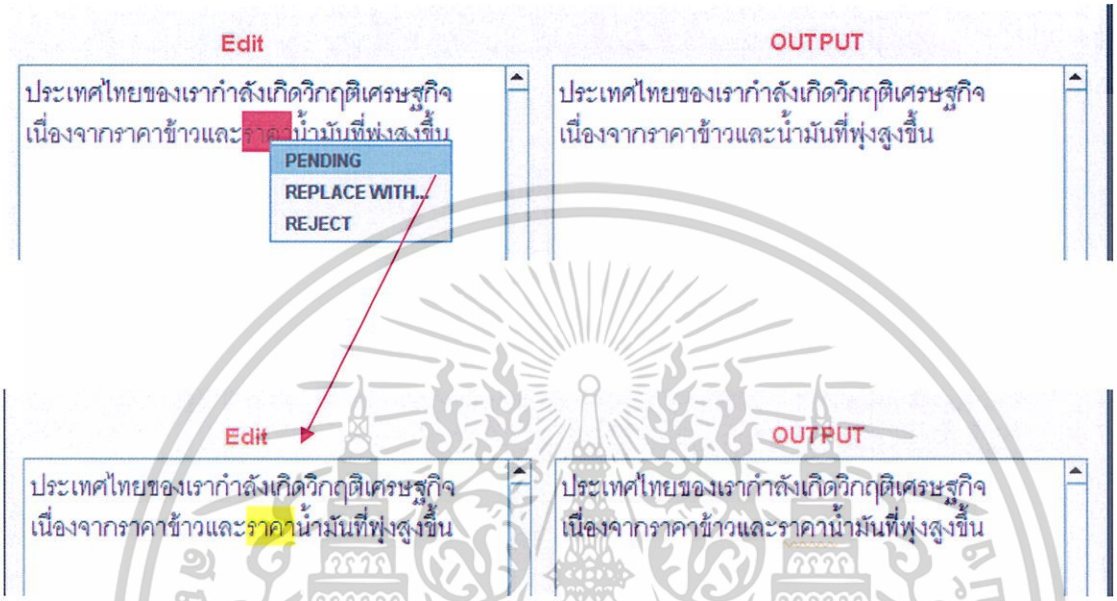
- ใช้เมาส์ชี้ภายในบริเวณที่เลือกไว้ (จาก 1) แล้วคลิกขวาจะแสดง ป๊อบอัพเมนู ดังรูป
- เลือกสถานะตามที่ต้องการ ซึ่งมีให้เลือก 3 สถานะ



รูปที่ 3.21 แสดงการเลือกวิธีสถานะในมุมมอง Edit's View และผลลัพธ์จากการทำงานของโปรแกรมช่วยสร้างคลังข้อมูลสำหรับการสรุปใจความสำคัญภาษาไทย

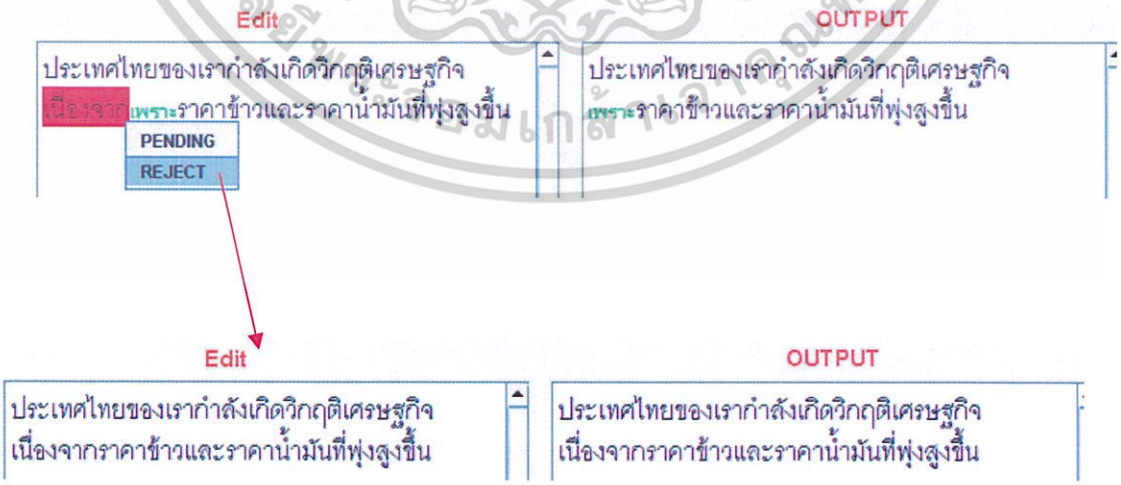
3) การเปลี่ยนและยกเลิกสถานะของคำ

3.1 การเปลี่ยนสถานะ ทำได้โดยการคลิกขวาในบริเวณคำที่ต้องการจะเปลี่ยนสถานะและทำการเลือกสถานะที่ต้องการจะเปลี่ยน



รูปที่ 3.22 แสดงการเปลี่ยนสถานะ ในมุมมอง Edit's View และผลลัพธ์จากการทำงานของโปรแกรมช่วยสร้างคลังข้อมูลสำหรับการสรุปใจความสำคัญภาษาไทย

3.2 การยกเลิกสถานะที่เลือกไว้ ทำได้โดยการคลิกขวาในบริเวณคำที่ต้องการจะยกเลิก และทำการเลือก Reject จะทำให้คำนั้นๆกลับมาเป็นสถานะตั้งต้น



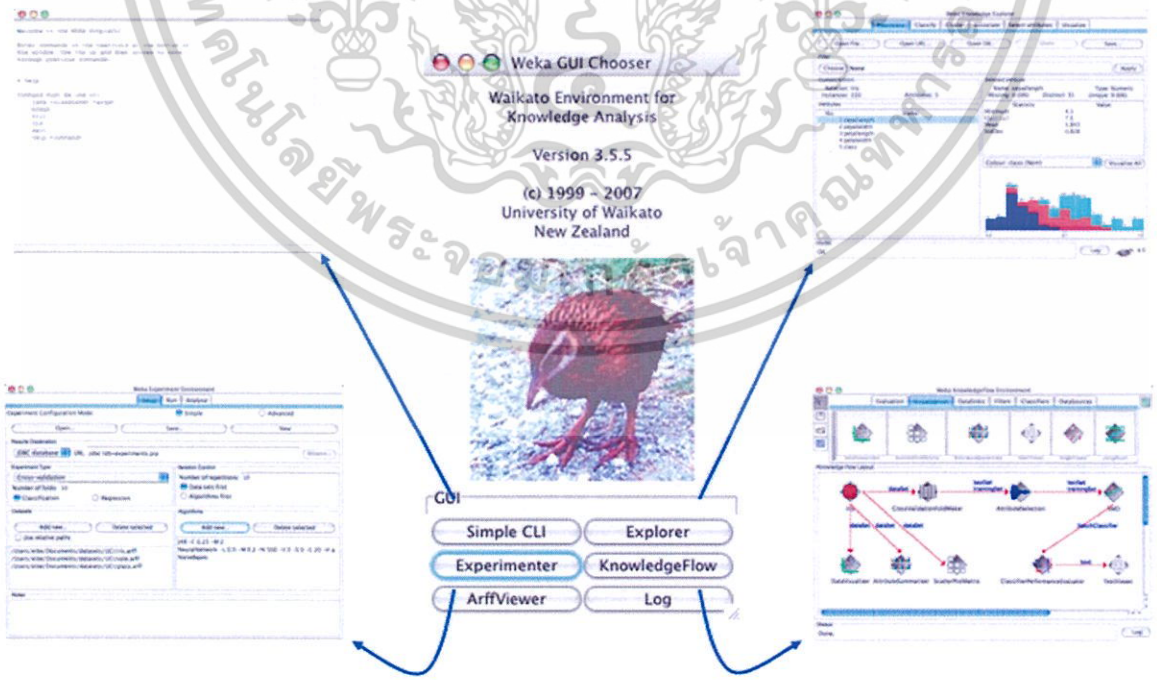
รูปที่ 3.23 แสดงการยกเลิกสถานะ ในมุมมอง Edit's View และผลลัพธ์จากการทำงานของโปรแกรมช่วยสร้างคลังข้อมูลสำหรับการสรุปใจความสำคัญภาษาไทย

3.4.2 เครื่องมือเวก้า (WEKA Tool)

รูปที่ 3.24 แสดงหน้าหลักของโปรแกรม WEKA

WEKA ย่อมาจาก Waikato Environment for Knowledge Analysis เป็นซอฟต์แวร์ฟรีที่แจกจ่ายภายใต้ GPL License ภาษาที่ใช้เขียนคือ จาวา โปรแกรมนี้เขียนมาโดยเน้นการเรียนรู้ด้วยเครื่อง แมชชินเลินนิ่งกับการทำเหมืองข้อมูล (Data Mining) มีโมดูลย่อยสำหรับการจัดการข้อมูลและใช้ GUI และคำสั่งในการสั่งให้ซอฟต์แวร์ประมวลผล

3.4.2.1 โปรแกรมการทำงานหลักของ WEKA มีดังนี้



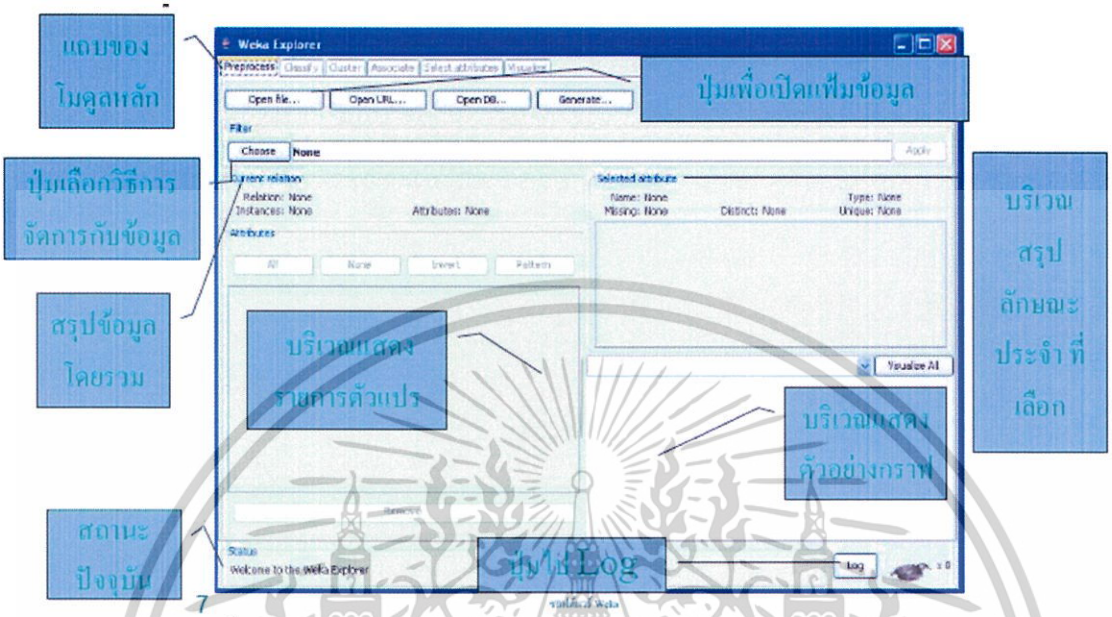
รูปที่ 3.25 ภาพรวมการทำงานทั้งหมดของโปรแกรม WEKA

1) **Simple CLI (Command Line Interface)** เป็นโปรแกรมรับคำสั่งการทำงานผ่านการพิมพ์
2) **Explorer** เป็นโปรแกรมที่ออกแบบในลักษณะเชื่อมต่อกับผู้ใช้แบบกราฟฟิค (GUI)
3) **Experimenter** เป็นโปรแกรมที่ออกแบบการทดลองและการทดสอบผลลัพธ์
4) **KnowledgeFlow** เป็นโปรแกรมออกแบบผังการไหลของความรู้
5) **ArffViewer** เป็นโปรแกรมที่ใช้สำหรับแก้ไขแฟ้มประเภท ARFF
6) **Log** เป็นโปรแกรมที่ใช้อ่านข้อความบันทึกเก็บระหว่างการทำงาน

โปรแกรมแบ่งเป็น 2 ชนิด ตามการทำงาน ดังนี้

1. **Program**
   - ***Log Window*** เป็นหน้าต่างที่จับข้อมูลที่ถูกปริ๊นจาก stdout ไปยังstderr มีประโยชน์กับการใช้ MS Windows ซึ่ง WEKA จะไม่เริ่มต้นการทำงานจากส่วนท้าย
   - ***Exit*** ปิดหน้าต่างโปรแกรม WEKA
2. **Applications** แสดงการทำงานหลักของโปรแกรมหลักใน WEKA
   - ***Explorer*** ใช้สำหรับสำรวจข้อมูลด้วย WEKA (ส่วนที่เหลือของข้อมูลเกี่ยวข้องกับโปรแกรมส่วนนี้ในส่วนของข้อมูลมากกว่า)
   - ***Experimenter*** ใช้สำหรับแสดงการทดลองและการจัดการทางสถิติระหว่างกระบวนการเรียนรู้
   - ***Knowledge Flow*** จะสนับสนุนการทำงานหลักเหมือนกับการทำงานของ Explorer แต่จะทำผ่านการทำ drag-and-drop กระบวนการนี้ช่วยเพิ่มการเรียนรู้ให้มากขึ้น
   - ***SimpleCLT*** สร้างคำสั่งแบบ Command Line อย่างง่าย ที่อนุญาติให้คำสั่งของ WEKA ดำเนินการได้โดยตรงสำหรับระบบปฏิบัติการที่ไม่สามารถสร้างคำสั่ง Command Line ได้ด้วยตัวเอง

3.4.2.2 ส่วนประกอบของ Explorer



รูปที่ 3.26 แสดงส่วนประกอบของ Explorer ใน โปรแกรม WEKA

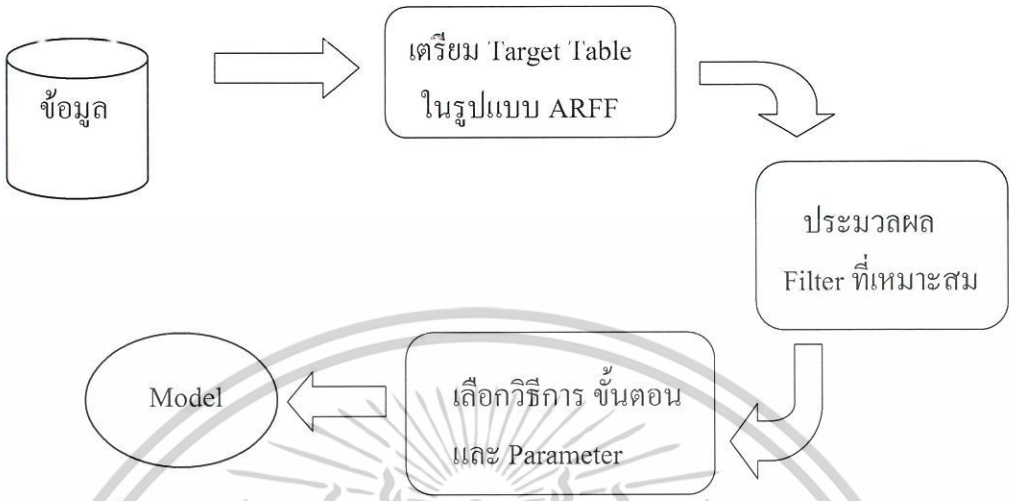
1) เมนูหลักของ Explorer

- *Preprocess* การเตรียมข้อมูล
- *Classify* รวมโมดูลการทำเหมืองข้อมูลแบบจัดแบ่งประเภท
- *Cluster* รวมโมดูลการทำเหมืองข้อมูลแบบการเกาะกลุ่ม
- *Associate* รวมโมดูลการทำเหมืองข้อมูลแบบกฎเชื่อมโยง
- *Select attributes* รวมโมดูลสำหรับการวิเคราะห์ความเกี่ยวพันของลักษณะประจำ
- *Visualize* นำเสนอข้อมูลด้วยภาพนามธรรมสองมิติ

2) ส่วนประกอบอื่นของ Explorer

- *Log box* แสดงบันทึกการเรียกใช้งานซอฟต์แวร์ WEKA ทั้งหมด ความผิดพลาดที่เกิดขึ้นจะแสดงในส่วนนี้
- *Status box* แสดงการประมวลผลปัจจุบันของซอฟต์แวร์ มีการแจ้งการผิดพลาดแต่ไม่มีรายละเอียด
- *Bird Icon* แสดงรูปนกกีวี ถ้ามีการประมวลผลนกกีวีจะขยับตัวไปมา ถ้าไม่มีการประมวลผลนกกีวีจะนั่งเฉยๆ

3) หลักการทำงานของ Explorer ใน WEKA



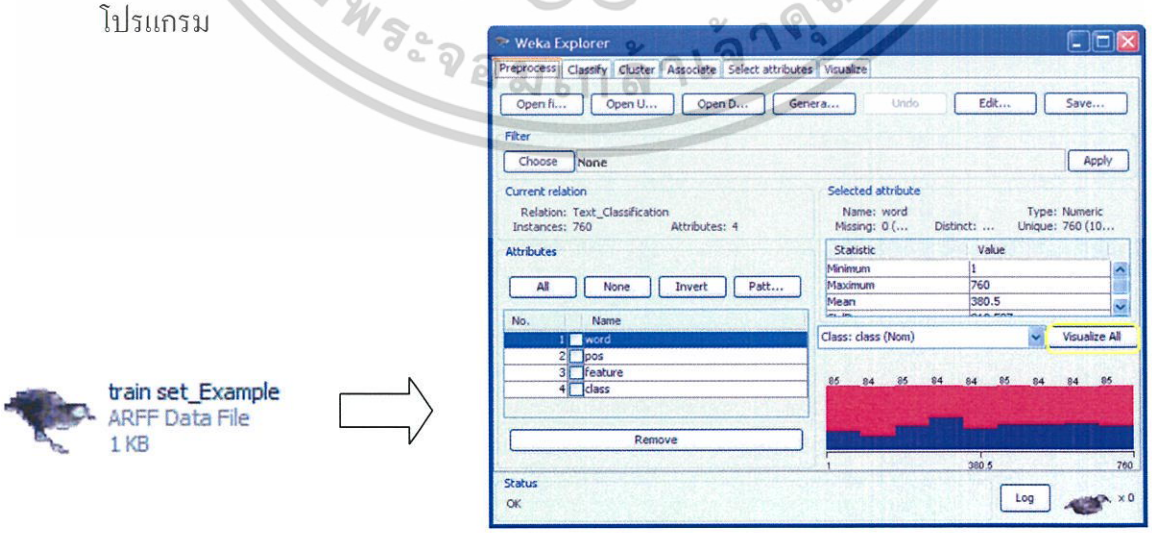
รูปที่ 3.27 แสดงหลักการทำงานของ Explorer ในโปรแกรม WEKA

