การจำแนกความรู้สึกระดับประโยคสำหรับความคิดเห็นของนักเรียนมัธยมศึกษาตอนต้น

วิทยานิพนธ์

ของ

อรทัย จันท่าม่วง

เสนอต่อมหาวิทยาลัยมหาสารคาม เพื่อเป็นส่วนหนึ่งของการศึกษาตามหลักสูตร

ปริญญาวิทยาศาสตรมหาบัณฑิต สาขาวิชาวิทยาการคอมพิวเตอร์

พฤษภาคม 2567

การจำแนกความรู้สึกระดับประโยคสำหรับความคิดเห็นของนักเรียนมัธยมศึกษาตอนต้น

วิทยานิพนธ์

ของ

อรทัย จันท่าม่วง

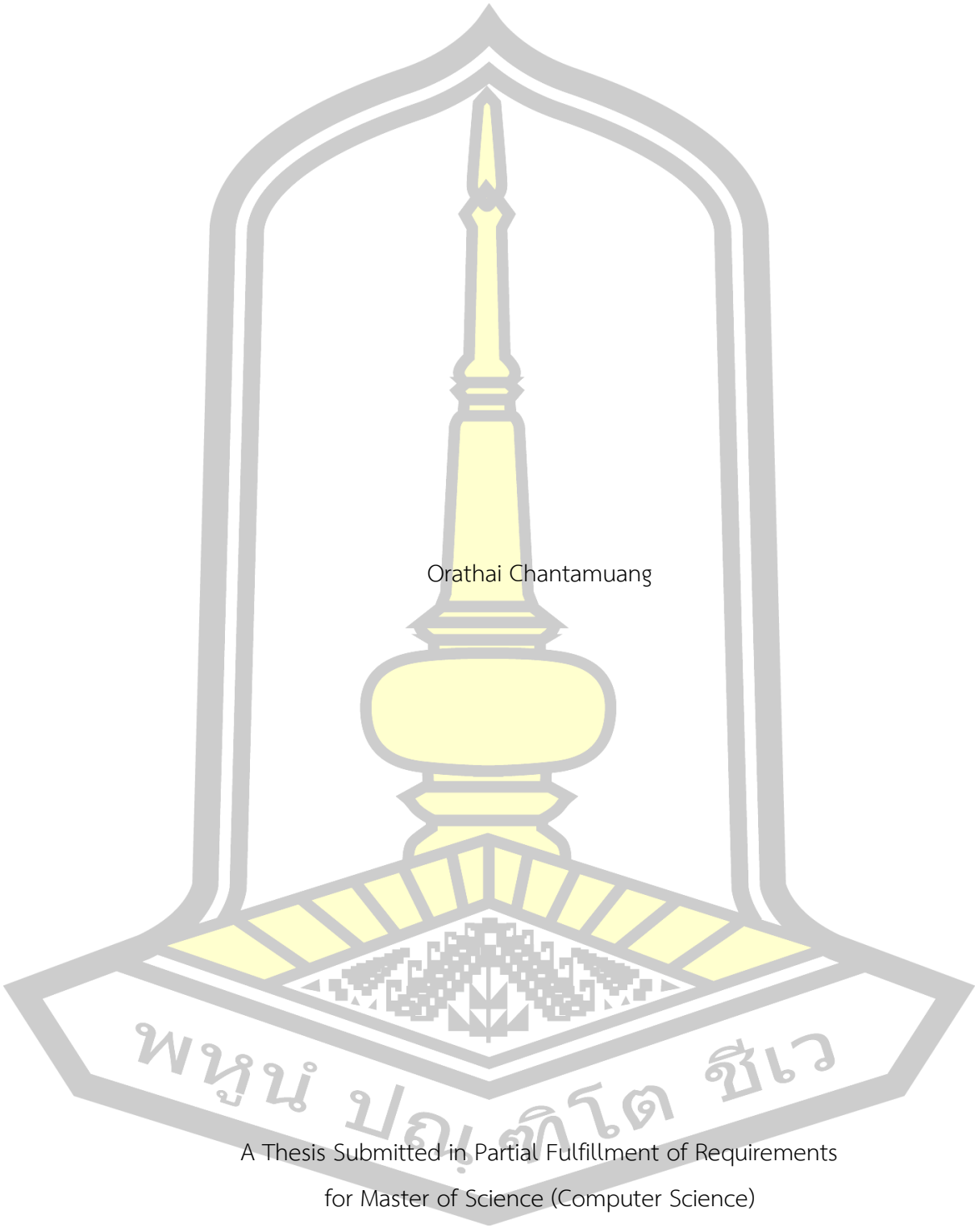
เสนอต่อมหาวิทยาลัยมหาสารคาม เพื่อเป็นส่วนหนึ่งของการศึกษาตามหลักสูตร

ปริญญาวิทยาศาสตรมหาบัณฑิต สาขาวิชาวิทยาการคอมพิวเตอร์

พฤษภาคม 2567

Sentence-level Sentiment Classification for Middle-school Student Comments

Orathai Chantamuang



A Thesis Submitted in Partial Fulfillment of Requirements

for Master of Science (Computer Science)

May 2024

คณะกรรมการสอบวิทยานิพนธ์ ได้พิจารณาวิทยานิพนธ์ของนางสาวอรทัย จันท่าม่วง แล้วเห็นสมควรรับเป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต สาขาวิชาวิทยาการคอมพิวเตอร์ ของมหาวิทยาลัยมหาสารคาม

คณะกรรมการสอบวิทยานิพนธ์

.....ประธานกรรมการ
(รศ. ดร. ศุภกานต์ พิมลธเรศ)

.....อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก
(รศ. ดร. จันทิมา พลพินิจ)

.....กรรมการ
(ผศ. ดร. ฉัตรเกล้า เจริญผล)

.....กรรมการ
(ผศ. ดร. แกมกาญจน์ สมประเสริฐศรี)

มหาวิทยาลัยอนุมัติให้รับวิทยานิพนธ์ฉบับนี้ เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญา วิทยาศาสตรมหาบัณฑิต สาขาวิชาวิทยาการคอมพิวเตอร์ ของมหาวิทยาลัยมหาสารคาม

.....

(รศ. ดร. จันทิมา พลพินิจ)
คณบดีคณะวิทยาการสารสนเทศ

(รศ. ดร. กริสน์ ชัยมูล)
คณบดีบัณฑิตวิทยาลัย

ชื่อเรื่อง การจำแนกความรู้สึกระดับประโยคสำหรับความคิดเห็นของนักเรียนมัธยมศึกษาตอนต้น
ผู้วิจัย อรทัย จันท่าม่วง
อาจารย์ที่ปรึกษา รองศาสตราจารย์ ดร. จันทิมา พลพินิจ
ปริญญา วิทยาศาสตรมหาบัณฑิต **สาขาวิชา** วิทยาการคอมพิวเตอร์
มหาวิทยาลัย มหาวิทยาลัยมหาสารคาม **ปีที่พิมพ์** 2567

บทคัดย่อ

งานวิจัยฉบับนี้นำเสนอการจำแนกความรู้สึกระดับประโยคสำหรับความคิดเห็นของนักเรียนมัธยมศึกษาตอนต้นในสามมุมมอง (Aspect) คือ มุมมองด้านผู้สอน มุมมองด้านบทเรียน และมุมมองด้านสภาพแวดล้อมในการเรียน โดยในกระบวนการที่นำเสนอเป็นกระบวนการแบบมีผู้สอน โดยใช้ข้อมูลสามชุด ข้อมูลชุดที่หนึ่งจะเก็บในรูปแบบประโยคของแต่ละมุมมอง จากนั้นจะให้ผู้เชี่ยวชาญกำหนดคำเบื้องต้นในแต่ละด้านเพื่อเป็นคำเป้าหมาย (Target Word) ในการเรียนรู้ด้วย Skip gram ของ Word2Vec เพื่อรวบรวมคลังคำของแต่ละมุมมอง โดยได้จำนวนคำสำหรับมุมมองด้านผู้สอน 43 คำ จำนวนคำสำหรับมุมมองด้านบทเรียน 37 คำ และจำนวนคำสำหรับมุมมองด้านสภาพแวดล้อมในการเรียน 28 คำ จากนั้นจะใช้ข้อมูลชุดที่สองซึ่งเก็บเป็นบทวิจารณ์ของนักเรียนเพื่อใช้ในการสร้างโมเดลเพื่อการจำแนกความรู้สึกจากบทวิจารณ์แบบสองกลุ่มคือความรู้สึกเชิงบวกและความรู้สึกเชิงลบ โดยอาศัยคลังคำที่ได้จากข้อมูลชุดที่หนึ่งเป็นคุณลักษณะในการสร้างโมเดล ในการศึกษานี้ได้ศึกษาเชิงเปรียบเทียบกับวิธีการให้น้ำหนักสองตัวคือ tf-idf และ tf-igm นอกจากนั้นยังมีการเปรียบเทียบอัลกอริทึมการเรียนรู้แบบมีผู้สอนจำนวนสี่ตัวคือ ขั้นตอนวิธีเพื่อนบ้านที่ใกล้ที่สุด ซัพพอร์ตเวกเตอร์แมชชีน ป่าสุ่ม และมัลติโนเมียลนาอีฟเบย์ เมื่อได้โมเดลการจำแนกความรู้สึกจากบทวิจารณ์แล้ว โมเดลดังกล่าวจะถูกใช้ทดสอบด้วยข้อมูลชุดที่สามซึ่งเก็บเป็นแบบบทวิจารณ์ จากนั้นจะทำการแยกประโยคด้วยเว้นวรรคและจะใช้วิธีการวิเคราะห์ความคล้ายคลึงโดยเทียบประโยคที่แยกได้กับคลังคำของแต่ละมุมมองที่ได้จากข้อมูลชุดที่หนึ่งเพื่อวิเคราะห์ว่าประโยคดังกล่าวควรจัดอยู่ในกลุ่มของมุมมองใด ท้ายที่สุดโมเดลเพื่อการจำแนกความรู้สึกจะถูกใช้ในการวิเคราะห์ขั้วความรู้สึกของแต่ละประโยค ภายหลังจากการทดสอบด้วยค่าความระลึก ค่าความแม่นยำ ค่าเอฟ และค่าความถูกต้อง พบว่าอัลกอริทึมป่าสุ่มที่ทำงานร่วมกับการให้น้ำหนักแบบ tf-idf ให้ประสิทธิภาพที่ดีที่สุด

คำสำคัญ : การจำแนกความรู้สึกระดับประโยค, ความคิดเห็นของนักเรียน, การจำแนกเอกสาร, สคริป

แกรม, โมเดลการจำแนกความรู้สึก, อัลกอริทึมการเรียนรู้ของเครื่อง

TITLE Sentence-level Sentiment Classification for Middle-school Student Comments

AUTHOR Orathai Chantamuang

ADVISORS Associate Professor Jantima Polpinij , Ph.D.

DEGREE Master of Science **MAJOR** Computer Science

UNIVERSITY Mahasarakham University **YEAR** 2024

ABSTRACT

The objective of this research is to identify the sentiments expressed by junior high school students at the sentence-level based sentiment analysis on three aspects: the instructor, the lesson, and the learning environment. This study has utilized the supervised learning method. The first dataset is gathered in sentence format, with each aspect represented. Linguistic specialists then choose the fundamental words within each aspect to serve as the target words for learning. Word2Vec's Skip Gram algorithm is utilized to generate a corpus of words from each aspect. The instructor aspect consisted of 43 words, the lesson aspect consisted of 37 words, and the learning environment aspect consisted of 28 words. The second dataset of student reviews was subsequently utilized to construct a model for classifying the sentiments of the reviews into two distinct groups: positive and negative classes. This sentiment classifier model was developed by using a corpus of words obtained from the first dataset, which served as features for the modeling process. This study conducted a comparative analysis of two weighing algorithms, namely tf-idf and tf-igm. Furthermore, four supervised learning algorithms are compared. The algorithms used are *k*-Nearest Neighbor, Support Vector Machines, Random Forests, and Multinomial Naïve Bayes. After acquiring the sentiment classification model from the student reviews, it was evaluated using the third dataset of student reviews. This method involves segmenting sentences with white-spaces and utilizing Euclidean distance to group each sentence from a student review into particular aspect groups. The first dataset's word corpus was employed

for defining sentences into relevant groups. Finally, the sentiment classification model was applied to determine the sentiment polarity of each review's sentence. Following testing via the recall, precision, F1, and accuracy, It was discovered that the random forest algorithm together with tf-idf term weighting produced the most impressive results.

Keyword : Sentence-level Sentiment Analysis, Student Reviews, Text Classification, Skip Gram, Sentment Classifier Model, Machine Learning Algorithm

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จสมบูรณ์ได้ด้วยความกรุณาและความช่วยเหลืออย่างสูงยิ่งจาก รองศาสตราจารย์ ดร. ศุภกานต์ พิมลธเรศ ประธานกรรมการควบคุมวิทยานิพนธ์ ผู้ช่วยศาสตราจารย์ ดร. ฉัตรเกล้า เจริญผล และ ผู้ช่วยศาสตราจารย์ ดร. แกมกาญจน์ สมประเสริฐศรี กรรมการควบคุมวิทยานิพนธ์ ที่ให้คำชี้แนะในการปรับปรุงกระบวนการวิจัยในวิทยานิพนธ์

ขอขอบพระคุณ รองศาสตราจารย์ ดร. จันทิมา พลพินิจ อาจารย์ที่ปรึกษาวิทยานิพนธ์ ที่ให้คำปรึกษาคำแนะนำและควบคุมการดำเนินงานในวิทยานิพนธ์และเอกสารวิจัยเพื่อการตีพิมพ์จนสำเร็จแล้วเสร็จ

ขอขอบพระคุณอาจารย์ ดร. บัญชา เหลือผล ที่ให้ความกรุณาในการสอนเขียนโปรแกรมและให้คำแนะนำในการแก้ปัญหาเกี่ยวกับการเขียนโปรแกรม จนสามารถเขียนโปรแกรมได้อย่างครบถ้วนสมบูรณ์

อรทัย จันท่าม่วง

สารบัญ

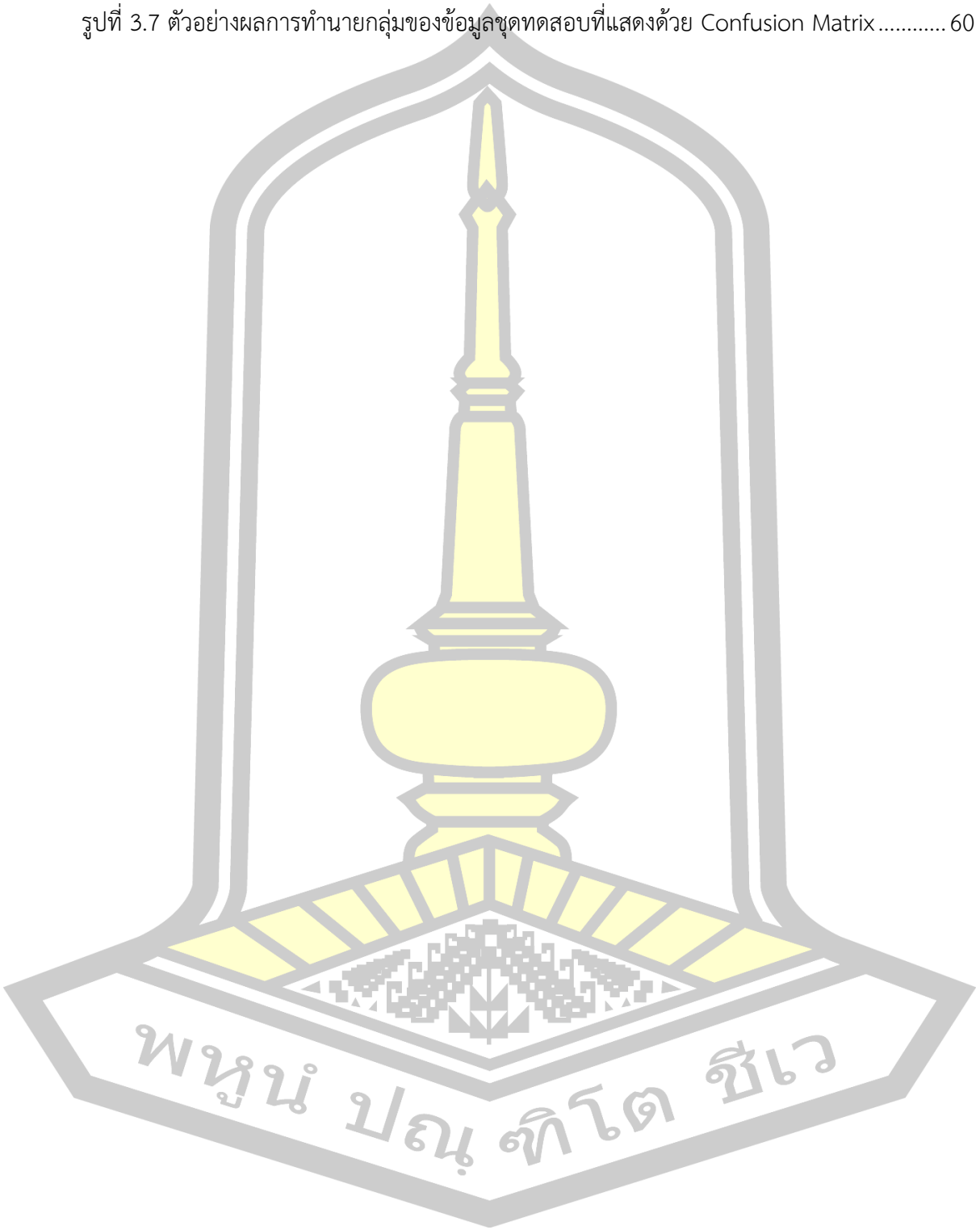
หน้า

สารบัญตาราง

หน้า

สารบัญรูป

หน้า



บทที่ 1
บทนำ

1.1 หลักการและเหตุผล

กระบวนการเรียนการสอน เป็นยุทธศาสตร์และกลวิธีในการกระตุ้นผู้เรียนให้เกิดการเรียนรู้และแสดงพฤติกรรมที่พึงประสงค์โดยใช้เทคนิควิธีสอน การจัดกิจกรรมและสื่อการสอน [1] กระบวนการเรียนการสอนที่เหมาะสมจะช่วยให้ผู้เรียนมีการเรียนรู้ตามที่คาดหวังและตรงตามมาตรฐานของการเรียนรู้ อย่างไรก็ตาม เพื่อให้กระบวนการการจัดการเรียนเป็นรูปแบบที่สามารถส่งเสริมให้ผู้เรียนเกิดการเรียนรู้ที่มีมาตรฐานและมีประสิทธิภาพ และช่วยให้ผู้สอนทราบเกี่ยวกับความรู้สึกของผู้เรียนที่มีต่อตนเอง การประเมินการเรียนการสอน (Learning Process Assessment) และผู้สอน (Teacher Evaluation) โดยผู้เรียนจึงกลายเป็นสิ่งที่จำเป็น เพราะจะช่วยทำให้เกิดการปรับปรุงกระบวนการของการเรียนการสอน

คำติชมหรือความคิดเห็นจากผู้เรียน (Student Feedback) เป็นเครื่องมือสำคัญอย่างหนึ่งในการวิเคราะห์เพื่อประเมินรูปแบบการเรียนการสอน เพราะความคิดเห็นจากผู้เรียนจะเป็นการแสดงอารมณ์และความรู้สึกของผู้เรียนที่แสดงให้เห็นถึงความพึงพอใจที่ผู้เรียนมีต่อผู้สอนและรูปแบบการเรียนการสอนมากน้อยเพียงใด [2] นอกจากนี้ ในงานวิจัย [2] ยังได้กล่าวว่า ความคิดเห็นจากผู้เรียนสามารถนำมาใช้ในการปรับปรุงรูปแบบการเรียนการสอน เพื่อให้มีประสิทธิภาพที่ดีหรือเหมาะสมกับผู้เรียนมากขึ้น

อย่างไรก็ตามการวิเคราะห์ความคิดเห็นจากผู้เรียน เพื่อนำผลของการวิเคราะห์มาใช้ในการปรับปรุงรูปแบบการเรียนการสอนนั้น ในช่วงแรกๆ จะเป็นการวิเคราะห์ด้วยคน แต่การวิเคราะห์ด้วยคนนั้นมีโอกาสที่จะเกิดข้อผิดพลาดหรืออคติ [3] ดังนั้นในช่วงทศวรรษที่ผ่านมา เครื่องมือในการวิเคราะห์คำติชมจากผู้เรียนแบบอัตโนมัติจึงได้รับความสนในการศึกษามาอย่างต่อเนื่อง [4] แต่ก็ยังมีจำนวนไม่มากนักเนื่องจากข้อจำกัดเรื่องของข้อมูลที่ใช้ในการศึกษา เพราะข้อมูลเหล่านั้นจัดว่าเป็นข้อมูลปกปิดของหน่วยงานด้านการศึกษานั้นๆ และเครื่องมือที่สามารถประยุกต์เข้ามาใช้การวิเคราะห์คำติชมจากผู้เรียนแบบอัตโนมัติเรียกว่าการวิเคราะห์ความรู้สึก (Sentiment Analysis)

การวิเคราะห์ความรู้สึก [5] เป็นสาขาหนึ่งด้านการประมวลผลภาษาธรรมชาติ (Natural Language Processing: NLP) โดยเป็นการวิเคราะห์ความรู้สึกหรืออารมณ์จากข้อความเพื่อให้รู้ว่าความรู้สึกหรืออารมณ์ของผู้คนที่มีต่อสิ่งใดสิ่งหนึ่ง เช่น สินค้าและบริการ นั้นมีแนวโน้มบวก (Positive) เป็นกลาง (Neutral) หรือเป็นลบ (Negative) และนักวิจัยหลายท่านได้ประยุกต์กระบวนการของการวิเคราะห์ความรู้สึกมาใช้ในการวิเคราะห์คำติชมจากผู้เรียนในโดเมนของการศึกษา [6]

อย่างไรก็ตาม จากการทบทวนวรรณกรรมเบื้องต้น [7] พบว่าโดยทั่วไปแล้ว การประยุกต์กระบวนการของการวิเคราะห์ความรู้สึกมาใช้ในการวิเคราะห์คำติชมหรือความคิดเห็นจากผู้เรียนในโดเมนของการศึกษา มักจะมองภาพรวมของเอกสารที่แสดงคำติชมนั้นว่ามีความพึงพอใจต่อรูปแบบการเรียนการสอนหรือไม่ ทำให้การวิเคราะห์ความรู้สึกอาจจะไม่ใช่ความรู้สึกที่แท้จริงของผู้เรียน

เพราะในเอกสารที่แสดงความคิดเห็นของผู้เรียนเกี่ยวกับการเรียนการสอนนั้น อาจจะมีทั้งส่วนที่ผู้เรียนแสดงความคิดเห็นเชิงบวกและเชิงลบปะปนกันมา แต่สมมติว่าหากเอกสารหนึ่งมีประโยคที่เป็น “คำชม” มีจำนวนมาก ก็อาจจะทำให้วิเคราะห์ว่าเอกสารนี้แสดงความพึงพอใจต่อการเรียนการสอนทั้ง ๆ ที่ในเอกสารนั้นๆ อาจจะมีบางประโยคที่แสดงถึงความไม่พึงพอใจต่อการเรียนการสอน ซึ่งประเด็นนี้อาจจะทำให้ผลลัพธ์ลัพธ์ของการวิเคราะห์ความรู้สึกของผู้เรียนที่มีต่อการเรียนการสอนไม่ถูกต้อง

ดังนั้นในงานวิจัยฉบับนี้ จึงนำเสนอการประยุกต์กระบวนการของการวิเคราะห์ความรู้สึกมาใช้ในการวิเคราะห์คำติชมหรือความคิดเห็นจากผู้เรียนในโดเมนของการศึกษาระดับมัธยม โดยเป็นการวิเคราะห์ความรู้สึกในระดับประโยค (Sentence Level) เพื่อวิเคราะห์ความรู้สึกทีละประโยคว่าเป็นความรู้สึกขั้วบวก (Positive) หรือเป็นความรู้สึกขั้วลบ (Negative) จากนั้นจึงนำผลลัพธ์ของการวิเคราะห์ประโยคที่ได้มาสรุปสุดท้าย ว่าเอกสารที่แสดงคำติชมหรือความคิดเห็นจากผู้เรียนนั้นมีแนวโน้มความคิดเห็นไปในทิศทางใด

1.2 วัตถุประสงค์ของการวิจัย

เพื่อนำเสนอกระบวนการการจำแนกความรู้สึกระดับประโยคแบบ 2 ขั้ว คือ ขั้วบวก (Positive) และขั้วลบ (Negative) ในการวิเคราะห์คำติชมหรือความคิดเห็นจากผู้เรียนระดับมัธยมศึกษาตอนต้น

1.3 ความสำคัญของการวิจัย

ปัจจุบันในระบบการศึกษาหลายโรงเรียนได้มีการวัดและประเมินผลการเรียนรู้โดยได้นำเอากระบวนการวิเคราะห์ความรู้สึกของผู้เรียนเป็นตัวช่วยในการปรับปรุงรูปแบบการเรียนการสอนเพื่อให้มีประสิทธิภาพที่ดีหรือเหมาะสมกับผู้เรียนมากขึ้น การวิเคราะห์ความรู้สึก เป็นกระบวนการในการระบุและจัดประเภทความคิดเห็นของผู้ใช้จากข้อความเป็นความรู้สึกต่างๆ เช่น เชิงบวก เชิงลบ หรือเป็นกลาง ด้านอารมณ์ เช่น มีความสุข เศร้า โกรธ การประยุกต์ใช้กระบวนการของการวิเคราะห์ความรู้สึกมาใช้ในการวิเคราะห์คำติชม หรือความคิดเห็นจากผู้เรียนในโดเมนของการศึกษาระดับมัธยมจึงมีความสำคัญเป็นอย่างยิ่ง

1.4 ขอบเขตขอวการวิจัย

1. นำเสนอการประยุกต์กระบวนการการจำแนกความรู้สึกระดับประโยคในภาษาไทยของการวิเคราะห์ความรู้สึกมาใช้ในการวิเคราะห์คำติชมหรือความคิดเห็นจากผู้เรียนในโดเมนของการศึกษาระดับมัธยมศึกษา

2. คุณลักษณะในความคิดเห็นของนักเรียนจะพิจารณาใน 3 คุณลักษณะ คือ ผู้สอน (Teacher) บทเรียน (Lesson) และสภาพแวดล้อมในการเรียน (Learning Environment)

3. ข้อมูลที่ใช้ในการศึกษา จะรวบรวมมาจากความคิดเห็นของนักเรียนอย่างน้อย 3 โรงเรียนในจังหวัดร้อยเอ็ด ซึ่งเป็นข้อคิดเห็นในรูปแบบภาษาไทย โดยเก็บข้อมูล 3 ชุดข้อมูล

(1) ข้อมูลชุดที่ 1: เป็นชุดข้อมูลความรู้สึกในแต่ละคุณลักษณะ เพื่อใช้ในการสร้างตัววิเคราะห์คุณลักษณะ (Aspect Analyzer)

(2) ข้อมูลชุดที่ 2: เป็นชุดข้อมูลที่จะใช้ในการสร้างตัววิเคราะห์ความรู้สึก (Sentiment Analyzer)

(3) ข้อมูลชุดที่ 3: เป็นชุดข้อมูลที่จะใช้ในการทดลอง (Experiment)

4. ข้อมูลที่ใช้ในการศึกษามีผู้เชี่ยวชาญด้านภาษาไทยในระดับโรงเรียนช่วยในการจัดทำ Ground truth จำนวนอย่างน้อย 3 คน

5. กระบวนการวิเคราะห์ความรู้สึกที่ใช้ในการศึกษานี้จะเป็นกระบวนการแบบมีผู้สอน (Supervised Method) ที่เป็นการวิเคราะห์ความรู้สึกแบบ 2 ขั้ว คือ ขั้วบวก (Positive) และ ขั้วลบ (Negative) โดยจะศึกษาอัลกอริทึมแบบมีผู้สอน (Supervised Machine Learning) สำหรับการจำแนกความรู้สึกจากเอกสารความคิดเห็นแบบข้อความอย่างน้อย 2 ตัว

6. การตัดคำภาษาไทยจะใช้การตัดคำแบบพจนานุกรม (Dictionary-based Word Segmentation) โดยในงานวิจัยนี้เลือกใช้แบบวิธีการตัดคำด้วยพจนานุกรมแบบสอดคล้องมากที่สุด (Dictionary-based Word Segmentation with Maximal Matching)

7. ในการประเมินกระบวนการวิเคราะห์ความรู้สึกจะใช้ค่าความระลึก (Recall) ค่าความแม่นยำ (Precision) ค่าเอฟ (F-measure หรือ F-score หรือ F1) และค่าความถูกต้อง (Accuracy)