3.4.2.3 รูปแบบข้อมูลที่โปรแกรม WEKA รองรับ

รูปแบบของข้อมูลที่จะป้อนลงไปต้องอยู่ในรูปแบบ ASCII อาจจะเป็น ARFF, CSV, C45 ในกรณีที่แฟ้มข้อมูลอยู่ในเครือข่าย Internet ผู้ใช้สามารถเรียกใช้ได้โดยอาศัยการเรียกผ่าน URL หรืออาจจะใช้ข้อมูลที่อยู่ในฐานข้อมูลที่เชื่อมโยงผ่าน JDBC โดยรูปแบบที่นิยมใช้มากสุด ได้แก่ "ARFF"

3.4.2.4 ตัวอย่าง ขั้นตอนการใช้โปรแกรม WEKA เพื่อให้ตัวจำแนก ทำการเรียนรู้ โดยในตัวอย่างจะเลือกใช้ Naïve Bayes เป็นตัวจำแนก

1) คลิกไฟล์ ARFF ที่เป็น Training Set :WEKA จะเปิดไฟล์โหลดข้อมูลเข้าไปในโปรแกรม

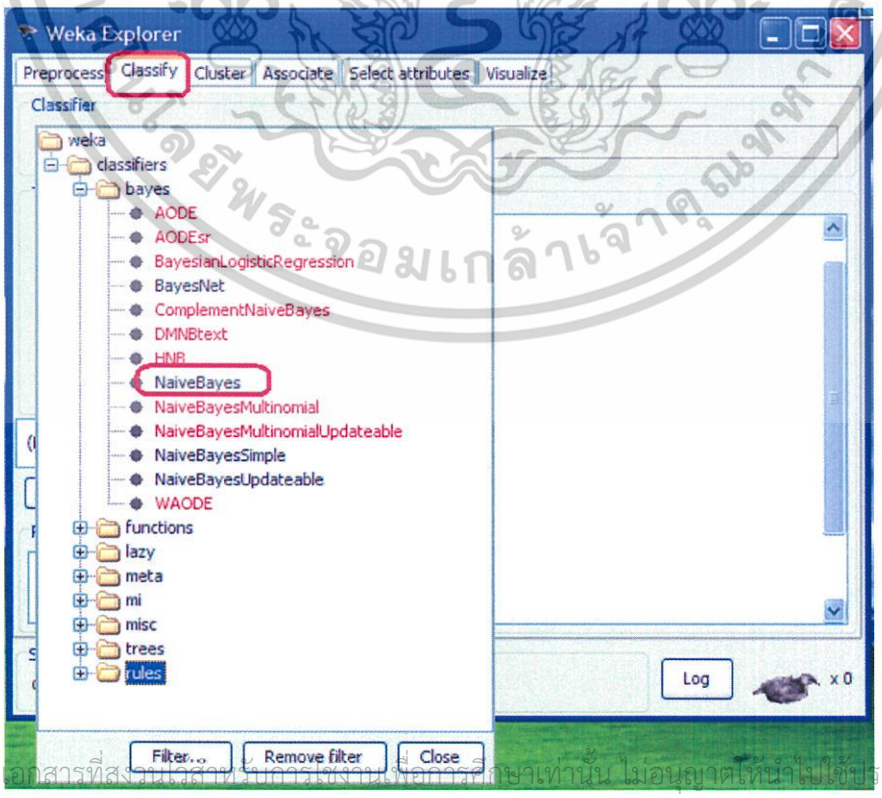


รูปที่ 3.28 แสดงการเปิดไฟล์โหลดข้อมูลเข้าไปในโปรแกรม WEKA

เมื่อกดปุ่ม Visualize all จะเห็นหน้าต่างซึ่งแสดงถึงความสัมพันธ์ของ Attribute ต่างๆ

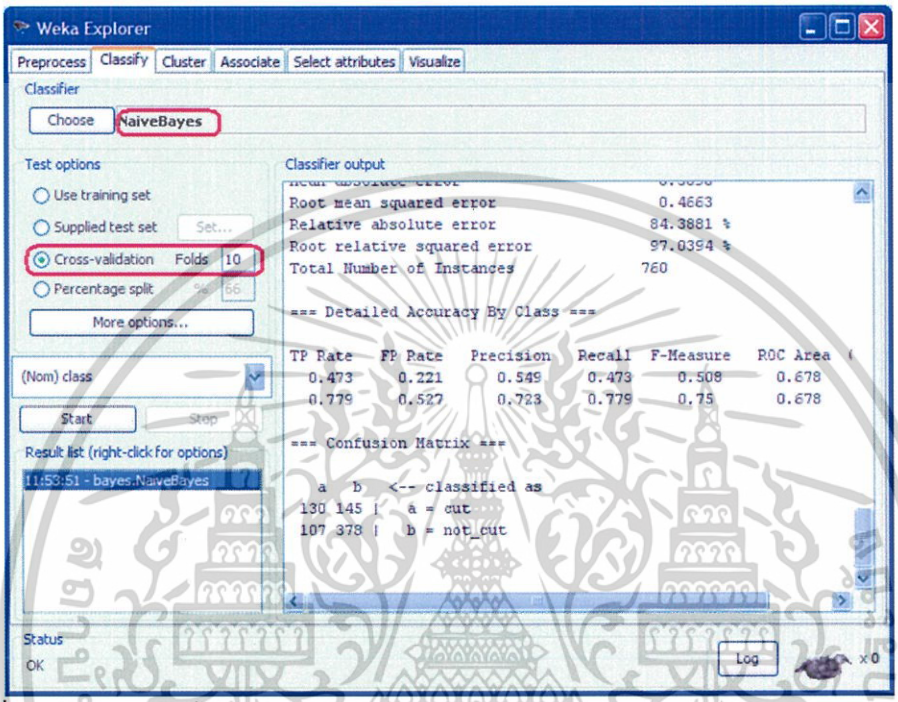
รูปที่ 3.29 แสดงหน้าต่าง หลังจากการกดปุ่ม Visualize All ของโปรแกรม WEKA

2) เลือก Classify Tab และทำการเลือกตัวจำแนก โดยจะเลือกเนอีฟเบย์



รูปที่ 3.30 แสดงการเลือกตัวจำแนก ในโปรแกรม WEKA

3) คลิกปุ่ม Start: ตัวจำแนก ที่เลือกไว้ จะทำการเรียนรู้และได้ผลลัพธ์ของการเรียนรู้ (Classifier Output)

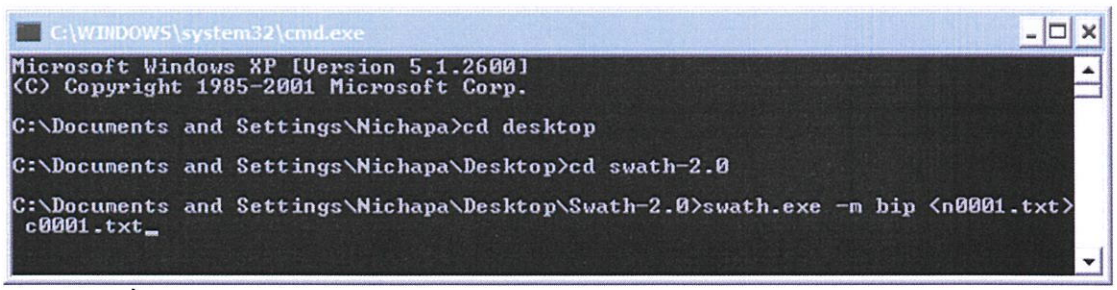


รูปที่ 3.31 แสดงการสั่งให้เกิดการเรียนรู้ของตัวจำแนก (Classifier) ในโปรแกรม WEKA

3.4.3 โปรแกรม SWATH (Smart Word Analysis for Thai)

เป็นโปรแกรมที่พัฒนาโดยหน่วยปฏิบัติการวิจัยวิทยาการมนุษยภาษา (HLT; Human Language Technology Laboratory) ศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติ (NECTEC; National Electronics and Computer Technology Center)

ใช้ในการตัดคำและบอกหน้าที่คำของคำในประโยค จะได้ผลลัพธ์เป็นไฟล์ .txt ทำงานในรูปแบบของ Command Line



รูปที่ 3.32 แสดงตัวอย่างการทำงานด้วยโปรแกรม SWATH ผ่านทาง Command Line

บทที่ 4
การทดลองเพื่อหาโมเดลที่มีประสิทธิภาพสูงสุด

ในบทนี้จะนำเสนอการทดลองแบบต่างๆ เพื่อวิเคราะห์หาโมเดลที่มีประสิทธิภาพสูงสุดในการตัดคำฟุ่มเฟือยออกจากบทความ การทดลองออกจะแบ่งออกเป็น 2 ช่วง โดยแบ่งตามขนาดของคลังข้อมูล (Corpus) คือ ช่วงที่ 1 คลังข้อมูล ประกอบไปด้วยบทความด้าน "อาหารและสุขภาพ" จำนวน 300 เอกสาร การทดลองช่วงที่ 2 มีคลังข้อมูล ประกอบไปด้วยบทความด้าน "อาหารและสุขภาพ" จำนวน 500 เอกสาร โดยแต่ละช่วงของการทดลองมีการวิเคราะห์ผลลัพธ์ที่ได้และสรุปผลการทดลองเพื่อหาโมเดลที่มีประสิทธิภาพสูงสุดในการตัดคำฟุ่มเฟือย

4.1 การทดลองช่วงที่ 1

4.1.1 เทรนนิ่ง ที่ใช้ในการทดลองช่วงที่ 1

เทรนนิ่งเซตที่ใช้ในการทดลองช่วงที่ 1 มีข้อมูลทั้งหมด 66,011 เรคคอร์ดสร้างจากคลังข้อมูลของบทความด้าน "อาหารและสุขภาพ" จำนวน 300 เอกสาร และในการทดลองช่วงที่ 1 มีการทดลอง 2 แบบ โดยแต่ละการทดลองมีการใช้เทรนนิ่งเซต ที่ลักษณะของข้อมูลที่ใช้ในการทดลอง (Attributes) แตกต่างกัน ดังนี้

ตารางที่ 4.1 แสดงลักษณะต่างๆ ของข้อมูลที่ใช้ในการทดลองในเทรนนิ่งเซตของแต่ละการทดลองในการทดลองช่วงที่ 1

Attribute	การทดลองแบบที่ 1	การทดลองแบบที่ 2
Word_id	✓	✓
Own_Pos	✓	✓
Feature	✓	✓
Pre_ Pos		✓
Post_ Pos		✓
Class	✓	✓

จากตารางที่ 4.1 สามารถอธิบายลักษณะประจำ (attribute) ที่ใช้ในเทรนนิ่งเซต (Training Set) ได้ดังนี้

1. Word_id คือ หมายเลขประจำคำหนึ่งๆ
2. Own_pos คือ หน้าที่ของคำในประโยค

3. Feature คือ ลักษณะของคำ มี 2 ลักษณะคือ Function และ Content
4. Pre_pos คือ หน้าที่ของคำที่อยู่ก่อนหน้าคำที่พิจารณา 1 ตำแหน่ง
5. Post_pos คือ หน้าที่ของคำที่อยู่หลังคำที่พิจารณา 1 ตำแหน่ง
6. Class คือ บอกว่าคำนี้ถูกตัดหรือไม่ถูกตัด (Cut, Not_cut)

4.1.1.1 ตัวอย่างเทรนนิ่งเซต (Training Set) ไฟล์ ARFF

การทดลองแบบที่ 1

```
@relation Text_Classification1
@attribute word_id numeric
@attribute
pos{NPRP,NCNM,NONM,NLBL,NCMN,NTTL,PPRS,PDMN,PNTR,PREL,VACT,VSTA,
VATT,XVBM,WVAM,XVMM,XVBB,XVAE,DDAN,DDAC,DDBQ,DDAQ,DIAC,DIBQ,DIA
Q,DCNM,DONM,ADVN,ADVI,ADVP,ADVS,CNIT,CLTV,CMTR,CFQC,CVBL,JCRG,JCMP,
JSBR,RPRE,
INT,FIXN,FIXV,EAFF,EITT,NEG,PUNC,XVAM}
@attribute feature{Content, Function}
@attribute class{cut, not_cut}

@data
1,NCMN,Content,not_cut
2,VSTA,Content,not_cut
3,NCMN,Content,not_cut
4,VACT,Content,cut
5,VATT,Content,not_cut
6,VSTA,Content,not_cut
7,NCMN,Content,not_cut
8,VACT,Content,cut
9,VATT,Content,not_cut
10,VSTA,Content,not_cut
11,NCMN,Content,not_cut
12,VACT,Content,cut
```

4.1.2 ผลการทดลองช่วงที่ 1

ผลการทดลองที่ได้ หลังจากที่นำเทรนนิ่งเซต ของการทดลองทั้ง 2 แบบไปให้ตัวจำแนก (Classifier) ทั้ง 4 ตัวใน WEKA ทำการเรียนรู้

ตารางที่ 4.2 Confusion Matrix และค่าความถูกต้องในการทำงานของเนอีฟเบย์โมเดลและเบย์เซียนเน็ตเว็ร์กโมเดลของการทดลองช่วงที่ 1 มีการทดลอง 2 แบบ

การทดลองแบบที่		Naïve Bayes			Bayesian Network		
		ตัด	ไม่ตัด	รวม	ตัด	ไม่ตัด	รวม
1	ตัด	6254	7128	**13462**	6147	7235	**13462**
	ไม่ตัด	10141	42488	**52549**	8567	44062	**52549**
	รวม	**16395**	**49616**	**66011**	**14714**	**51297**	**66011**
2	ตัด	6298	7164	**13462**	6283	7179	**13462**
	ไม่ตัด	9855	42694	**52549**	8206	44343	**52549**
	รวม	**16153**	**49858**	**66011**	14489	51522	**66011**

ตารางที่ 4.3 Confusion Matrix และค่าความถูกต้องในการทำงานแม็กซิมั่ม เอน โทรปีโมเดลซับพอตเวกเตอร์แมชชีนโมเดลของการทดลองช่วงที่ 1 มีการทดลอง 2 แบบ

การทดลองแบบที่		Maximum Entropy			Support Vector Machine		
		ตัด	ไม่ตัด	รวม	ตัด	ไม่ตัด	รวม
1	ตัด	2150	11312	**13462**	1488	11974	**13462**
	ไม่ตัด	1378	51171	**52549**	1255	51294	**52549**
	รวม	**3528**	**62483**	**66011**	**2743**	**63268**	**66011**
2	ตัด	3129	10253	**13462**	1488	11974	**13462**
	ไม่ตัด	1765	50864	**52549**	1255	51294	**52549**
	รวม	4894	61117	**66011**	**2743**	**63268**	**66011**

ตารางที่ 4.4 แสดงผลการทดลองค่าความถูกต้องในการตัดและไม่ตัดคำของโมเดลที่ใช้ในการทดลองช่วงที่ 1 แบบที่ 1

ชนิดของตัวจำแนก		Naïve Bayes	Bayesian Network	Maximum Entropy	Support Vector Machine
ตัด	Precision	0.381	0.418	0.609	0.542
	Recall	0.467	0.459	0.16	0.111
	F-Measure	0.42	0.438	0.253	0.184
ไม่ตัด	Precision	0.856	0.859	0.819	0.811
	Recall	0.807	0.837	0.974	0.976
	F-Measure	0.831	0.848	0.89	0.886

ตารางที่ 4.5 แสดงผลการทดลองค่าความถูกต้องในการตัดและไม่ตัดคำของโมเดลที่ใช้ในการทดลองช่วงที่ 1 แบบที่ 2

ชนิดของตัวจำแนก		Naïve Bayes	Bayesian Network	Maximum Entropy	Support Vector Machine
ตัด	Precision	0.39	0.434	0.639	0.542
	Recall	0.468	0.467	0.234	0.111
	F-Measure	0.425	0.439	0.342	0.184
ไม่ตัด	Precision	0.856	0.861	0.832	0.811
	Recall	0.812	0.844	0.966	0.976
	F-Measure	0.834	0.852	0.894	0.886

<u>คำอธิบาย ค่าความถูกต้องต่างๆที่ได้จากการตารางผลทดลอง</u>

1. Precision คือ อัตราส่วนที่โมเดลทำการพยากรณ์ผลการตัดคำหรือไม่ตัดคำที่ถูกต้องตรงตามข้อมูลค่าจริงที่ป้อนเข้าไปต่อข้อมูลที่โมเดลพยากรณ์ได้ทั้งหมดในผลการตัดหรือไม่ตัดคำประเภทนั้นๆ