1.5 นิยามศัพท์เฉพาะ

1. การวิเคราะห์ความรู้สึก คือ เป็นการวิเคราะห์อารมณ์และความรู้สึกจากข้อความ เช่น ความรู้สึกด้านบวก (Positive) ความรู้สึกด้านลบ (Negative)

2. ความคิดเห็นของนักเรียน คือ ความคิดเห็นเกี่ยวกับการจัดกิจกรรมการเรียนการสอนของนักเรียนระดับชั้นมัธยมศึกษาตอนต้น

3. การวิเคราะห์ความคิดเห็นของนักเรียน คือ การวิเคราะห์เพื่อให้ทราบถึงความรู้สึกของนักเรียนที่เกี่ยวข้องกับการเรียนการสอนว่าออกมาในเชิงบวกหรือเชิงลบ

4. การวิเคราะห์ความคิดเห็นเชิงบวก คือ กระบวนการทางความคิดของนักเรียนที่เกิดจากการรับรู้และแปลความหมายจากการเรียนไปในทางที่ดี

5. การวิเคราะห์ความคิดเห็นเชิงลบ คือ กระบวนการทางความคิดของนักเรียนที่เกิดจากการรับรู้และแปลความหมายจากการเรียนไปในทางที่ไม่ดี

6. กระบวนการแบบมีผู้สอน (Supervised Method) คือกระบวนการที่ใช้อัลกอริทึมการเรียนรู้แบบมีผู้สอนในการสร้างโมเดลเพื่อการจำแนกความรู้ออกเป็น 2 ขั้ว คือ ขั้วบวก (Positive) และ ขั้วลบ (Negative)

บทที่ 2
ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

ในบทนี้จะกล่าวถึงแนวคิด วิธีการ และอัลกอริทึมที่เกี่ยวข้อง ต่อการวิจัยและการวิเคราะห์ความรู้สึกของนักเรียนมัธยมที่เกี่ยวข้องกับกระบวนการการเรียนการสอน โดยแนวคิดและอัลกอริทึมที่เกี่ยวข้องมีดังต่อไปนี้

2.1 การวิเคราะห์ความรู้สึก (Sentiment Analysis)

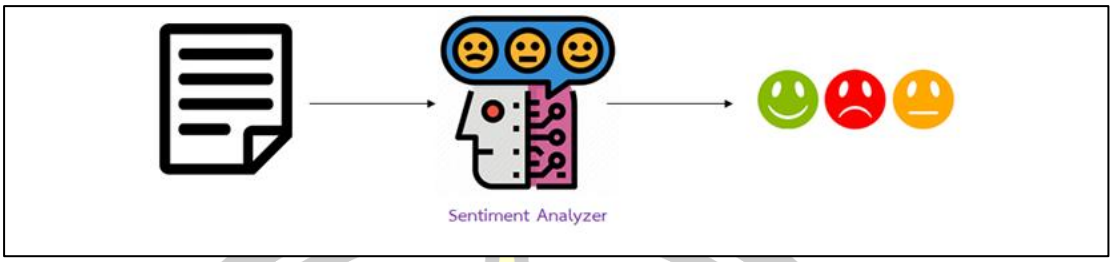
การวิเคราะห์ความรู้สึก [8] คือ งานวิจัยที่อยู่ในกลุ่มของการประมวลผลภาษาธรรมชาติ (Natural Language Processing: NLP) ที่มีกระบวนการมุ่งเน้นการวิเคราะห์และตรวจสอบความรู้สึก (Opinion) ของผู้คนจากข้อความ (Text) ที่คนเหล่านั้นเขียนหรือโพสต์เอาไว้ เพื่อบ่งบอกความรู้สึกของตนเองที่มีต่อบางสิ่งบางอย่าง เช่น ความรู้สึกดี (Positive หรือ Good) หรือความรู้สึกที่ไม่ดีหรือไม่ชอบ (Negative หรือ Bad)

ส่วนใหญ่งานวิจัยด้านการวิเคราะห์ความรู้สึกนั้น มีการนำไปประยุกต์ใช้กับข้อมูลที่เป็นข้อความที่เรียกว่า “Product Review” จุดประสงค์ก็เพื่อการวิเคราะห์ความรู้สึกของลูกค้าที่มีต่อสินค้าของบริษัทหรือองค์กรต่างๆ เนื่องจากการวิเคราะห์ความพึงพอใจจากข้อมูล Rating บางครั้งจะไม่สามารถระบุถึงปัญหาหรือความรู้สึกของลูกค้าที่มีต่อสินค้าที่แท้จริงได้ เช่น การที่ลูกค้าซื้อคอมพิวเตอร์ไป 1 เครื่อง ลูกค้าอาจจะให้คะแนนเฉลี่ยเกี่ยวกับคอมพิวเตอร์ยี่ห้อนั้นๆ ไว้ที่ 3 จากคะแนนเต็ม 5 แต่หัวข้อต่างๆ ที่สอบถามไปยังลูกค้าอาจจะไม่ครอบคลุมในทุกๆ กรณี ที่เป็นความต้องการหรือความคาดหวังต่อสินค้าของลูกค้า ดังนั้นลูกค้าก็อาจจะไปเขียนแสดงความรู้สึกเกี่ยวกับสินค้ายี่ห้อนั้นๆ ไว้ในส่วนอื่นๆ เช่น ใน Blog, Twitter หรือ Facebook ของตน โดยความรู้สึกเหล่านั้นอาจจะเป็นมุมมองที่เจ้าของสินค้ามองข้ามหรือคาดไม่ถึง ซึ่งหากนำความรู้สึกเหล่านั้นมาพิจารณา ก็อาจจะสามารถนำมาปรับปรุงสินค้าของตนเองได้

ปัจจุบันเทคโนโลยีด้านการวิเคราะห์ความรู้สึก เริ่มเข้ามามีบทบาทเป็นอย่างมากในหลายๆ องค์กร ทั้งธุรกิจที่เกี่ยวข้องการภาพยนตร์ การศึกษา และการให้บริการด้านการแพทย์ โดยเทคโนโลยีด้านการวิเคราะห์ความรู้สึกได้ถูกรวมเข้าไว้ในระบบ commercial website หรือ Customer Relationship Management (CRM) ของแต่ละบริษัทหรือองค์กร เพื่อให้ง่ายต่อการวิเคราะห์ความรู้สึกของลูกค้าหรือผู้ใช้บริการ จนนำไปสู่การแก้ปัญหาอย่างรวดเร็ว

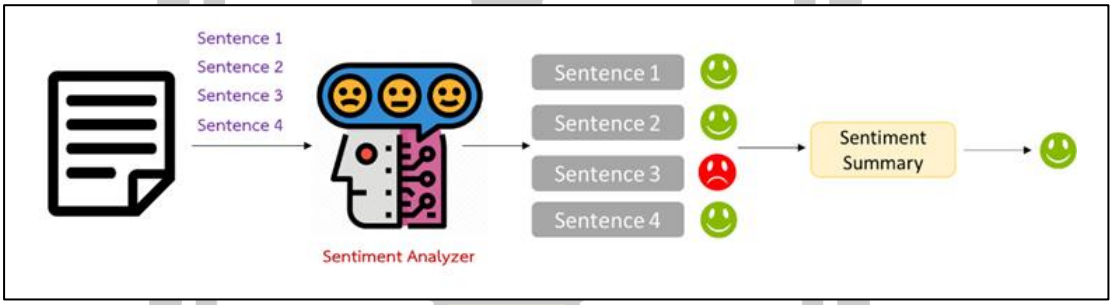
การวิเคราะห์ความรู้สึกสามารถแบ่งได้ 3 ระดับ [9] ดังนี้

(1) การวิเคราะห์ความรู้สึกระดับเอกสาร (Document Level Analysis) เป็นการวิเคราะห์ข้อความแสดงความคิดเห็นในแบบหยาบ เนื่องจากเป็นการนำข้อความแสดงความคิดเห็นทั้งหมดจากเอกสารมาสรุป แยกขั้วความคิดเห็นเป็นขั้วบวก ขั้วลบ หรือเป็นกลาง



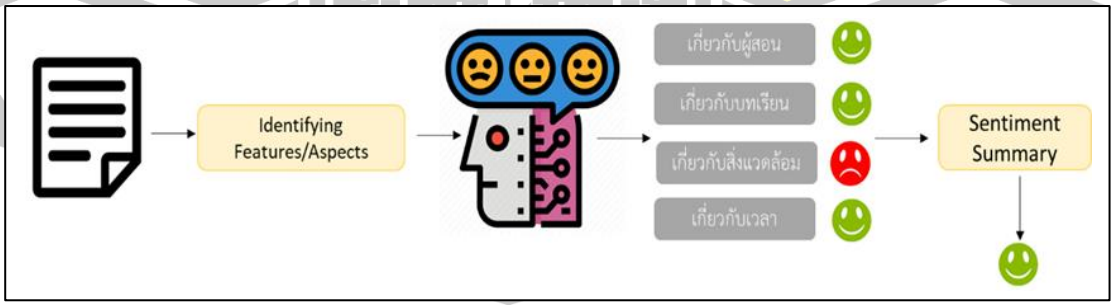
รูปที่ 2.1 แสดงขั้นตอนการวิเคราะห์ความรู้สึกระดับเอกสาร (Document Level Analysis)

(2) การวิเคราะห์ความรู้สึกระดับประโยค (Sentence Level Analysis) เป็นการวิเคราะห์ข้อความแสดงความคิดเห็น โดยแยกข้อความที่เป็นข้อความแสดงความคิดเห็นออกมาจากข้อความที่เป็นข้อเท็จจริงในระดับที่เป็นประโยค แล้วนำมาแยกขั้วความคิดเห็นเป็นขั้วบวก ขั้วลบ หรือเป็นกลาง



รูปที่ 2.2 แสดงขั้นตอนการวิเคราะห์ความรู้สึกระดับประโยค (Sentence Level Analysis)

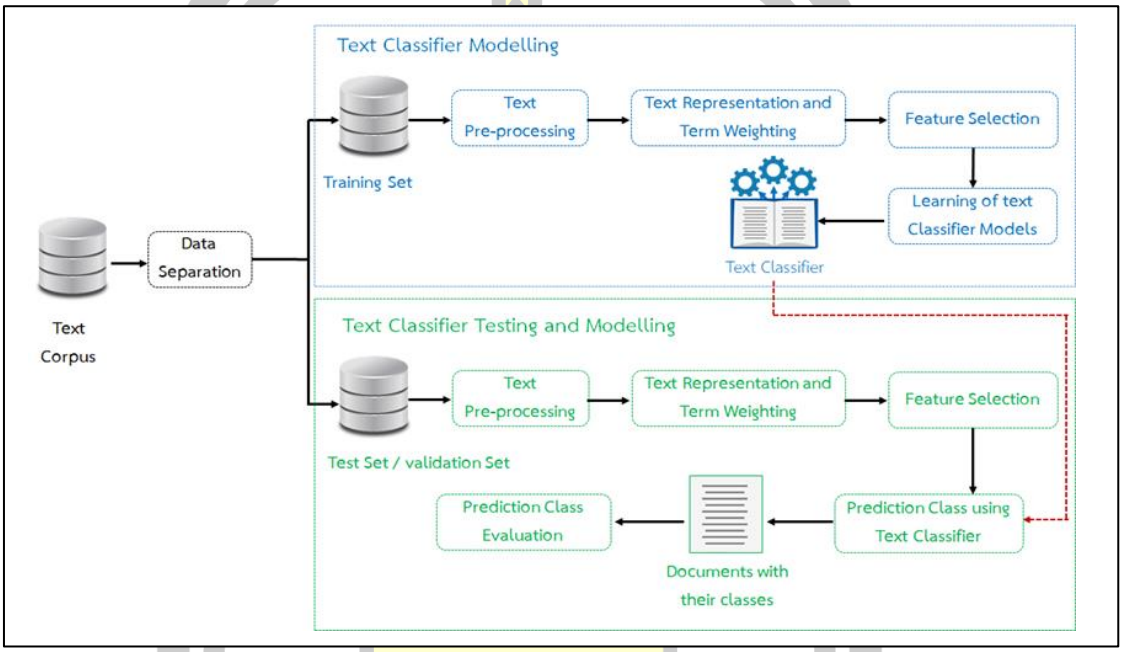
(3) การวิเคราะห์ความรู้สึกระดับคุณลักษณะ (Aspect/Feature Level Analysis) เป็นการวิเคราะห์ข้อความแสดงความคิดเห็น โดยแยกคุณลักษณะที่สนใจหรือหัวข้อที่ถูกแสดงความคิดเห็นออกมาก่อน แล้วจึงนำมาแบ่งขั้วความคิดเห็นเป็นขั้วบวก ขั้วลบ หรือเป็นกลาง และนำมาจัดกลุ่มเข้ากับคำที่มีความหมายเหมือนกันในแต่ละคุณลักษณะ ซึ่งระบบจะวิเคราะห์ข้อความแสดงความคิดเห็นในระดับคุณลักษณะ แล้วนำข้อมูลที่ได้มาประมวลผลและแสดงผลลัพธ์ที่ให้ผู้ใช้งานเข้าใจได้ง่ายขึ้น



รูปที่ 2.3 แสดงขั้นตอนการวิเคราะห์ความรู้สึกระดับคุณลักษณะ (Aspect/Feature Level Analysis)

2.2 กรอบการดำเนินงานทั่วไปของการจำแนกความรู้สึก (Generic Method of Sentiment Classification)

การสร้างโมเดลเพื่อการจำแนกเอกสารข้อความ (Text Classifier Modelling) คือ การเตรียมเอกสารข้อความ การแทนเอกสารและการให้น้ำหนักคำ การเลือกฟีเจอร์ และการเรียนรู้ตัวจำแนกเอกสารข้อความ

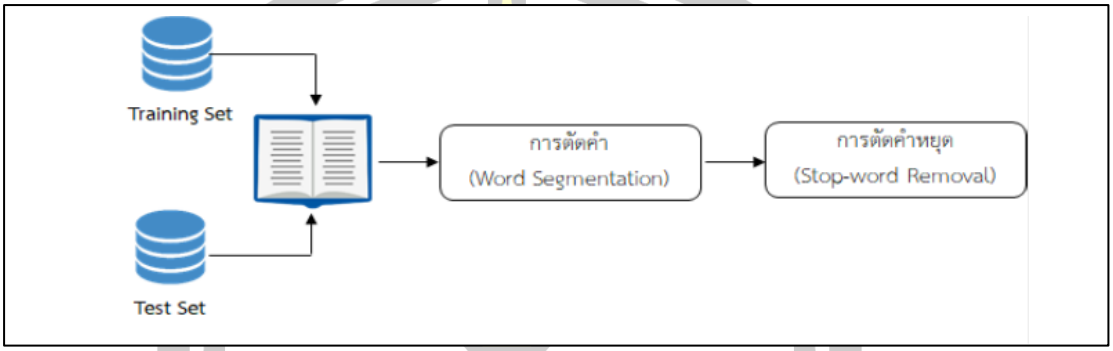


รูปที่ 2.4 กรอบการดำเนินงานสำหรับการจำแนกเอกสารข้อความ

การประเมินโมเดลเพื่อจำแนกเอกสารข้อความ (Text Classifier Evaluation) คือ การวัดประสิทธิภาพโมเดล เพื่อทดสอบว่าโมเดลที่สร้างขึ้นมามีความถูกต้องมากน้อยเพียงใดมีประสิทธิภาพในการจำแนกกลุ่มหรือคลาสของข้อมูลหรือไม่และเพื่อทำการเปรียบเทียบกับโมเดลชนิดอื่น มีขั้นตอนได้แก่ การเตรียมเอกสารข้อความ การแทนเอกสารและการให้น้ำหนักคำ และการคัดเลือกฟีเจอร์ ผลลัพธ์ที่ได้เอกสารจะถูกแสดงในรูปแบบของแบบจำลองปริภูมิเวกเตอร์ที่แสดงเอกสารและฟีเจอร์ของเอกสาร จากนั้นนำไปทดสอบกับโมเดลโดยใช้ข้อมูลชุดทดสอบหรือข้อมูลชุดตรวจสอบ และนำผลลัพธ์ที่ได้ไปประเมินประสิทธิภาพ เครื่องมือที่นิยมใช้ในการประเมินประสิทธิภาพของโมเดลเพื่อการจำแนกเอกสารข้อความ ได้แก่ ค่าความระลึก (Recall) ค่าความแม่นยำ (Precision) ค่า F-measure: F1 หรือ F-score และค่าความถูกต้อง (Accuracy)

2.2.1 การเตรียมเอกสารข้อความ (Text Pre-processing)

การเตรียมเอกสารข้อความ คือ ขั้นตอนของการเตรียมเอกสารข้อความให้อยู่ในรูปแบบที่เหมาะสม ก่อนที่จะถูกนำไปสร้างโมเดลเพื่อการจำแนกเอกสารข้อความ



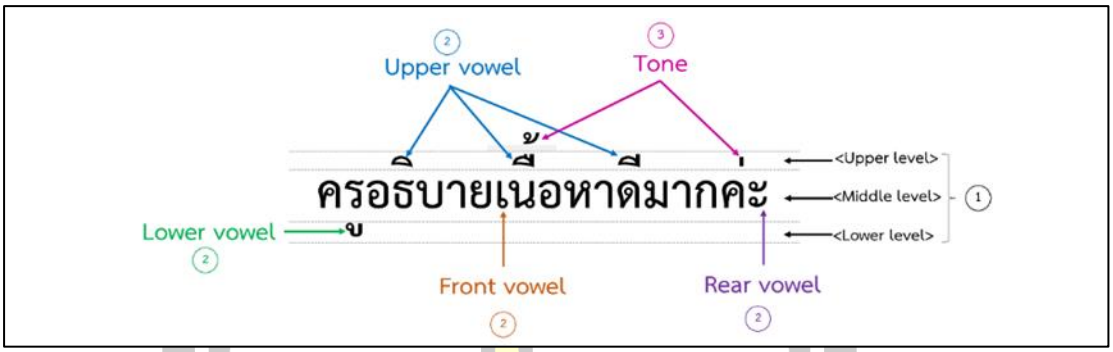
รูปที่ 2.5 ภาพรวมของขั้นตอนการเตรียมเอกสารข้อความ

2.2.1.1 การตัดคำ (Word Segmentation)

การตัดคำ [10] คือ การแยกประโยคในเอกสารออกเป็นส่วนย่อยๆ ที่มีความหมายทางภาษา เรียกว่า “คำ (Word)” ซึ่งการเลือกเทคนิคในการตัดคำนั้นจะขึ้นอยู่กับภาษาที่ใช้ในเอกสารที่ต้องการจำแนกข้อความ เช่น ถ้าเอกสารข้อความที่เป็นภาษาอังกฤษ จะทำการแยกประโยคจากอักขระ ที่เป็นตัวคั่น (Word Delimiter) เช่น เครื่องหมายยัติภังค์ (-) หรือมหัพภาค (.) หรือจุลภาค (,) หรือ ช่องว่างระหว่างคำ (White Space) แต่ถ้าเป็นภาษาอื่นๆ เช่น ภาษาจีน ภาษาญี่ปุ่น หรือภาษาไทย จะไม่สามารถใช้เทคนิคการตัดคำแบบภาษาอังกฤษได้ เนื่องจากภาษาเหล่านี้ไม่มีสัญลักษณ์ที่แสดงขอบเขตของคำที่ชัดเจนและมีรูปแบบการเขียนที่ตัวอักขระจะเรียงติดต่อกัน จึงทำให้ยากต่อการระบุขอบเขตของคำ

กระบวนการในการตัดคำภาษาไทยนั้นจะค่อนข้างซับซ้อน เพราะการเขียนข้อความภาษาไทย มีลักษณะการเขียนเรียงต่อเนื่องกันทั้งประโยคโดยไม่มีเครื่องหมายวรรคตอน แต่ในบางครั้งก็อาจจะมีการแบ่งวรรคตอนบ้าง เพื่อเป็นการแบ่งแยกประโยคหรือวลีด้วยช่องว่าง ดังนั้นจึงใช้เทคนิคเดียวกันกับการตัดคำแบบภาษาอังกฤษไม่ได้ เนื่องจากการเขียนหรือแสดงประโยคในภาษาไทย จะไม่มีเครื่องหมายแสดงขอบเขตของ “คำ” หรือ “ประโยค” ที่ชัดเจนเหมือนภาษาอังกฤษ หรือ ภาษาทางฝั่งยุโรป

โครงสร้างประโยคในภาษาไทย เป็นการเขียนด้วยอักขระที่เรียงต่อกันจนจบประโยคโดยไม่มีสัญลักษณ์ของการแยกคำที่ชัดเจน (No Explicit Word Boundary) โดยรูปแบบการเขียนในภาษาไทยจะมีลักษณะดังรูปที่ 2.6



รูปที่ 2.6 รูปแบบลักษณะการเขียนในภาษาไทย

2.2.1.1.1 การตัดคำด้วยพจนานุกรมแบบเลือกคำที่ยาวที่สุด (Dictionary-based Word Segmentation with Longest Matching)

ได้นำเอากฎทางด้านอักขระมาใช้ร่วมกับพจนานุกรมในการตัดคำ มีขั้นตอนดังนี้

(1) อ่านข้อความจากเอกสารเข้ามาทีละประโยค

(2) อ่านประโยคจากซ้ายไปขวา

1. อ่านอักขระตามกฎอักขระไทยและเทียบกับพจนานุกรมไปเรื่อยๆ จนกว่าจะพบ “คำ” แม้จะพบ “คำ” แล้ว แต่ให้อ่านอักขระแล้วเทียบกับพจนานุกรมต่อไป เพื่อหา “คำที่ยาวที่สุด”
2. หากอ่านไปเรื่อยๆ แล้วไม่พบ “คำ” ใดๆ ที่ตรงตามพจนานุกรมเลย ให้ย้อนกลับ (Backtrack) ไปยังตำแหน่งที่พบคำก่อนหน้า แล้วตัด “คำ” นั้นออกมา
3. ดำเนินการตามขั้นตอนที่ 1 - 2 กับประโยคที่เหลือ

(3) ดำเนินการตามขั้นตอนที่ 2 จนกว่าจะหมดข้อความในเอกสาร

2.2.1.1.2 การตัดคำด้วยพจนานุกรมแบบที่เลือกคำที่มากที่สุด (Dictionary-based Word Segmentation with Maximal Matching)

การตัดคำด้วยพจนานุกรมแบบที่เลือกคำที่สอดคล้องมากที่สุด ถูกนำมาใช้เพื่อแก้ปัญหาการตัดคำภาษาไทยด้วยพจนานุกรมแบบเลือกคำที่ยาวที่สุด การเลือกคำยาวมากที่สุดตั้งแต่ครั้งแรกที่พบ อาจทำให้ได้ผลลัพธ์ของการตัดคำผิดพลาด เช่น “เพลาชาย” มีความหมายว่า “ช่วงเวลาบ่าย”เมื่อทำการตัดคำด้วยพจนานุกรมแบบเลือกคำที่ยาวที่สุดจะตัดคำได้ 2 คำ คือคำว่า “เพลา” และ “ชาย” ซึ่งในความเป็นจริงแล้วควรจะตัดคำได้คำว่า “เพ”, “ลา” และ “ชาย” เพราะคำที่ขีดเส้นใต้ในประโยค “เพลา” ถ้าเลือกใช้การตัดคำด้วยพจนานุกรมแบบเลือกคำที่ยาวที่สุด จะสามารถตัดคำได้ 2 แบบ คือ ได้คำว่า “เพลา” และ คำว่า “เพ”, “ลา” เมื่อเปรียบเทียบความยาวของคำแล้วพบว่า คำว่า “เพลา” มีความยาวมากกว่าคำว่า “เพ”, “ลา” ผลลัพธ์ที่ได้จึงเป็นคำว่า “เพลา” ซึ่งเป็นประโยคที่ไม่ถูกต้องตรงตามความหมาย ประโยค “เพลาชาย” จะสามารถตัดคำ ที่เป็นไปได้ใน 2 รูปแบบ คือ

รูปแบบที่ 1: จะได้คำว่า “เพลา”และ “ชาย”
รูปแบบที่ 2: จะได้คำว่า “เพ”, “ลา”, และ “ชาย”

และบางครั้งยังพบว่า การตัดคำในบางประโยคจะได้รูปแบบการตัดคำที่แตกต่างกัน แต่พบว่ามีจำนวนคำที่ตัดได้เท่ากัน จึงทำให้ยากต่อการตัดสินใจว่าควรเลือกรูปแบบการตัดคำแบบใดเป็นรูปแบบที่ถูกต้อง เช่น ในประโยค “ช่างวัดรอบอก” สามารถตัดคำได้ รูปแบบดังนี้

รูปแบบที่ 1: จะได้คำว่า “ช่าง”, “วัด”, “รอบ” และ “อก”
รูปแบบที่ 2: จะได้คำว่า “ช่าง”, “วัด”, “รอ” และ “บอก”

จากปัญหาที่พบดังกล่าว จึงได้มีการปรับปรุงพัฒนาโดยใช้อัลกอริทึมการประยุกต์วิธีการแบบละโมภ (Greedy Algorithm) เข้ามาช่วย คือนอกจากการตัดคำจากรูปแบบคำที่สอดคล้องมากที่สุดแล้ว ยังเลือกคำที่ยาวที่สุดอีกด้วย ดังนั้นหากใช้อัลกอริทึมการตัดคำด้วยพจนานุกรมแบบที่เลือกคำที่สอดคล้องมากที่สุดที่ได้รับการปรับปรุง จากประโยค “ช่างวัด รอบอก” จะได้การตัดคำตามรูปแบบที่ 1 (“ช่าง”, “วัด”, “รอบ” และ “อก”) จึงเป็นการตัดคำที่ถูกต้องสำหรับประโยค “ช่างวัดรอบอก”

2.2.1.2 การตัดคำหยุด (Stop-word Removal)

การตัดคำหยุด [11] คือ กระบวนการตัดคำหรือสัญลักษณ์ที่พบบ่อยมากในเอกสาร แต่คำหรือสัญลักษณ์เหล่านั้นไม่ได้ส่งผลต่อการจัดกลุ่มเอกสาร ดังนั้นเมื่อทำการตัดออกแล้วไม่ทำให้ใจความในเอกสารนั้นๆ เปลี่ยนไป ตัวอย่างคำที่เป็นคำหยุด เช่น คำบุพบทคำสันธาน คำสรรพนาม เป็นต้น

สมมติคำว่า คำว่า “ฉัน”, “อยาก”, “ไป”, “มี” และ “ที่” เป็นคำหยุดในภาษาไทย ดังนั้นสามารถยกตัวอย่างการตัดคำหยุดทั้งในเอกสารภาษาไทยได้ดังแสดงใน ตารางที่ 2.1

ตารางที่ 2.1 ตัวอย่างการตัดคำหยุดในภาษาไทย

ประโยคต้นฉบับ	ประโยคที่ผ่านการตัดคำ	ภายหลังตัดคำหยุด
ฉันอยากไปเรียนคอมพิวเตอร์	ฉัน/ อยาก/ ไป/ เรียน/ คอมพิวเตอร์	เรียน, คอมพิวเตอร์
วิทยาศาสตร์มีการทดลองที่สนุก	วิทยาศาสตร์/ มี/ การ/ ทดลอง/ ที่/ สนุก	วิทยาศาสตร์, การ, ทดลอง, สนุก

2.2.2 การแทนเอกสารข้อความ (Text Representation)

การแทนเอกสารข้อความ คือ การแสดงข้อความที่ไม่มีโครงสร้างข้อมูลให้อยู่ในรูปแบบที่ง่ายต่อการประมวลผล เนื่องจากคอมพิวเตอร์ไม่สามารถเรียนรู้ และจำแนกหมวดหมู่ของเอกสารที่เป็นภาษาธรรมชาติได้โดยตรง จึงต้องแปลงเอกสารให้อยู่ในรูปแบบที่คอมพิวเตอร์สามารถเรียนรู้ได้ โดยขั้นตอนนี้เรียกว่า การทำดัชนี (Indexing) [12] เพื่อสร้างตัวแทนเนื้อหาของเอกสาร (Document Representation) สำหรับใช้ในกระบวนการเรียนรู้ วัตถุประสงค์ของการสร้างดัชนี คือ

การคำนวณหาค่าที่จะมาใช้เป็นค่าคุณลักษณะของเอกสาร หรืออาจจะเรียกได้ว่าการหาค่าน้ำหนัก (Term Weighting)

	W_1	W_2	...	W_k	...	W_v
D_1	W_{11}	W_{12}	...	W_{1k}	...	W_{1v}
D_2	W_{21}	W_{22}	...	W_{2k}	...	W_{2v}
D_3	W_{31}	W_{32}	...	W_{3k}	...	W_{3v}
....	....	....	....	....	....	....
D_N	W_{N1}	W_{N2}	...	W_{Nk}	...	W_{Nv}