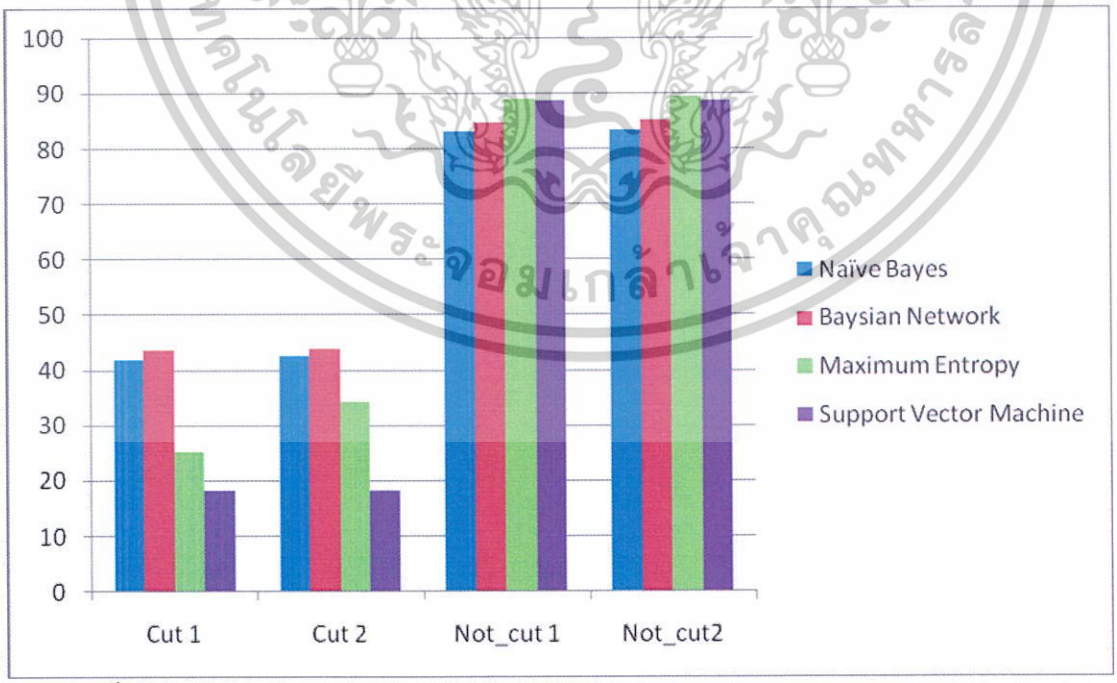
2. Recall คือ อัตราส่วนระหว่างจำนวนข้อมูลผลลัพธ์ที่ถูกต้องและจำนวนข้อมูลจริงในเทรนนิ่งเซต

3. F-Measure คือ ค่าเฉลี่ยที่เกิดจากการคำนวณค่า Precision และ Recall รวมกันเพื่อให้ได้ค่าที่บ่งบอกถึงประสิทธิภาพของการตัดคำหรือไม่ตัดคำที่ถูกต้องของโมเดลประเภทต่างๆ โดยใช้สมการ ดังนี้

$$F-measure = \frac{2 * \Pr ecision * \mathrm{Re} call}{\Pr ecision + \mathrm{Re} call} \quad (4.1)$$

4.1.3 การวิเคราะห์ผลการทดลองช่วงที่ 1

จากการทดลองช่วงที่ 1 สามารถเปรียบเทียบผลการเรียนรู้ของแมชชินเลินนิ่งได้จากค่าความถูกต้องในการทำงานของโมเดลที่ได้จากการทดลอง โดยจะดูค่า F-measure ซึ่งเป็นค่าเฉลี่ยระหว่างค่า Prediction และค่า Recall ว่าโมเดลทำการพยากรณ์การตัดคำหรือไม่ตัดคำของคำ จากเทรนนิ่งเซตได้ถูกต้องคิดเป็นสัดส่วนเปอร์เซ็นต์ ดังกราฟ ในรูปที่ 4.1



รูปที่ 4.1 แสดงสัดส่วนความถูกต้องของการตัดคำของโมเดลแต่ละประเภทจากการทดลองแบบที่ 1 และการทดลองแบบที่ 2

จากกราฟ เมื่อดูจากแท่งกราฟ cut1 และ cut2 (cut1 หมายถึงความถูกต้องของการตัดคำของการทดลองแบบที่ 1, cut2 หมายถึงความถูกต้องของการตัดคำของการทดลองแบบที่ 2) แสดงให้เห็นว่าโมเดลที่มีประสิทธิภาพในการตัดคำที่ถูกต้อง คือ เบย์เซียน เน็ตเวิร์ก (Baysian Network) รองลงมาคือ เนอีฟเบย์ (Naïve Bayes) อันดับต่อมาคือแม็กซิมั่ม เอนโทรปี (Maximum Entropy) และโมเดลที่มีประสิทธิภาพแย่ที่สุดคือ ซับพอท เวกเตอร์ แมชชีน (Support Vector Machine) ซึ่งค่อนข้างจะขัดแย้งกับทฤษฎีที่ได้ศึกษามา เพราะจากทฤษฎีกล่าวว่าโมเดลการทำงานของซับพอท เวกเตอร์ แมชชินนั้นมีประสิทธิภาพดีที่สุด รองลงมาคือแม็กซิมั่ม เอนโทรปี แต่จากผลการทดลองนั้นมีค่ากลับกัน

จากผลการทดลองแบบที่ 1 และ 2 เมื่อเราเปรียบเทียบค่าผลลัพธ์ที่ได้จากการทำงานของโมเดลจากตัวจำแนกทั้ง 4 ประเภทพบว่าผลการทดลองแบบที่ 2 ที่มีการเพิ่มตัวแปร pre_pos และ pos_pos คือการตรวจสอบหน้าที่ของคำที่อยู่ด้านหน้าและด้านหลังของคำที่สนใจอยู่นั้น มีค่าความถูกต้อง F-Measure เพิ่มขึ้นจากการทดลองแบบที่ 1 เพียงเล็กน้อย แสดงให้เห็นว่าการพิจารณาหน้าที่ของคำที่อยู่ก่อนและหลังคำที่สนใจที่ทำการทดลองเพิ่มมานั้นมีค่าไม่ส่งผลต่อประสิทธิภาพการทำงานของโมเดลเลย ซึ่งอาจจะเป็นเพราะขนาดของคลังข้อมูลที่มีขนาดเล็กและคุณสมบัติที่กำหนดให้เทรนนิ่งเซตมีน้อย หรือยังไม่เหมาะสมมากพอ ดังนั้น ในการทดลองช่วงที่ 2 จึงจะทำการเพิ่มขนาดเทรนนิ่งเซตและเพิ่มการทดลองขึ้นเพื่อทดสอบหาว่าคุณสมบัติประเภทใดของคำที่ส่งผลต่อการแยกประเภทข้อมูลมากที่สุด จึงทำการออกแบบการทดลองเพิ่มขึ้นเป็น 22 การทดลองในการทดลองช่วงที่ 2

4.2 การทดลองช่วงที่ 2

คลังข้อมูลที่นำมาใช้สร้างเทรนนิ่งเซตเกิดจากบทความด้าน “อาหารและสุขภาพ” จำนวน 500 เอกสาร (162,125 เรคคอร์ด)

4.2.1 เทรนนิ่งเซต ที่ใช้ในการทดลองช่วงที่ 2

เทรนนิ่งเซต ที่ใช้ในการทดลองช่วงที่ 2 มีข้อมูลทั้งหมดจำนวน 162,125 เรคคอร์ดโดยสร้างจากคลังข้อมูลของบทความด้าน “อาหารและสุขภาพ” จำนวน 500 เอกสาร และในการทดลองช่วงที่ 1 มีการทดลอง 22 แบบ ซึ่งแต่ละการทดลองมีการใช้เทรนนิ่งเซตที่มีลักษณะประจำต่างๆ ที่แตกต่างกัน ดังนี้

ตารางที่ 4.6 แสดงลักษณะประจำต่างๆ (attributes) ในเทรนนิ่งเซต (Training Set) ของแต่ละการทดลองในการทดลองช่วงที่ 2

Attribute	#1	#2	#3	#4	#5	#6	#7	#8	#9	#10	#11	#12	#13	#14	#15	#16	#17	#18	#19	#20	#21	#22
Word_id	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Own_pos	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓												
Feature		✓		✓		✓		✓		✓		✓		✓		✓		✓		✓		✓
Pre3_pos									✓	✓											✓	✓
Pre2_pos							✓	✓	✓	✓									✓	✓	✓	✓
Pre1_pos			✓	✓	✓	✓	✓	✓	✓	✓	✓						✓	✓	✓	✓	✓	✓
Post1_pos					✓	✓	✓	✓	✓	✓	✓						✓	✓	✓	✓	✓	✓
Post2_pos							✓	✓	✓	✓									✓	✓	✓	✓
Post3_pos									✓	✓											✓	✓
Pre3_id															✓	✓					✓	✓
Pre2_id													✓	✓	✓	✓			✓	✓	✓	✓
Pre1_id												✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Post1_id												✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Post2_id													✓	✓	✓	✓			✓	✓	✓	✓
Post3_id															✓	✓					✓	✓
Class	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

จากตารางที่ 4.6 สามารถอธิบายลักษณะประจำ ที่ใช้ในเทรนนิ่งเซต ได้ดังนี้

คำอธิบายลักษณะประจำ (attribute) ที่ใช้ในเทรนนิ่งเซต

1. Word_id คือ หมายเลขประจำคำหนึ่งๆ
2. Own_pos คือ หน้าที่ของคำในประโยค
3. Feature คือ ลักษณะของคำ มี 2 ลักษณะคือ Function และ Content
4. Pre3_pos คือ หน้าที่ของคำที่อยู่ก่อนหน้าคำที่พิจารณา 3 ตำแหน่ง
5. Pre2_pos คือ หน้าที่ของคำที่อยู่ก่อนหน้าคำที่พิจารณา 2 ตำแหน่ง
6. Pre1_pos คือ หน้าที่ของคำที่อยู่ก่อนหน้าคำที่พิจารณา 1 ตำแหน่ง
7. Post1_pos คือ หน้าที่ของคำที่อยู่หลังคำที่พิจารณา 1 ตำแหน่ง
8. Post2_pos คือ หน้าที่ของคำที่อยู่หลังคำที่พิจารณา 2 ตำแหน่ง
9. Post3_pos คือ หน้าที่ของคำที่อยู่หลังคำที่พิจารณา 3 ตำแหน่ง
10. Pre3_id คือ หมายเลขประจำของคำที่อยู่ก่อนหน้าคำที่พิจารณา 3 ตำแหน่ง
11. Pre2_id คือ หมายเลขประจำของคำที่อยู่ก่อนหน้าคำที่พิจารณา 2 ตำแหน่ง
12. Pre1_id คือ หมายเลขประจำของคำที่อยู่ก่อนหน้าคำที่พิจารณา 1 ตำแหน่ง
13. Post1_id คือ หมายเลขประจำของคำที่อยู่หลังคำที่พิจารณา 1 ตำแหน่ง
14. Post2_id คือ หมายเลขประจำของคำที่อยู่หลังคำที่พิจารณา 2 ตำแหน่ง
15. Post3_id คือ หมายเลขประจำของคำที่อยู่หลังคำที่พิจารณา 3 ตำแหน่ง
16. Class คือ บอกว่าคำนี้ถูกตัดหรือไม่ถูกตัด (Cut, Not_cut)

4.2.2 ค่าสถิติที่ได้จากการพิจารณาเทรนนิ่งเซต

จากการนำเทรนนิ่งเซตมาพิจารณาสามารถสรุปเป็นค่าสถิติของความสัมพันธ์ต่างๆ ได้ดังนี้

ตารางที่ 4.7 ตัวอย่างการแสดงค่าความถี่ของการตัดและไม่ตัดของตัวอย่างคำในเทรนนิ่งเซต

หมายเลขประจำคำ	คำ	จำนวนครั้งที่ถูกตัด	จำนวนครั้งที่ไม่ถูกตัด	% การถูกตัด	% การไม่ตัด
1	สับปะรด	2	14	12.50%	87.50%
2	เป็น	456	2095	17.88%	82.12%
3	ผลไม้	17	162	9.50%	90.50%
4	อม	3	42	6.67%	93.33%
5	หวาน	31	210	12.86%	87.14%
6	เปรี้ยว	5	75	6.25%	93.75%

ตารางที่4.7 (ต่อ) ตัวอย่างการแสดงค่าความถี่ของการตัดและไม่ตัดของตัวอย่างคำในเทรนนิ่งเซต

หมายเลขประจำคำ	คำ	จำนวนครั้งที่ถูกตัด	จำนวนครั้งที่ไม่ถูกตัด	% การถูกตัด	% การไม่ตัด
7	ที่	1672	2622	38.94%	61.06%
8	สามารถ	189	127	59.81%	40.19%
9	นำ	110	401	21.53%	78.47%
10	ไป	324	800	28.83%	71.17%
11	ทำ	192	1127	14.56%	85.44%
12	อาหาร	80	1024	7.25%	92.75%
13	ทั้ง	200	269	42.64%	57.36%
14	คาว	2	36	5.26%	94.74%
15	และ	513	1859	21.63%	78.37%
16	ได้	501	1511	24.90%	75.10%
17	อร่อย	15	76	16.48%	83.52%
18	หลาย	58	160	26.61%	73.39%
19	ชนิด	78	220	26.17%	73.83%
20	ให้	375	1964	16.03%	83.97%
21	ประโยชน์	11	178	5.82%	94.18%
22	ต่อ	39	433	8.26%	91.74%
23	สุขภาพ	11	237	4.44%	95.56%
24	ร่างกาย	43	490	8.07%	91.93%

ตารางที่ 4.8 แสดงค่าความถี่ของการตัดและไม่ตัดของหน้าที่ของคำ (POS) ในเทรนนิ่งเซต

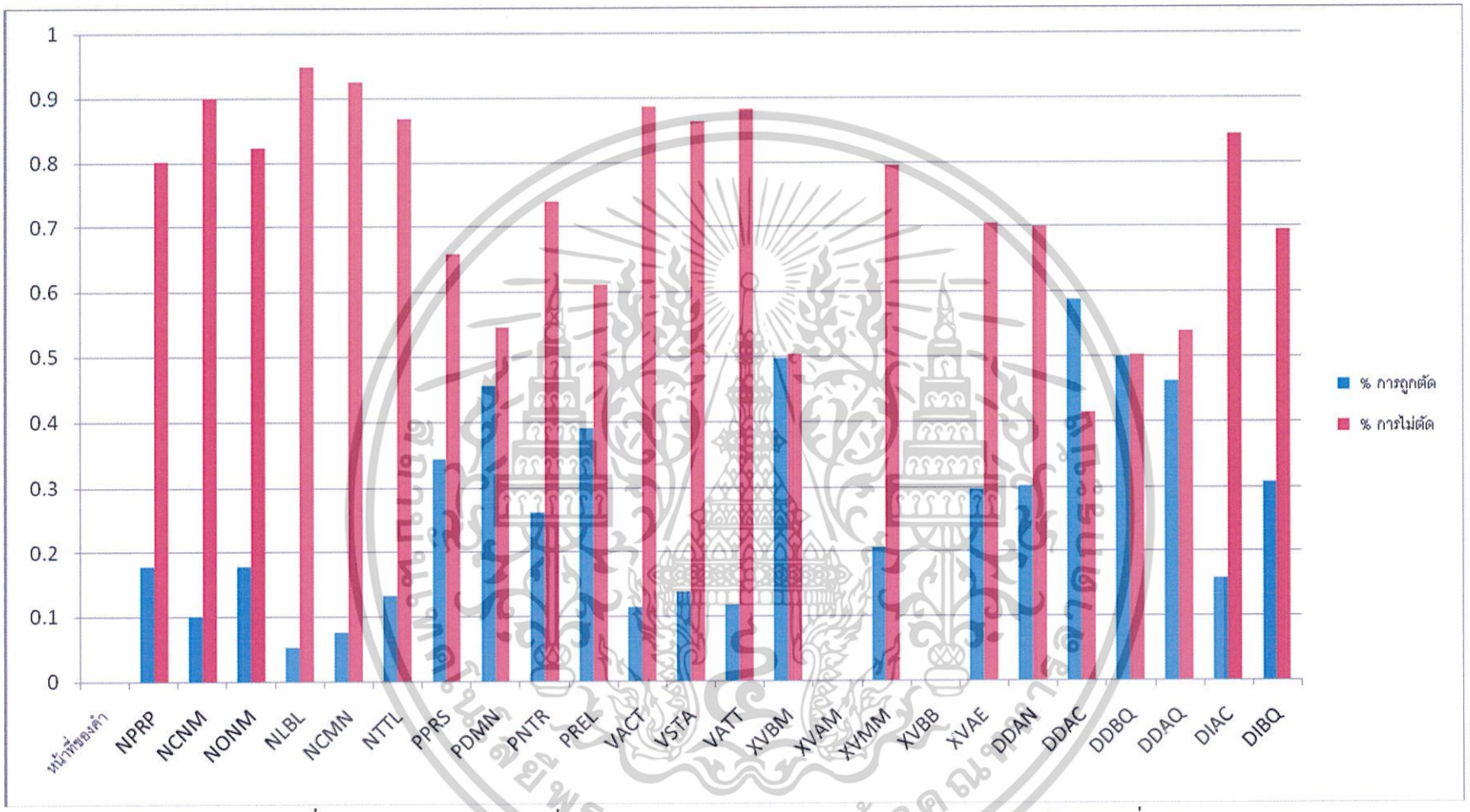
หน้าที่ของคำ	ตัวอย่างคำ	จำนวนครั้งที่ถูกตัด	จำนวนครั้งที่ไม่ถูกตัด	% การถูกตัด	% การไม่ตัด
NPRP	โคโรน่า, พระอาทิตย์,วินโดวน์ 95	2189	8804	17.65%	80.09%
NCNM	หนึ่ง, สอง, สิบ, 1, 2, 10	1	9	10.00%	90.00%
NONM	ที่หนึ่ง, ที่สอง, ที่สาม	3	14	17.65%	82.35%

ตารางที่ 4.8 (ต่อ) แสดงค่าความถี่ของการตัดและไม่ตัดของหน้าที่ของคำ (POS) ในเทรนนิ่งเซต