รูปที่ 2.7 Bag of Words

การสร้างดัชนีโดยทั่วไปที่นิยมใช้กัน จะเริ่มจากการสร้างเวกเตอร์ตัวแทนเอกสาร จากนั้นจะสร้างเมตริกซ์ของกลุ่มเอกสารขึ้นจากเวกเตอร์เอกสารทั้งหมดในกลุ่ม ซึ่งใช้วิธีหาความถี่ของคำที่ปรากฏในเอกสารที่ผ่านการตัดคำมาเป็นค่าน้ำหนัก ถ้าคำใดที่ผ่านการตัดคำมีปริมาณมาก ก็จะมีค่าความถี่มาก ซึ่งจะส่งผลให้ได้ค่าน้ำหนักที่มีค่าสูงมากตาม เมื่อถึงขั้นตอนนี้จะได้รูปแบบที่มีลักษณะของการแสดงความสัมพันธ์ระหว่างคำ (Words: W) และเอกสารสารทั้งหมด (Documents: D) ด้วยเวกเตอร์ 2 มิติ ซึ่งคำที่ได้นั้นต้องผ่านการทำดัชนีและการตัดคำหยุด (Stop-words) ออกไปและเอกสารทั้งหมดอยู่ในรูปแบบ Vector Space Model หรือบางครั้งเรียกรูปแบบนี้ว่า Bag of Words โดยสามารถแสดงได้ดังรูปที่ 2.7

ในงานวิจัยรูปแบบของการแทนเอกสารข้อความที่ได้รับความนิยม คือ แบบจำลองปริภูมิเวกเตอร์ (Vector Space Model: VSM) และ VSM จะกำหนดให้เอกสารแต่ละฉบับเปรียบเสมือนเวกเตอร์ของฟีเจอร์ (Features) โดยที่ขนาดของเวกเตอร์จะขึ้นอยู่กับจำนวนของฟีเจอร์ที่ปรากฏอยู่ในเอกสารฉบับนั้น และส่วนมากคุณลักษณะหรือฟีเจอร์ของเอกสารข้อความที่นิยมใช้เป็น “คำ” ด้วยเหตุนี้เราจึงนิยมเรียกโครงสร้างแบบ VSM นี้ว่า “ถุงคำ (Bag-of-Words: BOW)” แต่ก็มีสิ่งที่ทำให้ถุงคำ และ VSM มีความแตกต่างกันก็คือ ในเรื่องมุมมองของการแทนเอกสารข้อความนั่นคือถุงคำเป็นโครงสร้างที่ใช้จัดเก็บ “คำ” หรือ “เทอม (Term)” สามารถสกัดได้จากเอกสารข้อความโดยไม่มีการเรียงลำดับคำ (Unordered List of Words) และปราศจากสารสนเทศอื่นๆ เช่น ชนิดของคำ (Part of Speech: POS) โครงสร้าง (Syntax) หรือความหมาย (Semantics) แต่ในส่วนของ VSM ก็คือถุงคำได้มีการเพิ่มสารสนเทศที่เรียกว่า “น้ำหนักคำ (Term Weight)” เข้าไปด้วย นอกจากจะใช้ VSM ในการแทนข้อมูลเอกสารข้อความ VSM ยังสามารถใช้ในการแทนข้อมูลประเภทอื่นๆ ได้ด้วย เช่น ข้อมูลภาพ (Image) หรือข้อมูลไมโครอาร์เรย์ ของดีเอ็นเอ (DNA Microarray) เป็นต้น

สมมติว่ามีเอกสารข้อความ 2 เอกสาร ได้แก่

D1: ฉันอยากไปเรียนคอมพิวเตอร์

D2: วิทยาศาสตร์มีการทดลองที่สนุก

เมื่อผ่านการเตรียมเอกสารข้อความด้วยขั้นตอนของการตัดคำแล้วเอกสารเหล่านี้สามารถแสดงในรูปแบบของถุงคำได้ดังตารางที่ 2.2

ตารางที่ 2.2 ตัวอย่างการแทนเอกสารข้อความด้วย BOW

	คอมพิวเตอร์	วิทยาศาสตร์	คณิตศาสตร์	เรียน	การ	ทดลอง	ไม่	สนุก	Class
D1	1	0	0	1	0	0	0	0	YES
D2	0	1	0	0	1	1	0	1	YES

ในตารางที่ 2.2 ค่า 0 หมายถึง คำที่ไม่ปรากฏ (Absence) ในเอกสาร ในขณะที่ค่า 1 หมายถึงคำที่ปรากฏ (Presence) ในเอกสารนั่นเอง

2.2.3 การให้น้ำหนักคำ (Term Weighting)

2.2.3.1 การให้น้ำหนักคำ (Term Weighting)

การให้น้ำหนักคำ [13] คือ การกำหนดค่าน้ำหนักให้กับ “คำ” เพื่อแสดงให้เห็นว่าเอกสารหรือข้อความของคำที่ปรากฏในเอกสารหรือในแต่ละคลาสมีนัยสำคัญมากน้อยเพียงใด แสดงให้เห็นถึงความสัมพันธ์ระหว่างเอกสารและข้อความได้มากขึ้น ก่อนที่เราจะให้น้ำหนักคำและสามารถนำคำนั้นไปวิเคราะห์หาความถี่ได้ เราต้องแปลงข้อความเอกสารให้อยู่ในรูปแบบของ Vector Space Model (VSM) หรือ Bag-of-Word: BOW ก่อน การให้น้ำหนักคำมีหลายวิธีดังนี้

การให้น้ำหนักคำแบบไม่มีผู้สอน (Unsupervised Term Weighting) [14] คือ การให้น้ำหนักของคำภายในเอกสารทั้งหมดโดยไม่มีการแบ่งคลาสอาจทำให้ประสิทธิภาพในการจำแนกข้อความลดลง การให้น้ำหนักคำแบบไม่มีผู้สอนมีหลายรูปแบบ เช่น การให้น้ำหนักแบบพิจารณาความถี่ของคำ (Term Frequency: tf) และ ความถี่ของคำ - ความถี่เอกสารผกผัน (Term Frequency – Inverse Document Frequency: tf-idf)

การให้น้ำหนักคำแบบมีผู้สอน (Supervised Term Weighting) [15] จะใช้สำหรับเอกสารชุดฝึกเพื่อคำนวณหาน้ำหนักของคำ การให้น้ำหนักคำแบบมีผู้สอนมีหลายรูปแบบ เช่น ความถี่ของคำ – ช่วงเวลาแรงโน้มถ่วงผกผัน (Term Frequency-Inverse Gravity Moment: tf-igm)

ตัวอย่างในที่นี้จะสมมุติให้เอกสารเป็นเอกสาร ภาษาไทย 4 ฉบับ ดังนี้

D1: ฉันอยากไปเรียนคอมพิวเตอร์
D2: วิทยาศาสตร์มีการทดลองที่สนุก
D3: ฉันไม่อยากเรียนคณิตศาสตร์
D4: วิชาภาษาไทยน่าเบื่อ

จากทั้ง 4 เอกสาร เมื่อผ่านขั้นตอนการตัดคำหยุด การคัดเลือกคำด้วยพจนานุกรม สามารถแสดงได้ดังตารางที่ 2.3

ตารางที่ 2.3 แสดงคำสำคัญที่ได้ภายหลังการคัดเลือกคำด้วยพจนานุกรม

ประโยคต้นฉบับ	ประโยคที่ผ่านการตัดคำ	ภายหลังตัดคำหยุด
ฉันอยากไปเรียนคอมพิวเตอร์	ฉัน/ อยาก/ ไป/ เรียน/ คอมพิวเตอร์	เรียน, คอมพิวเตอร์
วิทยาศาสตร์มีการทดลองที่สนุก	วิทยาศาสตร์/ มี/ การ/ ทดลอง/ ที่/ สนุก	วิทยาศาสตร์, การ, ทดลอง, สนุก
ฉันไม่อยากเรียนคณิตศาสตร์	ฉัน/ ไม่/ อยาก/ เรียน/ คณิตศาสตร์	ไม่, เรียน, คณิตศาสตร์
วิชาภาษาไทยน่าเบื่อ	วิชา/ ภาษาไทย/ น่าเบื่อ	วิชา/ ภาษาไทย/ น่าเบื่อ

เมื่อได้ผลลัพธ์ดังตาราง 2.3 จะสามารถแสดงความสัมพันธ์ระหว่าง “คำสำคัญ” และ “เอกสาร” ในรูปแบบของ VSM หรือ BOW โดย POS แสดงคลาสที่เป็น Positive และ NEG แสดงคลาสที่เป็น NEGATIVE

ตารางที่ 2.4 แสดงความสัมพันธ์ระหว่างคำสำคัญและเอกสาร

	คอมพิวเตอร์	วิทยาศาสตร์	คณิตศาสตร์	เรียน	การ	ทดลอง	ไม่	สนุก	วิชา	ภาษาไทย	น่าเบื่อ	Class
D1	1	0	0	1	0	0	0	0	0	0	0	POS
D2	0	1	0	0	1	1	0	1	0	0	0	POS
D3	0	0	1	1	0	0	1	0	0	0	0	NEG
D4	0	0	0	0	0	0	0	0	1	1	1	NEG

2.2.3.2 การให้น้ำหนักคำแบบไม่มีผู้สอน (Unsupervised Term Weighting) การให้น้ำหนักคำแบบไม่มีผู้สอน

2.2.3.2.1 ให้น้ำหนักแบบ Term Frequency: tf

การให้น้ำหนักแบบ tf [13] เมื่อ tf เป็นความถี่ของคำหนึ่งๆ ที่พบในแต่ละเอกสาร ในที่นี้จะหาจากสมการที่เรียกว่า ค่าความถี่ที่สเกลแบบลอการิทึม (Logarithmic- scale frequency) นั่นคือ

$$\boldsymbol{tf(t,d) = log(1 + f_{t,d})} \quad (2.1)$$

โดยที่ $tf(t,d)$ หมายถึง จำนวนครั้งที่พบ t ในเอกสาร d

2.2.3.2.2 ให้น้ำหนักแบบ $tf - idf$

การให้น้ำหนักแบบ $tf - idf$ เป็นวิธีการสร้างตัวแทนเอกสารในรูปแบบของเวกเตอร์เพื่อใช้ในการจัดกลุ่มของเอกสารให้ตรงกับหมวดหมู่ที่ถูกกำหนดไว้ [14] โดย tf เป็นการหาความถี่ของคำหนึ่งๆ ที่พบในแต่ละเอกสาร และ idf ก็คือ global weight ที่เป็นการหาส่วนกลับของความถี่ของคำในเอกสาร หรือที่เรียกว่าระบบน้ำหนักความถี่เอกสารผกผัน ซึ่งหาได้จากสมการที่ 2

$$idf = \log N/df \quad (2.2)$$

โดยที่ N คือจำนวนเอกสารทั้งหมดในคลัง และ df คือจำนวนเอกสารที่มีคำๆ นั้นปรากฏอยู่

$$tf - idf = tf \times idf \quad (2.3)$$

จากสมการดังกล่าวเป็นวิธีการหาตัวแทนเวกเตอร์เพื่อนำไปค้นคืนสารสนเทศที่เป็นกลุ่มของเอกสาร ซึ่งวิธีนี้เป็นการให้น้ำหนักอย่างง่ายแต่ก็ได้รับการยอมรับว่ามีประสิทธิภาพที่น่าพอใจกับการจัดกลุ่มเอกสาร

จากข้อมูลข้างต้นจะเห็นว่ามีเอกสารทั้งหมด 4 เอกสาร เมื่อนำมาคำนวณหาค่าน้ำหนักด้วยสมการ $tf - idf$ จะสามารถแสดงขั้นตอนได้ดังนี้ คือ

ขั้นตอนที่ 1 : หาค่า tf ที่เป็นความถี่ของคำแต่ละคำที่อยู่ในเอกสารนั้นๆ ว่าพบกี่ครั้ง

ขั้นตอนที่ 2 : หาค่า idf คือการหาค่าส่วนกลับของแต่ละคำในเอกสารนั้นๆ

การคำนวณหา idf ทำได้โดยใช้สมการ $idf = \log N/df$ โดย N คือจำนวนเอกสารทั้งหมดในคลังเอกสาร และ df คือจำนวนเอกสารที่มีคำนั้นๆ ปรากฏอยู่ และสามารถคำนวณหาค่า idf ได้ดังนี้ในที่นี้จะให้ $N = 4$

$idf_{\text{คอมพิวเตอร์}} = \log(4/1) = 0.602$

$idf_{\text{วิทยาศาสตร์}} = \log(4/1) = 0.602$

$idf_{\text{คณิตศาสตร์}} = \log(4/1) = 0.602$

$idf_{\text{เรียน}} = \log(4/2) = 0.301$

$idf_{\text{การ}} = \log(4/1) = 0.602$

$idf_{\text{ทดลอง}} = \log(4/1) = 0.602$

$idf_{\text{ไม่}} = \log(4/1) = 0.602$

$idf_{\text{สนุก}} = \log(4/1) = 0.602$

$idf_{\text{วิชา}} = \log(4/1) = 0.602$

$idf_{\text{ภาษาไทย}} = \log(4/1) = 0.602$

$idf_{\text{น่าเบื่อ}} = \log(4/1) = 0.602$

ขั้นตอนที่ 3 : การคำนวณหาค่า $tf - idf$

ในขั้นตอนนี้จะเป็นการนำเอาค่า tf ที่ได้คูณเข้ากับค่า idf เช่น ในเอกสารที่ 1 คำที่พบได้แก่ “คอมพิวเตอร์” และ “เรียน” โดยคำเหล่านี้ ที่ปรากฏในเอกสารที่ 1 มีค่า tf เป็น 1 และ 1 ตามลำดับ เมื่อนำมาหาค่า $tf - idf$ จะได้ผลลัพธ์ ดังต่อไปนี้

tf-idf "คอมพิวเตอร์", D1 = 1 x 0.602 = 0.602
tf-idf "เรียน", D1 = 1 x 0.301 = 0.301
tf-idf "วิทยาศาสตร์", D2 = 1 x 0.602 = 0.602
tf-idf "การ", D2 = 1 x 0.602 = 0.602
tf-idf "ทดลอง", D2 = 1 x 0.602 = 0.602
tf-idf "คณิตศาสตร์", D3 = 1 x 0.602 = 0.602
tf-idf "เรียน", D3 = 1 x 0.301 = 0.301
tf-idf "ไม่", D3 = 1 x 0.301 = 0.301
tf-idf "วิชา", D4 = 1 x 0.602 = 0.602
tf-idf "ภาษาไทย", D4 = 1 x 0.602 = 0.602
tf-idf "น่าเบื่อ", D4 = 1 x 0.602 = 0.602

ดังนั้นผลการคำนวณค่า $tf - idf$ ของคำในแต่ละเอกสาร จึงสามารถแสดงได้ดังตารางที่ 2.5

ตารางที่ 2.5 แสดงเอกสารที่ผ่านการให้น้ำหนักด้วย $tf - idf$

	คอมพิวเตอร์	วิทยาศาสตร์	คณิตศาสตร์	เรียน	การ	ทดลอง	ไม่	สนุก	วิชา	ภาษาไทย	น่าเบื่อ	Class
D1	0.602	0	0	0.301	0	0	0	0	0	0	0	POS
D2	0	0.602	0	0	0.602	0.602	0	0.602	0	0	0	POS
D3	0	0	0.602	0.301	0	0	0.602	0	0	0	0	NEG
D4	0	0	0	0	0	0	0	0	0.602	0.602	0.602	NEG

2.2.3.2.3 ให้น้ำหนักแบบ $tf - igm$

การให้น้ำหนักแบบ $tf - igm$ คือวิธีการสร้างตัวแทนเอกสารในรูปแบบของความถี่ของคำ-ส่วนกลับของช่วงเวลาแรงโน้มถ่วง [15] เพื่อเป็นการวัดค่าน้ำหนักของคำ คือคำต่างๆ ที่มีการกระจายระหว่างคลาสไม่สม่ำเสมอ เป็นแนวคิดที่คล้ายกับแนวคิดเรื่อง "โมเมนต์แรงโน้มถ่วง (Gravity Moment: GM)" ในทางฟิสิกส์ ซึ่งสมการของ igm สามารถแสดงได้ดังนี้

$$\boldsymbol{igm(t_i) = \frac{f_{i1}}{\sum_{r=f_{ir}}^{m} \times r}} \quad (2.4)$$

เมื่อ f_{ir} คือจำนวนของเอกสารที่มีคำ t_i ในคลาสที่ $r - th$ โดยเรียงลำดับจากมากไปหาน้อย และ f_{i_1} คือจำนวนความถี่ของคำ t_i ที่ปรากฏในคลาส

การคำนวณค่าน้ำหนักแบบ $tf - igm$ จะมีการใช้ตัวแปรแลมด้า (λ) เป็นค่าสัมประสิทธิ์ที่ปรับได้ (Adjustable Coefficient) โดยทั่วไปค่า λ จะถูกกำหนดไว้ที่ 7.0 เพราะในงานวิจัยหลายๆ งานให้ความคิดเห็นว่าเป็นค่าที่เหมาะสมที่สุด (Chen et al., 2016) นอกจากนี้ λ ยังสามารถกำหนดให้อยู่ระหว่างค่า 5.0 ถึง 9.0 สมการ $tf - igm$ สามารถแสดงได้ดังนี้

$$\boldsymbol{w_{tf-igm(t_k)} = tf(t_i, d_j) \times (1 + \lambda \times igm(t_i))} \quad (2.5)$$

เมื่อ $tf(t_i, d_j)$ คือความถี่ของคำ t_i ที่พบในเอกสาร d_j

$$\boldsymbol{w_{sqrt\_tf-igm(t_k)} = sqrt\_tf(t_i, d_j) \times (1 + \lambda \times igm(t_i))} \quad (2.6)$$

$$igm_{imp}(t_i) = \left| \frac{f_{i_1}}{\sum_{r=f_{ir}}^{m} \times r + long_{10}\left(\frac{D_{total}(t_{i\_max})}{f_{i_1}}\right)} \right| \quad (2.7)$$

จากตารางที่ 2.5 และสมการที่ 5 เมื่อกำหนดค่า λ =7.0 จะสามารถคำนวณค่าน้ำหนักของคำว่า “คณิตศาสตร์” ด้วย $tf - igm$ ได้ตัวอย่างต่อไปนี้

$$w_{tf-igm(t_{=\text{“คณิตศาสตร์”}})} = tf(t_i, d_j) \times (1 + \lambda \times igm(t_i))$$

$$w_{tf-igm(t_{=\text{“คณิตศาสตร์”}})} = tf(t_i, d_j) \times \left(1 + \lambda \times \frac{f_{i_1}}{\sum_{r=f_{ir}}^{m} \times r}\right)$$

$$w_{tf-igm(t_{=\text{“คณิตศาสตร์”}})} = 1 \times \left(1 + 7.0 \times \frac{1}{(1 \times 1) \times (0 \times 2)}\right)$$

$$w_{tf-igm(t_{=\text{“คณิตศาสตร์”}})} = 8$$

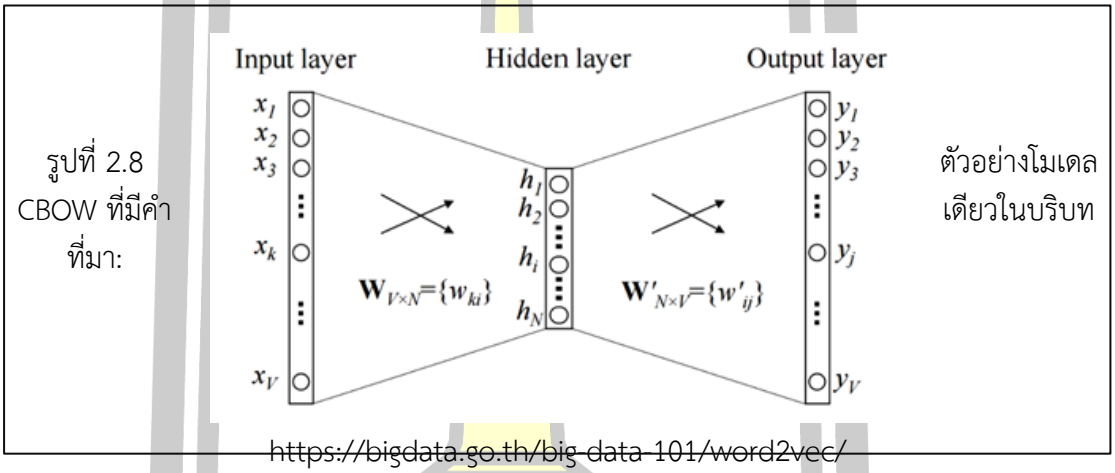
2.2.4 Word2Vec

Word2vec [16] คือ โมเดล Word Embedding ที่มีการแปลง “คำ” เป็น “ตัวเลข” ซึ่งเป็นหนึ่งในวิธีในการสร้างคุณลักษณะจาก “คำ” วิธีหนึ่ง โดยจะทำการลดขนาด vector space ลง โดย Word2Vec พัฒนาโดยทีมนักวิจัยของ Google นำโดย Tomas Mikolov แนวคิดของ Word2Vec ก็คือการแสดง “คำ” ให้อยู่ในรูปของ “vector” ที่ใช้วิธีการคำนวณตัวเลขของคำนั้นๆ จาก context รอบๆ คำนั่นคือ แทนที่จะนับว่าแต่ละคำเกิดขึ้นพร้อม target word กี่ครั้ง แต่เราจะทำการสร้างตัวจำแนก (Classifier) เพื่อทำนายว่า “คำ” ที่กำลังพิจารณาใน context word นี้ มีโอกาสเกิดใกล้ๆ กับ target word หรือไม่ แทน ในความเป็นจริงแล้ว เราจะไม่ได้สนใจงานทำนายนี้หรอก เราเพียงต้องการจะเอาค่าน้ำหนัก หรือ Weight ที่ได้จากการสร้างตัวจำแนกมาเป็น Word Embedding

Word2Vec ประกอบด้วยแนวคิดหลัก 2 แนวคิด คือ (1) ความหมายของคำคำหนึ่งนั้นอาจสามารถถูกทำนายได้จากบริบทของคำที่อยู่รอบข้าง และ (2) คำที่อยู่รอบข้างของคำคำหนึ่งก็อาจสามารถถูกทำนายได้จากคำคำนั้นเช่นกัน และแนวคิดทั้งสองนี้ถูกนำไปสร้างเป็นเครื่องมือหลักใน Word2Vec คือ Continuous Bag of Words (CBOW) และ Skip-gram ตามลำดับ

2.2.4.1 โมเดล Continuous Bag of Words (CBOW)

ในการใช้งานโมเดล CBOW [17] นั้น ผู้ใช้จะต้องทำการสร้างเครือข่ายสมองแบบตื้น (Shallow Neural Network) โดยจะใช้เวกเตอร์ของคำ (Word Vector) เป็นชั้นอินพุต (Input Layer) ซึ่งจะเชื่อมต่อเข้ากับชั้นซ่อน (Hidden Layer) จำนวน 1 ชั้น โดยที่เราสามารถกำหนดจำนวนโหนดในชั้นนี้ได้ตามต้องการ แต่โดยทั่วไปแล้ว โหนดในชั้นซ่อนควรจะมีจำนวนน้อยกว่าจำนวนมิติของเวกเตอร์ที่เป็นชั้นอินพุต และทำการต่อเข้าสู่ชั้นเอาต์พุต (Output Layer) ที่มีจำนวนมิติเท่ากับชั้นอินพุต จากนั้นจึงทำการเรียนรู้โมเดล (Training) ในรูปแบบของการจำแนกข้อมูล (Classification)



รูปที่ 2.8 ตัวอย่างโมเดล CBOW ที่มีคำเดียวในบริบท

ที่มา: https://bigdata.go.th/big-data-101/word2vec/

สำหรับแต่ละคำในประโยคหรือข้อความที่จะวิเคราะห์นั้น ในขั้นตอนของการเรียนรู้โมเดลจะนำเอาเวกเตอร์ของคำที่อยู่รอบๆ “คำที่กำลังพิจารณา” ในระยะหรือขนาดของบริบท (Context Size) ที่กำหนดมาใช้เป็นอินพุตสำหรับการทำการจำแนกข้อมูล จากนั้นจะใช้เวกเตอร์ของ “คำที่กำลังพิจารณา” ซึ่งมีตำแหน่งอยู่ที่ตรงกลาง (Center Word) ของบริบท เป็นเป้าหมายในการทำนายผลลัพธ์ในชั้นเอาต์พุต ซึ่งรูปที่ 2.8 แสดงรูปแบบของการทำนายคำที่อยู่รอบๆ “คำที่กำลังพิจารณา” ที่ทำหน้าที่เป็นคำตรงกลาง

รูปที่ 2.9 ตัวอย่างที่แสดงรูปแบบของการทำนายคำที่อยู่รอบๆ “คำที่กำลังพิจารณา” ที่ทำหน้าที่เป็นคำตรงกลาง

ที่มา: https://bigdata.go.th/big-data-101/word2vec/

2.2.4.2 โมเดล Skip-gram

โมเดล Skip-gram [18] จะมีการสร้างเครือข่ายสมองแบบตื้นเหมือนกับที่ใช้งานในโมเดล CBOW โดยชั้นอินพุตและชั้นเอาต์พุตจะมีขนาดเท่ากัน และมีชั้นซ่อนจำนวน 1 ชั้น เช่นเดียวกับที่แสดงในรูปที่ 2.8

แต่การเรียนรู้ของโมเดล Skip-gram นั้นจะต่างจากการโมเดล CBOW เพราะจะไม่ใช้ เวกเตอร์ของคำต่างๆ ในบริบทของคำแต่ละคำเพื่อการทำนายคำดังกล่าว แต่โมเดล Skip-gram เลือกใช้คำหนึ่งๆ ในการทำนายคำทุกคำที่อยู่ในบริบทของคำนั้นแทน (รูปแบบการทำงานของ Skip-gram ดูได้ในรูปที่ 2.9) จึงทำให้การพิจารณาคำจะแตกต่างจาก CBOW นั่นคือ “คำ” ในประโยคหรือข้อความที่นำมาพิจารณานั้น การเรียนรู้โมเดลของ Skip-gram โดยการนำเอาเวกเตอร์ของคำที่จะพิจารณามาใช้เป็นคำตรงกลาง หรือ Center Word จากนั้นจะทำนายการกระจายตัวเชิงความน่าจะเป็น (Probability Distribution) ของคำที่เป็นไปได้ที่น่าจะเป็นบริบทของคำที่กำลังพิจารณา

รูปที่ 2.10 ตัวอย่างที่แสดงรูปแบบรูปแบบการทำงานของ Skip-gram
ที่มา: https://bigdata.go.th/big-data-101/word2vec/

2.2.5 โมเดลการจำแนกความรู้สึก (Sentiment Classifier Model)

หลังจากที่ได้เตรียมเอกสารข้อความ การทำความสะอาดเอกสารข้อความ การทำโทเค็นไนเซชั่น การตัดคำหยุด และได้ถุงคำ ที่บรรจุ “คำ” มีการให้น้ำหนักและทำการคัดเลือกคำที่จะเป็นคุณลักษณะที่เหมาะสมเรียบร้อยแล้ว ก็จะนำถุงคำนั้นเข้าสู่ขั้นตอนของการประมวลผลเพื่อสร้างตัวแบบสำหรับใช้ในการการจำแนกเอกสารข้อความแบบอัตโนมัติ หรือเรียกสั้นๆ ว่า “ตัวจำแนกเอกสารข้อความ (Text Classifier)” อัลกอริทึมที่ใช้สร้างตัวจำแนกเอกสารนั้น จะเป็นอัลกอริทึมการเรียนรู้ของเครื่องแบบมีผู้สอน (Supervised Machine Learning Algorithm) เนื่องจากได้รับความนิยม นำมาประยุกต์ใช้เพื่อการสร้างตัวจำแนกเอกสารข้อความนั้นมีหลายตัว เช่น พหุนามอีฟเบย์ (Multinomial Naïve Bayes: MNB) ขั้นตอนวิธีเพื่อนบ้านใกล้ที่สุด (K-Nearest Neighbor: K-NN) ซัพพอร์ตเวกเตอร์แมชชีน (Support Vector Machines: SVM) ป่าสุ่ม (Random Forest: RF) และ K Fold เป็นต้น

2.2.5.1 โมเดลการจำแนกความรู้สึก (Sentiment Classifier Model)

ในการจำแนกเอกสารมักจะใช้อัลกอริทึมพหุนาอีฟเบย์ [19] โดยมีพื้นฐานแนวคิดมาจากทฤษฎีของเบย์ (Bayes' Therem) ที่จะจัดข้อมูลในรูปแบบเวกเตอร์ สมมติกำหนดเวกเตอร์ของเอกสารอยู่ในรูปแบบ dy = {wy1, wy2, ..., wyn} นั่นคือ เวกเตอร์ของเอกสาร d ในคลาส y โดยมีคุณลักษณะ wจำนวน n ตัว ดังนั้นจะเขียนสมการของความน่าจะเป็นของ w ในเอกสาร d ที่ทำให้เกิดคลาส y

$$d_{yi} = P(x_i|y) \tag{2.8}$$

d_y เป็นการประมาณค่าสูงสุดหรือ Likeliihood ซึ่งเขียนสมการได้คือ

$$\hat{d}_y = \frac{N_{yi} + \alpha}{N_y + \alpha_n} \tag{2.9}$$

$$N_{yi} = \sum_{x \in T} w_i \tag{2.10}$$

จากสมการที่ (2.10) เป็นจำนวนความถี่ของคำ *w* ที่ *i* ที่อยู่ในคลาส *y*

$$N_y = \sum_{i=1}^{|T|} N_{yi} \tag{2.11}$$

จากสมการที่ (2.11) เป็นจำนวนคำ *w* ทั้งหมดในคลาส *y*

เพื่อป้องกันค่าความน่าจะเป็นมีค่าเป็น 0 จึงมีการบวกค่า α เข้ามาในสมการ โดยค่า α ที่บวกเข้ามานั้นจะมีค่ามากกว่า 0 เรียกวิธีการนี้ว่า Laplace Smooting ถ้าหากค่า α มีค่าน้อยกว่า 0 จะเรียกว่า Lidstone Smoothing โดยทั่วไปมักจะใช้วิธีการแบบ Laplace Smooting

ในการสร้างตัวจำแนกเอกสาร สามารถแสดงขั้นตอนในการสร้างตัวจำแนกเอกสารได้ดังนี้

ขั้นตอนที่ 1: หาความน่าจะเป็นของคลาส

$$\hat{P}(c) = \frac{N_c}{N_{doc}} \tag{2.12}$$

จากสมการที่ (2.12) N_c คือ จำนวนเอกสารในคลาส *c* ที่มีในคลังเอกสาร และ N_{doc} คือ จำนวนเอกสารทั้งหมดในคลังเอกสาร

ขั้นตอนที่ 2: หาความน่าจะเป็นของ "คำ" แต่ละคำ ที่ปรากฏในคลาสนั้นๆ

$$\hat{P}(w_i|c) = \frac{count(w_i, c) + 1}{\sum_{w \in V}(count(w_i, c) + 1)} \tag{2.13}$$

หรืออาจจะใช้สมการ (2.14) ซึ่ง คือขนาดหรือจำนวนคำทั้งหมดที่ใช้เป็นคุณลักษณะ

$$\hat{P}(w_i|c) = \frac{count(w_i, c) + 1}{\sum_{w \in V}(count(w_i, c) + V)} \quad (2.14)$$

2.2.5.2 ขั้นตอนวิธีเพื่อนบ้านใกล้เคียงที่สุด (K-nearest Neighbors: KNN)

KNN [20] นับเป็นเทคนิคที่มีวิธีการไม่ซับซ้อนและเข้าใจได้ง่ายที่สุดที่ใช้ ในการจำแนกประเภทข้อมูล โดยหลักการทำงาน คือ จะใช้หลักการเปรียบเทียบความคล้ายคลึงกันของข้อมูลที่สนใจกับข้อมูลอื่นว่ามีความคล้ายคลึงหรืออยู่ใกล้กับข้อมูลใดมากที่สุด *k* ตัว จากนั้นจะทำการตัดสินใจว่า คำตอบของข้อมูลที่สนใจนั้นควรเป็นคำตอบเดียวกับข้อมูลที่อยู่ใกล้ที่สุด *k* ตัวนั้น ทั้งนี้ *k* คือความถี่ของข้อมูลที่อยู่ใกล้กับข้อมูลที่สนใจ KNN มีขั้นตอนดังนี้

ขั้นตอนที่ 1: กำหนดค่า *k* ซึ่งโดยทั่วไปจะกำหนดให้เป็นเลขคี่ เช่น 3, 5, 7 และ 9 เป็นต้น

ขั้นตอนที่ 2: นำวัตถุที่ต้องการจำแนกมาวัดหาความคล้ายคลึงหรือความต่างกับข้อมูลทั้งหมดในชุดข้อมูล โดยทั่วไปจะใช้มาตรวัดระยะห่างที่นิยม ได้แก่ ระยะยูคลิด (Euclidean Distance)

$$distance(d_x, d_y) = \sqrt{\sum_{i=1}^{n}(d_x - d_y)^2} \quad (2.15)$$

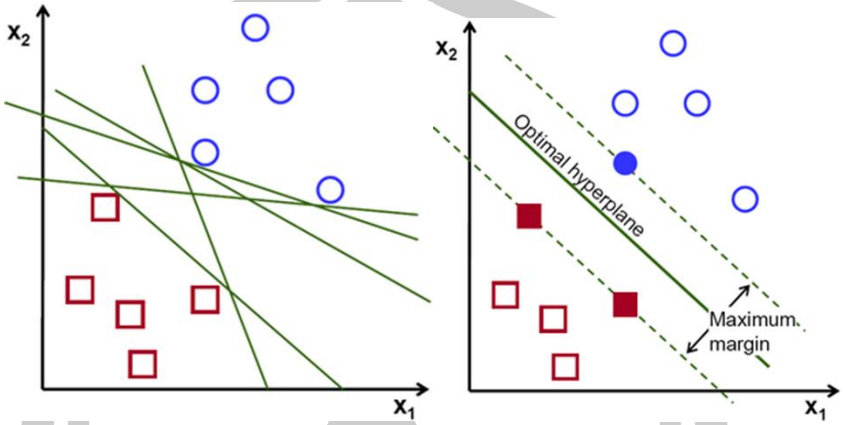
ขั้นตอนที่ 3: เรียงลำดับวัตถุตามความคล้ายหรือความแตกต่าง โดยทั่วไปจะเรียงลำดับจากน้อยไปมาก เพราะข้อมูลที่มีระยะห่างกันยิ่งน้อย แสดงว่ามีความคล้ายคลึงกันมาก

ขั้นตอนที่ 4: พิจารณาคำตอบจากจำนวนคลาสของคำตอบที่มีมากที่สุดใน *k* ตัว ยกตัวอย่างเช่น หากพิจารณาข้อมูลแบบ 2 คลาสคือ คลาส a และคลาส b โดยพิจารณาจาก *k = 5* ถ้าหากในจำนวนคำตอบของ unknown data จาก 5 คำตอบ พบว่าเป็นคำตอบของคลาส a จำนวน 3 ตัว และเป็นคำตอบของคลาส b จำนวน 2 ตัว ก็หมายความว่า unknown data ตัวนั้นจะถูกพิจารณาให้อยู่ในคลาส b

2.2.5.3 ซัพพอร์ตเวคเตอร์แมชชีน (Support Vector Machine : SVM)

SVM [21] เป็นหนึ่งในอัลกอริธึมการเรียนรู้ภายใต้การดูแลที่ได้รับความนิยมมากที่สุด สำหรับการจำแนกประเภทและปัญหาการถดถอย มีการประยุกต์ใช้ 2 รูปแบบด้วยกัน คือ การจำแนกข้อมูล (Classification) และการวิเคราะห์การถดถอย (Regression) และถ้ามีการนำ SVM มาใช้ในการจำแนกข้อมูลจะเรียกว่า “การจำแนกข้อมูลด้วยซัพพอร์ตเวกเตอร์ (Support Vector Classification: SCV)”และหากใช้ในการวิเคราะห์การถดถอยจะเรียกว่า “การถดถอยด้วยซัพพอร์ตเวกเตอร์ (Support Vector Regression: SVR)” โดยการทำงานของ SVM จะทำการ

วิเคราะห์ข้อมูลและจำแนกข้อมูล โดยการแมปข้อมูลกับสเปซคุณลักษณะมิติสูง ซึ่งจะอาศัยหลักการของการหาสัมประสิทธิ์ของสมการเพื่อสร้างเส้นแบ่งแยกกลุ่มข้อมูลที่ถูกป้อนเข้าสู่กระบวนการสอนให้ระบบเรียนรู้ โดยเน้นไปยังเส้นแบ่งแยกแยะกลุ่มข้อมูลได้ดีที่สุด ดังรูปที่ 2.11



รูปที่ 2.11 ตัวอย่างการแยกกลุ่มของข้อมูลของ SVM
ที่มา: https://medium.com/@pradyasin/support-vector-machines-svm-943f9a732a69

SVM จะทำการแบ่งชุดข้อมูลออกเป็นคลาสเพื่อค้นหาไฮเปอร์เพลนระยะขอบสูงสุด (Maximum Margin Hyperplane: MMH) ซึ่งจุดข้อมูลที่ใกล้กับไฮเปอร์เพลนมากที่สุดเรียกว่าเวกเตอร์สนับสนุน เส้นแบ่งจะถูกกำหนดด้วยความช่วยเหลือของจุดข้อมูล โดยมีสมการดังนี้

SVM ใช้ Hypothesis function แบบเส้นตรง โดยมีสมการดังนี้:

$$\begin{aligned} h_\theta(x) &= w_1x_1 + w_2x_2 + \cdots + w_nx_n + b \\ &= w^Tx + b \end{aligned} \tag{2.16}$$

ถ้าในกรณีผลลัพธ์เป็นบวก จะทำนาย Class $\hat{y}$ ว่าเป็น 1 ส่วนถ้าเป็นลบ ทำนายว่าเป็น 0 เราสามารถเขียนวิธีการตัดสินใจตามเงื่อนไขดังกล่าวได้ดังนี้:

$$\hat{y} = \begin{cases} 0 \text{ if } w^Tx + b < 0, \\ 1 \text{ if } w^Tx + b \geq 0 \end{cases} \tag{2.17}$$

เมื่อได้ค่าของเส้นแบ่งการตัดสินใจ (เส้น Maximum Margin Hyperplane) ในขั้นตอนต่อไปจะเป็นการกำหนดเส้น Positive Hyperplane และ Negative Hyperplane โดยเส้นประแต่ละด้านคือตำแหน่งที่ $h\theta(x)$ เท่ากับ -1 และ 1

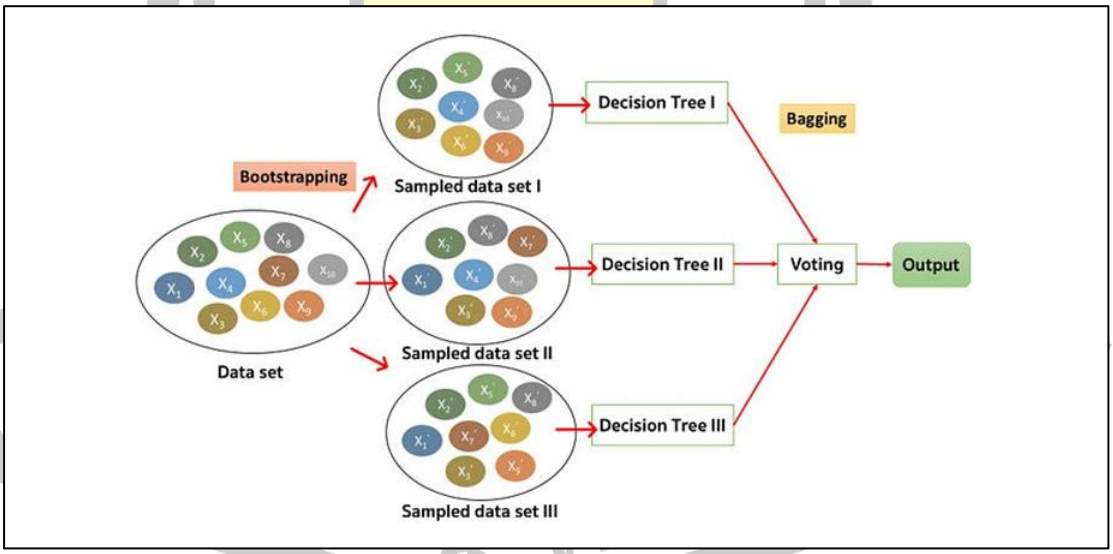
ความชันของฟังก์ชันการตัดสินใจ เท่ากับ Norm ของ Vector ค่าน้ำหนัก

$$\begin{aligned}\frac{\partial}{\partial x} h_\theta(x) &= \sum_{i=1}^{m} |w_i| \\ &= |w_1| + |w_2| + \ldots + |w_i| \\ &= ||w||_1\end{aligned} \tag{2.18}$$

2.2.5.4 ป่าสุ่ม (Random Forest: RF)

Rabdon Forest (RF) [22, 23] เป็นอัลกอริธึมการเรียนรู้ของเครื่องที่ใช้กันอย่างแพร่หลาย พัฒนาโดย Leo Breiman และ Adele Cutle ซึ่ง RF เป็นการรวมเอาต์พุตของแผนผังการตัดสินใจหลายรายการเพื่อให้ได้ผลลัพธ์เดียว และ RF ยังมีความสามารถในการจัดการกับปัญหาการจำแนกประเภทและการถดถอยได้เนื่องจากแบบจำลองของ RF ประกอบด้วยแผนผังการตัดสินใจหลายแผนผัง จึงควรเริ่มต้นด้วยการอธิบายอัลกอริทึมแผนผังการตัดสินใจโดยย่อ แผนผังการตัดสินใจจึงพยายามหาการแยกที่ดีที่สุดเพื่อย่อยข้อมูลและ โดยทั่วไปแล้วแผนผังการตัดสินใจจะได้รับการฝึกผ่านอัลกอริทึม Classification and Regression Tree (CART) เมตริก โดยใช้ การให้น้ำหนักคำ (Term Weighting) เพื่อใช้ในการประเมินคุณภาพของการแยกได้ จุดแข็งของ RF คือสามารถจัดการกับชุดข้อมูลที่ซับซ้อนและลดการโอเวอร์ฟิต

แม้ว่าแผนผังการตัดสินใจจะเป็นอัลกอริธึมการเรียนรู้แบบมีผู้สอนทั่วไป แต่ก็อาจมีแนวโน้มที่จะเกิดปัญหา เช่น อคติและการติดตั้งมากเกินไป อย่างไรก็ตาม เมื่อแผนผังการตัดสินใจหลายแผนผังรวมกันในอัลกอริธึมฟอเรสต์แบบสุ่ม แผนผังเหล่านี้จะทำนายผลลัพธ์ที่แม่นยำยิ่งขึ้น โดยเฉพาะอย่างยิ่งเมื่อแผนผังแต่ละแผนผังไม่มีความสัมพันธ์กัน ดังรูปที่ 2.12



รูปที่ 2.12 หลักการทำงานของ RF

ที่มา: https://www.ibm.com/topics/random-forest

จากรูปที่ 2.12 RF มีไฮเปอร์พารามิเตอร์หลักสามตัว ขนาดโหนด จำนวนแผนผัง และจำนวนคุณลักษณะที่สุ่มตัวอย่าง ซึ่ง RF ประกอบด้วยคอลเลกชันของแผนผังการตัดสินใจ และแต่ละแผนผังในชุดประกอบด้วยตัวอย่างข้อมูลที่ดึงมาจากชุดการฝึกที่มีการทดแทนเรียกว่าตัวอย่างบูตสแตรป (Bootstrapping)คือเทคนิคการสุ่มตัวอย่างทางสถิติของการสุ่มตัวอย่างของชุดข้อมูลด้วยการแทนที่ สำหรับการฝึก หนึ่งในสามถูกกันไว้เป็นข้อมูลทดสอบ หรือที่เรียกว่าตัวอย่างที่ไม่อยู่ในถุง (oob) จากนั้นการสุ่มอีกตัวอย่างหนึ่งจะถูกแทรกเข้าไปในฟีเจอร์การบรรจุถุง เพื่อเพิ่มความหลากหลายให้กับชุดข้อมูล และลดความสัมพันธ์ระหว่างแผนผังการตัดสินใจ การทำนายแต่ละครั้งจะแตกต่างกันขึ้นอยู่กับประเภทของปัญหา โดย RF มีสมการดังนี้

$$\begin{aligned} Gini\ Index &= {}_1 - \sum_{i=_1}^{n} (P_i)^2 \\ &= [(P_+)^2 + (P_-)^2] \end{aligned} \tag{2.19}$$

โดยค่า P_+ คือความน่าจะเป็นของคลาสที่เป็นบวก และP_- คือความน่าจะเป็นของคลาสที่เป็นลบ

2.2.5.5 K-Fold Cross Validation

K-Fold Cross Validation [24] คือ วิธีการที่ใช้ในการคาดเดาค่าความผิดพลาดของแบบจำลอง (model) คิดค้นโดย Dunlap K. และPopper K. R โดยจะเริ่มจากการแบ่งชุดข้อมูล training set ออกเป็นส่วนๆ ทั้งหมด K ส่วนเท่าๆ กัน เช่น 5 ส่วน 10 ส่วน ถ้าเราเลือกแบ่งข้อมูล 5 ส่วน แปลว่า k=5 เพื่อสร้างและทดสอบโมเดล ในการแบ่งข้อมูลแต่ละส่วนจะต้องมาจากการสุ่ม (random) เพื่อที่จะให้ข้อมูลมีการกระจายเท่าๆ กัน นำไปสร้างและทดสอบโมเดล (train + validate) ทำการคำนวณค่าความผิดพลาดทั้งหมด K รอบ โดยแต่ละรอบคำนวณข้อมูล ชุดหนึ่งจากข้อมูล K ชุด เลือกออกมาเพื่อเป็นข้อมูลทดสอบและเหลือข้อมูลอีก K-1 ชุด จะถูกใช้เป็นข้อมูลสำหรับการเรียนรู้และการศึกษางานวิจัยในครั้งนี้ใช้ K-fold cross validation โดยใช้ K=10

ตัวอย่าง ถ้าเราแบ่งข้อมูล k=5 เราจะสร้างและทดสอบโมเดลทั้งหมด 6 รอบ เพราะการ train รอบสุดท้าย คือรอบที่ 6 เราจะใช้ all data {1, 2, 3, 4, 5} เพื่อสร้าง final model กับค่า hyperparameters ที่ดีที่สุดจากการทดสอบ 5 รอบ ดังรูปที่ 2.13

รอบที่	train				test
1	1	2	3	4	5
2	1	2	3	4	5
3	1	2	3	4	5
4	1	2	3	4	5
5	1	2	3	4	5

รูปที่ 2.13 5-fold cross-validation

สรุป K-Fold Cross Validation

- ใช้เปรียบเทียบว่าข้อมูลชุดไหนดีที่สุดในโมเดล
- ใช้เปรียบเทียบระหว่างโมเดลได้ว่าโมเดลไหนดีกว่ากัน

หลักการทำงานการวัดประสิทธิภาพของโมเดลแบบ K-Fold Cross Validation

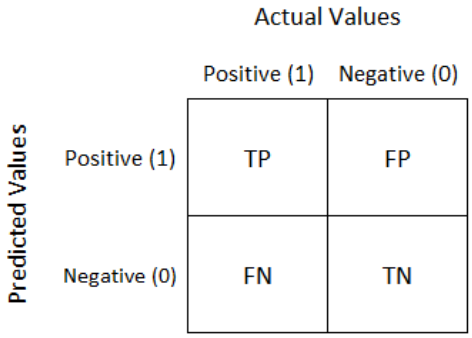
- แบ่งข้อมูลเรียนรู้ออกเป็น k ชุดเท่าๆ กัน
- ใช้ข้อมูลส่วนที่เหลือ (k -1 ชุด) เพื่อทำการสร้างโมเดล
- เก็บข้อมูลที่แบ่งไว้ 1 ชุด เพื่อทำการ Evaluate
- วนทำซ้ำจนข้อมูลทุกส่วนถูกนำมาทดสอบ

ค่าความถูกต้องของ Model

$$\text{overall accuracy} = \text{Average}(\sum_{i=1}^{k} \text{Accuracy}_i) \tag{2.20}$$

2.2.6 การวัดประสิทธิภาพโมเดล (Evaluation)

เป็นขั้นตอนการประเมินโมเดลเพื่อใช้ในการจัดกลุ่มเอกสารก่อนการนำไปใช้งานจริงที่โดยทั่วไปจะใช้เทคนิคมาตรฐาน [22] ที่เรียกว่า ค่าความถูกต้อง (Accuracy) การวัดค่าความระลึก $(Recall)$ การวัดค่าความแม่นยำ $(Precision)$ และการวัดค่า $F\text{-}measure$



รูปที่ 2.14 ตาราง Confusion Matrix
ที่มา: https://towardsdatascience.com/understanding-confusion-matrix-a9ad42dcfd62.

- True Positive (TP) คือ สิ่งที่โปรแกรมทำนายว่าจริง และคนบอกว่ามันจริง
- True Negative (TN) คือ สิ่งที่โปรแกรมทำนายว่าไม่จริง และคนบอกว่ามันไม่จริง
- False Positive (FP) คือ สิ่งที่โปรแกรมทำนายว่าจริง แต่คนบอกว่าไม่จริง
- False Negative (FN) คือ สิ่งที่โปรแกรมทำนายว่าไม่จริง แต่คนบอกว่าจริง

การวัดค่าความถูกต้อง $(Accuracy)$ [25] คือ เป็นอัตราส่วนของเอกสารที่จัดกลุ่มได้จากเอกสารทั้งหมดที่มีอยู่ โดยจะนำค่าจากตาราง Confusion matrix มาใช้ในการคำนวณหาค่าความระลึกได้ดังนี้

$$Accuracy = \frac{TP + TN}{TP + TN + FN + FP} \tag{2.21}$$

การวัดค่าความระลึก $(Recall)$ [25] คือ เป็นอัตราส่วนของเอกสารที่จัดกลุ่มได้จากเอกสารทั้งหมดที่มีอยู่ โดยจะนำค่าจากตาราง Confusion matrix มาใช้ในการคำนวณหาค่าความระลึกได้ดังนี้

$$Recall = \frac{TP}{TP + FN} \tag{2.22}$$

การวัดค่าความแม่นยำ $(Precision)$ [25] คือ เป็นอัตราส่วนของเอกสารที่จัดกลุ่มได้และถูกต้อง ส่วนด้วยจำนวนของเอกสารที่จัดกลุ่มได้

$$Precision = \frac{TP}{TP + FP} \tag{2.23}$$

การวัดค่า $F-measure$ เป็นการพิจารณาค่าความสัมพันธ์ระหว่างค่าความระลึกและค่าความแม่นยำ

$$F-measure = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (2.24)$$

โดยที่ค่า $F-measure$ จะมีค่าระหว่าง 0 ถึง 1 ซึ่งถ้าหากค่า F มีค่าเข้าใกล้ 1 มากเท่าไหร่ก็จะหมายถึง การจัดกลุ่มเอกสารนั้นมีประสิทธิภาพและมีความถูกต้องมากขึ้นเท่านั้น

2.3 งานวิจัยที่เกี่ยวข้อง

ในส่วนนี้จะเป็นการทบทวนงานวิจัยด้านการวิเคราะห์ความรู้สึกในโดเมนของการศึกษา ซึ่งมีดังต่อไปนี้

ในงานวิจัยของ Ulfa และคณะ [26] ผู้วิจัยได้อ้างอิงถึงงานวิจัยของ Brenan และ Williams [27] และงานวิจัยของ Shankararaman และคณะ [28] ที่กล่าวไว้ว่าคำติชม (Feedback) จากผู้เรียนจะสามารถใช้ในการปรับปรุงคุณภาพของการเรียนการสอนและหลักสูตรได้ ดังนั้นนักวิจัยจึงสนใจการใช้ประโยชน์จากคำติชมและกระบวนการในการวิเคราะห์ คำติชมที่ได้มา โดยเฉพาะคำติชมที่อยู่ในรูปแบบของ free text ภายใต้คำถามปลายเปิด (Open Question) ซึ่งนักวิจัยจึงเริ่มศึกษาว่าในงานวิจัยที่เกี่ยวข้องกับ “*analysis of student feedback on online learning using sentiment analysis*” มีมากน้อยเพียงใดและมีกระบวนการในการวิเคราะห์ความรู้สึกอย่างไร โดยนักวิจัยพบว่าจากบทความในช่วง 6 ปี ระหว่าง พ.ศ. 2557-2562 มีงานวิจัยที่เกี่ยวข้องกับเรื่องนี้เพียง 12 เรื่อง และนอกนั้นเขายังพบว่าเทคนิคในการวิเคราะห์ความรู้สึกมี 2 รูปแบบคือ วิธีการเชิงสถิติ (Statistical Technique) และวิธีการเชิงความหมาย (Semantic Technique) ในขณะที่อัลกอริทึมการเรียนรู้ของเครื่อง (Machine Learning) ที่นิยมใช้ ได้แก่ Naive Bayes, Support Vector Machine, (SVM), Random Forest, และ Neural Networks ส่วนวิธีการที่นิยมใช้ในการวิเคราะห์ความรู้สึกก็ได้แก่ การใช้การวิเคราะห์ความรู้สึกด้วยคำ (The Lexicon Based Approach) การใช้การวิเคราะห์ความรู้สึกด้วยกฎ (The Rule-based Approach) และการใช้การวิเคราะห์ความรู้สึกเชิงสถิติ (Statistical Methods)

พิชญะ พรมลา และจรัญ แสนราช [29] ได้พบว่าการติชมเรื่องการเรียนการสอนภายใต้คำถามปลายเปิดทำได้ยากและมีความซับซ้อน รวมทั้งอาจเกิดความไม่แม่นยำเนื่องจากอคติจากผู้วิเคราะห์ข้อมูล ดังนั้นนักวิจัยจึงประยุกต์กระบวนการของการวิเคราะห์ความคิดเห็นเข้ามาใช้ในการวิเคราะห์คำติชมดังกล่าว ซึ่งในการศึกษานี้เป็นการวิเคราะห์คำติชมของผู้เรียนในวิทยาลัยเทคนิคแห่งหนึ่งที่มีต่อการเรียนการสอนในรายวิชาต่างๆ ระหว่างปีการศึกษา 2560 จนถึงปีการศึกษา 2561 โดยเป็นการรวบรวมคำติชมจากแบบสอบถามปลายเปิด และรวบรวมข้อมูลคำติชมได้ทั้งสิ้น 1,577 ข้อความ ซึ่งสามารถแบ่งเป็นข้อความในขั้วบวกจำนวน 1,037 ข้อความ และข้อความในขั้วลบจำนวน 540 ข้อความ ก่อนขั้นตอนของการเตรียมข้อมูลนั้น ข้อความทั้งหมดได้ผ่านการทำ Text Cleaning