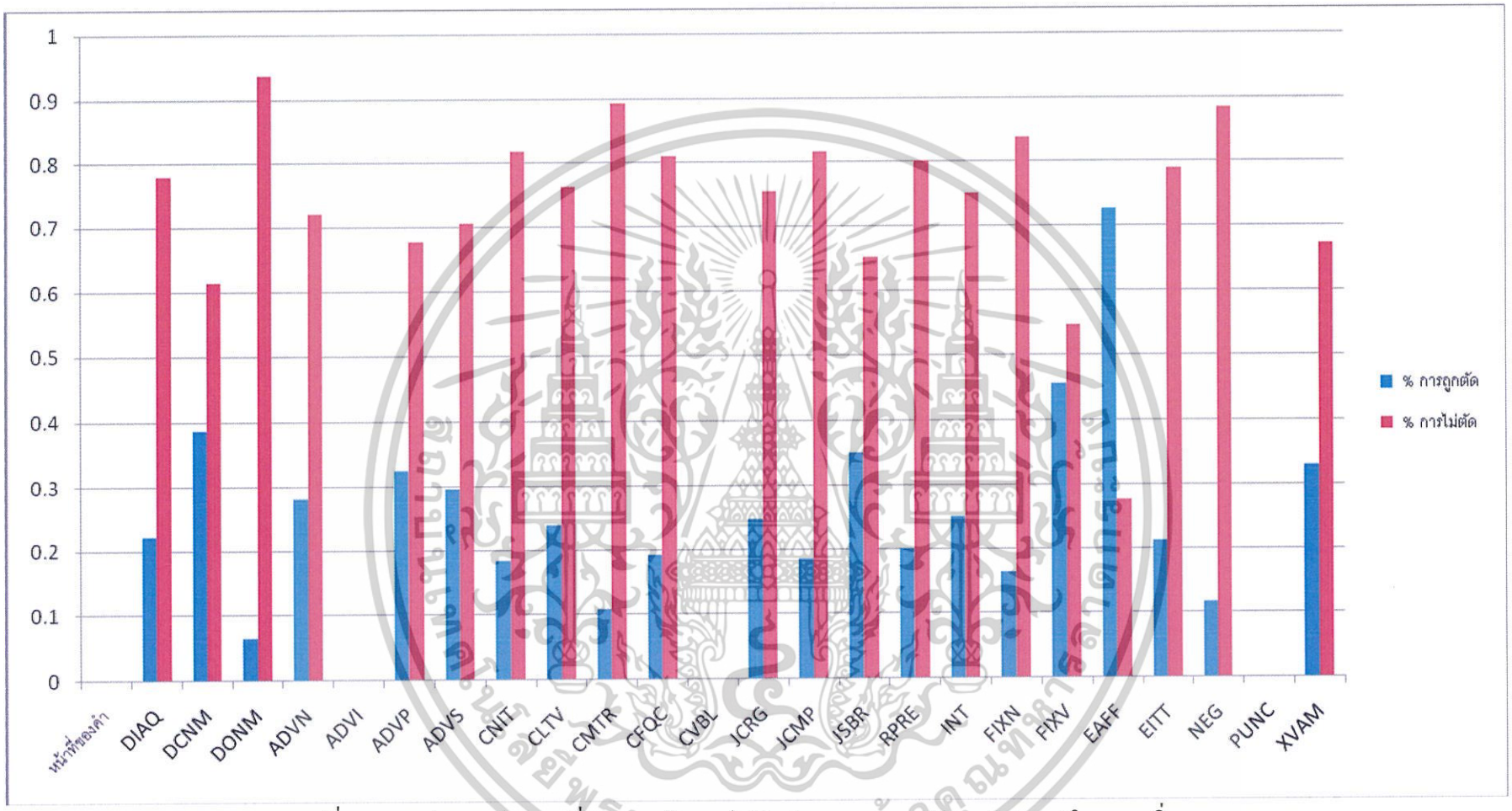
หน้าที่ของคำ	ตัวอย่างคำ	จำนวนครั้งที่ถูกตัด	จำนวนครั้งที่ไม่ถูกตัด	% การถูกตัด	% การไม่ตัด
NLBL	1, 2, 3, ก, ข, a, b	1	18	5.26%	94.74%
NCMN	หนังสือ, อาหาร, น้ำ, อากาศ	2432	29639	7.58%	92.42%
NTTL	ดร., พลเอก, นาย	33	217	13.20%	86.80%
PPRS	คุณ, เขา, ฉัน	609	1168	34.27%	65.73%
PDMN	นั่น, ที่นี่, ที่นั่น, ที่โน่น	315	377	45.52%	54.48%
PNTR	ใคร, อะไร, อย่างไร	108	307	26.02%	73.98%
PREL	ที่, ซึ่ง, อัน, ผู้	1686	2635	39.02%	60.98%
VACT	ทำงาน, เดิน, กิน	2455	19078	11.40%	88.60%
VSTA	เห็น, รู้, คือ	2333	14625	13.76%	86.24%
VATT	อ้วน, ดี,สวย	922	6905	11.78%	88.22%
XVBM	เกิด, เกือบ, กำลัง	1846	1869	49.69%	50.31%
XVAM	ค่อย, น่า, ได้	0	0	0.00%	0.00%
XVMM	ควร, เคย, ต้อง	354	1368	20.56%	79.44%
XVBB	กรุณา, อย่า, ห้าม, เชิญ	0	0	0.00%	0.00%
XVAE	ไป, มา, ขึ้น	2512	5983	29.57%	70.43%
DDAN	นี่, นั่น, โน่น, ทั้งหมด	85	197	30.14%	69.86%
DDAC	นี่, นั้น, โน้น, นู้น	869	611	58.72%	41.28%
DDBQ	ทั้ง, อีก, เพียง	607	613	49.75%	50.25%
DDAQ	พอดี, ถ้วน	30	35	46.15%	53.85%
DIAC	ไหน, อื่น, ต่างๆ	29	155	15.76%	84.24%
DIBQ	บาง, ประมาณ, เกือบ	277	630	30.54%	69.46%
DIAQ	กว่า, เศษ	2	7	22.22%	77.78%
DCNM	หนึ่งคน, เสือ 2 ตัว	168	267	38.62%	61.38%
DONM	ที่หนึ่ง, ที่สอง, ที่สุดท้าย	15	218	6.44%	93.56%

ตารางที่ 4.8 (ต่อ) แสดงค่าความถี่ของการตัดและไม่ตัดของหน้าที่ของคำ (POS) ในเทรนนิ่งเซต

หน้าที่ของคำ	ตัวอย่างคำ	จำนวนครั้งที่ถูกตัด	จำนวนครั้งที่ไม่ถูกตัด	% การถูกตัด	% การไม่ตัด
ADVN	เก่ง, เร็ว, ช้า, สม่ำเสมอ	1582	4079	27.95%	72.05%
ADVI	เร็วๆ, เสมอๆ, ช้าๆ	0	0	0.00%	0.00%
ADVP	โดยเร็ว	49	102	32.45%	67.55%
ADVS	โดยปกติ, ธรรมดา	59	141	29.50%	70.50%
CNIT	ตัว, คน, เล่ม	618	2770	18.24%	81.76%
CMTR	กิโลกรัม, แก้ว, ชั่วโมง	558	4656	10.70%	89.30%
CFQC	ครั้ง, เที่ยว	52	219	19.19%	80.81%
CVBL	ม้วน, มัด	0	0	0.00%	0.00%
JCRG	และ, หรือ, แต่	936	2866	24.62%	75.38%
JCMP	กว่า, เหมือน, เท่ากับ	108	475	18.52%	81.48%
JSBR	เพราะว่า, เนื่องจาก, ที่, แม้ว่า	3239	6056	34.85%	65.15%
RPRE	จาก, ละ, ของ, ใต้, บน	1644	6585	19.98%	80.02%
INT	โอ๊ย, โอ้, เออ, เฮ้, อ๊อ	2	6	25.00%	75.00%
FIXN	การทำงาน, ความสนุกสนาน	600	3074	16.33%	83.67%
FIXV	อย่างเร็ว	397	477	45.42%	54.58%
EAFF	จ๊ะ, ค่ะ, ครับ, นะ, น่า, เถอะ	166	63	72.49%	27.51%
EITT	หรือ, เหรอ, ไหม, มั้ย	140	523	21.12%	78.88%
NEG	ไม่, มิได้, ไม่ได้, มิ	201	1545	11.51%	88.49%
PUNC	( ), *, .., :, “ ”	0	0	0.00%	0.00%
XVAM	ถูก, น่า, สามารถ	325	661	32.96%	67.04%



รูปที่ 4.2 กราฟแสดงค่าความถี่ของการตัดและไม่ตัดคำของคุณสมบัติของคำ(POS)ในเทรนนิ่งเซต



รูปที่ 4.3 กราฟแสดงค่าความถี่ของการตัดและไม่ตัดคำของคุณสมบัติของคำ(POS)ในเทรนนิ่งเซต

4.2.3 ผลการทดลองช่วงที่ 2

นำเสนอผลการทดลองที่ได้ หลังจากที่นำเทรนนิ่งเซต ของการทดลองทั้ง 22 แบบไปให้ตัวจำแนกทั้ง 3 ตัวใน WEKA ทำการเรียนรู้

ตารางที่ 4.9 แสดงผลการทดลองค่าความถูกต้องเป็นเปอร์เซ็นต์ในการตัดคำของโมเดลที่ใช้ในการทดลองช่วงที่ 2 มีการทดลองทั้งหมด 22 รูปแบบ

ชนิดของตัวจำแนก		Naïve Bayes	Bayesian Network	Maximum Entropy
1	Precision	62.7	60.5	73
	Recall	14.4	33.4	24.5
	F-Measure	23.4	43	36.7
2	Precision	40.8	49.8	71.9
	Recall	47.2	41.4	21
	F-Measure	43.8	45.2	32.5
3	Precision	57.2	57.2	66
	Recall	30.4	30.4	21.9
	F-Measure	39.7	39.7	32.9
4	Precision	43.4	48.8	66
	Recall	45.8	42.6	21.9
	F-Measure	44.6	45.3	32.9
5	Precision	54.3	58.5	67.4
	Recall	32.1	36.8	18.6
	F-Measure	40.4	45.2	29.1
6	Precision	44.1	46.1	67.4
	Recall	42.6	44.7	18.6
	F-Measure	43.3	45.4	29.1
7	Precision	54	57.8	68
	Recall	34.6	37.6	18.5
	F-Measure	42.2	45.5	29
8	Precision	45.9	49	68
	Recall	46.3	43.8	18.5

ตารางที่ 4.9 (ต่อ) แสดงผลการทดลองค่าความถูกต้องเป็นเปอร์เซ็นต์ในการตัดคำของโมเดลที่ใช้ในการทดลองช่วงที่ 2 มีการทดลองทั้งหมด 22 รูปแบบ

ชนิดของตัวจำแนก		Naïve Bayes	Bayesian Network	Maximum Entropy
8	F-Measure	46.1	46.3	29
9	Precision	54	57.6	68
	Recall	35	37.8	18.5
	F-Measure	42.5	45.7	29
10	Precision	46.3	49.4	69
	Recall	46.3	44.1	18.5
	F-Measure	46.3	46.6	29
11	Precision	0	67.2	75.1
	Recall	0	25.8	22.2
	F-Measure	0	37.3	34.3
12	Precision	32.2	50.4	77
	Recall	0.5	35.4	24.4
	F-Measure	1.1	41.6	37.1
13	Precision	0	58.3	83.1
	Recall	0	28.5	24.4
	F-Measure	0	38.3	37.7
14	Precision	34	50.2	77.6
	Recall	1.4	36.1	24.2
	F-Measure	2.7	42	37
15	Precision	0	58.4	84
	Recall	0	28.6	25.6
	F-Measure	0	38.4	39.3
16	Precision	42.6	55.4	79.5
	Recall	7.6	35.2	26.3
	F-Measure	13	43	39.5

ตารางที่ 4.9 (ต่อ) แสดงผลการทดลองค่าความถูกต้องเป็นเปอร์เซ็นต์ในการตัดคำของโมเดลที่ใช้ในการทดลองช่วงที่ 2 มีการทดลองทั้งหมด 22 รูปแบบ

ชนิดของตัวจำแนก		Naïve Bayes	Bayesian Network	Maximum Entropy
17	Precision	48.6	59.8	80.6
	Recall	5.2	31.4	12.1
	F-Measure	9.4	41.2	21.1
18	Precision	49.3	54	76.5
	Recall	23	38.3	16
	F-Measure	31.3	44.8	26.5
19	Precision	4.7	57.9	86.7
	Recall	11.5	31.8	9.3
	F-Measure	18.5	41	16.7
20	Precision	47.1	53.6	80.5
	Recall	27.1	38.8	14.3
	F-Measure	34.4	45	24.3
21	Precision	44.8	57.4	86.4
	Recall	14.4	33.7	10
	F-Measure	21.8	42.4	18
22	Precision	46.7	52.9	82.2
	Recall	28.7	39.3	17.2
	F-Measure	35.6	45.1	28.4

ตารางที่ 4.10 แสดงผลการทดลองค่าความถูกต้องเป็นเปอร์เซ็นต์ในการไม่ตัดคำของโมเดลที่ใช้ในการทดลองช่วงที่ 2 มีการทดลองทั้งหมด 22 แบบ

ชนิดของตัวจำแนก		Naïve Bayes	Bayesian Network	Maximum Entropy
1	Precision	82.1	85.1	83.9
	Recall	97.9	94.6	97.8

ตารางที่ 4.10 (ต่อ) แสดงผลการทดลองค่าความถูกต้องเป็นเปอร์เซ็นต์ในการไม่ตัดคำของโมเดลที่ใช้ในการทดลองช่วงที่ 2 มีการทดลองทั้งหมด 22 แบบ

ชนิดของตัวจำแนก		Naïve Bayes	Bayesian Network	Maximum Entropy
1	F-Measure	89.3	89.6	90.3
2	Precision	86.3	86	83.3
	Recall	83	89.6	98
	F-Measure	84.6	87.8	90
3	Precision	84.5	84.5	83.3
	Recall	94.3	94.3	97.2
	F-Measure	89.1	98.1	89.7
4	Precision	86.3	86.2	83.3
	Recall	85.2	88.9	97.2
	F-Measure	85.7	87.5	89.7
5	Precision	84.7	85.6	82.8
	Recall	93.3	93.5	97.8
	F-Measure	88.8	89.4	89.7
6	Precision	85.8	86.4	82.8
	Recall	86.6	87	97.8
	F-Measure	86.2	86.7	89.7
7	Precision	85.1	85.7	82.8
	Recall	92.7	93.2	97.8
	F-Measure	88.7	89.3	89.7
8	Precision	86.6	86.4	82.8
	Recall	86.4	88.7	97.8
	F-Measure	86.5	87.5	89.7

ตารางที่ 4.10 (ต่อ) แสดงผลการทดลองค่าความถูกต้องเป็นเปอร์เซ็นต์ในการไม่ตัดคำของโมเดลที่ใช้ในการทดลองช่วงที่ 2 มีการทดลองทั้งหมด 22 แบบ

ชนิดของตัวจำแนก		Naïve Bayes	Bayesian Network	Maximum Entropy
10	Precision	86.6	86.5	82.8
	Recall	86.6	88.8	97.8
	F-Measure	86.6	87.6	98.7
11	Precision	80.1	84	83.5
	Recall	100	96.9	98.2
	F-Measure	88.9	90	90.3
12	Precision	81.1	85.9	83.9
	Recall	99.7	91.8	98.2
	F-Measure	89.4	88.7	90.5
13	Precision	81	85	84
	Recall	100	95.2	98.8
	F-Measure	89.5	89.9	90.8
14	Precision	81.1	86	83.9
	Recall	99.4	91.6	98.3
	F-Measure	89.3	88.7	90.5
15	Precision	81	85.1	84.2
	Recall	100	95.2	98.8
	F-Measure	89.5	89.9	90.9
16	Precision	80.9	85.2	84.3
	Recall	97.4	93	98.3
	F-Measure	88.4	88.9	90.8
17	Precision	80.7	84.7	82
	Recall	98.6	94.7	99.3
	F-Measure	88.8	89.5	89.8

ตารางที่ 4.10 (ต่อ) แสดงผลการทดลองค่าความถูกต้องเป็นเปอร์เซ็นต์ในการไม่ตัดคำของโมเดลที่ใช้ในการทดลองช่วงที่ 2 มีการทดลองทั้งหมด 22 แบบ

ชนิดของตัวจำแนก		Naïve Bayes	Bayesian Network	Maximum Entropy
18	Precision	83.1	85.7	82.5
	Recall	94.1	91.9	98.8
	F-Measure	88.3	88.7	89.9
19	Precision	81.5	84.7	81.5
	Recall	96.8	94.2	99.6
	F-Measure	88.5	89.2	89.7
20	Precision	83.6	85.8	82.3
	Recall	92.4	91.7	99.1
	F-Measure	87.8	88.6	89.9
21	Precision	81.8	85	81.7
	Recall	95.6	93.8	99.6
	F-Measure	88.1	89.2	89.7
22	Precision	83.8	85.8	82.8
	Recall	91.9	91.3	82.8
	F-Measure	87.7	88.5	99.1

ตารางที่ 4.11 Confusion Matrix และค่าความถูกต้องในการทำงานของแต่ละโมเดลของการทดลองช่วงที่ 2 มีการทดลอง 22 แบบ

รูปแบบการทดลอง		Naïve Bayes			Bayesian Network			Maximum Entropy		
		ตัด	ไม่ตัด	รวม	ตัด	ไม่ตัด	รวม	ตัด	ไม่ตัด	รวม
1	ตัด	465	2764	**3229**	1077	2152	**3229**	791	2438	**3229**
	ไม่ตัด	277	12707	**12984**	703	12281	**12984**	292	12692	**12984**
	รวม	**742**	**15471**	**16213**	**1780**	**14433**	**16213**	**1083**	**15130**	**16213**
2	ตัด	1524	1705	**3229**	1338	1891	**3229**	677	2552	**3229**
	ไม่ตัด	2213	10771	**12984**	1350	11634	**12984**	264	12720	**12984**
	รวม	**3737**	**12476**	**16213**	**2688**	**13525**	**16213**	**941**	**15272**	**16213**
3	ตัด	981	2248	**3229**	981	2248	**3229**	707	2522	**3229**
	ไม่ตัด	735	12249	**12984**	735	12249	**12984**	364	12620	**12984**
	รวม	**1716**	**14497**	**16213**	**1716**	**14497**	**16213**	**1071**	**15142**	**16213**
4	ตัด	1479	1750	**3229**	1377	1852	**3229**	707	2522	**3229**
	ไม่ตัด	1926	11058	**12984**	1443	11541	**12984**	364	12620	**12984**
	รวม	3405	**12808**	**16213**	**2820**	**13393**	**16213**	**1071**	**15142**	**16213**
5	ตัด	1038	2191	**13462**	1188	2041	**3229**	599	2630	**3229**
	ไม่ตัด	873	12111	**52549**	844	12140	**12984**	290	12694	**12984**

ตารางที่ 4.11 (ต่อ) Confusion Matrix และค่าความถูกต้องในการทำงานของแต่ละโมเดลของการทดลองช่วงที่ 2 มีการทดลอง 22 แบบ

รูปแบบการทดลอง		Naïve Bayes			Bayesian Network			Maximum Entropy		
		ตัด	ไม่ตัด	รวม	ตัด	ไม่ตัด	รวม	ตัด	ไม่ตัด	รวม
5	รวม	**1911**	**14302**	**66011**	**2032**	**14181**	**16213**	**889**	**15324**	**16213**
6	ตัด	1375	1854	**3229**	1443	1786	**3229**	599	2630	**3229**
	ไม่ตัด	1742	11242	**12984**	1685	11299	**12984**	290	12694	**12984**
	รวม	**3117**	**13096**	**16213**	**3128**	**13085**	**16213**	**889**	**15324**	**16213**
7	ตัด	1118	2111	**3229**	1213	2016	**3229**	596	2633	**3229**
	ไม่ตัด	954	12030	**12984**	885	12099	**12984**	280	12704	**12984**
	รวม	**2072**	**14141**	**16213**	**2098**	**14115**	**16213**	**876**	**15337**	**16213**
8	ตัด	1496	1733	**3229**	1414	1815	**3229**	596	2633	**3229**
	ไม่ตัด	1762	11222	**12984**	1469	11515	**12984**	280	12740	**12984**
	รวม	**3258**	**12955**	**16213**	**2883**	**13330**	**16213**	**876**	**15337**	**16213**
9	ตัด	1129	2100	**3229**	1221	2008	**3229**	596	2633	**3229**
	ไม่ตัด	961	12023	**12984**	898	12086	**12984**	280	12704	**12984**
	รวม	**2090**	**14123**	**16213**	**2119**	**14094**	**16213**	**876**	**15337**	**16213**
10	ตัด	1496	1733	**3229**	1425	1804	**3229**	596	2633	**3229**

ตารางที่ 4.11 (ต่อ) Confusion Matrix และค่าความถูกต้องในการทำงานของแต่ละโมเดลของการทดลองช่วงที่ 2 มีการทดลอง 22 แบบ

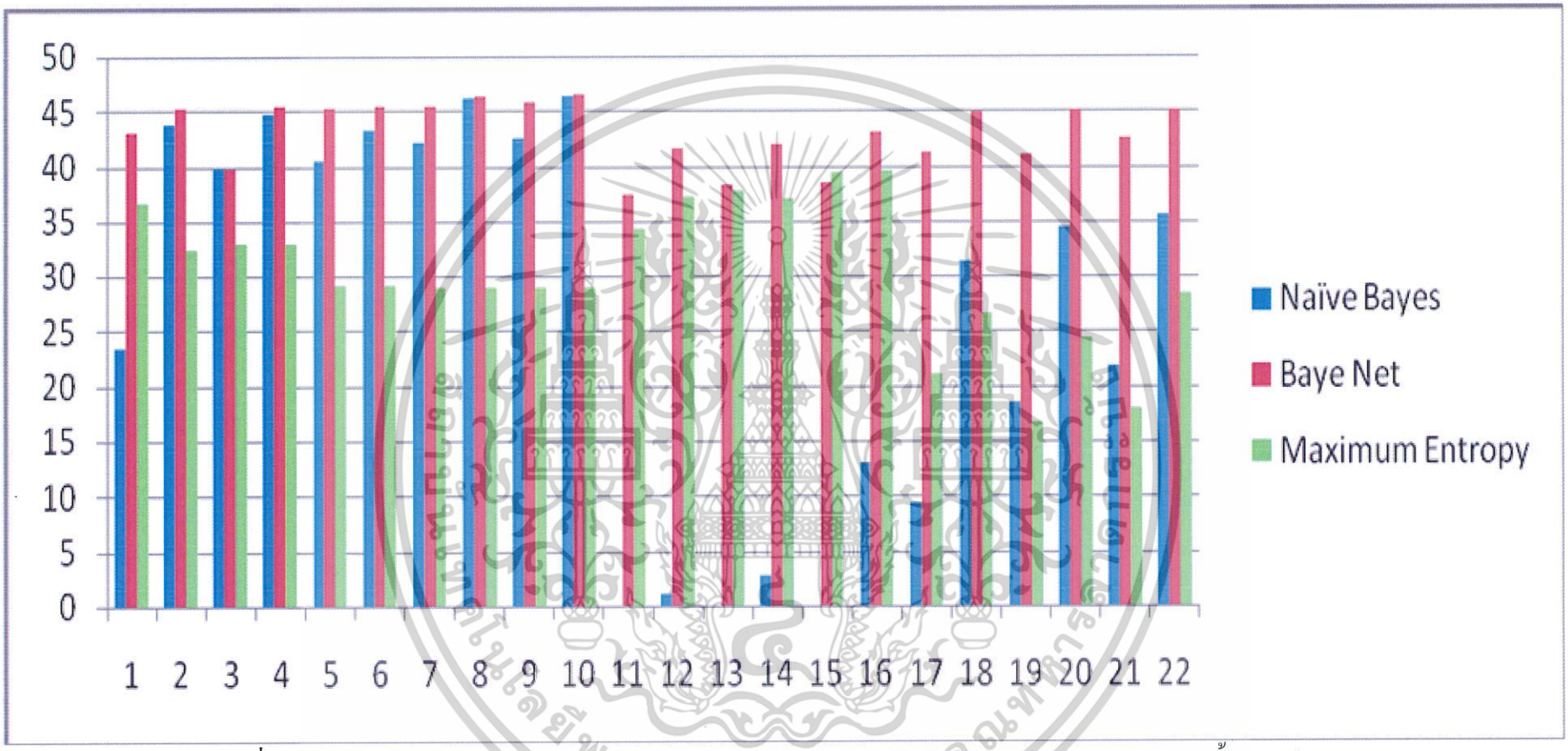
รูปแบบการทดลอง		Naïve Bayes			Bayesian Network			Maximum Entropy		
		ตัด	ไม่ตัด	รวม	ตัด	ไม่ตัด	รวม	ตัด	ไม่ตัด	รวม
10	ไม่ตัด	1738	11246	12984	1457	11527	12984	280	12704	12984
	รวม	3234	12979	16213	2882	13331	16213	876	15337	16213
11	ตัด	0	3229	3229	834	2395	3228	718	2511	3229
	ไม่ตัด	0	12984	12984	407	12577	12984	238	12746	12984
	รวม	0	16213	16213	1241	14972	16213	956	15257	16213
12	ตัด	149	27542	27691	9813	17878	27691	788	2441	3299
	ไม่ตัด	314	117907	118221	9654	108567	118221	235	12749	12984
	รวม	463	15750	145912	19467	126445	145912	1023	15190	16213
13	ตัด	0	27691	27691	7886	19805	27691	787	2442	3229
	ไม่ตัด	0	118221	118221	5632	112589	118221	160	12824	12984
	รวม	0	145912	145912	13518	132394	145912	947	15266	16213
14	ตัด	383	27308	27691	10001	17690	27691	783	2446	3229
	ไม่ตัด	744	117477	118221	9903	108318	118221	226	12758	12984
	รวม	1127	144785	145912	19904	116008	145912	1009	15204	16213

ตารางที่ 4.11 (ต่อ) Confusion Matrix และค่าความถูกต้องในการทำงานของแต่ละโมเดลของการทดลองช่วงที่ 2 มีการทดลอง 22 แบบ

รูปแบบการทดลอง		Naïve Bayes			Bayesian Network			Maximum Entropy		
		ตัด	ไม่ตัด	รวม	ตัด	ไม่ตัด	รวม	ตัด	ไม่ตัด	รวม
15	ตัด	0	27691	**27691**	7920	19771	**27691**	827	2402	**3229**
	ไม่ตัด	0	118221	**118221**	5633	112588	**118221**	158	12825	**12984**
	รวม	**0**	**145912**	**145912**	**13553**	**132359**	**145912**	**985**	**15228**	**16213**
16	ตัด	247	2982	**3229**	1135	2094	**3229**	848	2381	**3229**
	ไม่ตัด	333	12651	**12984**	915	12069	**12984**	219	12765	**12984**
	รวม	**580**	**15633**	**16213**	**2050**	**14163**	**16213**	**1067**	**15146**	**16213**
17	ตัด	168	3061	**3229**	1015	2214	**3229**	391	2838	**3229**
	ไม่ตัด	178	12806	**12984**	682	12302	**12984**	94	12890	**12984**
	รวม	**346**	**15867**	**16213**	**1697**	**14516**	**16213**	**485**	**15728**	**16213**
18	ตัด	742	2487	**3229**	1237	1992	**3229**	517	2712	**3229**
	ไม่ตัด	763	12221	**12984**	1053	11931	**12984**	159	12825	**12984**
	รวม	**1505**	**14708**	**16213**	**2290**	**13923**	**16213**	**676**	**15537**	**16213**
19	ตัด	372	2857	**3229**	1026	2203	**3229**	299	2930	**3229**
	ไม่ตัด	420	12564	**12984**	747	12237	**12984**	446	12938	**12984**

ตารางที่ 4.11 (ต่อ) Confusion Matrix และค่าความถูกต้องในการทำงานของแต่ละโมเดลของการทดลองช่วงที่ 2 มีการทดลอง 22 แบบ

รูปแบบการทดลอง		Naïve Bayes			Bayesian Network			Maximum Entropy		
		ตัด	ไม่ตัด	รวม	ตัด	ไม่ตัด	รวม	ตัด	ไม่ตัด	รวม
19	**รวม**	**792**	**15421**	**16213**	**1773**	**14440**	**16312**	**745**	**15468**	**16213**
20	**ตัด**	876	2353	**3229**	1252	1977	**3229**	463	2766	**3229**
	ไม่ตัด	982	12002	**12984**	1084	11900	**12984**	112	12872	**12984**
	รวม	**1858**	**14355**	**16213**	**2336**	**13877**	**16213**	**575**	**15638**	**16213**
21	**ตัด**	466	2763	**3229**	1087	2142	**3229**	324	2905	**3229**
	ไม่ตัด	575	12409	**12984**	807	12177	**12984**	51	12933	**12984**
	รวม	**1041**	**15172**	**16213**	**1894**	**14319**	**16213**	**375**	**15838**	**16213**
22	**ตัด**	927	2302	**3229**	1268	1961	**3229**	554	2675	**3229**
	ไม่ตัด	1057	11927	**12984**	1128	11856	**12984**	120	12864	**12984**
	รวม	**1984**	**14229**	**16213**	**2396**	**13817**	**16213**	**674**	**15539**	**16213**



รูปที่ 4.4 กราฟแสดงผลการทดลองเปรียบเทียบประสิทธิภาพการตัดคำจากค่า F-Measure ของการทดลองทั้ง 22 รูปแบบ

จากผลการทดลองให้เห็นว่าประสิทธิภาพของเบย์เซียน เน็ตเวิร์กโมเดลมีค่าความถูกต้องโดยเฉลี่ยสูงที่สุด รองลงมาคือ เนอีฟเบย์ และโมเดลแม็กซิมั่ม เอนโทรปีซึ่งมีค่าความถูกต้องเฉลี่ยต่ำที่สุด เมื่อพิจารณาที่ค่า F-Measure แต่ละโมเดล จะได้โมเดลที่มีค่า F-Measure ในการทดลองของแต่ละโมเดลสูงสุด ดังนี้

เบย์เซียน เน็ตเวิร์กโมเดลการทดลองที่ 10 มีค่า F-Measure ที่ 46.6 ซึ่งมีคุณสมบัติต่างๆ (Attributes) ที่ใช้ในการทดลอง ดังนี้

1. Word_id (หมายเลขประจำคำหนึ่งๆ)
2. Own_pos (หน้าที่ของคำในประโยค)
3. Feature (ลักษณะของคำ มี 2 ลักษณะคือ Function และ Content)
4. Pre3_pos (หน้าที่ของคำที่อยู่ก่อนหน้าคำที่พิจารณา 3 ตำแหน่ง)
5. Pre2_pos (หน้าที่ของคำที่อยู่ก่อนหน้าคำที่พิจารณา 2 ตำแหน่ง)
6. Pre1_pos (หน้าที่ของคำที่อยู่ก่อนหน้าคำที่พิจารณา 1 ตำแหน่ง)
7. Post1_pos (หน้าที่ของคำที่อยู่หลังคำที่พิจารณา 1 ตำแหน่ง)
8. Post2_pos (หน้าที่ของคำที่อยู่หลังคำที่พิจารณา 2 ตำแหน่ง)
9. Post3_pos (หน้าที่ของคำที่อยู่หลังคำที่พิจารณา 3 ตำแหน่ง)
10. Class (บอกว่าคำนี้ถูกตัดหรือไม่ถูกตัด (Cut, Not_cut)

โมเดลสามารถพยากรณ์การตัดของคำในเทรนนิ่งเซตได้ทั้งหมด 2882 คำ มีความถูกต้องจำนวน 1425 คำ และพยากรณ์การไม่ตัดของคำได้ทั้งหมด 13331 คำ มีจำนวนที่ถูกต้อง 11527 คำ ค่าการพยากรณ์ที่ได้นี้มีค่าความถูกต้องสูงสุดเมื่อเทียบกับโมเดลอื่นๆ ในการทดลองของโมเดลประเภทเดียวกัน

เนอีฟเบย์โมเดลการทดลองที่ 10 มีค่า F-Measure ที่ 46.3 ซึ่งมีคุณสมบัติต่างๆ ที่ใช้ในการทดลอง ดังนี้

1. Word_id (หมายเลขประจำคำหนึ่งๆ)
2. Own_pos (หน้าที่ของคำในประโยค)
3. Feature (ลักษณะของคำ มี 2 ลักษณะคือ Function และ Content)
4. Pre3_pos (หน้าที่ของคำที่อยู่ก่อนหน้าคำที่พิจารณา 3 ตำแหน่ง)
5. Pre2_pos (หน้าที่ของคำที่อยู่ก่อนหน้าคำที่พิจารณา 2 ตำแหน่ง)
6. Pre1_pos (หน้าที่ของคำที่อยู่ก่อนหน้าคำที่พิจารณา 1 ตำแหน่ง)
7. Post1_pos (หน้าที่ของคำที่อยู่หลังคำที่พิจารณา 1 ตำแหน่ง)

8. Post2_pos (หน้าที่ของคำที่อยู่หลังคำที่พิจารณา 2 ตำแหน่ง)
9. Post3_pos (หน้าที่ของคำที่อยู่หลังคำที่พิจารณา 3 ตำแหน่ง)
10. Class (บอกว่าคำนี้ถูกตัดหรือไม่ถูกตัด (Cut, Not cut)

โมเดลสามารถพยากรณ์การตัดของคำในเทรนนิ่งเซตได้ทั้งหมด 3234 คำ พยากรณ์ถูกต้องจำนวน 1496 คำ และพยากรณ์การไม่ตัดของคำได้ทั้งหมด 12979 คำ มีจำนวนที่ถูกต้องทั้งหมด 11246 คำ ค่าการพยากรณ์ที่ได้นี้มีค่าความถูกต้องสูงสุดเมื่อเทียบกับโมเดลอื่นๆ ในการทดลองของโมเดลประเภทเดียวกัน

แม็กซิมัมเอนโทรปีโมเดลการทดลองที่ 16 มีค่า F-Measure ที่ 39.5 ซึ่งมีคุณสมบัติต่างๆ (Attributes) ที่ใช้ในการทดลอง ดังนี้

1. Word_id (หมายเลขประจำคำหนึ่งๆ)
2. Feature (ลักษณะของคำ มี 2 ลักษณะคือ Function และ Content)
3. Pre3_id (หมายเลขประจำของคำที่อยู่ก่อนหน้าคำที่พิจารณา 3 ตำแหน่ง)
4. Pre2_id (หมายเลขประจำของคำที่อยู่ก่อนหน้าคำที่พิจารณา 2 ตำแหน่ง)
5. Pre1_id (หมายเลขประจำของคำที่อยู่ก่อนหน้าคำที่พิจารณา 1 ตำแหน่ง)
6. Post1_id (หมายเลขประจำของคำที่อยู่หลังคำที่พิจารณา 1 ตำแหน่ง)
7. Post2_id (หมายเลขประจำของคำที่อยู่หลังคำที่พิจารณา 2 ตำแหน่ง)
8. Post3_id (หมายเลขประจำของคำที่อยู่หลังคำที่พิจารณา 3 ตำแหน่ง)
9. Class (บอกว่าคำนี้ถูกตัดหรือไม่ถูกตัด (Cut, Not_cut)