เพื่อแก้ไขคำผิด กำจัดอักขระพิเศษ และข้อมูลซ้ำ ข้อความไม่สมบูรณ์ และชื่อเฉพาะต่างๆ จากนั้นนำข้อความที่ได้เข้าสู่กระบวนการเตรียมข้อมูล ที่เริ่มด้วยการตัดคำ (Word Segmentation) ซึ่งในงานวิจัยนี้เลือกใช้วิธีการตัดคำแบบเหมือนมากที่สุด (Maximum Matching) ต่อมาทำการคัดเลือกคำที่จะใช้เป็นคุณลักษณะ (Feature Selection) ในที่นี้สามารถเลือกคำที่จะใช้เป็นคุณลักษณะได้ทั้งสิ้น 1,542 คำ อย่างไรก็ตามเนื่องจากชุดข้อมูลที่ใช้มีความไม่สมดุล (Imbalanced Data) ดังนั้นผู้วิจัยจึงแก้ปัญหาด้วยวิธีการแบบการสุ่มเพิ่มตัวอย่างกลุ่มน้อย (SMOTE) ในอัตราส่วน 190% จนได้ข้อมูลในแต่ละขั้วเท่ากันคือขั้วละ 1,037 ข้อความ รวมเป็น 2,074 ข้อความ สำหรับใช้สร้างตัวแบบการจำแนกข้อมูล เมื่อเตรียมข้อมูลเรียบร้อยก็นำข้อมูลที่ผ่านการเตรียมข้อมูลนั้นเข้าสู่ขั้นตอนของการสร้างโมเดลเพื่อการวิเคราะห์ความรู้สึกของผู้เรียนจากคำติชมแบบ 2 ขั้ว นั่นคือ พอใจและไม่พอใจในการเรียนการสอน อย่างไรก็ตามในการสร้างโมเดลเพื่อวิเคราะห์ความรู้สึกของผู้เรียนนั้น ผู้วิจัยแยกการศึกษาออกเป็น 2 รอบ โดยในรอบแรกเป็นการศึกษาเชิงเปรียบเทียบการสร้างโมเดลเพื่อวิเคราะห์ความรู้สึกด้วยอัลกอริทึม 3 ตัว ได้แก่ Decision Tree (DT), Naïve Bayes (NB) และ K-Nearest Neighbor (K-NN) ด้วยการใช้ด้วยซอฟต์แวร์ RapidMiner Studio Education ทดสอบประสิทธิภาพตัวแบบด้วยวิธีการ K – Fold Cross Validation โดยกำหนดค่า K เท่ากับ10 ใช้ค่าความถูกต้อง (Accuracy), ค่าความแม่นยำ (Precision), ค่าความระลึก (Recall), ค่า F-measure, เส้นโค้ง ROC (Receiver Operating Characteristic Curve) และ AUC (Area Under Curve) ในการเปรียบเทียบประสิทธิภาพ จากผลการทดลองพบว่า Naïve Bayes ให้ผลลัพธ์ที่ดีที่สุดของค่าความถูกต้องและค่าความแม่นยำอยู่ที่ 86.88% และ 86.94% ตามลำดับ และเมื่อพิจารณาด้วย AUC ก็พบว่าให้ผลลัพธ์สอดคล้องกัน สำหรับรอบที่สองผู้วิจัยศึกษาเชิงเปรียบเทียบการสร้างเพื่อวิเคราะห์ความรู้สึกของผู้เรียนด้วยวิธีการแบบร่วมกันตัดสินใจ (Ensemble Method) โดยวิธีการแบบร่วมกันตัดสินใจแบบโหวต (Voting) จะเป็นการพิจารณาโหวตผลลัพธ์จาก 3 โมเดล คือ DT, NB และ K-NN ในขณะที่วิธีการแบบร่วมกันตัดสินใจแบบแบ็กกิ้ง (Bagging) จะใช้อัลกอริทึม NB ในการสร้างโมเดล และสำหรับวิธีการแบบร่วมกันตัดสินใจแบบป่าสุ่ม (Random Forest) ก็จะใช้อัลกอริทึม DT ในการสร้างต้นไม้ในป่าสุ่มจากผลการทดลองพบว่าวิธีการแบบร่วมกันตัดสินใจแบบโหวตให้ผลลัพธ์ที่ดีที่สุดของค่าความถูกต้องและค่าความแม่นยำอยู่ที่ 89.06% และ 89.53% ตามลำดับ และเมื่อพิจารณาด้วย AUC ก็พบว่าให้ผลลัพธ์สอดคล้องกัน

ในงานวิจัยของ Gottipati และคณะ [28] ให้ความคิดเห็นว่าสามารถให้ข้อมูลเชิงลึกที่แฝงอยู่ในคำติชมของนักศึกษามีคุณค่าทั้งในเชิงคุณภาพ (Qualitative) และเชิงปริมาณ (Quantitative) ต่อการปรับปรุงในเรื่องของกระบวนการสอนและหลักสูตร ดังนั้นนักวิจัยจึงนำเสนอกรอบแนวคิดในการวิเคราะห์คำติชมจากจากนักศึกษาในมหาวิทยาลัยแห่งหนึ่ง ซึ่งเน้นการอธิบายถึงการพัฒนาโครงสร้างและเครื่องมือในการวิเคราะห์คำติชมจากนักศึกษา พร้อมทั้งมีการอธิบายถึงเครื่องมือในโปรแกรมประยุกต์ที่สามารถใช้ในการวิเคราะห์คำติชมจากนักศึกษาได้ โดยข้อมูลที่ใช้ในการศึกษาจะรวบรวมมาจากระบบสำหรับการแสดงความคิดเห็นของนักศึกษาแบบออนไลน์ที่เรียกว่า FACETS เป็นข้อมูลคำติชมจาก 1 หลักสูตรที่อยู่ใน School ซึ่งการแสดงความคิดเห็นของนักศึกษาจะเป็นการแสดงความคิดเห็นตามหัวข้อ (Topic หรือ Aspect) ภายใต้คำถาม (Question) เช่น คำติชมในประเด็นเรื่องกระบวนการเรียนการสอน หรือคำติชมในประเด็นเรื่องผู้สอน เป็นต้น

สำหรับกรอบแนวคิดของการวิเคราะห์คำติชมจากจากนักศึกษาที่นักวิจัยนำเสนอนั้น ผู้วิจัยเริ่มจากการทำความเข้าใจพื้นฐาน 5 ประเด็นคือ คำติชม (Comment) นั้น อยู่ในหัวข้อ (Topic หรือ Aspect) ไหน และข้อความในหัวข้อนั้นแสดงความรู้สึก (Sentiment) เป็นอย่างไร ในคำติชมนั้นมีข้อเสนอแนะ (Suggestion) หรือไม่ และคำติชมนั้นมีความสัมพันธ์ (Correlation) กับคะแนน (Score) ของคำติชมของนักศึกษาหรือไม่ จากความเข้าใจพื้นฐาน 5 ประเด็นนี้ จะเป็นเป้าหมายของผลลัพธ์ที่ได้จากการวิเคราะห์คำติชมจากนักศึกษา ซึ่งกรอบแนวคิดของการวิเคราะห์คำติชมจากนักศึกษาที่นักวิจัยนำเสนอจะประกอบด้วย 5 ขั้นตอนหลักคือ

1. โมเดลในการวิเคราะห์ข้อความ (Text Analytics Model) ในขั้นตอนนี้ นักวิจัยได้ทำการรวบรวมเครื่องมือด้านการประมวลผลภาษาธรรมชาติ (Natural Language Processing: NLP) ที่น่าจะจำเป็นต่อการวิเคราะห์คำติชมของนักศึกษา เครื่องมือเหล่านั้น เช่น การกำจัดคำหยุด (Stop-word) การทำสเต็มมิ่ง (Stemming) ตัววิเคราะห์คำเฉพาะ (Named Entity Taggers: NER) การสกัดวลีสำคัญ (Key-phrase Extraction)
   การโมเดลหัวข้อ (Topic Modeling) การสรุปความ (Text Summarization) เป็นต้น
2. การประมวลผลข้อมูล (Data Processing) เป็นขั้นตอนในการรวบรวมข้อมูลและการเตรียมข้อมูล พร้อมทั้งการจัดรูปแบบคำติชมให้เหมาะสมต่อการวิเคราะห์ในขั้นตอนถัดไป งานในขั้นตอนนี้ เช่น การกำจัดคำที่ไม่จำเป็นหรือการกำจัดคำหยุดออกไป เช่น คำว่า a, an, the, หรือ for เป็นต้น การพิจารณาคำที่เกิดร่วมกันที่อาจจะสื่อถึงความรู้สึก (Sentiment) เช่น คำว่า too fast, not easy เป็นต้น จากนั้นก็จัดรูปแบบเอกสารคำติชมเพื่อให้เหมาะสมต่อการวิเคราะห์ในขั้นตอนถัดไป
3. การสกัดสาระสำคัญ (Extraction) ขั้นตอนนี้ นักวิจัยอธิบายว่าเป็นขั้นตอนที่สำคัญที่สุดของกรอบการดำเนินงานที่นำเสนอ เพราะเป็นการขั้นตอนของการใช้เทคนิคเหมืองข้อความ (Text Mining) และอัลกอริทึมการเรียนรู้ของเครื่อง (Machine Learning)
   เพื่อค้นหาข้อมูลที่เป็นประโยชน์จากคำติชมของนักศึกษา ซึ่งข้อมูลที่เป็นประโยชน์ที่จะมองหาจากคำติชมได้แก่ หัวข้อ (Topic หรือ Aspect), ความรู้สึก (Sentiment) เป็นขั้วบวก (Positive) หรือขั้วลบ (Negative), และข้อเสนอแนะ (Suggestion)
4. การเชื่อมโยงความคิดเห็นเชิงปริมาณและเชิงคุณภาพ (Correlating Quantitative and Qualitative Feedback) เป็นการวิเคราะห์เพื่อพิจารณาความสัมพันธ์ระหว่างข้อความแสดงคำติชมกับคะแนนความพึงพอใจ (Rating Score) ที่นักศึกษาให้มานั้นว่ามีความสัมพันธ์กันหรือไม่
5. การสรุป (Summarization) เป็นขั้นตอนที่นำผลของการวิเคราะห์ข้างต้นมาสรุปรวมกัน เพื่อให้เห็นภาพรวมของการวิเคราะห์คำติชม

จากกรอบแนวคิดข้างต้น นักวิจัยได้พัฒนาต้นแบบระบบที่เรียกว่า “Student Feedback Mining Systems (SFMS)” และทำการทดสอบระบบด้วยคำติชมจาก 7 วิชา ซึ่งเริ่มจากการสร้างเมทริกซ์ของเอกสาร ซึ่งในที่นี้นักวิจัยใช้ doc2mat perl scripts ที่อยู่ใน Cluto library จากนั้นใช้

vcluster ในชุดเครื่องมือ (Toolkit) ในการจัดกลุ่ม (Clustering) เนื้อหาคำติชมตามหัวข้อหรือ Topic ที่ต้องการคือ ซึ่งในการจัดกลุ่มคำติชมใช้การจัดแบบ Cosine Similarity จากนั้นใช้การจำแนกเอกสาร (Text Classification) เพื่อวิเคราะห์ขั้วความรู้สึกของข้อความว่าให้ความรู้สึกเป็นขั้วบวกหรือขั้วลบ ซึ่งอัลกอริทึมที่ใช้ในการสร้างโมเดลเพื่อการจำแนกเอกสารคือ Logistic Regression ซึ่งจากการทดสอบพบว่าให้ผลลัพธ์ในแต่ละขั้นตอนเมื่อประเมินด้วยค่าความระลึก (Recall) ค่าความแม่นยำ (Precision) และค่า F1 ในระดับที่น่าพอใจ นั่นคือให้ค่าความระลึกเฉลี่ยที่ 86.4% ค่าความแม่นยำเฉลี่ยที่ 80.1% และค่า F1 เฉลี่ยที่ 83.5% และสุดท้ายนักวิจัยนำผลการวิเคราะห์คำติชมทั้งในส่วนของการวิเคราะห์หัวข้อ และการวิเคราะห์ความรู้สึก มาสรุปรวมกัน เพื่อให้เห็นภาพรวมของผลลัพธ์ของการวิเคราะห์คำติชมของนักศึกษา ท้ายที่สุด นักวิจัยให้ข้อสังเกตว่ากรอบงานที่นำเสนอมีประโยชน์ในการวิเคราะห์ความคิดเห็นของนักศึกษาในแต่ละหัวข้อที่สนใจและความรู้สึกของนักศึกษาในแต่ละในหัวข้อเหล่านั้น ผลลัพธ์แสดงให้เห็นว่าการใช้อัลกอริทึมการวิเคราะห์ข้อความและเทคนิคการสรุปผลมีประโยชน์ในการค้นหาข้อมูลเชิงลึกจากคำติชมเชิงคุณภาพ

ในงานวิจัยของ Qi และ Liu [30] ได้นำเสนอการทำเหมืองบทวิจารณ์ (Reviews Mining) ของหลักสูตรที่เรียนออนไลน์ (On-line Course) ของระบบการจัดการเรียนการสอนออนไลน์ระบบเปิดสำหรับมหาชน (Massive Open Online Courses: MOOC) ของประเทศจีน เนื่องจากว่าความคิดเห็นเหล่านี้สามารถสะท้อนทัศนคติของผู้เรียนที่มีต่อหลักสูตรออนไลน์ ที่สามารถจะช่วยให้ผู้เรียนคนอื่นเลือกหลักสูตรที่เหมาะสมกับตนเอง อีกทั้งยังช่วยให้ผู้สอนสามารถนำความคิดเห็นเหล่านั้นไปปรับปรุงหลักสูตรของตนเอง ดังนั้นนักวิจัยจึงใช้ประโยชน์จากบทวิจารณ์ของหลักสูตรที่เรียนออนไลน์จากระบบ MOOC ในการวิเคราะห์ทัศนคติของผู้เรียนที่มีต่อหลักสูตรดังกล่าว โดยชุดข้อมูลสำหรับใช้ในการศึกษานี้มาจากการสอนออนไลน์ในวิชา Python Language Programming ของ Beijing Institute of Technology และวิชา The Essence of C Language Programming ของ Harbin Institute of Technology สำหรับกระบวนการของเหมืองบทวิจารณ์ที่นักวิจัยนำเสนอนักวิจัยเริ่มจากการวิเคราะห์บทวิจารณ์ด้วย Latent Dirichlet Allocation (LDA) เพื่อให้ได้หัวข้อคำ (Topic-word) ได้แก่ Instructor, Course Content, Course Assessment, MOOC platform, และ Hot Courses นอกจากนี้ LDA ยังช่วยในการวิเคราะห์ความคิดเห็นที่สอดคล้องในแต่ละหัวข้อ (Comment-topic) อีกด้วย เมื่อได้หัวข้อความและความคิดเห็นที่สอดคล้องแล้ว ก็จะนำหัวข้อคำและความคิดเห็นเหล่านั้นไปแสดงในรูปแบบของเมทริกซ์การกระจาย (Distribution Matrix) จากนั้นจะทำการประมวลผลข้อความแสดงความรู้สึกที่สอดคล้องในแต่ละหัวข้อเพื่อให้ได้คะแนนความรู้สึก (Emotion Score) ด้วยเริ่มจากการศึกษาโมเดลการจำแนกเอกสารข้อความเชิงเปรียบเทียบหลายตัว ได้แก่ Random Forest, AdaBoost, SVM, BiLSTM, LSTM, GRU, LSTM with auto-encoder, Bi-LSTM with auto-encoder และ GRU with auto-encoder ในการจำแนกความรู้สึกของผู้เรียนว่าเป็นบวกหรือลบ ซึ่งจากผลการทดลองพบว่า Bi-LSTM with auto-encoder ให้ผลลัพธ์ที่ดีที่สุดโดยให้ความแม่นยำอยู่ 97.21% ซึ่งใช้การวนรอบการทำงานทั้งสิ้น 180 ครั้ง และหลังจากนั้นใช้วิธีการที่เรียกว่า weight coefficient ในการคำนวณคะแนนความรู้สึกของบทวิจารณ์ของแต่ละหัวข้อ โดยคะแนนที่คำนวณได้จะมีดีกรีอยู่ระหว่าง 0-5 ซึ่งคะแนนดังกล่าวจะถูกใช้ในการแนะนำหลักสูตรที่เรียนออนไลน์ให้กับนักศึกษาต่อไป

Wongkar และ Angdresey [31] นำเสนอการประยุกต์การวิเคราะห์ความรู้สึก (Sentiment Analysis) สำหรับการวิเคราะห์ความรู้สึกของประชาชนที่มีต่อผู้สมัครเป็นประธานาธิบดีสาธารณรัฐอินโดนีเซียในปี ค.ศ. 2019 ว่าเป็นบวก (Positive) หรือเป็นลบ (Negative) โดยข้อมูลที่ใช้ในการศึกษาเป็นข้อมูลจากทวิตเตอร์ (Twitter) ที่มีการรวบรวมในระหว่างเดือนมกราคม – พฤษภาคม ค.ศ. 2019 ซึ่งนักวิจัยได้ใช้เครื่องมือในไพทอนในการศึกษานี้ เริ่มจากขั้นตอนการเตรียมข้อมูล (Pre-processing) ได้แก่การตัดคำ (Tokenization) การทำความสะอาด (Cleaning) และการเลือกฟีเจอร์เป็นคำที่มีความหมาย (Meaningful Words) จากนั้นนำข้อความจากทวิตเตอร์เหล่านี้เข้าสู่ขั้นตอนการสร้างโมเดลการวิเคราะห์ความรู้สึกด้วยอัลกอริทึม Naïve Bayes ภายหลังจากที่โมเดลการวิเคราะห์ความรู้สึกด้วยอัลกอริทึม Naïve Bayes แล้ว นักวิจัยได้นำไปเปรียบเทียบกับโมเดลการวิเคราะห์ความรู้สึกที่พัฒนาด้วยอัลกอริทึม KNN และ SVM จากการทดสอบพบว่าโมเดลการวิเคราะห์ความรู้สึกด้วยอัลกอริทึม Naïve Bayes ให้ประสิทธิภาพที่ดีกว่าโมเดลการวิเคราะห์ความรู้สึกด้วยอัลกอริทึม KNN และ SVM ทั้งในด้านค่าความแม่นยำ (Precision) และค่าความถูกต้อง (Accuracy)

ในงานวิจัยของ Onan [32] ได้มีการนำเสนอวิธีการขุดข้อความเพื่อวิเคราะห์บทวิจารณ์ MOOC เพื่อนำเสนอรูปแบบการจัดประเภทความรู้สึกนึกคิดที่มีประสิทธิภาพพร้อมประสิทธิภาพเชิงคาดการณ์สูงในด้านการศึกษาด้วยการเรียนรู้ของเครื่อง (Machine Learning) การเรียนรู้ทั้งมวล (Ensemble Learning) และวิธีการเรียนรู้เชิงลึก (Deep Learning) สำหรับชุดข้อมูลที่ใช้ในการศึกษานี้ ถูกรวบรวมจากคลังข้อความเกี่ยวกับบทวิจารณ์ MOOC จากเว็บไซต์ coursetalk.com เป็นแพลตฟอร์มสำหรับการรีวิวหลักสูตรออนไลน์แบบเปิด โดยมีหลากหลายสาขาวิชา มีข้อมูลทั้งสิน 93,000 รายการ ในแพลตฟอร์มนี้จะใช้การแบ่งคะแนนออกเป็น 5 คะแนน ซึ่งคะแนนคุณภาพโดยรวมจะถูกคำนวณสำหรับหลักสูตรหนึ่งๆ และเพื่อให้ได้คลังข้อมูลที่มีป้ายกำกับ นักวิจัยจะทำการทบทวนประเมินผลด้วยคะแนนคุณภาพ 1 และ 2 จะถูกระบุว่าเป็น "เชิงลบ" ในขณะที่การทบทวนประเมินผลที่มีคะแนนคุณภาพ 4 และ 5 จะถูกระบุว่าเป็น "เชิงบวก" หลังจากกระบวนการติดฉลาก (Labeling Process) ซึ่งมีข้อมูลในบทวิจารณ์เชิงลบประมาณ 33,000 รายการ และบทวิจารณ์เชิงบวกประมาณ 37,000 รายการ เพื่อแก้ไขปัญหาข้อมูลที่ไม่สมดุล นักวิจัยได้ทำการแบ่งข้อมูล โดยมีความคิดเห็นเชิงลบ 33,000 รายการ และความคิดเห็นเชิงบวก 33,000 ซึ่งมีข้อมูลทั้งหมด 66,000 รายการ ในส่วนของการเตรียมข้อมูล นักวิจัยจะทำข้อความทั้งหมดให้อยู่ในรูปแบบมาตรฐาน ในขั้นตอนนี้ตัวอักษรทั้งหมดในคลังข้อความจะถูกแปลงเป็นอักษรตัวพิมพ์เล็กและเครื่องหมายวรรคตอนทั้งหมดถูกนำออก นอกจากนี้จะใช้กระบวนการ Tokenization เพื่อแยกประโยคที่กำหนดและคำในเอกสารออกและได้มีการใช้อัลกอริทึม Snowball เพื่อการใช้รากศัพท์ในคลังข้อความเพื่อลดจำนวนคำให้เป็นต้นกำเนิดของคำ นักวิจัยได้ทำการแบ่งข้อมูล 80% จากข้อมูลทั้งหมดเป็นชุดสอน (Training) และใช้ข้อมูล 20% จากข้อมูลทั้งหมดเป็นข้อมูลชุดทดสอบ (Testing)

ในส่วนของการสร้างการสร้างคุณลักษณะและการสร้างโมเดล เพื่อประเมินประสิทธิภาพการทำนายของแบบจำลองสำหรับการวิเคราะห์ความเชื่อมั่นใน EDM นักวิจัยได้ทำการแบ่งการทดลองออกเป็น 3 ชุด ได้แก่ การวิเคราะห์ความรู้สึกตามการเรียนรู้ของเครื่อง การเรียนรู้ทั้งมวล และการวิเคราะห์ความรู้สึกตามการเรียนรู้เชิงลึก

1. สำหรับงานประเมินเกี่ยวกับวิธีการเรียนรู้แบบมีผู้สอน นักวิจัยได้ทำการสร้างคุณลักษณะโดยการดึงข้อมูลจะใช้วิธีรูปแบบถุงคำ (Bag-Of-Words: BOW) โดยการใช้รูปแบบการถ่วงน้ำหนัก (Weighting Schemes) 3 ประเภท ได้แก่ TP, TF และ TF-IDF สำหรับในงานเหมืองข้อความ จะใช้โมเดล N-gram เพื่อแยกส่วนของอักขระ n ของเอกสารข้อความแล้ว แบบจำลอง N-gram ทั่วไปที่ใช้ในการวิเคราะห์ความรู้สึก ได้แก่ แบบจำลอง Unigram (N=1), bigram (N=2) และ Trigram (N=3) และนักวิจัยได้การเรียนรู้แบบมีผู้สอน 5 อัลกอริทึม ได้แก่ Naïve Bayes (NB), Support vector machines (SVM), Logistic Regression (LR), K-nearest neighbor (KNN), Random Forest (RF) สำหรับงานประเมินเกี่ยวกับวิธีการเรียนรู้แบบมีผู้สอน นักวิจัยได้ทำการสร้างคุณลักษณะโดยการดึงข้อมูลจะใช้วิธีรูปแบบถุงคำ (Bag-Of-Words: BOW) โดยการใช้รูปแบบการถ่วงน้ำหนัก (Weighting Schemes) 3 ประเภท ได้แก่ TP, TF และ TF-IDF สำหรับในงานเหมืองข้อความ จะใช้โมเดล N-gram เพื่อแยกส่วนของอักขระ n ของเอกสารข้อความแล้ว แบบจำลอง N-gram ทั่วไปที่ใช้ในการวิเคราะห์ความรู้สึก ได้แก่ แบบจำลอง Unigram (N=1), bigram (N=2) และ Trigram (N=3) และนักวิจัยได้การเรียนรู้แบบมีผู้สอน 5 อัลกอริทึม ได้แก่ Naïve Bayes (NB), Support vector machines (SVM), Logistic Regression (LR), K-nearest neighbor (KNN), Random Forest (RF)

2. สำหรับงานประเมินเกี่ยวกับวิธีการเรียนรู้ทั้งมวล มีจุดมุ่งหมายเพื่อระบุรูปแบบการเรียนรู้ที่มีประสิทธิภาพการทำนายที่สูงกว่า โดยจะใช้วิธี ดังนี้ AdaBoost, Bagging, Random Subspace, Voting และ Stacking

3. สำหรับงานประเมินเกี่ยวกับการเรียนรู้เชิงลึก นักวิจัยได้ทำการสร้างคุณลักษณะโดยการใช้รูปแบบการฝังคำ (Word-Embedding) 3 รูปแบบ ได้แก่ word2vec, fastText และ GloVe สำหรับการฝังคำ word2vec และ fastText ในวิธี continuous bag of words (CBOW) ได้รับการประเมินด้วยขนาดเวกเตอร์ที่แตกต่างกัน โดยมีขนาดเวกเตอร์ 200 และ 300 นอกจากนี้ยังมิติที่แตกต่างกันสำหรับ projection layers โดยมีขนาดมิติที่ 100 และ 200 การฝังคำยังได้ทำงานร่วมกับการเรียนรู้เชิงลึก 5 อัลกอริทึม ได้แก่ Convolutional Neural Network (CNN), Recurrent Neural Network (RNN), Gated Recurrent Unit (GRU), Long Short-Term Memory (LSTM) และ Recurrent neural network with attention mechanism (RNN-AM) ในการปรับการทดสอบและฝึกอบรม จะใช้ Tensorflow และ Keras เพื่อให้ได้ประสิทธิภาพการคาดการณ์ที่เหมาะสมที่สุดจากแบบจำลอง

หลังจากการทดสอบข้อมูล ได้ทำการประเมินประสิทธิภาพโดยใช้ ความแม่นยำในการจำแนกประเภท (Classification Accuracy) และค่าการวัด F (F-Measure) สำหรับการวิเคราะห์เชิงประจักษ์ (Empirical Analysis) ระบุว่าวิธีการเรียนรู้ทั้งมวลให้ประสิทธิภาพการทำนายที่สูงขึ้นในการทำเหมืองข้อมูลเมื่อเทียบกับวิธีการเรียนรู้แบบมีผู้สอน ผลลัพธ์เชิงประจักษ์ที่ได้จากรูปแบบจำลอง N-gram พบว่า Unigram ได้ประสิทธิภาพการทำนายสูงสุด สำหรับงานประเมิน TF แบบแผนการให้น้ำหนักแบบคำ มีประสิทธิภาพดีกว่าการนำเสนอแบบ TF-IDF และ TP สำหรับผลการเรียนรู้เชิงลึก LSTM มีประสิทธิภาพการทำนายสูงสุด และ RNN-AM มีประสิทธิภาพรองลงมาผลจากการทบทวน MOOC ระบุว่าการเรียนรู้เชิงลึก มีประสิทธิภาพเหนือกว่าวิธีการเรียนรู้แบบมีผู้สอน

และวิธีการเรียนรู้ทั้งมวล สำหรับการวิเคราะห์การฝังคำ GloVe ให้ประสิทธิภาพการคาดการณ์ที่สูงขึ้น เมื่อเทียบกับรูปแบบการฝังคำอื่นๆ ในส่วนของโมเดล fastText CBOW ได้ประสิทธิภาพการคาดคะเนสูงสุดเป็นอันดับสอง ซึ่งตามมาด้วยโมเดล fastText skip-gram เมื่อเปรียบเทียบผลจากการทบทวน MOOC พบว่าการเรียนรู้เชิงลึกมีประสิทธิภาพเหนือกว่าวิธีการเรียนรู้แบบมีผู้สอนและวิธีการเรียนรู้ทั้งมวล

ในงานวิจัยของ Osmanoglu และคณะ [33] ได้ทำการศึกษาวิเคราะห์ความรู้สึกนักเรียนในมหาวิทยาลัย Giga ที่อยู่ในระบบ eCampus Learning Management ซึ่งเป็นนักเรียนทางไกล โดยมีการเรียนการสอนแบบออนไลน์ (Online) ชุดข้อมูลในงานวิจัยจะถูกรวบรวมจากผลตอบรับจากนักเรียน โดยนักเรียน 1 คนสามารถตอบจำกัดเพียง 250 ตัวอักษร ซึ่งมีผลตอนรับทั้งหมด 6,059 รายการ ในงานวิจัยจะทำการข้อมูลโดยการปรับขนาดแบบ Triple Likert ซึ่งจะมีการแบ่งออกเป็น 3 คลาส ได้แก่ เชิงลบ เชิงกลาง และเชิงบวก แบบอัตโนมัติได้ดำเนินการตามคำในความคิดเห็นที่เขียนโดยผู้ใช้สำหรับ ตัวอย่างเช่น หากมีคำ เช่น “ไม่ดี”, “น้อย”, “ต่ำ” ในเนื้อหาของความคิดเห็นจะอยู่ในคลาสเชิงลบ หากในความคิดเห็น คำว่า “ควร”, “ถ้ามี”, “มากกว่า”, “น่าจะเป็น” จะถูกระบุว่าเป็นเชิงกลาง หากมีคำเช่น “ดี”, “ถูกใจ”, “สุดยอด” จะถูกระบุว่าเป็นเชิงบวก แต่เนื่องจากชุดข้อมูลไม่มีความสมดุลหลังจากการแบ่งคลาส การกระจายชุดข้อมูลจะเป็นดังนี้ : 4438, 815, 806 นักวิจัยได้ทำการใช้เทคนิคการสุ่มตัวอย่างแบบสุ่ม และกระจายไปถึง 800, 815, 806 ตามลำดับ ซึ่งจะเหลือผลตอบรับ 2,421 รายการ โดยจะทำการแบ่งข้อมูล 80% จากข้อมูลทั้งหมดเป็นชุดสอน (Training) และใช้ข้อมูล 20% จากข้อมูลทั้งหมดเป็นข้อมูลชุดทดสอบ (Testing) นอกจากนี้นักวิจัยยังได้ทำการประมวลผลล่วงหน้า โดยความคิดเห็นทั้งหมดจะถูกแปลงเป็นตัวพิมพ์เล็ก อักขระตุรกีจะถูกแปลงเป็นตัวอักษรละติน ละเว้นตัวเลข อักขระพิเศษ อีโมจิ คำที่ไม่จำเป็น และการตีความที่ไม่ใช่ภาษาละติน ซึ่งจะวิเคราะห์ข้อมูลด้วย 3เทคนิค ได้แก่ ข้อความสะอาด (Clean Text: CT) ตัวตรวจสอบการสะกด(Spell Checker: SC) และ หยุดคำ (Stop Words: SW) สำหรับในเกณฑ์ที่ใช้ในการเปรียบเทียบอัลกอริทึมการจำแนกประเภทเมทริกซ์ความสับสน จากนั้นนักวิจัยได้ศึกษาเชิงเปรียบเทียบการสร้างโมเดลเพื่อวิเคราะห์ความรู้สึกของนักเรียนด้วยอัลกอริทึม 7 ตัว ได้แก่ Decision Tree Classifier, MLP Classifier, XGB Classifier, Support Vector Classifier, Multinomial Logistic Regression, Gaussian NB และ KNeighbors Classifier ผลการศึกษาพบว่า ผลลัพธ์ที่ดีที่สุดคือ 0.775 ความแม่นยำของการทดสอบแบบจำลองของอัลกอริทึม Logistic Regression