โมเดลสามารถพยากรณ์การตัดของคำในเทรนนิ่งเซตได้ทั้งหมด 1067 คำ ซึ่งพยากรณ์ถูกต้องจำนวน 848 คำ และพยากรณ์การไม่ตัดของคำได้ทั่งหมด 15146 เป็นการพยากรณ์ที่ถูกต้องจำนวน 12765 คำ ค่าการพยากรณ์ที่ได้นี้มีค่าความถูกต้องสูงสุดเมื่อเทียบกับโมเดลอื่นๆ ในการทดลองของโมเดลประเภทเดียวกัน

จากรูปที่ 4.4 พบว่าเนอีฟเบย์โมเดลและเบย์เซียน เน็ตเวิร์กโมเดลมีคุณสมบัติต่างๆ ที่มีผลต่อประสิทธิภาพของการทำงานของโมเดลที่เหมือนกัน (ในการทดลองที่ 10) แต่สำหรับแม็กซิมัมเอนโทรปีนั้น มีคุณสมบัติต่างๆ ที่มีผลต่อประสิทธิภาพของการทำงานของโมเดลต่างกับ 2 โมเดลข้างต้น โดยในแม็กซิมัมเอนโทรปี ค่าเลขประจำตัวของคำที่อยู่บริเวณข้างเคียงคำที่สนใจจะเป็นตัวที่ส่งผลต่อประสิทธิภาพของโมเดล แต่ในเนอีฟเบย์โมเดลและเบย์เซียน เน็ตเวิร์กโมเดลค่าของหน้าที่ข้างคำที่อยู่บริเวณข้างเคียงคำที่สนใจจะเป็นตัวที่ส่งผลต่อประสิทธิภาพของโมเดล

และเมื่อพิจารณาค่า ความถูกต้องของการตัดคำที่ได้จากค่า F-measure แม็กซิมัมเอนโทรปีก็ยังมีประสิทธิภาพที่ต่ำกว่าเนอีฟเบย์โมเดลและเบย์เซียน เน็ตเวิร์กโมเดลในการทดลองอื่นๆอยู่

ดังนั้นจึงกล่าวได้ว่า คุณสมบัติประจำตัวของคำที่สนใจและคุณสมบัติประจำตัวของคำข้างเคียงคำที่สนใจส่งผลกระทบต่อประสิทธิภาพการตัดคำของโมเดลมากกว่าค่าเลขประจำตัวของคำที่อยู่บริเวณข้างเคียงคำที่สนใจ

จากการเปรียบเทียบการทดลองทั้งหมด พบว่า คุณสมบัติของคำที่มีผลต่อประสิทธิภาพการตัดคำของโมเดล คือ

1. ชนิดของคำในประโยคของคำที่สนใจ
2. คุณสมบัติของคำ
3. ชนิดของคำในประโยคของคำข้างเคียงคำที่สนใจ โดยยิ่งมาค่าชนิดของคำในประโยคของคำข้างเคียงมากขึ้นก็ยิ่งทำให้ประสิทธิภาพในการตัดคำดีขึ้น

4.3 การทดลองเพื่อเปรียบเทียบประสิทธิภาพของโมเดลที่ดีที่สุดกับเบสไลน์

การทดลองเปรียบเทียบประสิทธิภาพของโมเดลกับเบสไลน์ เพื่อทดสอบประสิทธิภาพของโมเดลที่ได้จากการทดลองว่ามีประสิทธิภาพในการตัดคำดีพอ เมื่อเปรียบเทียบกับการผลการทำงานจริงหรือไม่ ทำการทดลองโดยนำโมเดลแต่ละโมเดลที่มีประสิทธิภาพสูงสุดที่ได้จากการทดลอง 4.2 ได้แก่ โมเดลของเบย์เซียน เน็ตเวิร์ก ในการทดลองที่ 10, เนอีฟเบย์โมเดลการทดลองที่ 10, แม็กซิมัมเอนโทรปีโมเดลการทดลองที่ 16 มาทำการทดสอบเปรียบเทียบประสิทธิภาพการทำงานกับเบสไลน์

ซึ่งการทดสอบนั้น ได้นำข้อมูลทั้งหมด 162,125 เรคคอร์ด มาแบ่งเป็นเทรนนิ่งเซต เพื่อให้เกิดการเรียนรู้ จำนวน 90% (145,912 เรคคอร์ด) และเป็นเซตข้อมูลที่ใช้ทดสอบ เพื่อทำการทดสอบประสิทธิภาพของการตัด จำนวน 10% (16,213 เรคคอร์ด) โดยใช้วิธีการ 10% Cross-Validation ในการสร้างและทดสอบประสิทธิภาพของโมเดล

4.3.1 เบสไลน์

เบสไลน์ คือ โมเดลที่เกิดขึ้นจากการพิจารณาค่าความถี่ที่มากกว่าของการตัดและไม่ตัดของคำในเทรนนิ่งเซต

4.3.2 ผลการเปรียบเทียบ

การทดลองดังกล่าวให้ผลของการทดลอง ดังต่อไปนี้

ตารางที่ 4.12 ตารางเปรียบเทียบค่าความถูกต้องในการตัดคำของโมเดลและ Baseline

	Precision	**Recall**	**F-Measure**
โมเดลเบย์เซียน เน็ตเวิร์กการทดลองที่ 10	49.4	44.1	**46.6**
โมเดลเนอีฟเบย์การทดลองที่ 10	46.3	46.3	**46.3**
โมเดลแม็กซิมัมเอนโทรปี การทดลองที่ 16	79.5	26.3	**39.5**
เบสไลน์	39.6	16.4	**23.2**

จากตารางจะเห็นว่า เมื่อเปรียบเทียบค่า Precision, Recall และค่า F-Measure ระหว่างโมเดลของเบย์เซียน เน็ตเวิร์ก ในการทดลองที่ 10, เนอีฟเบย์โมเดลการทดลองที่ 10, แม็กซิมัมเอนโทรปีโมเดลการทดลองที่ 16 และเบสไลน์ เมื่อพิจารณาถึงค่า Precision ของแต่ละโมเดลกับ เบสไลน์ พบว่ามีค่าสูงกว่าเบสไลน์ แสดงให้เห็นว่าความถูกต้องในการพยากรณ์การตัดคำของโมเดลสูงกว่าความผิดพลาดในการพยากรณ์การตัดคำของโมเดล สำหรับค่า Recall พบว่าทุกๆ โมเดลมีค่า Recall สูงกว่าของเบสไลน์ แสดงให้เห็นว่าอัตราส่วนของการตัดคำของโมเดลที่ถูกต้องเมื่อเทียบกับคำที่ถูกตัดมีในเทรนนิ่งเซตมีค่าค่อนข้างสูง และเมื่อเปรียบเทียบประสิทธิภาพการตัดคำของโมเดลทั้ง 3 โดยรวมจากค่า F-Measure กับเบสไลน์ พบว่าโมเดลมีค่าเฉลี่ยสูงกว่าค่าของเบสไลน์ จึงสรุปได้ว่าโมเดล ทั้ง 3 ชนิดนี้มีประสิทธิภาพในการตัดคำถูกต้องน่าเชื่อถือสูงกว่าความน่าเชื่อถือพื้นฐานของการตัดคำ

4.4 การทดสอบประสิทธิภาพการทำงานระบบ

เนื่องจากการทดสอบประสิทธิภาพการทำงานของโมเดลว่ามีการทำงานที่สอดคล้องเหมือนกับการทำงานจริงของคนหรือไม่นั้น ไม่มีหลักเกณฑ์ตายตัวมาวัดได้ว่าโมเดลชนิดใดที่ทำงานได้ผลลัพธ์น่าพึงพอใจตรงตามความต้องการของมนุษย์มากที่สุด เพราะโครงสร้างทางภาษาไทยไม่มีโครงสร้างที่ตายตัว การมองเห็นคำฟุ่มเฟือยของคนแต่ละคนแตกต่างกัน ทำให้การตัดคำแตกต่างกันไปด้วย ดังนั้น การทดสอบประสิทธิภาพสามารถทำได้โดยการนำผลลัพธ์จากการตัดคำของโมเดลชนิดต่างๆ มาเปรียบเทียบกับบทความที่มนุษย์ทำการตัดไว้ เพื่อหาค่าความถูกต้องและเปรียบเทียบเวลาในการประมวลผลของแต่ละโมเดลเพื่อหาโมเดลที่ใช้เวลาในการประมวลผลน้อยที่สุด

การทดสอบประสิทธิภาพของโมเดลจะนำโมเดลทั้ง 3 ชนิดมาทำการทดสอบ คือ เบย์เซียนเน็ตเวิร์ก โมเดลการทดลองที่ 10 เนอีฟเบย์โมเดลการทดลองที่ 10 และแม็กซิมัมเอนโทรปีโมเดลการทดลองที่ 16

4.4.1 การทดสอบประสิทธิภาพการทำงานของโมเดล

ทดสอบโดยนำบทความด้าน “อาหารและสุขภาพ” ที่เป็นบทความทั้งหมดที่ยังไม่ได้รับการตัดโดยมนุษย์มาทำการป้อนให้ระบบทำการตัดคำอัตโนมัติ ต่อจากนั้นนำผลลัพธ์มาเปรียบเทียบประสิทธิภาพความถูกต้องกับบทความทั้งหมดที่มนุษย์ทำการตัดเพื่อหาประสิทธิภาพค่าถูกต้องของทั้ง 3 โมเดล

4.4.1.2 ตัวอย่างการทำงานของระบบ

ตัวอย่างผลลัพธ์การทำงานของระบบโดยใช้ โมเดลทั้ง 3 ชนิดในการตัดคำ

<u>ตัวอย่างบทความ</u>

สาหร่ายผมนาง" คุณค่าจากท้องทะเล

เมื่อเร็วๆนี้ "108 เคล็ดกิน" เพิ่งได้ไปเยือนเกาะยอ ที่จังหวัดสงขลามา และได้ไปลองลิ้มชิมรสอาหารพื้นบ้านขึ้นชื่อของชาวเกาะยออย่าง "ยำสาหร่ายผมนาง" ที่มีรสชาติเอร็ดอร่อย และประโยชน์ของสาหร่ายผมนางนั้นก็มีมากจนต้องขอนำมาบอกต่อเสียหน่อย

"สาหร่ายผมนาง" นั้น คนเกาะยอเรียกกันสั้นๆ ว่า "สาย" มีลักษณะทัลลัสกลมแข็ง กรอบ อวบน้ำ แตกแขนงอิสระ โดยแขนงแตกออกจากแกนกลาง ต้นพุ่มขนาดใหญ่มีส่วนคล้ายรากเกาะกับหินหรือเปลือกหอยมองดูคล้ายเส้นผมปลิวอยู่ในน้ำ จึงมีชื่อว่า "สาหร่ายผมนาง"

สาหร่ายชนิดนี้ถือเป็นพืชชนิดหนึ่งที่มีคุณค่าทางโภชนาการสูง โดยคุณค่าทางอาหารที่ได้จากสาหร่ายผมนางนั้นก็เช่น โปรตีน คาร์โบไฮเดรต เกลือแร่ โดยเฉพาะธาตุไอโอดีนและวิตามินต่างๆ อีกทั้งยังช่วยป้องกันโรคหลายอย่าง เช่น เบาหวาน ความดันโลหิตสูง โรคตับ โรคโลหิตจาง และโรคมะเร็ง เป็นต้น และไม่เพียงแต่ยำสาหร่ายผมนางเท่านั้นที่กินได้อร่อย แต่สามารถนำมาทำยำ แกงจืด ก็กินได้อร่อยด้วยเช่นกัน

บทความที่ถูกตัดโดยมนุษย์

สาหร่ายผมนาง" คุณค่าจากท้องทะเล

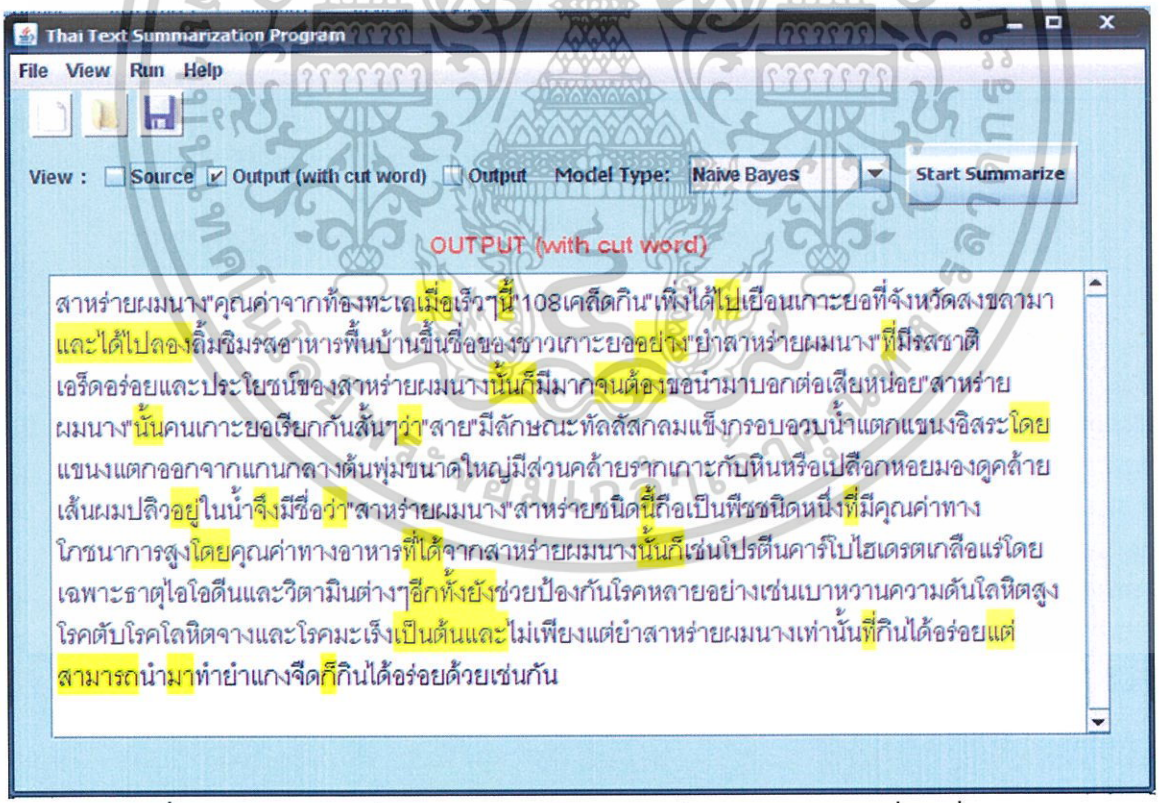
เมื่อเร็วๆนี้ "108 เคล็ดกิน" เพิ่งได้ไปเยือนเกาะยอ ที่จังหวัดสงขลามา และได้ไปลองลิ้มชิมรสอาหารพื้นบ้านขึ้นชื่อของชาวเกาะยอออย่าง "ยำสาหร่ายผมนาง" ที่มีรสชาติเอร็ดอร่อย และประโยชน์ของสาหร่ายผมนางนั้นก็มีมากจนต้องขอนำมาบอกต่อเสียหน่อย

"สาหร่ายผมนาง" นั้น คนเกาะยอเรียกกันสั้นๆ ว่า "สาย" มีลักษณะทัลลัสกลมแข็ง กรอบ อวบน้ำ แตกแขนงอิสระ โดยแขนงแตกออกจากแกนกลาง ต้นพุ่มขนาดใหญ่มีส่วนคล้ายรากเกาะกับหินหรือเปลือกหอย มองดูคล้ายเส้นผมปลิวอยู่ในน้ำ จึงมีชื่อว่า "สาหร่ายผมนาง"

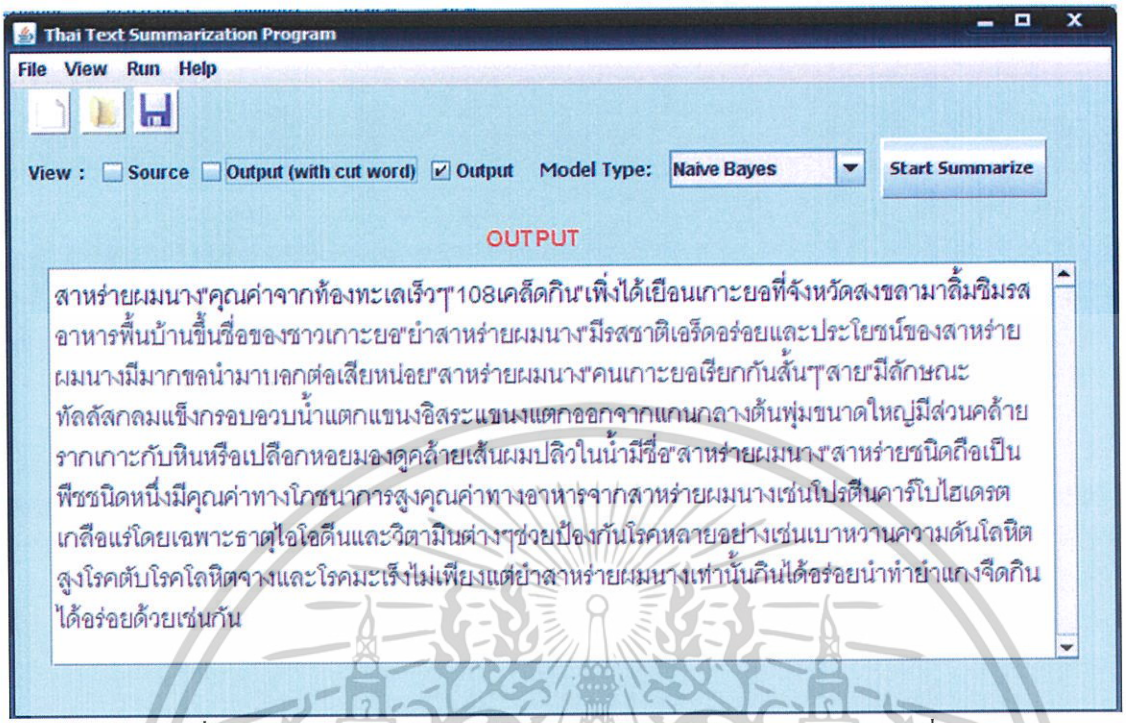
สาหร่ายชนิดนี้ถือเป็นพืชชนิดหนึ่งที่มีคุณค่าทางโภชนาการสูง โดยคุณค่าทางอาหารที่ได้จากสาหร่ายผมนางนั้นก็เช่น โปรตีน คาร์โบไฮเดรต เกลือแร่ โดยเฉพาะธาตุไอโอดีนและวิตามินต่างๆ อีกทั้งยังช่วยป้องกันโรคหลายอย่าง เช่น เบาหวาน ความดันโลหิตสูง โรคตับ โรคโลหิตจาง และโรคมะเร็ง เป็นต้น และไม่เพียงแต่ยำสาหร่ายผมนางเท่านั้นที่กินได้อร่อย แต่สามารถนำมาทำ ยำ แกงจืด ก็กินได้อร่อยด้วยเช่นกัน

รูปที่ 4.5 บทความที่ถูกตัด โดยมนุษย์

บทความที่ถูกตัดโดยเนอีฟเบย์โมเดลการทดลองที่ 10

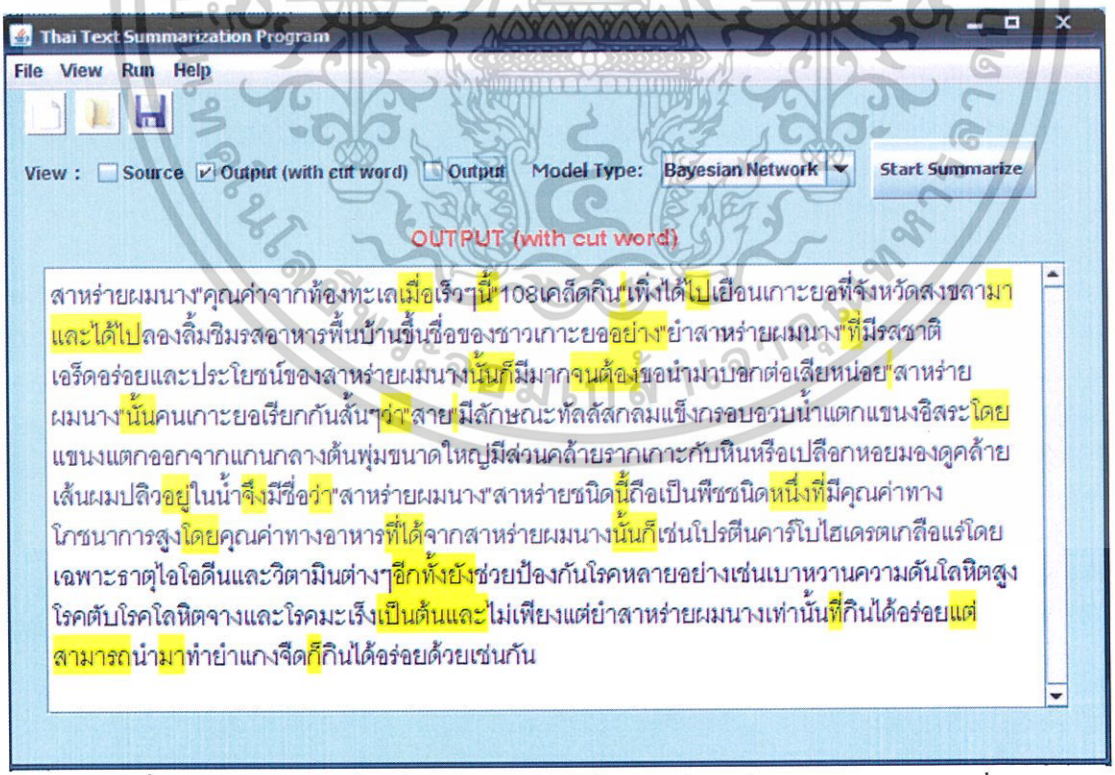


รูปที่ 4.6 แสดงผลลัพธ์ของการตัดคำโดยเนอีฟเบย์โมเดลการทดลองที่ 10 ซึ่งยังแสดงคำที่จะตัดออกอยู่



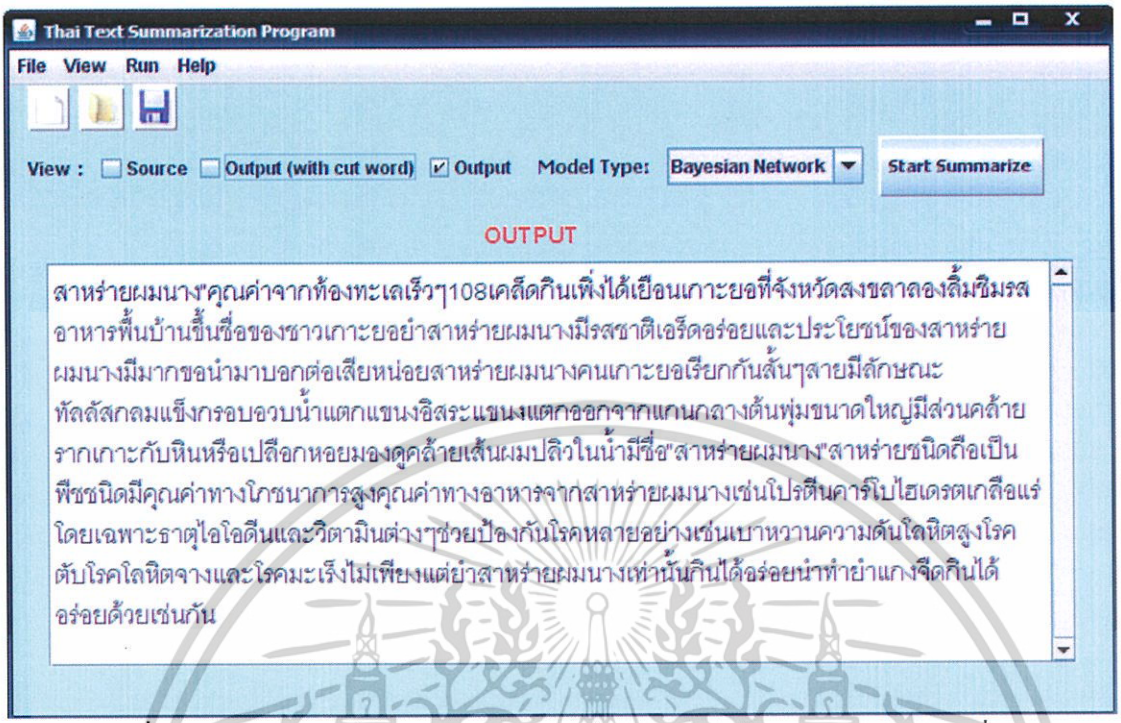
รูปที่ 4.7 แสดงผลลัพธ์ของการตัดคำโดยเนอีฟเบย์โมเดลการทดลองที่ 10

บทความที่ถูกตัดโดยเบย์เซียน เน็ตเวิร์ก โมเดลการทดลองที่ 10



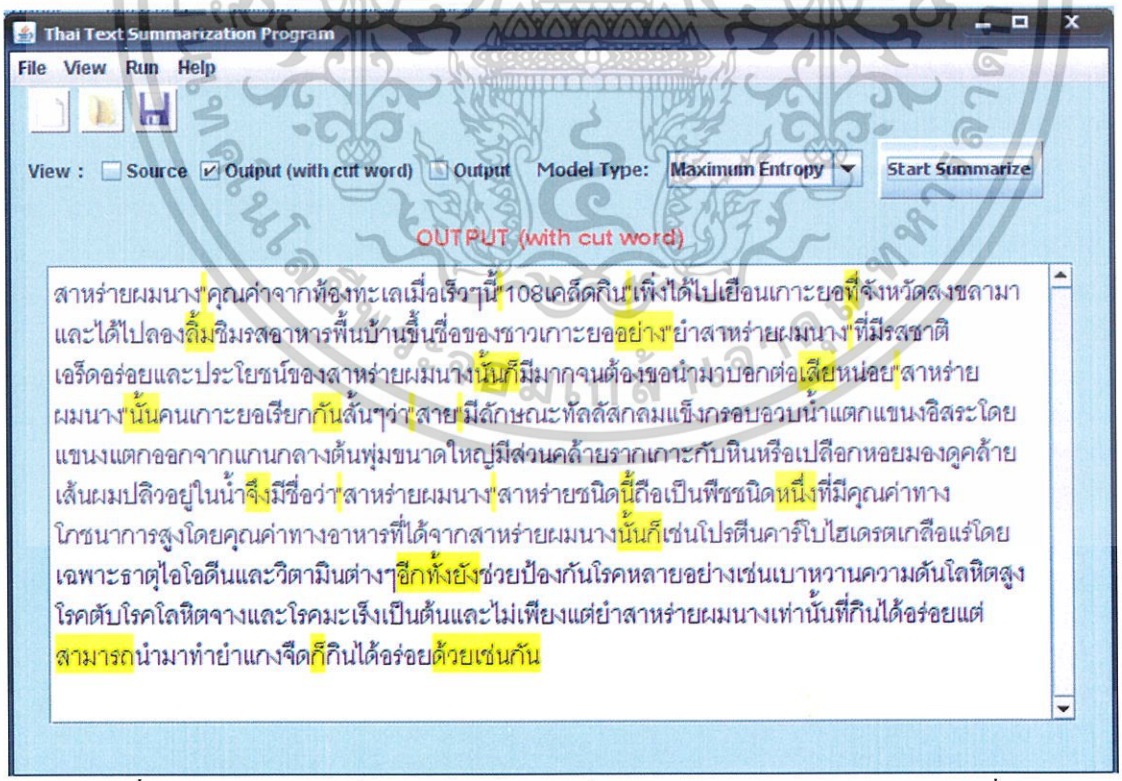
รูปที่ 4.8 แสดงผลลัพธ์ของการตัดคำโดยเบย์เซียน เน็ตเวิร์ก โมเดลการทดลองที่ 10

ซึ่งยังแสดงคำที่จะตัดออกอยู่



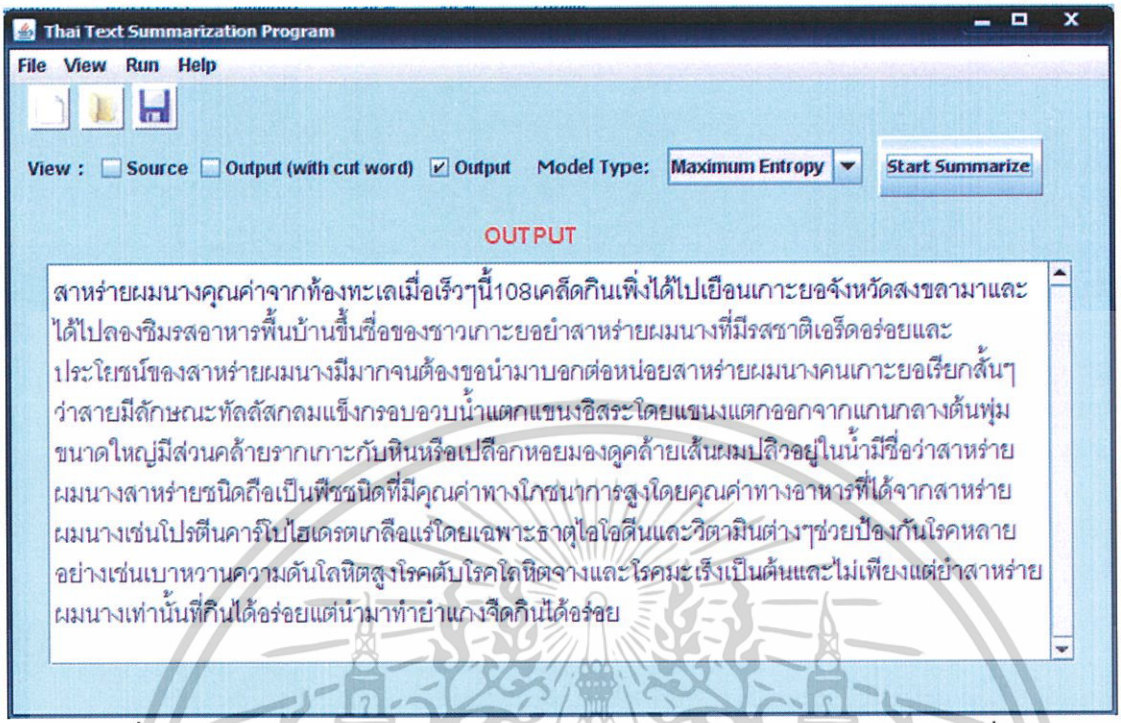
รูปที่ 4.9 แสดงผลลัพธ์ของการตัดคำโดยเบย์เซียน เน็ตเวิร์ก โมเดลการทดลองที่ 10

บทความที่ถูกตัดโดยแม็กซิมัมเอนโทรปีโมเดลการทดลองที่ 16



รูปที่ 4.10 แสดงผลลัพธ์ของการตัดคำโดยแม็กซิมัมเอนโทรปีโมเดลการทดลองที่ 16

ซึ่งยังแสดงคำที่จะตัดออกอยู่



รูปที่ 4.11 แสดงผลลัพธ์ของการตัดคำโดยแม็กซิมัมเอนโทรปีโมเดลการทดลองที่ 16

4.4.1.3 ผลการเปรียบเทียบประสิทธิภาพการตัดคำของโมเดล 3 รูปแบบ

ตารางที่ 4.13 แสดงประสิทธิภาพการตัดคำของโมเดลทั้ง3 ชนิดเปรียบเทียบกับบทความที่มนุษย์ตัด

โมเดลที่ใช้ในการทดสอบประสิทธิภาพ	ความถูกต้องของการตัดคำ	ความผิดพลาดของการตัดคำ	เวลาเฉลี่ยในการตัดคำ
เนอีฟเบย์ การทดลองที่ 10	78.5913	21.4087	24.98
เบย์เซียน เน็ตเวิร์ก การทดลองที่ 10	79.8865	20.1135	40.93
แม็กซิมัมเอนโทรปี การทดลองที่ 16	63.9635	36.0365	32.64

จากการทดลองเปรียบเทียบประสิทธิภาพการตัดคำของโมเดลทั้ง 3 รูปแบบ พบว่าแม็กซิมัมเอนโทรปี มีประสิทธิภาพในการตัดคำน้อยกว่าโมเดลตัวอื่นเมื่อเปรียบเทียบผลลัพธ์จากการตัดคำโดยมนุษย์ ซึ่งเห็นได้ชัดเจนจากผลลัพธ์ที่ได้ในการตัดคำและเวลาที่ใช้ในการตัดคำ โดยเวลาที่ใช้ในการตัดคำมีค่าเฉลี่ยน้อยกว่าเบย์เซียน เน็ตเวิร์ก แต่ก็ยังมีค่าเฉลี่ยมากกว่าเนอีฟเบย์ สำหรับเบย์เซียน เน็ตเวิร์กโมเดล และเนอีฟเบย์โมเดลมีการทำงานที่ได้ผลลัพธ์คล้ายคลึงกันมาก แตกต่างกันเพียงนิดเดียวตรงที่ว่าเบย์เซียน เน็ตเวิร์กโมเดลมีประสิทธิภาพในการตัดคำมากกว่าการทำงานของเนอีฟเบย์ จากตัวอย่างการตัดคำของบทความจะเห็นได้ชัดเจนว่าเบย์เซียน เน็ตเวิร์กสามารถตัด

คำที่มีคุณสมบัติของคำได้ละเอียดกว่าเนอีฟเบย์ ไม่ว่าจะเป็นคุณสมบัติของคำประเภท EAFF, DDAC, XVBM, DDBQ เป็นต้น ที่เป็นคุณสมบัติของคำที่ได้รับการตัดมากที่สุดจากกราฟแสดงความถี่ของคำในการตัดและไม่ตัดคำในเทรนนิ่งเซตจากรูปที่ 4.2 และ 4.3 ซึ่งแสดงให้เห็นว่าประสิทธิภาพการทำงานของเบย์เซียน เน็ตเวิร์กมีประสิทธิภาพดีกว่า และมีการเรียนรู้ในการตัดคำได้ดีกว่าเนอีฟเบย์โมเดลและแม็กซิมัมเอนโทรปีโมเดล

4.4.3 การนำโมเดลมาใช้ในระบบช่วยสรุปใจความสำคัญภาษาไทยอัตโนมัติ

เราจะเลือกใช้โมเดลทั้ง 3 โมเดล ได้แก่ เบย์เซียน เน็ตเวิร์กโมเดลของการทดลองที่ 10, เนอีฟเบย์ โมเดลของการทดลองที่ 10 และแม็กซิมัมเอนโทรปีโมเดลการทดลองที่ 16 เพื่อให้ผู้ใช้มีตัวเลือกในการใช้งานที่มากขึ้น เพราะโมเดลแต่ละตัวมีความสามารถในการตัดคำที่แตกต่างกัน

บทที่ 5

สรุปผลการวิจัย และข้อเสนอแนะ

การสื่อสารด้วยภาษาไทย ประกอบด้วยโครงสร้างทางภาษาที่มีความซับซ้อน เพื่อส่งเสริมให้บทความเหล่านั้นมีความสละสลวยทางภาษา การสรุปบทความเหล่านั้นโดยการตัดคำฟุ่มเฟือยออกให้อยู่ในรูปของบทความที่มีเพียงใจความสำคัญจึงช่วยลดเวลาในการอ่านบทความของผู้อ่านได้มาก การสร้างระบบช่วยสรุปใจความสำคัญภาษาไทยอัตโนมัติ โดยใช้เทคนิค Binary Classification จึงเป็นอีกวิธีหนึ่งที่สามารถช่วยลดระยะเวลาในการอ่านบทความของมนุษย์ได้