ในงานวิจัยของ Kechaou และคณะ [34] ได้นำเสนออัลกอริทึมการจำแนกความรู้สึกตามการเรียนรู้เพื่อการจำแนกความคิดเห็นของผู้เรียนเกี่ยวกับบริการระบบอีเลิร์นนิง (E-Learning) โดยแบ่งออกเป็นเชิงบวกและเชิงลบเพื่อการปรับปรุงประสิทธิภาพในการจำแนกความรู้สึกในส่วนชุดข้อมูล นักวิจัยได้ทำการสร้างชุดข้อมูลโดยการดึงคลังข้อมูลจากบทวิจารณ์อีเลิร์นนิงที่ถูกรวบรวมจากบล็อกอีเลิร์นนิงจำนวนมาก และยังมีการรวบรวมความคิดเห็นบางส่วนจากฟอรัมของ Moodle จากเว็บไซต์ http://docs.moodle.org/en/Forums ซึ่งเป็นเว็บไซต์ที่ให้ผู้เรียนและผู้สอนสามารถแลกเปลี่ยนความคิดเห็นได้โดยการโพสต์ข้อความ ซึ่งมีจำนวนข้อความทั้งหมด 2,000 ข้อความ จะแบ่งเป็นข้อความเชิงบวก 1,000 ข้อความ และข้อความเชิงลบ 1,000 ข้อความ นักวิจัยได้ทำการแบ่ง

ข้อมูล 80% จากข้อมูลทั้งหมดเป็นชุดสอน (Training) และใช้ข้อมูล 20% จากข้อมูลทั้งหมดเป็นข้อมูลชุดทดสอบ (Testing) ในส่วนของขั้นตอนการเตรียมเอกสาร เอกสารทั้งหมดจะถูกปรับสภาพโดยการลบคำหยุด (Stop Words) การแยกประโยค (Stemming) และการเลือกคุณสมบัติจะใช้เกณฑ์ในการศึกษา ได้แก่ ข้อมูลที่ได้รับ (IG) ข้อมูลร่วม (MI) และสถิติ CHI (CHI) เพื่อจะใช้ในการเลือกเงื่อนไขสำหรับอบรมและการจัดหมวดหมู่ และในงานวิจัยยังใช้เทคนิคTerm Frequency (TF) เพื่อหาความถี่ของคำ และ Inversed Document Frequency TFIDF (IDF) เพื่อหาความถี่ของเอกสารผกผันที่กำหนดโดยสมการด้านล่าง: IDF=Log (N/n) โดยที่ N คือจำนวนเอกสารการฝึกอบรมทั้งหมด และ n คือจำนวนเอกสารที่คำศัพท์ ในการศึกษาจะตั้ง n = 3 สำหรับ n-grams (Unigrams, Bigrams และ Trigrams) และนักวิจัยยังได้ทำการประมวลผลล่วงหน้าก่อนที่จะเริ่มการจัดประเภท เนื่องจากการเรียนรู้อัลกอริทึมไม่สามารถจัดเก็บข้อความได้โดยตรง การประมวลผลล่วงหน้าจะเปลี่ยนเอกสารให้เป็นตัวอย่างที่เหมาะสม และเพื่อเตรียมความพร้อมของเอกสารไปสู่ในขั้นตอนการจัดหมวดหมู่

หลังจากนั้นนักวิจัยได้ทดสอบนำข้อมูลไปสร้างโมเดลในส่วนนี้จะใช้เทคนิคแมชชีนเลิร์นนิง (Machine Learning) ในการดูแลแบบผสมผสาน โดยใช้อัลกอริทึม Hidden Model Markov (HMM) และ Support Vector Machine (SVM) เพื่อระบุแนวโน้มความรู้สึกที่เป็นเชิงบวกหรือ เชิงลบ การใช้ HMM ในการจัดการกับการจัดประเภทข้อความความคิดเห็น จากชุดของบทวิจารณ์ข้อความที่เป็นของคลาส Cj เพื่อคำนวณพารามิเตอร์ต่างๆ ของโมเดล HMM ที่ควบคุมพื้นที่ ที่ซ่อนอยู่มีลักษณะเฉพาะด้วยชุดของ Q และ V ภายในสัญลักษณ์ที่สังเกตได้ของเวกเตอร์ และใช้โมเดล HMM มาใช้เป็นคำสั่งที่ยอมรับเงื่อนไขประสมทั้งหมด และเป็นสัญลักษณ์ที่แสดงถึงเงื่อนไขที่เกี่ยวข้องทั้งหมด (Unigrams) สำหรับ SVM เพื่อใช้ในการจำแนกอารมณ์ โดยข้อความจะถูกแบ่งออกเป็นสองประเภท เชิงบวกและเชิงลบ SVM จะหาพื้นหลังที่ชัดเจนซึ่งช่วยในการแบ่งจุดข้อมูลการฝึกอบรมออกเป็นสองประเภท (บวกและลบ) และตัดสินใจตามเวกเตอร์สนับสนุนที่เลือก นอกจากนี้ในงานวิจัยยังมีการรวมโมเดล HMM และ SVM ซึ่งจะเป็นการเพิ่มประสิทธิภาพดีว่าการแยกโมเดล และยังมีการรวมระหว่างUnigrams+Bigrams+Trigrams เพื่อบรรลุประสิทธิภาพโดยรวมสำหรับตัวแยกประเภทแบบรวมของนักวิจัย ซึ่งจะเป็นหลักฐานว่าการจำแนกขั้วสามารถได้รับประโยชน์จากลำดับ n-gram ที่สูงขึ้น จากผลการทดลองจะแบ่งออกเป็นการวัดประสิทธิภาพทั่วไปและการวัดประสิทธิภาพสำหรับการจำแนกประเภทความรู้สึก สำหรับการวัดประสิทธิภาพทั่วไปในการจำแนกประเภทจะขึ้นอยู่กับเมทริกซ์ความสับสน (Confusion Matrix) ที่ตรงข้ามกับคลาสที่กำหนด (คอลัมน์) ของตัวอย่างโดยตัวแยกประเภทที่มีคลาส (แถว) การวัดประสิทธิภาพสำหรับการจำแนกประเภทความรู้สึก ประเมินในแง่ของ Standard Precision (P) Recall (R) และ F-Measure (F) ผลการทดลองระบุว่า IG ทำงานได้ดีที่สุดสำหรับการเลือกเงื่อนไขทางอารมณ์และแสดงประสิทธิภาพที่ดีที่สุดสำหรับการจำแนกอารมณ์ในสถานการณ์ส่วนใหญ่ในแง่ของความแม่นยำ การเรียกคืน และการวัด F นอกจากนี้ MI ยังดีกว่า CHI เพียงเล็กน้อยเท่านั้น สำหรับความถูกต้องมีผลลัพธ์ที่ไม่ค่อยดีเนื่องจากลักษณะของคลังข้อมูลอีเลิร์นนิงที่รวบรวมโดยบล็อก มีความซับซ้อนของงานในการทำงานบนบล็อก บล็อกเกอร์ไม่ใช่นักเขียนมืออาชีพ ไม่ถูกต้องตามไวยากรณ์ของภาษาเป็นต้น ปัจจัยเหล่านี้มีผลอย่างมากต่อ ประสิทธิภาพของความแม่นยำในการจำแนกประเภท

ในงานวิจัยของปรีดาวรรณ เกษเมธีการุณ และคณะ [35] ได้พัฒนาระบบการวิเคราะห์ความรู้สึกจากวิดีโอบนโซเชียลมีเดียด้วยซัพพอร์ตเวกเตอร์แมชชีนบนพื้นฐานการวิเคราะห์ความรู้สึกของผู้บริโภคจากวิดีโอบนโซเชียลมีเดียเพื่อเป็นแนวทางในการตัดสินใจในการซื้อของผู้บริโภคและการจัดการข้อมูลของภาคธุรกิจต่อวิดีโอบนสื่อสังคมออนไลน์ทั้งเชิงบวกและเชิงลบโดยใช้เทคนิคการเรียนรู้ของเครื่อง งานวิจัยนี้ได้ทำการสกัดข้อความจากวิดีโอ จากนั้นข้อมูลจะผ่านกระบวนการตัดคำ และส่งไปเข้าสู่กระบวนการสกัดคุณลักษณะโดย Term Frequency Document Frequency (TF - IDF) และเข้าสู่ขั้นตอนการจำแนกประเภทด้วยซัพพอร์ตเวกเตอร์แมชชีน (Support Vector Machine: SVM) นอกจากนี้ยังได้ทำการศึกษางานวิจัยที่เกี่ยวข้องกับการวิเคราะห์ความรู้สึกบนสื่อสังคมออนไลน์ส่วนใหญ่จะอาศัยวิธีการเรียนรู้ของเครื่อง (Machine Learning) ด้วยการจำแนกประเภท เช่น Naïve Bayes, Decision Tree, Maximum Entropy และ Support Vector Machine (SVM) เป็นต้น ได้ดำเนินการวิจัยดังนี้

1. การรวบรวมข้อมูล จากวิดีโอบทวิจารณ์จาก Youtube จำนวน 500 วิดีโอ แต่ละวิดีโอมีความยาวอยู่ระหว่าง 2-5 นาที

2. การออกแบบระบบ ออกแบบระบบการวิเคราะห์ความรู้สึกจากวิดีโอบนโซเชียลมีเดียด้วยซัพพอร์ตเวกเตอร์แมชชีน โดยระบบประกอบไปด้วยส่วนการสกัดวิดีโอเป็นข้อความ ส่วนการตัดคำจากข้อความที่ได้จากการสกัดจากวิดีโอ ส่วนการสกัดคุณลักษณะของคำโดยอาศัยน้ำหนักของคำ นำมาคำนวณค่า Term Frequency (TF) และ Inverse Document Frequency (IDF) และการวิเคราะห์ความรู้สึกเชิงบวกหรือเชิงลบโดยทำการเปรียบเทียบจากฐานข้อมูล แสดงผลการวิเคราะห์ความรู้สึกเป็นเชิงบวกหรือเชิงลบ โดยฐานข้อมูลจะเก็บคุณลักษณะของคำว่าเป็นเชิงบวกหรือเชิงลบจะมีการปรับปรุงด้วยวิธีการจำแนกคุณลักษณะด้วย SVM นำคำที่ผ่านการจำแนกคุณลักษณะไปปรับปรุงและจัดเก็บในฐานข้อมูลเพื่อให้มีฐานข้อมูลมีความสามารถในการจำแนกความรู้สึกได้ดียิ่งขึ้น

3. การพัฒนาระบบ จะทำการพัฒนาเป็นระบบโดยเขียนด้วยภาษา PHP และฐานข้อมูล MySql ประกอบไปด้วย

   1) การสกัดข้อความจากวิดีโอ การพัฒนาระบบจะทำการเชื่อมโยงเพื่อเรียกใช้งาน Web Speech API ของกูเกิลโดยผู้ใช้สามารถนำไฟล์วิดีโอหรือใส่ลิงค์เข้าสู่ Web Speech API ซึ่งจะทำการสกัดออกมาเป็นข้อความจัดเก็บไว้เป็นไฟล์นามสกุล .txt

   2) การตัดคำ การพัฒนาระบบจะเขียนโปรแกรมเรียกใช้ฟังก์ชันการตัดคำ thsplitlib
   เพื่อทำการตัดคำไว้ในระบบ

   3) การสกัดคุณลักษณะด้วย TF และ IDF และการวิเคราะห์ความรู้สึกเชิงลบหรือเชิงบวก

4. การแสดงผลการวิเคราะห์ความรู้สึกเชิงลบหรือเชิงบวก จะทำการวิเคราะห์คำจากโปรแกรมตัดคำ thsplitlib ซึ่งมีคำ 76,314 คำ โดยผู้เชี่ยวชาญด้านภาษาจะทำการวิเคราะห์คำว่าคำใดมีความหมายเชิงบวกหรือเชิงลบและนำมาจัดเก็บเป็นฐานข้อมูล

5. การจำแนกคุณลักษณะด้วย SVM นำคุณลักษณะที่ได้นำไปฝึกฝนตัวจำแนกประเภทด้วยการใช้ SVM จากข้อมูลในการทดสอบจำนวน 500 ตัวอย่าง

จากการทดลองพบว่าระบบมีประสิทธิภาพดีทั้งด้านความถูกต้องอยู่ที่ร้อยละ 98 เมื่อเทียบกับผู้เชี่ยวชาญจำนวน 15 คน และทำการประเมินประสิทธิภาพด้านความพึงพอใจของผู้ใช้งานจำนวน 60 คน ต่อระบบ มีค่าเฉลี่ยเท่ากับ 4.79 ส่วนเบี่ยงเบนมาตรฐานที่ 0.14 กล่าวได้ว่าระบบการวิเคราะห์ความรู้สึกจากวิดีโอบนโซเชียลมีเดียด้วยซัพพอร์ตเวกเตอร์แมชชีนที่พัฒนาขึ้นสามารถวิเคราะห์ความรู้สึกจากวิดีโอภาษาไทยบนโซเชียลมีเดียได้อย่างมีประสิทธิภาพ

จากการศึกษางานวิจัยนี้ทำให้ทราบถึงขั้นตอนการออกแบบระบบการวิเคราะห์ความรู้สึกสามารถนำมาปรับใช้ในงานวิจัย เกี่ยวกับการสกัดข้อความ การตัดคำ การสกัดคุณลักษณะการวิเคราะห์ความรู้สึกเชิงลบหรือเชิงบวกด้วย TF และ IDF และการวิเคราะห์ความรู้สึกด้วย SVM

บทที่ 3
วิธีดำเนินการวิจัย

การดำเนินงานวิจัยในบทนี้ผู้วิจัยได้นำเสนอรูปแบบของการเก็บรวมรวมข้อมูลที่นำมาทดสอบ และขั้นตอนของการดำเนินงานของระบบการจำแนกความรู้สึกระดับประโยคสำหรับความคิดเห็นของนักเรียนมัธยมศึกษาตอนต้นในภาษาไทย ซึ่งจะมีกรอบการดำเนินงานดังนี้

3.1 ชุดข้อมูล (Dataset)

ในงานวิจัยนี้ใช้ชุดข้อมูลในการศึกษา โดยรวบรวมความคิดเห็นจากนักเรียนมัธยมศึกษาตอนต้น จำนวน 660 คน จากโรงเรียนมัธยมศึกษาในจังหวัดร้อยเอ็ด จำนวน 3 โรงเรียน ได้ทำการเก็บรวบรวมเป็น 3 ชุด ในระหว่างภาคเรียนที่ 1 และ ภาคเรียนที่ 2 ปีการศึกษา 2564 จำนวน 10 วิชา ซึ่งรายละเอียดแต่ละชุดข้อมูลสามารถอธิบายและแสดงตัวอย่างได้ดังนี้

ชุดข้อมูลที่ 1: จะเก็บเอกสารข้อมูลความคิดเห็นของนักเรียนแยกเป็นประโยค เป็นการแยกเก็บความคิดเห็นของนักเรียนในแต่ละ 3 Aspect ได้แก่ ด้านครูผู้สอน ด้านบทเรียน และด้านสภาพแวดล้อมการเรียนรู้ ตามลำดับ ซึ่งข้อมูลชุดนี้จะใช้ในการสร้าง “คลังคำ” ของแต่ละ Aspect ด้วย Skip gram ของ Word2Vec โดยตัวอย่างการจัดเก็บข้อมูลในชุดที่ 1 สามารถแสดงได้ดังตารางที่ 3.1

ตารางที่ 3.1 ตัวอย่างข้อมูลชุดที่ 1

Aspect	ID	ความคิดเห็นของนักเรียน
ด้านครูผู้สอน	1	ครูอธิบายเนื้อหาละเอียดเข้าใจง่าย
	2	ครูสอนเนื้อหาที่ทันสมัย
	...	ครูอธิบายเนื้อหาสับสนไม่เข้าใจ
ด้านบทเรียน	1	บทเรียนมีเนื้อหาที่ทันสมัยสอดคล้องเหตุการณ์ปัจจุบัน
	2	บทเรียนมีเนื้อหาตรงตามจุดประสงค์
	...	บทเรียนเนื้อหาไม่น่าสนใจไม่อัพเดทข้อมูล
ด้านสภาพแวดล้อม	1	ห้องเรียนสกปรกเหม็นอับไม่น่าเรียน
	2	โต๊ะเก้าอี้ชำรุดมีรอยขีดเขียน
	....	ห้องเรียนสะอาดดี

สำหรับข้อมูลในชุดที่ 1 นี้ เป็นประโยคที่เกี่ยวกับคุณครู 300 ประโยค ประโยคที่เกี่ยวกับบทเรียน 278 ประโยค และประโยคที่เกี่ยวข้องกับสภาพแวดล้อมในการเรียนอีก 250 ประโยค

ชุดข้อมูลที่ 2: เป็นชุดข้อมูลความคิดเห็นของนักเรียนใช้ในการสร้างโมเดลการวิเคราะห์ความรู้สึก โดยเป็นข้อมูลความคิดเห็นที่มีคลาสแสดงความรู้สึก แบ่งเป็น 2 กลุ่ม กลุ่มที่มีความคิดเห็น

เป็นบวก (Positive: POS) และกลุ่มที่มีความคิดเห็นเป็นลบ (Negative: NEG) ข้อมูลชุดนี้จะมีผู้เชี่ยวชาญด้านภาษาช่วยในการตรวจสอบความถูกต้องของขั้วความรู้สึก (Sentiment Polarity) โดยตัวอย่างการจัดเก็บข้อมูลในชุดที่ 2 สามารถแสดงได้ดังตารางที่ 3.2

ตารางที่ 3.2 ตัวอย่างข้อมูลชุดที่ 2

ID	ความคิดเห็นของนักเรียน	Sentiment Polarity (Class)
1	คุณครูน่ารักและสอนดีมาก เนื้อหาในบทเรียนค่อนข้างยากแต่ก็สนุกดี ห้องเรียนร้อนแต่สะอาด	Positive
2	คุณครูน่ารักและสอนดีมาก เนื้อหาที่เรียนก็โอเค แต่ห้องเรียนค่อนข้างร้อน	Positive
3	ครูดีมาก สอนก็ยาก	Negative
4	ครูก็ดี แต่สอนไม่ดี เนื้อหาวิชาก็ยาก บรรยากาศการเรียนก็เครียด	Negative

สำหรับข้อมูลความคิดเห็นในชุดที่ 2 นี้ เป็นความคิดเห็นเชิงบวกจำนวน 450 ความคิดเห็น และความคิดเห็นเชิงลบ 230 ความคิดเห็น

ชุดข้อมูลที่ 3: เป็นชุดข้อมูลทดสอบ โดยเป็นการเก็บข้อมูลความคิดเห็นของนักเรียน (Student Reviews) และจะมีผู้เชี่ยวชาญด้านภาษาช่วยในการตรวจสอบความถูกต้องของขั้วความรู้สึกของความคิดเห็นในแต่ละ Aspect หรือมุมมอง ได้แก่ความรู้สึกเกี่ยวกับครูผู้สอน ความรู้สึกเกี่ยวกับบทเรียน และความรู้สึกเกี่ยวกับสภาพแวดล้อม ว่าในแต่ละ “มุมมอง” ที่ปรากฏในความคิดเห็นนั้นมีขั้วความรู้สึกเป็น Positive หรือ Negative ตัวอย่างข้อมูลชุดที่ 3 ที่ใช้ในการทดสอบสามารถแสดงได้ดังตารางที่ 3.3

สำหรับข้อมูลความคิดเห็นในชุดที่ 3 นี้ เป็นความคิดเห็นของนักเรียนทั้งสิ้น 200 ความคิดเห็น โดย “กลุ่มคำ” ที่ไม่แสดงหรือเกี่ยวข้องกับ Aspect เช่น คำเชื่อม คำสันธาน จะถูกตัดออกไปในขั้นตอนที่ผู้เชี่ยวชาญคัดแยกประโยคและระบุขั้วความรู้สึกของประโยค เช่น ประโยค *“คุณครูน่ารัก เนื้อหาสนุก แต่ห้องเรียนร้อน”* คำว่า *“แต่”* จะถูกตัดออกไป

ตารางที่ 3.3 ตัวอย่างข้อมูลชุดที่ 3

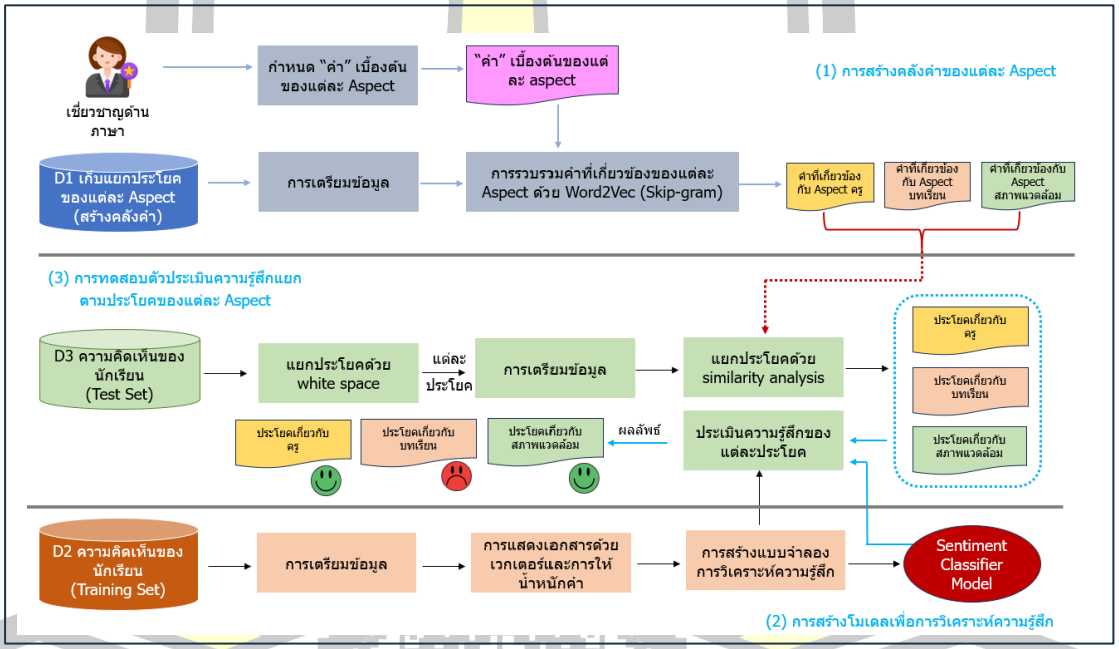
ID	ความคิดเห็นจากนักเรียน	Aspect	ประโยคของแต่ละ Aspect	ขั้วความรู้สึก
1	คุณครูน่ารักและสอนดีมาก เนื้อหาในบทเรียนค่อนข้างยาก ห้องเรียนร้อน	ครู	คุณครูน่ารักและสอนดีมาก	Positive
		บทเรียน	เนื้อหาในบทเรียนค่อนข้างยาก	Negative
		สภาพแวดล้อม	ห้องเรียนร้อน	Negative
2	คุณครูน่ารัก เนื้อหาสนุก แต่ห้องเรียนร้อน	ครู	คุณครูน่ารัก	Positive
		บทเรียน	เนื้อหาสนุก	Positive
		สภาพแวดล้อม	ห้องเรียนร้อน	Negative

3.2 เครื่องมือที่ใช้

ในการศึกษานี้จะใช้ไพทอน (Python) เป็นเครื่องมือในการศึกษา โดยชุดเครื่องมือหลักในไพทอนที่ใช้คือ Scikit-learn และ PyThaiNLP ซึ่ง Scikit-learn เป็นไลบรารีฟรีในภาษาไพทอนสำหรับการพัฒนาโปรแกรมโดยใช้การเรียนรู้ของเครื่อง ส่วน PyThaiNLP คือไลบรารี Python สำหรับงานด้านการประมวลผลข้อมูลภาษาไทย พัฒนาขึ้นมาโดยคนไทย มีฟังก์ชันที่มีประโยชน์มากมายสำหรับการประมวลผลภาษาไทย สามารถดาวน์โหลดได้ที่ https://pypi.org/project/pythainlp/

3.3 กรอบการดำเนินงานวิจัย

ในส่วนนี้จะแสดงกรอบการดำเนินงานวิจัยที่นำเสนอสำหรับการจำแนกความรู้สึกระดับประโยคสำหรับความคิดเห็นของนักเรียนมัธยมศึกษาตอนต้น ซึ่งสามารถแสดงได้ดังรูปที่ 3.1 โดยจะแสดงให้เห็นว่าในงานวิจัยมีการทำงานใน 3 ขั้นตอนหลัก ตามชุดข้อมูล 3 ชุดข้อมูลที่ใช้



รูปที่ 3.1 กรอบการดำเนินงานวิจัยที่นำเสนอสำหรับการจำแนกความรู้สึกระดับประโยคสำหรับความคิดเห็นของนักเรียนมัธยมศึกษาตอนต้น

ขั้นตอนที่ 1: การสร้างคลังคำของแต่ละ Aspect ด้วยชุดข้อมูลที่ 1

การสร้างคลังคำของแต่ละ Aspect ด้วยชุดข้อมูลที่ 1 จะเริ่มจากผู้เชี่ยวชาญด้านภาษากำหนดคำหลัก (Keywords) ของแต่ละ Aspect โดยคำหลักของแต่ละ Aspect สามารถแสดงได้ดังตารางที่ 3.4

ตารางที่ 3.4 คำหลักของแต่ละ Aspect ที่กำหนดโดยผู้เชี่ยวชาญด้านภาษา

Aspect	คำหลักของแต่ละ Aspect ที่กำหนดโดยผู้เชี่ยวชาญ
ครู	น่ารัก, สอนดี, ดุ, อธิบาย, ตรวจงาน, ใจดี, เข้มงวด
บทเรียน	เนื้อหา, ยาก, เข้าใจ, ซับซ้อน, น่าเบื่อ, คำนวณ
สภาพแวดล้อม	ร้อน, สกปรก, อับ, อบอ้าว, โต๊ะ, เก้าอี้, พัดลม

เมื่อได้ “คำหลัก” จากผู้เชี่ยวชาญ จะนำ “คำหลัก” เหล่านั้นมาวิเคราะห์หาคำที่มีใกล้เคียงด้วย Skip-gram ซึ่งเป็นโมเดลในการเรียนรู้ของเครื่องจากกลุ่ม Word2Vec มีวัตถุประสงค์เพื่อแปลงคำในข้อความเป็นเวกเตอร์ (Vector) ที่สามารถใช้ในการคำนวณทางคณิตศาสตร์ได้ Skip-gram ทำงานโดยการใช้ Target Word เพื่อทายคำรอบข้าง (Context Words) ในระยะที่กำหนด ในที่นี้ “คำหลัก” จากผู้เชี่ยวชาญ ก็คือ Target Word ที่ใช้ในการทำนายคำรอบข้างนั่นเอง โดยการทำงานของ Skip-gram มีดังนี้

1) การเลือกคำเป้าหมายและคำรอบข้าง: กำหนด Target Word และเลือกคำรอบข้างที่อยู่ในระยะที่กำหนด ตัวอย่างเช่น ถ้าคำว่า "น่ารัก" เป็นคำเป้าหมาย และระยะคำรอบข้างคือ 2 คำจากซ้ายและขวา คำรอบข้างอาจจะเป็น “มาก” และ “ใจดี”

2) การสร้างเวกเตอร์คำ: Skip-gram สร้างเวกเตอร์สำหรับแต่ละคำโดยใช้โครงสร้างของ neural network ที่มีเลเยอร์ซ่อน (hidden layer) เพียงชั้นเดียว โดยไม่ต้องการเลเยอร์ที่ซับซ้อนเนื่องจากไม่ได้ทำงานกับปัญหาที่มีโครงสร้างซับซ้อนเช่นการจำแนกประเภทประโยคหรือการแปลภาษา

3) การฝึกอบรมโมเดล: ในระหว่างการฝึกอบรม, โมเดลพยายามทายคำรอบข้างของคำเป้าหมาย โดยการปรับเปลี่ยนเวกเตอร์คำให้สามารถทายคำรอบข้างได้ดีขึ้น ค่าสูญเสีย (Loss) จะถูกคำนวณจากความแตกต่างระหว่างคำที่ทายได้และคำรอบข้างจริง และใช้เพื่อปรับปรุงเวกเตอร์คำ