จากการทดลองทั้งหมด สามารถสรุปได้ว่าเบย์เซียน เน็ตเวิร์กโมเดลของการทดลองที่ 10 เป็นโมเดลที่มีประสิทธิภาพในการตัดคำสูงสุด ซึ่งมีคุณสมบัติต่างๆ (Attributes) ที่ใช้ในการทดลองได้แก่ หมายเลขประจำคำที่พิจารณา, หน้าที่ของคำที่พิจารณา ในประโยค, ลักษณะของคำที่พิจารณา มี 2 ลักษณะคือ Function และ Content, หน้าที่ของคำที่อยู่ก่อนหน้าคำที่พิจารณา 3 ตำแหน่ง, หน้าที่ของคำที่อยู่ก่อนหน้าคำที่พิจารณา 2 ตำแหน่ง, หน้าที่ของคำที่อยู่ก่อนหน้าคำที่พิจารณา 1 ตำแหน่ง, หน้าที่ของคำที่อยู่หลังคำที่พิจารณา 1 ตำแหน่ง, หน้าที่ของคำที่อยู่หลังคำที่พิจารณา 2 ตำแหน่ง, หน้าที่ของคำที่อยู่หลังคำที่พิจารณา 3 ตำแหน่ง, บอกว่าคำที่พิจารณา ถูกตัดหรือไม่ถูกตัด (Cut, Not_cut) และยังสามารถสรุปได้อีกว่า คุณสมบัติของคำที่มีผลต่อประสิทธิภาพการตัดคำของโมเดล คือ 1. ชนิดของคำในประโยคของคำที่พิจารณา, 2. คุณสมบัติของคำ, 3. ชนิดของคำในประโยคของคำข้างเคียงคำที่สนใจ โดยยิ่งมาค่าชนิดของคำในประโยคของคำข้างเคียงมากขึ้นก็ยิ่งทำให้ประสิทธิภาพในการตัดคำดีขึ้น

ในการพัฒนาระบบสรุปใจความสำคัญภาษาไทยอัตโนมัติ จะนำโมเดลโมเดลทั้ง 3 โมเดล ได้แก่ เบย์เซียน เน็ตเวิร์กโมเดลของการทดลองที่ 10, เนอีฟเบย์โมเดลของการทดลองที่ 10 และแม็กซิมัมเอนโทรปีโมเดลการทดลองที่ 16 มาใช้ ทั้งนี้เพื่อให้ผู้ใช้สามารถเลือกที่จะใช้โมเดลที่ต้องการได้ด้วยตัวเอง อันจะทำให้ ผู้ใช้มีตัวเลือกในการใช้งานที่มากขึ้น เพราะโมเดลแต่ละตัวมีความสามารถในการตัดคำที่แตกต่างกัน

โดยระบบช่วยสรุปใจความสำคัญภาษาไทยอัตโนมัติ โดยใช้เทคนิคการำแนกแบบไบนารีนั้น เมื่อได้รับบทความตั้งต้นเข้ามา และกดปุ่ม Start Summarize เพื่อสั่งให้ระบบทำการสรุปใจความ จะมีกระบวณการ การทำงานตังนี้ เริ่มต้นที่ ตัดคำและหาชนิดของคำ (Pos) โดยการเรียกใช้ Swath จากนั้นจะทำการสร้างชุดข้อมูลเพื่อการทดสอบ (Test set) โดย อาศัยไฟล์ที่ตัดคำและระบุชนิดของคำ (Pos) ที่ได้จาก Swath ในขั้นตอนแรก และจาก Word list เพื่ออ้างอิง ID ของคำ ขั้นที่สาม นำ Test set ให้โมเดลพยากรณ์ว่าจะตัดหรือไม่ตัดคำใดบ้าง โดยเรียกใช้ Weka จะได้ผลลัพธ์ที่

อยู่ในรูปแบบ .txt และสุดท้านจะนำไฟล์ผลลัพธ์จากขั้นตอนที่สาม มาเปรียบเทียบกับไฟล์ที่ตัดคำและระบุชนิดของคำ (Pos) ที่ได้จาก swath และแสดงผลที่หน้าจอผลลัพธ์ให้ผู้ใช้เห็นว่าคำใดที่ถูกตัดไปบ้าง

5.1 เปรียบเทียบประสิทธิภาพการทำงานของระบบกับมนุษย์

5.1.2 การทดสอบการใช้งานระบบช่วยสรุปใจความสำคัญภาษาไทยอัตโนมัติ

การทดลองให้ผู้ใช้งานระบบสรุปใจความสำคัญอัตโนมัติ จากผู้ใช้ 30 คน เป็นวิธีที่สามารถทดสอบระบบที่ทำขึ้นมาได้ว่าระบบมีประสิทธิภาพและความสามารถเพียงพอกับความต้องการของผู้ใช้พอหรือไม่ ผลการทดสอบการใช้งานเป็นดังนี้

บทความที่ถูกตัดจากระบบอ่านรู้เรื่องได้สาระ	67.5	เปอร์เซ็นต์
คำที่ถูกตัดจากระบบสมควรถูกตัดจากบทความ	50	เปอร์เซ็นต์
ระบบควรตัดคำในบทความเพิ่มเติม	47.5	เปอร์เซ็นต์
ระบบตัดคำฟุ่มเฟือยออกเพียงพอแล้ว	72.5	เปอร์เซ็นต์
ความเชื่อมั่นในความาสามารถการตัดคำของระบบ	65	เปอร์เซ็นต์
ความพึงพอใจในประสิทธิภาพการทำงานของระบบ	50	เปอร์เซ็นต์

เมื่อทำการทดลองตัดคำโดยใช้ระบบตัดคำอัตโนมัติพบว่า คำส่วนใหญ่ที่ถูกตัดออกจากบทความนั้น เป็นคำฟุ่มเฟือยจริง ไม่ว่าจะเป็น คำบุพบท คำอุทาน คำขยายคำนาม สรรพนาม กริยาหรือคำสันธาน ดังนั้นจึงถือได้ว่าการทำงานของระบบมีประสิทธิภาพตรงตามวัตถุประสงค์ของการพัฒนา

5.2 ปัญหาและอุปสรรคที่พบ

5.2.1 การเก็บคลังข้อมูลมีความลำบากและเกิดความล่าช้า

เนื่องจากในการสร้างคลังข้อมูล ต้องอาศัยเวลาค่อนข้างมากในการตัดคำฟุ่มเฟือยออกจากแต่ละบทความ

5.2.2 ความจำกัดด้านทรัพยากร

การเรียนรู้ของบางตัวจำแนกต้องใช้คอมพิวเตอร์ที่มีสเปกสูงมาก และใช้เวลาในการเรียนรู้ค่อนข้างนาน ทำให้บางทีหากเกิดระบบไฟฟ้าขัดข้อง ก็จะทำให้ต้องเสียเวลาในการเรียนรู้ใหม่

5.2.3 ความไม่สอดคล้องกันของข้อมูล

การตัดคำของมนุษย์และโปรแกรม Swath ยังไม่สอดคล้องกัน ทำให้เกิดความไม่สมบูรณ์ของ Training Set เช่น คำว่า "เอร็ดอร่อย" มนุษย์ต้องการตัดคำว่า "เอร็ด" ออก เนื่องจากเห็นว่าเป็นคำฟุ่มเฟือย แต่ใน swath นั้น ตัดคำว่า "เอร็ดอร่อย" เป็นคำเดียว

5.2.4 ข้อจำกัดของเครื่องมือ

โปรแกรม Weka สามารถกำหนดหน่วยคำจำสูงสุดในการประมวลผลได้เพียง 1.5 GB ซึ่งไม่เพียงพอในการเรียนรู้ของ ซับพอท เวกเตอร์ แมชชีนในหลายๆการทดลอง

5.3 ประโยชน์ที่ได้รับจากการพัฒนาโครงงาน

การศึกษาค้นคว้าเพื่อพัฒนาระบบสรุปใจความสำคัญภาษาไทยอัตโนมัติ ผู้พัฒนาได้ทำการศึกษาทฤษฎีที่เกี่ยวข้องกับการพัฒนา เครื่องมือและอุปกรณ์ที่ช่วยในการพัฒนาระบบทำให้มีความเข้าใจในทฤษฎีและโครงสร้างรูปแบบการทำงานของเครื่องมือและอุปกรณ์ที่เป็นส่วนหนึ่งในการพัฒนาระบบมากขึ้น ซึ่งสามารถนำความรู้และประสบการณ์เหล่านี้ไปใช้ในการพัฒนาระบบและทำงานวิจัยอื่นๆ ต่อไปได้ในอนาคต

การนำโปรแกรมมาทดสอบการตัดคำฟุ่มเฟือยในบทความหนึ่งๆ พบว่า ประสิทธิภาพในการตัดคำอยู่ในเกณฑ์ที่น่าพอใจ โปรแกรมมีการเรียนรู้การตัดคำจากโมเดล และมีความสามารถในการแยกประเภทการตัดของคำได้เป็นอย่างดี เมื่อเทียบกับบทความต้นแบบที่ถูกตัดโดยมนุษย์ ดังนั้นการนำโปรแกรมไปใช้งานจริงจึงมีแนวโน้มที่เป็นไปได้ แต่อาจต้องอาศัยการทดลองเปรียบเทียบโมเดลจากบทความประเภทอื่นๆ เพื่อหาความเสถียรของโปรแกรมเพิ่มเติม

นอกจากนี้การพัฒนาระบบสรุปใจความสำคัญภาษาไทยอัตโนมัติ สามารถเป็นแนวทางที่สำคัญให้กับบุคคลที่สนใจพัฒนาต่อไปในอนาคต

5.4 แนวทางในการพัฒนาโครงงาน

5.4.1 สร้างคลังข้อมูลจากข้อมูลด้านอื่นๆ

คลังข้อมูลเป็นปัจจัยสำคัญที่ทำให้โมเดลของระบบมีความสามารถในการสรุปใจความสำคัญ ซึ่งเครื่องมือในการเรียนรู้หรือแมชชินเลินนิ่งต้องทำการเรียนรู้เพื่อสร้างโมเดลในการตัดสินใจ ยิ่งข้อมูลที่นำมาสร้างมีหลายประเภท เราจะสามารถทำการทดลองเปรียบเทียบกับโมเดลที่สร้างจากบทความเกี่ยวกับอาหารและสุขภาพได้ว่า บทความประเภทใดที่มีความสามารถในการนำมาสร้างเป็นโมเดลได้ดีกว่า นั่นแสดงให้เห็นว่า คำในกลุ่มบทความประเภทนั้นๆ มีความ

สอดคล้องกันของข้อมูลและความหมายมากกว่า จึงมีประสิทธิภาพในการแยกประเภทของคำได้ดีกว่า ส่งผลต่อการพัฒนาในหัวข้อถัดไป

5.4.2 พิจารณาการแทนที่คำ ด้วยคำที่มีความหมายและสั้นที่สุด

คำในภาษาไทยคำหนึ่งๆ สามารถมีความหมายตรงกับคำอื่นๆ ในภาษาไทยได้ เช่นคำว่า กิน รับประทาน ฉันท์ เป็นต้น ซึ่งการใช้จะแตกต่างกันไปตามโอกาสและวัตถุประสงค์ในการใช้งาน อย่างไรก็ตาม เราจะสังเกตเห็นว่าคำที่มีความหมายเหมือนกันในภาษาไทยส่วนใหญ่เป็นคำที่พบอยู่ในกลุ่มบทความประเภทเดียวกัน ซึ่งถ้าเราสามารถนำมาวิเคราะห์เปรียบเทียบหาความหมายของคำได้แล้วนั้น เราสามารถแบ่งระดับของคำในภาษาไทยที่มีความหมายเหมือนกันเป็นระดับต่างๆ กัน เพื่อใช้เป็นคำแทนคำที่มีความหมายซับซ้อนและยาว ด้วยคำที่มีความหมายเหมือนเดิมแต่มีความกระทัดรัดและสั้นที่สุด เพื่อให้การสรุปบทความนั้นมีความสามารถและประสิทธิภาพสูงสุด

5.4.3 ทดลองกับการแยกประเภทของคำแบบอื่นๆ

นอกจากการแยกประเภทของคำโดยใช้ทฤษฎีของ เนอีฟเบย์ เบย์เซียน เน็ตเวิร์ก แม็กซิมั่ม เอนโทรปี และซับพอท เวกเตอร์ แมชชีน แล้ว ยังมีหลักการแยกประเภทของคำประเภทอื่นๆที่มีความน่าสนใจเพื่อหาการแยกประเภทของคำที่สามารถนำมาพัฒนาระบบให้มีประสิทธิภาพในการทำงานสูงสุด

บรรณานุกรม

Andrew W.Moore, **Support Vector Machines,** School of Computer Science, Carnegie Mellon University.

Andrew W.Moore, **Bayes Nets for representing and resoning about uncertainty**, School of Computer Science, Carnegie Mellon University.

Ben Hachey and Claire Grover, **Sentence Extraction for Legal Text Summarization,** Edinburgh, University of Edinburgh, Edinbutgh EH8 9LW, UK.

Bianca Zadrozny Charles Elkan, **Calibrated Naïve Bayes Classifier scores,** San Diego, University of California, , La Jolla, California.

Bo Pang, Lillian Lee, **Thumbs up? Sentiment Classification using Machine Learning Techniques,** NY, Cornell University Ithaca.

Charles Elkan, **Naïve Bayesian Learning,** San Diego, Adapted from Technical Report No. CS97-557, University of California, September 1997.

Daryle Niedermayer, **An Introduction to Bayesian Network and their Contemporary Application,** December 1, 1998.

Florian WOLF, **A comparison of algorithm and human performance,** Paragraph-. Word-, and coherence-base approaches to sentence renking, Massachusetts Institute of Technology Cambridge, MA 02139.

John Hutchins, **Summarization: Some problems and methods,** University of East Anglia.

Kamal Nigam, John Lafferty, Andrew McCallum, **Using Maximum Entropy for Text Classification,** PA, Carnegie Mellon University, Pittsburgh, PA 15213.

Mihai Rotaru, **Maximum Entropy Theory and Example,** Sennott Hall 5313, May 19,2005.

Morgan, Bruce W., **An Introduction to Bayesian Statistical Decision Processes**, Prentice-Hall Inc., Englewood Cliffs, N.J. 1968. p. 15.

Nello Cristianini, **Learning the Kernel Matrix with Semi-Definite Programming,** Leuven, ICML 2002.

Nevin Lianwen Zhang and David Poole, **A simple approach to Bayesian to Bayesian network computation**, Banff, AB, May 1994, 171-178.

Paul N. Bennett, **CMU-css.** CMU-CS-00-155, School of Computer Science Carnegie Mellon University Pittsburgh, PA 15213, September 12, 2000.

Paul Penfield Jr., **Principle of Maximum Entropy,** Massachusetts, Massachusetts Institute of Technology, April 4,2003.

R. Berwick, **An Idiot's guide to Support vector machines,** Village Idiot, 2001.

Steve Gunn, **Support Vector Machines for Classification and Regression,** Image speech and Intelligent Systems group, University of Southampton, November 10, 1997.

Tom M. Mitchell, **Generative and discriminative classifier: Naïve Bayes and Logistic regression,** Ng, A.Y., and Jordan, M. (2002).

ประวัติผู้เขียน

ชื่อ-นามสกุล	นางสาวกาญจนิจ กิจกสิวัฒน์
วัน เดือน ปีเกิด	7 พฤศจิกายน 2528 ที่จังหวัดขอนแก่น
ที่อยู่	89/315 หมู่1 ถนนบางขุนเทียนชายทะเล แขวงแสมดำ เขตบางขุนเทียน กรุงเทพมหานคร 10150
โทรศัพท์เคลื่อนที่	086-556-4066
ประวัติการศึกษา	
พ.ศ. 2551	วิทยาศาสตรบัณฑิต คณะเทคโนโลยีสารสนเทศ สาขาเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ประสบการณ์	
พ.ศ. 2545	นักเรียนแลกเปลี่ยน โครงการแลกเปลี่ยนวัฒนธรรมไทยนานาชาติ (AFS) ประเทศเบลเยี่ยม
พ.ศ. 2551	โครงงานวิจัยระบบสรุปใจความสำคัญภาษาไทยอัตโนมัติ โดยใช้เทคนิค Binary Classification

ประวัติผู้เขียน

ชื่อ-นามสกุล	นางสาวนิชาภา ถนอมสิงห์
วัน เดือน ปีเกิด	27 กรกฎาคม 2529 โรงพยาบาลราชวิถี จังหวัดกรุงเทพมหานคร
ที่อยู่	6/358 ซอยพระยาสุเรนทร์ 33 แขวงบางชัน เขตคลองสามวา กทม.10510
โทรศัพท์เคลื่อนที่	085-8055860
อีเมล	BES_NICH@HOTMAIL.COM
ประวัติการศึกษา	
พ.ศ. 2551	วิทยาศาสตรบัณฑิต คณะเทคโนโลยีสารสนเทศ สาขาเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ประสบการณ์	
พ.ศ. 2551	-นักศึกษาฝึกงาน ที่หน่วยปฏิบัติการวิจัยวิทยาการมนุษยภาษา ศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์ -นักศึกษาโครงการทุนสนับสนุนการวิจัยปริญญานิพนธ์ (Senior Project) โครงการสร้างปัญญาวิทย์ ผลิตนักเทคโน (Young Scientist and Technologist Programme : YSTP) -โครงงานวิจัยระบบสรุปใจความสำคัญภาษาไทยอัตโนมัติ โดยใช้เทคนิค Binary Classification
รางวัลที่ได้รับ	
พ.ศ. 2551	นักศึกษาดีเด่น ด้านทำชื่อเสียงทางด้านวิชาการให้กับคณะ
พ.ศ. 2551	นักศึกษาดีเด่น ด้านกิจกรรมนักศึกษา