4) การใช้เวกเตอร์คำ: เมื่อโมเดลได้รับการฝึกฝนอย่างเพียงพอ, เวกเตอร์คำที่ได้จะแสดงถึงความหมายของคำในพื้นที่เวกเตอร์ คำที่มีความหมายใกล้เคียงกันจะมีเวกเตอร์ที่อยู่ใกล้กันในพื้นที่นี้ ดังนั้นเมื่อได้เวกเตอร์คำ ก็สามารถใช้การคำนวณความคล้ายคลึงกัน (เช่น Cosine Similarity) เพื่อหาคำที่สอดคล้องกับคำใดคำหนึ่งได้

ขั้นตอนที่ 2: การสร้างโมเดลเพื่อการวิเคราะห์ความรู้สึก (Developing of Sentiment Classifier Model)

ในขั้นตอนนี้เป็นขั้นตอนที่ใช้ข้อมูลชุดที่ 2 ในการสร้างโมเดลเพื่อการวิเคราะห์ความรู้สึกของแต่ละประโยคแบบสองขั้วคือ Positive และ Negative โดยใช้อัลกอริทึมการเรียนรู้ของเครื่อง ซึ่งมีขั้นตอนการดำเนินงานดังนี้

การเตรียมข้อมูล (Text Pre-processing) – ขั้นตอนนี้ช่วยให้ข้อมูลมีรูปแบบที่เหมาะสมกับการวิเคราะห์และสามารถนำไปสู่ผลลัพธ์ที่ดีขึ้นเมื่อใช้กับโมเดลการเรียนรู้ของเครื่อง ขั้นตอนหลักๆ ในการเตรียมข้อมูลข้อความ เช่น

1) Tokenization: การแบ่งข้อความเป็นคำหรือ Token เพื่อทำให้ง่ายต่อการวิเคราะห์
2) Stop-word Removal: ลบคำที่ไม่มีความหมายมากในข้อความ เช่น “และ”, “ก็”, “ได้” เพื่อลดขนาดข้อมูล เพื่อเหลือคำที่มีความสำคัญเท่านั้น

การเปลี่ยนข้อความเป็นเวกเตอร์ (Vectorization) และการให้น้ำหนักคำ (Term Weighting) – เป็นกระบวนการเพื่อแปลงข้อความให้เป็นรูปแบบเวกเตอร์ (Vector) ของตัวเลขที่อัลกอริทึมการเรียนรู้ของเครื่องสามารถประมวลผลได้ โดยเวกเตอร์ดังกล่าวจะเรียกว่าแบบจำลองปริภูมิเวกเตอร์ (Vector Space Model: VSM) ในงานวิจัยนี้ รูปแบบเวกเตอร์คือ

1. TF-IDF (Term Frequency-Inverse Document Frequency): เป็นเทคนิคการให้น้ำหนักคำแบบไม่มีผู้สอนใช้เพื่อประเมินความสำคัญของคำภายในเอกสารเมื่อเปรียบเทียบกับชุดข้อมูลเอกสาร (Corpus) ทั้งหมด โดย TF-IDF ประกอบด้วย 2 ส่วนหลัก

1) Term Frequency (TF): ค่านี้เป็นการวัดความถี่ที่คำปรากฏในเอกสารเดียว ซึ่งบ่งบอกถึงความสำคัญของคำนั้นๆ ภายในเอกสารนั้นๆ เมื่อคำนั้นปรากฏบ่อยครั้ง ถือว่าคำนั้นๆ มีความสำคัญมาก อย่างไรก็ตาม ในการนับ TF อาจจะต้องมีการปรับน้ำหนักของคำที่ปรากฏบ่อยในภาษาทั่วไปด้วย
2) Inverse Document Frequency (IDF): ค่านี้เป็นการวัดความสำคัญของคำโดยดูจากความถี่ที่คำนั้นปรากฏในชุดข้อมูลเอกสารทั้งหมด คำที่ปรากฏในเอกสารจำนวนน้อยจะมีค่า IDF สูง ซึ่งบ่งบอกถึงความเฉพาะเจาะจงและความสำคัญที่มากขึ้น การคำนวณ IDF มักจะใช้สูตรที่ตรงกันข้ามกับความถี่ที่คำนั้นๆ ปรากฏในเอกสารต่างๆ ในชุดข้อมูล

TF-IDF ช่วยให้สามารถประเมินความสำคัญของคำได้ดีขึ้นโดยไม่เพียงแค่พิจารณาจากความถี่ของคำในเอกสารเดียว (ซึ่งอาจนำไปสู่การให้ความสำคัญกับคำทั่วไปที่ไม่มีความหมายมาก) แต่ยังพิจารณาจากความถี่ของคำนั้นในชุดข้อมูลเอกสารทั้งหมดด้วย ช่วยให้คำที่มีความเฉพาะเจาะจงมากขึ้นสามารถถูกน้ำหนักให้มีความสำคัญมากขึ้น

2. TF-IGM (Term Frequency-Inverse Gravity Moment): เป็นเทคนิคการให้น้ำหนักคำแบบมีผู้สอนที่ปรับปรุงหรือขยายมาจากเทคนิค TF-IDF แต่แนวคิดหลักของ TF-IGM คือการให้น้ำหนักกับคำตามความสำคัญและความเฉพาะเจาะจงของคำนั้นในชุดข้อมูลหรือเอกสารทั้งหมด แต่ด้วยวิธีการที่ต่างออกไปจาก IDF โดยพิจารณาจาก “ความหนัก (Gravity)” ของคำในการแบ่งประเภทหรือกลุ่มข้อมูล โดย TF-IGM ประกอบด้วย 2 ส่วนหลัก

1) Term Frequency (TF): คือความถี่ของคำ *t* ในเอกสาร *d*
2) Inverse Gravity Moment (IGM): ค่านี้เป็นการคำนวณความหนักหรือความเฉพาะเจาะจงของคำ *t* โดยพิจารณาจากการปรากฏในเอกสารต่างๆ ในแต่ละคลาสของชุดข้อมูล

แทนที่จะใช้ความถี่ที่คำปรากฏในเอกสารต่างๆ เพื่อคำนวณค่า IDF แบบที่ TF-IDF ทำ แต่ TF-IGM จะคำนวณค่า “Inverse Gravity Moment” โดยพิจารณาความหนักหรือความเฉพาะเจาะจงของคำนั้นในการแบ่งประเภทเอกสารหรือชุดข้อมูล นั่นคือมันพยายามวัดว่าคำนั้นมีบทบาทหรือความสำคัญในแต่ละคลาส เพื่อการแยกแยะหรือเข้าใจเนื้อหาหรือหัวข้อของเอกสารในแต่ละคลาสให้มากขึ้นนั่นเอง จากการคำนวณเช่นนี้เอง TF-IGM จึงมีประโยชน์โดยเฉพาะในสถานการณ์ที่คำบางคำมีความสำคัญสูงในการระบุคลาสของเอกสาร แต่อาจไม่ได้ปรากฏบ่อยในเอกสารนั้นๆ เท่าที่ควร โดยการให้น้ำหนักแก่คำเหล่านี้อย่างเหมาะสม โมเดลหรือระบบที่ใช้ TF-IGM สามารถปรับปรุงความสามารถในการเรียกค้นหรือจำแนกเอกสารได้อย่างแม่นยำมากขึ้น

การสร้างโมเดลเพื่อการวิเคราะห์ความรู้สึก – ในขั้นตอนนี้จะเริ่มจากแบ่งชุดข้อมูลแบบ 10-fold Cross Validation โดยแบ่งข้อมูลออกเป็นข้อมูลชุดสอน (Training Set) และข้อมูลชุดตรวจสอบ (Validation Set) จากนั้นนำข้อมูลชุดสอนมาสร้างโมเดลเพื่อการวิเคราะห์ความรู้สึกด้วยอัลกอริทึมการเรียนรู้ของเครื่อง ในงานวิจัยนี้ศึกษาเชิงเปรียบเทียบ 4 อัลกอริทึม คือ ขั้นตอนวิธีการเพื่อนบ้านใกล้ที่สุด (K-Nearest Neighbors: KNN) ซัพพอร์ตเวกเตอร์แมชชีน (Support Vector Machine: SVM) มัลติโนเมียลนาอีฟเบย์ (Multinomial Naïve Bayes: MNB) และป่าสุ่ม (Random Forest: RF) สำหรับข้อมูลชุดตรวจสอบใช้สำหรับปรับพารามิเตอร์และป้องกัน overfitting

การประเมินเพื่อเลือกโมเดลที่ดีที่สุด – ในการทดสอบการสร้างโมเดลเพื่อการวิเคราะห์ความรู้สึกด้วยชุดข้อมูลชุดตรวจสอบจะใช้ค่า Accuracy, F1 และ AUC หลังจากทำการทดลองหลายรอบ โมเดลเพื่อการวิเคราะห์ความรู้สึกที่ให้ผลลัพธ์ที่ดีที่สุดบนชุดตรวจสอบจะถูกเลือกเพื่อใช้ต่อไป

ขั้นตอนที่ 3: การนำโมเดลเพื่อการวิเคราะห์ความรู้สึกไปใช้กับชุดข้อมูลทดสอบ

ในขั้นตอนนี้จะใช้ข้อมูลชุดที่ 3 มาใช้ในการทดสอบโมเดลเพื่อการวิเคราะห์ความรู้สึกจากในระดับประโยคที่แสดงแต่ละ Aspect ที่ปรากฏในบทวิจารณ์ โดยมีขั้นตอนดังต่อไปนี้

ขั้นที่ 1: อ่านเอกสารบทวิจารณ์เข้ามา และทำการแยกประโยคด้วย white space ซึ่งหากตัดได้เป็นกลุ่มคำสั้นๆ เช่นคำว่า “ดังนั้น” หรือ “แต่ละ” จะถูกละทิ้ง

ขั้นที่ 2: เตรียมประโยคที่ตัดได้จากเอกสารบทวิจารณ์ด้วยการตัดคำและตัดคำหยุด

ขั้นที่ 3: นำประโยคในเอกสารที่ตัดได้แสดงในรูปแบบเวกเตอร์และแปลงค่าของคำแบบ One Hot Encoding นั่นคือ ค่า 0 (ไม่พบในคลังคำ) และ 1 (ไม่พบในคลังคำ) สมมติว่ามี

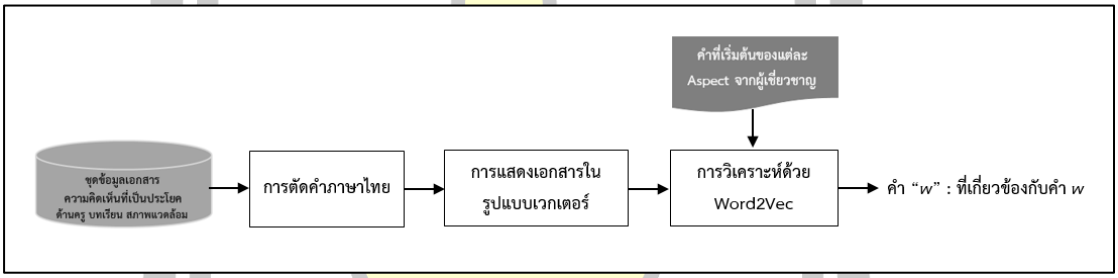
ประโยคคือ *“ครูใจดีและสอนดี”* ประโยคนี้จะถูกนำไปพิจารณาจากคลังคำของแต่ละ Aspect ที่ได้จากชุดข้อมูลชุดที่ 1 หลังจาก “คำ” ในประโยคถูกแปลงด้วย One Hot Encoding จะทำการจัดกลุ่มด้วยการวิเคราะห์ความคล้ายคลึงด้วย Euclident Distance เพื่อพิจารณาว่าประโยคนั้นจะอยู่ใน Aspect ที่เกี่ยวกับครู บทเรียน หรือสภาพแวดล้อม

ขั้นที่ 4: นำโมเดลเพื่อการวิเคราะห์ความรู้สึกที่สร้างจากข้อมูลชุดที่ 2 มาวิเคราะห์ว่าประโยคในแต่ละกลุ่ม Aspect เพื่อวิเคราะห์ว่าประโยคแต่ละประโยคนั้น ให้ความรู้สึกแบบ Positive หรือ Negative

ขั้นที่ 5: ประเมินความถูกต้องด้วย Recall, Precision, F1, และค่า Accuracy โดยเทียบกับผลเฉลยที่ผู้เชี่ยวชาญกำหนดไว้ให้

3.4 การสร้างคลังคำที่เกี่ยวข้องในแต่ละ Aspect ด้วย Skip-gram ของ Word2Vec

ชุดข้อมูลที่ 1 จะใช้สำหรับสร้างคลังข้อมูลของคำหลักที่เกี่ยวข้องกับความคิดเห็นแต่ละ Aspect เพื่อใช้เสมือนเป็น “คำ” ที่เกี่ยวข้องในแต่ละด้านด้วย Word2Vec โดยมีขั้นตอนดังนี้



รูปที่ 3.2 กรอบในการดำเนินงานเพื่อคัดเลือกคำที่เกี่ยวข้องในแต่ละ Aspect ด้วย Word2Vec

1. ผู้เชี่ยวชาญกำหนดคำเริ่มต้นให้ในแต่ละ Aspect เพื่อจะในการหาคำที่เกี่ยวข้องในแต่ละ Aspect จากชุดข้อมูลที่ 1 ดังที่แสดงในตารางที่ 3.4

2. อ่านเอกสารความคิดเห็นของนักเรียนเข้ามาในโปรแกรม แล้วทำการตัดคำแบบพจนานุกรมที่มีการเทียบคำที่เหมาะสมที่สุด (Dictionary-based Maximal Matching Method) ซึ่งการตัดคำแบบนี้จะพิจารณาตัดคำในทุกๆ รูปแบบที่เป็นไปได้ จากนั้นจะเลือกรูปแบบที่ได้จำนวน “คำ” ที่น้อยที่สุด ยกตัวอย่างเช่น ประโยค “คุณครูน่ารัก” สามารถตัดคำได้ 2 รูปแบบคือ

รูปแบบที่ 1: คุณ/ ครู/ น่ารัก
รูปแบบที่ 2: คุณครู/ น่ารัก

เมื่อพิจารณาจากจำนวนของ “คำ” พบว่ารูปแบบที่ 2 มีจำนวนคำที่น้อยที่สุด ดังนั้นจึงเลือกรูปแบบที่ 2 เป็นผลลัพธ์ของการตัดคำ

3. คำที่ได้จะถูกแสดงในรูปแบบของเวกเตอร์ จากนั้นนำเข้าสู่ Word2Vec โดยในการศึกษานี้จะใช้ Skip-gram ใน Word2vec สำหรับการสร้างคลังคำที่เกี่ยวข้องในแต่ละด้าน โดยในที่นี้มิติข้อมูลของเวกเตอร์สูงสุดที่ใช้คือ 300 เพราะชุดข้อมูลที่ใช้ในการศึกษามีขนาดไม่ใหญ่นัก

แต่ถ้าหากจะใช้มิติข้อมูลที่มีขนาดใหญ่กว่า 300 จะต้องใช้ชุดการฝึกขนาดใหญ่ ตัวอย่างข้อมูลที่เกี่ยวข้องกับ “คำ” เริ่มต้นของแต่ละ Aspect สามารถแสดงรูปแบบการวิเคราะห์ดังรูปที่ 3.3 และแสดงตัวอย่างได้ดังรูปที่ 3.3

รูปที่ 3.3 รูปแบบการพิจารณา “คำ” ด้วยโมเดล Skip-gram

สมมุติให้ มีประโยคเกี่ยวกับความคิดเห็นของนักเรียนทั้งหมด 3 ด้าน จำนวนด้านละ 4 เอกสาร ดังตารางที่ 3.5

ตารางที่ 3.5 ตัวอย่างเอกสารความคิดเห็น

ด้าน	เอกสาร
ครูผู้สอน	D1 : ครูอธิบายเนื้อหาละเอียดเข้าใจง่าย D2 : ครูสอนเนื้อหาที่ทันสมัย D3 : ครูอธิบายเนื้อหาสับสนไม่เข้าใจ D4 : ครูใส่ใจเฉพาะคนเรียนเก่ง
บทเรียน	D1 : บทเรียนมีเนื้อหาที่ทันสมัยสอดคล้องเหตุการณ์ปัจจุบัน D2 : บทเรียนมีเนื้อหาตรงตามจุดประสงค์ D3 : บทเรียนเนื้อหาไม่น่าสนใจไม่อัพเดทข้อมูล D4 : ยึดติดกับหลักสูตรเดิมล้าหลัง
สภาวพแวดล้อมการเรียนรู้	D1 : จัดบรรยากาศในห้องเรียนเอื้อต่อการเรียน D2 : จัดป้ายนิเทศให้ความรู้ D3 : ห้องเรียนสกปรกเหม็นอับไม่น่าเรียน D4 : โต๊ะเก้าอี้ชำรุดมีรอยขีดเขียน

ตัวอย่างแสดงการตัดคำด้วยด้วยวิธีการแบบพจนานุกรมที่มีการเทียบคำที่เหมาะสมที่สุด ดังแสดงในตารางที่ 3.6

ตารางที่ 3.6 ตัวอย่างแสดงการตัดคำ

ด้าน	ขั้นตอนการตัดคำในเอกสาร
ครูผู้สอน	D1 : ครู/ อธิบาย/ เนื้อหา/ ละเอียด/ เข้าใจ/ ง่าย D2 : คุณ/ ครู/ สอน/ เนื้อหา/ ที่/ ทันสมัย D3 : คุณ/ ครู/ อธิบาย/ เนื้อหา/ สับสน/ ไม่/ เข้าใจ D4 : คุณ/ ครู/ ใส่ใจ/ เฉพาะ/ คน/ เรียนเก่ง

บทเรียน	D1 : บทเรียน/ มี/ เนื้อหา/ ที่/ ทันสมัย/ สอดคล้อง/ เหตุการณ์/ ปัจจุบัน D2 : บทเรียน/ มี/ เนื้อหา/ ตรง/ ตาม/ จุดประสงค์ D3 : บทเรียน/ เนื้อหา/ ไม่/ น่าสนใจ/ ไม่/ อัพเดท/ ข้อมูล D4 : ยึด/ ติด/ กับ/ หลักสูตร/ เดิม/ ล้าหลัง
สภาวพแวดล้อมการเรียนรู้	D1 : จัด/ บรรยากาศ/ ใน/ ห้องเรียน/ เอื้อ/ ต่อ/ การเรียน D2 : จัด/ ป้ายนิเทศ/ ให้/ ความรู้ D3 : ห้อง/ เรียน/ สกปรก/ เหม็นอับ/ ไม่/ น่าเรียน D4 : โต๊ะ/ เก้าอี้/ ชำรุด/ มี/ รอย/ ขีดเขียน

จากนั้นทำการการตัดคำหยุด สมมุติคำว่า “คุณ”, “ครู”, “ที่”, “มี”, “ใน”, “ต่อ”, “คน”, “ตาม”, “กับ”, “เดิม”, “เฉพาะ”, “ให้” เป็นคำหยุดในภาษาไทย ดังนั้นสามารถยกตัวอย่างการตัดคำหยุดทั้งในเอกสารได้ดังตารางที่ 3.7

ตารางที่ 3.7 ตัวอย่างการตัดคำหยุด

ด้าน	การตัดคำหยุดในเอกสาร
ครูผู้สอน	D1 : อธิบาย/ เนื้อหา/ ละเอียด/ เข้าใจ/ ง่าย D2 : สอน/ เนื้อหา/ ทันสมัย D3 : อธิบาย/ เนื้อหา/ สับสน/ ไม่/ เข้าใจ D4 : ใส่ใจ/ เรียนเก่ง
บทเรียน	D1 : บทเรียน/ เนื้อหา/ ทันสมัย/ สอดคล้อง/ เหตุการณ์/ ปัจจุบัน D2 : บทเรียน/ เนื้อหา/ ตรง/ จุดประสงค์ D3 : บทเรียน/ เนื้อหา/ ไม่/ น่าสนใจ/ ไม่/ อัพเดท/ ข้อมูล D4 : ยึดติด/ กับ/ หลักสูตร/ ล้าหลัง
สภาวแวดล้อมการเรียนรู้	D1 : จัด/ บรรยากาศ/ ห้องเรียน/ เอื้อ/ ต่อ/ การเรียน D2 : จัด/ ป้ายนิเทศ/ ให้/ ความรู้ D3 : ห้องเรียน/ สกปรก/ เหม็นอับ/ ไม่/ น่าเรียน D4 : โต๊ะ/ เก้าอี้/ ชำรุด/ รอย/ ขีดเขียน

ในขั้นตอนการสร้างคลังคำด้วย Word2Vec นั้น ภายหลังจากทำการตัดคำและแสดงเอกสารด้วยเวกเตอร์

สมมติว่าจะสร้างคลังคำของคำว่า “ครู” เมื่ออ่านคำว่า “ครู” เข้ามาในโปรแกรม จากนั้นจะใช้ Skip-gram ของ Word2vec ในการทำนายคำที่มีความหมายใกล้เคียงในบริบท โดยสามารถแสดงตัวอย่างผลลัพธ์ที่ได้จากการประมวลผลดังรูปที่ 3.4 และรูปที่ 3.5

```
model.similar_by_word('ครู', topn=20)
```

```
[('อาจารย์', 0.4513567388057709),
 ('นักเรียน', 0.3599328398704529),
 ('คุณครู', 0.337405800819397),
 ('ผู้บริหาร', 0.3365687429904938),
 ('ลูกศิษย์', 0.33063027262687683),
 ('ศิษย์', 0.3306127190589905),
 ('ครูใหญ่', 0.31495144963264465),
 ('ศิษย์เก่า', 0.3079361915588379),
 ('บุคลากร', 0.30088862776756287),
 ('นักดนตรี', 0.2897532284259796),
 ('บัณฑิต', 0.28801798820495605),
 ('อาจารย์ใหญ่', 0.2849419116973877),
 ('ผู้ปกครอง', 0.28044888377189636),
 ('ผู้สอน', 0.27553054690361023),
 ('นักการ', 0.2748059630393982),
 ('อาสาสมัคร', 0.27435481548309326),
 ('ผู้เรียน', 0.2728964686393738),
 ('บาร์', 0.26190605759620667),
 ('คณาจารย์', 0.26095521450042725),
 ('ทางวิชาการ', 0.25726550817489624)]
```

รูปที่ 3.4 ตัวอย่างคลังคำที่สอดคล้องกับคำว่า “ครู”

```
model.most_similar_cosmul(positive=['อธิบาย','เนื้อหา'], negative=['เข้าใจ'], topn=20)
```

```
[('เนื้อเรื่อง', 0.810615599155426),
 ('บรรยาย', 0.8101048469543457),
 ('เนื้อเพลง', 0.7970755100250244),
 ('โครงเรื่อง', 0.7921794056892395),
 ('เค้าโครง', 0.7682165503501892),
 ('ใจความ', 0.764640748500824),
 ('เรื่องราว', 0.7633863091468811),
 ('นำเสนอ', 0.7627251744270325),
 ('ธีม', 0.751206636428833),
 ('เป็นแนว', 0.7481614351272583),
 ('กล่าวถึง', 0.7480083107948303),
 ('ดำเนินเรื่อง', 0.7479840517044067),
 ('ออกอากาศ', 0.747575044631958),
 ('ฉาก', 0.7459218502044678),
 ('ถ่ายทำ', 0.7453972697257996),
 ('รายละเอียด', 0.739524245262146),
 ('บทร้อง', 0.7362083196640015),
 ('เนื้อร้อง', 0.735675573348999),
 ('คำร้อง', 0.7353343367576599),
 ('อ้างอิง', 0.7333250045776367)]
```

คำที่มีความหมายใกล้เคียงกับประโยค

รูปที่ 3.5 Word2vec Skip-gram

การกำหนดค่าพารามิเตอร์ในแบบจำลอง Skip-gram มีผลต่อประสิทธิภาพในการเรียนรู้และทำนายบริบทของคำเป้าหมายได้ดี สำหรับการกำหนดค่าพารามิเตอร์หลักๆ ในแบบจำลอง Skip-gram มีดังนี้

ขนาดของหน้าต่างบริบท (Context Window Size) - หน้าต่างขนาดเล็ก (เช่น , 1-2) เหมาะสำหรับงานที่ต้องการบริบทใกล้ชิด ในขณะที่หน้าต่างขนาดใหญ่ (เช่น 5-10) เหมาะสำหรับการจับคู่ความหมายที่กว้างขึ้น

ขนาดของเวกเตอร์คำ (Embedding Size) – ขนาดของเวกเตอร์มีผลต่อความสามารถในการจับคู่ความละเอียดของความหมาย ขนาดเวกเตอร์ที่ใหญ่กว่า (เช่น 100-300) มักให้ผลลัพธ์ที่ดีกว่าสำหรับฐานข้อมูลขนาดใหญ่และงานที่ซับซ้อน, แต่ต้องใช้เวลาและหน่วยความจำมากขึ้นในการฝึก

อัตราการเรียนรู้ (Learning Rate) - ค่าที่เหมาะสมขึ้นอยู่กับงานและข้อมูล การเริ่มต้นด้วยอัตราการเรียนรู้สูงและลดลงเมื่อเวลาผ่านไป (Decay) สามารถช่วยให้การฝึกฝนมีประสิทธิภาพดีขึ้น

จำนวนรอบการฝึก (Epochs) – จำนวนรอบทั้งหมดที่ชุดข้อมูลใช้ในการฝึกแบบจำลอง จำนวน epochs ที่เพียงพอสำหรับแบบจำลองในการเรียนรู้โดยไม่เกิดการเรียนรู้เกินจำเป็น (Overfitting)

Activation function – แบบจำลองที่มีชั้นเชิงเส้นอย่างเดียวอาจไม่เพียงพอในการจับความซับซ้อนของความสัมพันธ์ระหว่างคำและบริบทของมันในภาษาธรรมชาติ ฟังก์ชันการเปิดใช้งานที่ไม่เชิงเส้นเช่น ReLU (Rectified Linear Unit) หรือ tanh (Hyperbolic Tangent) สามารถเพิ่มความสามารถในการเรียนรู้ความสัมพันธ์ที่ซับซ้อนเหล่านี้ได้ ในบางกรณี เช่น ชั้นสุดท้ายของแบบจำลอง Skip-gram ฟังก์ชันการเปิดใช้งานเช่น sigmoid หรือ softmax อาจถูกใช้เพื่อจำแนกและทำนายความน่าจะเป็นของคำบริบทเฉพาะจากคำเป้าหมาย Sigmoid มักใช้ในกรณีของการทำนายบริบทเป็นงาน binary classification (เช่น คำนี้เป็นบริบทที่ถูกต้องหรือไม่?) ในขณะที่ softmax ใช้เมื่อต้องการเลือกจากหลายคลาส (หลายคำบริบท)

Optimizer – Optimizer ในแบบจำลอง Skip-gram มีหน้าที่หลักคือปรับค่าน้ำหนักและไบแอสในเครือข่ายประสาทเพื่อลดค่าของฟังก์ชันการสูญเสีย การปรับน้ำหนักเหล่านี้ช่วยให้แบบจำลองสามารถทำนายบริบทของคำเป้าหมายได้ดีขึ้น ป้องกัน over fitting optimizer เช่น Adam ช่วยป้องกันปัญหาเช่นการเรียนรู้เกินจำเป็น ซึ่งเป็นสถานการณ์ที่แบบจำลองทำนายข้อมูลการฝึกได้ดีมาก แต่ทำนายข้อมูลใหม่ได้ไม่ดี

Batch size -Batch size มีผลต่อการควบคุมความเร็วและคุณภาพของกระบวนการเรียนรู้ โดย batch size ขนาดเล็กอาจทำให้โมเดลมีความยืดหยุ่นมากขึ้นในการปรับตัวตามข้อมูล แต่อาจทำให้กระบวนการเรียนรู้มีความผันผวนมากขึ้น ในทางกลับกัน batch size ขนาดใหญ่สามารถลดความผันผวนในการเรียนรู้และช่วยให้การปรับแต่งน้ำหนักมีความเสถียรมากขึ้น แต่อาจทำให้โมเดลตอบสนองต่อความแปรปรวนในข้อมูลได้ไม่ดีเท่าที่ควร

ตารางที่ 3.8 การกำหนดค่าพารามิเตอร์ใน Skip-gram สำหรับข้อมูลขนาดเล็ก

พารามิเตอร์	ค่าพารามิเตอร์ใน Skip-gram
Window Size	2
Embedding Size	300
Learning Rate	0.01
Epochs	1-3
Activation function	Sigmoid
Optimizer	Adam
Batch size	256

ภายหลังจากที่ได้กลุ่มคำที่สอดคล้องกับแต่ละ Aspect แล้ว คำเหล่านั้นจะถูกใช้เสมือนเป็น “คลังคำ” ของการศึกษานี้ โดยสามารถสรุปจำนวนคำที่ได้ดังแสดงในตารางที่ 3.9

ตารางที่ 3.9 สรุปจำนวนคำของแต่ละ Aspect ภายหลังการวิเคราะห์ด้วย Skip-gram

Aspect	จำนวนคำของแต่ละ Aspect
ครู	43
บทเรียน	37
สภาพแวดล้อม	28

สาเหตุที่ได้จำนวนคำที่สอดคล้องกับแต่ละ Aspect ไม่มากนัก อาจจะเนื่องมาจากว่า จำนวนประโยคที่ใช้ในการเรียนรู้มีนำนวนไม่มากนัก และมักเป็นประโยคแบบสั้น

3.5 การสร้างโมเดลเพื่อการวิเคราะห์ความรู้สึก

ในส่วนนี้จะเป็นการอธิบายขั้นตอนในการสร้างโมเดลเพื่อการวิเคราะห์ความรู้สึกด้วยอัลกอริทึมการเรียนรู้ของเครื่อง ในงานวิจัยนี้ศึกษาเชิงเปรียบเทียบ 4 อัลกอริทึม คือ ขั้นตอนวิธีการเพื่อนบ้านใกล้ที่สุด (K-Nearest Neighbors: KNN) ซัพพอร์ตเวกเตอร์แมชชีน (Support Vector Machine: SVM) มัลติโนเมียลนาอีฟเบย์ (Multinomial Naïve Bayes: MNB) และป่าสุ่ม (Random Forest: RF)

ขั้นที่ 1: การเตรียมข้อมูล

ในงานวิจัยนี้ทำการเตรียมข้อมูลด้วยทำการตัดคำแบบพจนานุกรมที่มีการเทียบคำที่เหมาะสมที่สุด (Dictionary-based Maximal Matching Method) ซึ่งการตัดคำแบบนี้จะ

พิจารณาตัดคำในทุกๆ รูปแบบที่เป็นไปได้ จากนั้นจะเลือกรูปแบบที่ได้จำนวน “คำ” ที่น้อยที่สุด ยกตัวอย่างเช่น ประโยค “คุณครูสวยและน่ารัก” สามารถตัดคำได้ 2 รูปแบบคือ

รูปแบบที่ 1: คุณ/ ครู/ สวย/ และ/ น่ารัก
รูปแบบที่ 2: คุณครู/ สวย/ และ/ น่ารัก

รูปแบบที่ได้จำนวนคำน้อยที่สุดจะถูกเลือก ดังนั้นจากตัวอย่างการตัดคำในรูปแบบที่ 2 จะถูกนำมาใช้ จากนั้นจะทำการตัดคำหยุด ดังนั้นจากตัวอย่างประโยค “คุณครูสวยและน่ารัก” ผลลัพธ์ที่ได้เมื่อทำการตัดคำหยุดแล้วจะได้ คุณครู/ สวย/ น่ารัก

ขั้นที่ 2: การแสดงข้อความและการให้น้ำหนักคำ

หลังจากขั้นที่ 1 แต่ละความคิดเห็นจะแสดงในรูปแบบเวกเตอร์ เรียกว่า Vector Space Model (VSM) ตัวอย่างการแทนเอกสารสามารถแสดงได้ดังตารางที่ 3.10

ตารางที่ 3.10 ตัวอย่างการแทนเอกสารด้วย VSM

	คุณครู	น่ารัก	สอน	ดี	มาก	เนื้อหา	บทเรียน	ค่อนข้างยาก	สนุก	ห้องเรียน	ร้อน	สะอาด	Class
D1	1	1	1	1	1	0	0	0	0	0	0	0	YES
D2	0	0	0	1	0	1	1	1	1	0	0	0	YES
D3	0	0	0	0	0	0	0	0	0	1	1	1	NO

1. ตัวอย่างการให้น้ำหนักด้วย tf-idf

จากข้อมูลข้างต้นจะเห็นว่ามีเอกสารทั้งหมด 3 เอกสาร เมื่อนำมาคำนวณหาค่าน้ำหนักด้วยสมการ $tf - idf$ จะสามารถแสดงขั้นตอนได้ดังนี้ คือ

ขั้นตอนที่ 1 : หาค่า tf ที่เป็นความถี่ของคำแต่ละคำที่อยู่ในเอกสารนั้นๆ ว่าพบกี่ครั้ง

ขั้นตอนที่ 2 : หาค่า idf คือการหาค่าส่วนกลับของแต่ละคำในเอกสารนั้นๆ

การคำนวณหา idf ทำได้โดยใช้สมการ $idf = \log N/df$ โดย N คือจำนวนเอกสารทั้งหมดในคลังเอกสาร และ df คือจำนวนเอกสารที่มีคำนั้นๆ ปรากฏอยู่ และสามารถคำนวณหาค่า idf ได้ดังนี้ในที่นี้จะให้ $N = 3$

$idf_{\text{คุณครู}} = \log(3/1) = 0.477$
$idf_{\text{น่ารัก}} = \log(3/1) = 0.477$
$idf_{\text{สอน}} = \log(3/1) = 0.477$
$idf_{\text{ดี}} = \log(3/2) = 0.176$

$idf_{\text{มาก}}$ $= \log(3/1)$ $= 0.477$

$idf_{\text{เนื้อหา}}$ $= \log(3/1)$ $= 0.477$

$idf_{\text{บทเรียน}}$ $= \log(3/1)$ $= 0.477$

$idf_{\text{ค่อนข้างยาก}}$ $= \log(3/1)$ $= 0.477$

$idf_{\text{สนุก}}$ $= \log(3/1)$ $= 0.477$

$idf_{\text{ห้องเรียน}}$ $= \log(3/1)$ $= 0.477$

$idf_{\text{ร้อน}}$ $= \log(3/1)$ $= 0.477$

$idf_{\text{สะอาด}}$ $= \log(3/1)$ $= 0.477$

ขั้นตอนที่ 3 : การคำนวณหาค่า $tf - idf$

ในขั้นตอนนี้จะเป็นการนำเอาค่า tf ที่ได้คูณเข้ากับค่า idf เช่น ในเอกสารที่ 1 จะพบคำ 5 คำ คือ “คุณครู”, “น่ารัก” , “สอน” , “ดี” และ “มาก” โดยคำเหล่านี้ที่ปรากฏในเอกสารที่ 1 มีค่า tf เป็น 1 และ 1 ตามลำดับ เมื่อนำมาหาค่า $tf - idf$ จะได้ผลลัพธ์ดังต่อไปนี้

$idf_{\text{คุณครู}}$ ในเอกสารที่ 1 $= 1 \times 0.477$ $= 0.477$

$idf_{\text{น่ารัก}}$ ในเอกสารที่ 1 $= 1 \times 0.477$ $= 0.477$

$idf_{\text{สอน}}$ ในเอกสารที่ 1 $= 1 \times 0.477$ $= 0.477$

$idf_{\text{ดี}}$ ในเอกสารที่ 1 $= 1 \times 0.176$ $= 0.176$

$idf_{\text{ดี}}$ ในเอกสารที่ 2 $= 1 \times 0.176$ $= 0.176$

$idf_{\text{มาก}}$ ในเอกสารที่ 1 $= 1 \times 0.477$ $= 0.477$

$idf_{\text{เนื้อหา}}$ ในเอกสารที่ 2 $= 1 \times 0.477$ $= 0.477$

$idf_{\text{บทเรียน}}$ ในเอกสารที่ 2 $= 1 \times 0.477$ $= 0.477$

$idf_{\text{ค่อนข้างยาก}}$ ในเอกสารที่ 2 $= 1 \times 0.477$ $= 0.477$

$idf_{\text{สนุก}}$ ในเอกสารที่ 2 $= 1 \times 0.477$ $= 0.477$

$idf_{\text{ห้องเรียน}}$ ในเอกสารที่ 3 $= 1 \times 0.477$ $= 0.477$

$idf_{\text{ร้อน}}$ ในเอกสารที่ 3 $= 1 \times 0.477$ $= 0.477$

$idf_{\text{สะอาด}}$ ในเอกสารที่ 3 $= 1 \times 0.477$ $= 0.477$

ดังนั้น ผลการคำนวณค่า $tf - idf$ ของคำในแต่ละเอกสาร จึงสามารถแสดงได้ดังตารางที่ 3.11

ตารางที่ 3.11 แสดงเอกสารการให้น้ำหนักด้วย $tf - idf$

	คุณครู	น่ารัก	สอน	ดี	มาก	เนื้อหา	บทเรียน	คุ้มค่ามาก	สนุก	ห้องเรียน	ร้อน	หนาว	Class
D1	0.477	0.477	0.477	0.176	0.477	0	0	0	0	0	0	0	YES
D2	0	0	0	0.176	0	0.477	0.477	0.477	0.477	0	0	0	YES
D3	0	0	0	0	0	0	0	0	0	0.477	0.477	0.477	NO

2. ตัวอย่างการให้น้ำหนักด้วย tf-igm

ในการหาค่า *tf* ก็คือการหาค่า *tf* แบบเดียวกันกับ *tf-idf* ส่วนสมการของ *igm*

$$w_{tf\text{-}igm(t_k)} = tf(t_i, d_j) \times (1 + \lambda \times igm(t_i))$$

สามารถแสดงได้ดังนี้

$$igm(t_i) = \frac{f_{i1}}{\sum_{r=1}^{m} f_{ir} \times r}$$

เมื่อ f_{ir} แสดงให้เห็นถึงจำนวนของเอกสารที่มีคำ t_i ในคลาสที่ r-th และมีการเรียงลำดับจากมากไปน้อย ในขณะที่ f_{i1} จะแสดงถึงความถี่ของคำ t_i ที่ปรากฏในคลาส

สำหรับในสมการการคำนวณค่าน้ำหนักแบบ *tf-igm* จะมีการใช้ตัวแปรแลมด้า (λ) เป็นค่าสัมประสิทธิ์ที่ปรับได้ (Adjustable Coefficient) ที่ถูกใช้เพื่อรักษาสมดุลสัมพัทธ์ระหว่างปัจจัยภายนอก (Global Factor) และปัจจัยภายใน (Local Factor) ในน้ำหนักของคำ โดยทั่วไปค่า λ จะถูกกำหนดไว้ที่ 7.0 เพราะในหลายๆ งานวิจัยให้ความคิดว่าเป็นค่าที่เหมาะสมที่สุด อย่างไรก็ตามค่า λ สามารถกำหนดให้อยู่ระหว่างค่า 5.0 ถึง 9.0 ซึ่งสมการ *tf-igm* สามารถแสดงได้ดังนี้

เมื่อ $tf(t_i, d)$ คือความถี่ของคำ t_i ที่พบในเอกสาร d_j การกำหนดค่า λ = 7.0 จะสามารถแสดงตัวอย่างการคำนวนค่าน้ำหนักคำของคำว่า *"คุณครู"* ด้วย *tf-igm* ได้ดังนี้

เมื่อพิจารณาข้อมูลจากตารางที่ 3.9 และการกำหนดค่า λ = 7.0 จะสามารถแสดงตัวอย่างการคำนวนค่าน้ำหนักคำของคำว่า *"คุณครู"* ด้วย *tf-igm* ได้ดังนี้

$$w_{tf-igm(t_{=\text{"คณิตศาสตร์"}})} = tf(t_i, d_j) \times (1 + \lambda \times igm(t_i))$$

$$w_{tf-igm(t_{=\text{"คณิตศาสตร์"}})} = tf(t_i, d_j) \times \left(1 + \lambda \times \frac{f_{i1}}{\sum_{r=f_{ir}}^{m} \times r}\right)$$

$$w_{tf-igm(t_{=\text{"คณิตศาสตร์"}})} = 1 \times \left(1 + 7.0 \times \frac{1}{(1 \times 1) \times (0 \times 2)}\right)$$

$W_{tf-igm(t_{=\text{คณิตศาสตร์}})} = 8$

ขั้นที่ 3: การสร้างโมเดลเพื่อการวิเคราะห์ความรู้สึกด้วยอัลกอริทึมการเรียนรู้ของเครื่อง

สำหรับข้อมูลที่ใช้ในการสร้างโมเดลเพื่อการวิเคราะห์ความรู้สึกนี้คือข้อมูลความคิดเห็นในชุดที่ 2 ซึ่งเป็นความคิดเห็นเชิงบวกจำนวน 450 ความคิดเห็น และความคิดเห็นเชิงลบ 230 ความคิดเห็น ดังนั้นเพื่อไม่ให้ปัญหาข้อมูลไม่สมดุล (Imbalanced data) ซึ่งหมายถึงสถานการณ์ในชุดข้อมูลที่จำนวนตัวอย่างในแต่ละคลาสไม่เท่ากันอย่างมาก ซึ่งส่งผลให้มีคลาสหนึ่งหรือหลายคลาสที่มีจำนวนตัวอย่างน้อยมากเมื่อเทียบกับคลาสอื่นๆ ในชุดข้อมูลเดียวกัน สิ่งนี้สามารถส่งผลต่อประสิทธิภาพของโมเดลการเรียนรู้ของเครื่อง เนื่องจากโมเดลอาจมีแนวโน้มที่จะทำนายคลาสที่มีตัวอย่างมากกว่าได้ดีกว่าคลาสที่มีตัวอย่างน้อย ดังนั้นการศึกษานี้จึงใช้ความคิดเห็นเชิงบวกจำนวน 200 ความคิดเห็น และความคิดเห็นเชิงลบ 200 เพื่อให้ข้อมูลในแต่ละคลาสมีความสมดุล

โดยอัลกอริทึมที่ใช้ในการศึกษาสำหรับการสร้างเพื่อการวิเคราะห์ความรู้สึกด้วยอัลกอริทึมการเรียนรู้ของเครื่องประกอบไปด้วย

1) ขั้นตอนวิธีการเพื่อนบ้านใกล้ที่สุด (K-Nearest Neighbors: KNN)

สมมุติต้องการสร้างโมเดลการจำแนกเอกสารเพื่อการวิเคราะห์ข้อมูลแบบ 2 กลุ่มคือ ความคิดเห็นด้านบวก และความคิดเห็นด้านลบ สมมุติกำหนดค่า $k = 5$ นั่นคือภายหลังจากการประมวลผลข้อมูลแล้ว จะพิจารณาข้อมูล 5 อันดับแรกที่อยู่ใกล้ x มากที่สุด

สมมุติให้มีเอกสารการจำแนกความรู้สึกจากเอกสารความคิดเห็นแบบข้อความทั้งหมด 10 เอกสาร คือ

D1 : คุณครูสอนดีนักเรียนเข้าใจง่าย
D2 : คุณครูสอนดีแจ้งเนื้อหาบทเรียนชัดเจน
D3 : สอนเนื้อหาที่ทันสมัยครูพูดเพราะ
D4 : คุณครูอธิบายงานได้เข้าใจง่าย
D5 : ครูยุติธรรมใส่ใจนักเรียนเท่ากัน
D6 : ครูไม่เปิดโอกาสให้นักเรียนซักถาม
D7 : ครูใส่ใจเฉพาะคนเรียนเก่ง
D8 : ครูก็อธิบายดีแต่ดุมากไปหน่อย
D9 : ครูสอนดีแต่เร็วเกินไป
D10: ครูสอนไม่ดี

การกำหนดพารามิเตอร์ที่สำคัญสำหรับ KNN สามารถแสดงได้ดังดังนี้

- จำนวนของเพื่อนบ้าน (K): ชุดข้อมูลขนาดเล็กอาจทำให้การเลือกค่า *k* ที่สูงเกินไปนำไปสู่การ overfitting หรือ underfitting ได้ โดยทั่วไป ค่า *k* ที่เหมาะสมควรเลือกให้ต่ำ แต่ไม่ต่ำเกินไปจนทำให้โมเดลไวต่อ “noise” ในข้อมูลงานวิจัยนี้ทดสอบที่ค่า 3, 5 และ 7
- วิธีการคำนวณระยะห่าง (Distance Metric): ใช้ Euclidean distance สำหรับการจำแนกความรู้สึกจากเอกสารข้อความสามารถแสดงได้โดยการใช้คุณลักษณะของข้อความเป็นเวกเตอร์ ซึ่งสามารถแสดงความคล้ายคลึงระหว่างเอกสารได้ด้วยการคำนวณระยะห่างระหว่างเวกเตอร์ของข้อความทั้งสอง การใช้ Euclidean distance ในการจำแนกความรู้สึกอาจเกี่ยวข้องกับการวัดระยะห่างระหว่างเอกสารกับจุดศูนย์กลาง (centroid) ของแต่ละกลุ่มอารมณ์หรือความรู้สึก เช่น บวก หรือ ลบ เพื่อจำแนกว่าเอกสารนั้นมีความรู้สึกใดมากที่สุด

ภายหลังจากเตรียมข้อมูล การแทนเอกสาร และการให้น้ำหนักคำด้วย tf-idf แล้วจะได้ผลลัพธ์ดังที่แสดงในตารางที่ 3.12 และ 3.13 ตามลำดับ

ตารางที่ 3.12 โมเดลเพื่อการวิเคราะห์ความรู้สึกจากเอกสารความคิดเห็นแบบข้อความด้วย KNN โดยการให้น้ำหนักคำด้วย tf

w_i	คนเก่ง	งาน	ง่าย	ชอบ	ชื่นชม	ดี	ตุ่มกด	ทำเสมัย	นักเรียน	ประโยค	ยุติธรรม	สอน	อธิบาย	เข้าใจ	เนื้อหา	เปิด	เร็ว	แจ้ง	แต่	โอกาส	ใส่ใจ	ไม่
D1	0	0	0.301	0	0	0.301	0	0	0	0	0	0.301	0	0.301	0	0	0	0	0	0	0	0
D2	0	0	0	0.301	0	0.301	0	0	0.301	0	0	0.301	0	0	0.301	0	0	0.301	0	0	0	0
D3	0	0	0	0	0	0	0	0.301	0	0.301	0	0.301	0	0	0.301	0	0	0	0	0	0	0
D4	0	0.301	0.301	0	0	0	0	0	0	0	0	0	0.301	0.301	0	0	0	0	0	0	0	0
D5	0	0	0	0	0	0	0	0	0	0	0.301	0	0	0	0	0	0	0	0	0	0.301	0
D6	0	0	0	0	0.301	0	0	0	0	0	0	0	0	0	0	0.301	0	0	0	0.301	0	0.301
D7	0.301	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.301	0
D8	0	0	0	0	0	0.301	0.301	0	0	0	0	0	0.301	0	0	0	0	0	0.301	0	0	0
D9	0	0	0	0	0	0.301	0	0	0	0	0	0.301	0	0	0	0	0.301	0	0.301	0	0	0
D10	0	0	0	0	0	0.301	0	0	0	0	0	0.301	0	0	0	0	0	0	0	0	0	0.301

ตารางที่ 3.13 โมเดลเพื่อการวิเคราะห์ความรู้สึกจากเอกสารความคิดเห็นแบบข้อความด้วย KNN โดยการให้น้ำหนักคำด้วย $tf - idf$

w_i	คนเก่ง	งาน	ง่าย	ชัดเจน	ซักถาม	ดี	ดุมาก	ทันสมัย	บทเรียน	พูดเพราะ	ยุติธรรม	สอน	อธิบาย	เข้าใจ	เนื้อหา	เปิด	เร็ว	แจ้ง	แต่	โอกาส	ใส่ใจ	ไม่	Class
D1	0	0	0.698	0	0	0.301	0	0	0	0	0	0.301	0	0.698	0	0	0	0	0	0	0	0	Yes
D2	0	0	0	1	0	0.301	0	0	1	0	0	0.301	0	0	0.698	0	0	1	0	0	0	0	Yes
D3	0	0	0	0	0	0	0	1	0	1	0	0.301	0	0	0.698	0	0	0	0	0	0	0	Yes
D4	0	1	0.698	0	0	0	0	0	0	0	0	0	0.698	0.698	0	0	0	0	0	0	0	0	Yes
D5	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0.698	0	Yes
D6	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0.698	No
D7	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.698	0	No
D8	0	0	0	0	0	0.301	1	0	0	0	0	0	0.698	0	0	0	0	0	0.698	0	0	0	No
D9	0	0	0	0	0	0.301	0	0	0	0	0	0.301	0	0	0	0	1	0	0.698	0	0	0	No
D10	0	0	0	0	0	0.301	0	0	0	0	0	0.301	0	0	0	0	0	0	0	0	0	0.698	No

ภายหลังจากเตรียมข้อมูล การแทนเอกสาร และการให้น้ำหนักคำด้วย tf-idf แล้ว เวกเตอร์ของเอกสารที่ได้เรียนรู้เพื่อสร้างโมเดลการจำแนกเอกสารด้วย KNN โดยมีขั้นตอนดังนี้

ขั้นตอนที่ 1: กำหนดค่า k - ขั้นตอนการกำหนดค่า k เป็นการกำหนดค่าเพื่อใช้เป็นเป้าหมายในการเลือกค่าที่ใกล้เคียงกับข้อมูลที่สนใจ โดยค่า k ที่กำหนดต้องเป็นเลขคี่ เพื่อให้โปรแกรมสามารถใช้ในการตัดสินใจได้ว่า x ควรถูกจัดอยู่ในกลุ่มใด ในชุดข้อมูลขนาดเล็กอาจทำให้การเลือกค่า k ที่สูงเกินไปนำไปสู่การ overfitting หรือ underfitting ได้ โดยทั่วไป ค่า k ที่เหมาะสมควรเลือกให้ต่ำ แต่ไม่ต่ำเกินไปจนทำให้โมเดลไวต่อ “noise” ในข้อมูลงานวิจัยนี้ทดสอบที่ค่า 3, 5 และ 7 f ดังนั้นในการศึกษานี้ จึงศึกษาด้วยการกำหนดค่า $k = 3, k = 5, k = 7$ ตามลำลับ

ขั้นตอนที่ 2: คำนวณหาระยะทางระหว่าง $\boldsymbol{x}$ กับข้อมูลทุกตัวในคลังข้อมูลการคำนวณค่าระยะทางระหว่างข้อมูลที่สนใจ กับข้อมูลทุกตัวในคลังข้อมูลจะใช้การคำนวณระยะทางด้วย Euclidian distance

$$E(x,y) = \sqrt{\sum_{i=0}^{n}(x_i - y_i)^2}$$

โดยที่ E คือ ระยะทางระหว่างเอกสาร x กับ y ซึ่งเป็นเอกสารที่เป็น centroid หรือ Center Point

x_i คือ คุณลักษณะที่ i ของเอกสาร x

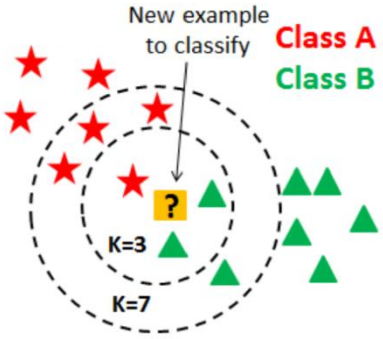
y_i คือ คุณลักษณะที่ i ของซึ่งเป็นเอกสารที่เป็น centroid

ซึ่งเอกสาร x จะถูกเปรียบเทียบกับเอกสาร y ที่เป็น centroid

ขั้นตอนที่ 3: จัดเรียงลำดับของระยะทาง - เมื่อวัดระยะทางระหว่างข้อมูลที่สนใจ x กับข้อมูลในคลังข้อมูลเสร็จเรียบร้อย จะมีการนำระยะทางที่วัดได้มาเรียงลำดับจากระยะทางที่น้อยที่สุดไปหามากที่สุด

ขั้นตอนที่ 4: พิจารณาข้อมูลที่ใกล้ที่สุด k ตัว - เมื่อทำการจัดเรียงลำดับของระยะทางแล้วจะเลือกค่าระยะทางที่น้อยที่สุดจำนวน k ตัวมาพิจารณาหาคำตอบ เช่น ถ้าหากค่า $k = 5$ ก็จะเลือกข้อมูลจากลำดับที่ 1 ถึง 5 มาพิจารณา

ขั้นตอนที่ 5: กำหนด Class ให้กับข้อมูล x - การกำหนด Class ให้กับข้อมูล x จะทำโดยการพิจารณาว่าข้อมูลจำนวน 5 ตัวที่อยู่ใกล้ x มากที่สุดอยู่กลุ่มใดบ้าง เช่น ถ้าข้อมูลในกลุ่มที่ 4 มีจำนวน 3 ตัว อยู่ในกลุ่ม 5 จำนวน 1 ตัว และกลุ่มที่ 2 จำนวน 1 ตัว ระบบจึงตัดสินใจให้ข้อมูล x อยู่ในกลุ่มที่ 4



รูปที่ 3.6 ตัวอย่างการจำแนกความรู้สึกจากเอกสารเมื่อ k=3 และ 7

อย่างไรก็ตาม มีข้อสังเกตว่าถ้าเลือกค่า k น้อยเกินไปอาจจะทำให้ไวต่อสัญญาณรบกวนได้ และถ้าหากเลือกค่า k มากเกินไปอาจจะทำให้มีกลุ่มข้อมูลอื่นๆ มาปะปนกับข้อมูลที่กำลังสนใจได้เช่นกัน ดังนั้นวิธีการนี้จึงมีทั้งข้อดีและข้อเสียซึ่ง ข้อดีคือเป็นวิธีการที่ง่ายและให้ประสิทธิภาพความถูกต้องสูง แต่ข้อเสียคือเวลาที่ใช้ในการประมวลผลค่อนข้างนาน เพราะการทำนายข้อมูลที่เข้ามาใหม่จะอาศัยการเปรียบเทียบข้อมูลใหม่กับข้อมูลเรียนรู้จำนวน k ตัวที่อยู่ใกล้ที่สุด

2) มัลติโนเมียลนาอีฟเบย์ (Multinomial Naïve Bayes: MNB)

ในการสร้างโมเดลเพื่อการจำแนกเอกสาร จะเป็นการยกตัวอย่างประยุกต์เอาอัลกอริทึมพหุนามนาอีฟเบย์มาใช้ในการจำแนกความรู้สึกจากเอกสารความคิดเห็นแบบข้อความ โดยจำแนกข้อมูลแบบ 2 กลุ่มคือ กลุ่มที่มีความคิดเห็นเป็นบวก และกลุ่มที่มีความคิดเห็นเป็นลบ

สำหรับการประยุกต์ใช้มัลติโนเมียลนาอีฟเบย์ในการสร้างโมเดลเพื่อการวิเคราะห์ความรู้สึก พารามิเตอร์ที่สำคัญคือ Alpha (α) (Laplace Smoothing Parameter) Alpha เป็นพารามิเตอร์สำหรับการปรับเรียบ (Smoothing) ซึ่งช่วยป้องกันปัญหาการคำนวณความน่าจะเป็นเป็นศูนย์ในกรณีที่คำหรือคุณลักษณะบางอย่างไม่ปรากฏในชุดข้อมูลการฝึก ค่าทั่วไปสำหรับ α อยู่ระหว่าง 0 ถึง 1 แต่ที่นิยมใช้คือ 1 ดังนั้นในงานวิจัยนี้จึงใช้ค่า Laplace smoothing = 1

สมมุติว่าให้เอกสารในตารางที่ 3.9 เป็นข้อมูลชุดสอน จะนำมาแสดงขั้นตอนการสร้างโมเดลเพื่อการจำแนกความคิดเห็นได้ดังนี้

ขั้นตอนที่ 1 : หาความน่าจะเป็นของแต่ละคลาส

จากตารางที่ 3.9 ข้อมูลมีจำนวน 3 เอกสาร เอกสารที่เป็นคลาส yes จำนวน 2 เอกสารและเอกสารที่เป็นคลาส no จำนวน 1 เอกสาร สามารถคำนวณหาค่าความน่าจะเป็นของแต่ละคลาสได้จากสมการที่ 1 ผลลัพธ์ที่ได้แสดงดังตารางที่ 3.14

ตารางที่ 3.14 คำนวณค่าความน่าจะเป็นของแต่ละคลาส

คลาส	ผลลัพธ์
yes	$\hat{P}(c_{yes}) = \frac{1 + Count(doc, c_j)}{Total\ of\ documents\ in\ corpus + NumClass} = \frac{1+2}{1+3} = 0.75$
no	$\hat{P}(c_{no}) = \frac{1 + Count(doc, c_j)}{Total\ of\ documents\ in\ corpus + NumClass} = \frac{1+1}{1+3} = 0.5$

ขั้นตอนที่ 2 : หาความน่าจะเป็นของ “คำ” ที่จะเกิดขึ้นในแต่ละคลาส

จากตารางที่ 3.10 ข้อมูลชุดสอน คำในถุงคำมีทั้งหมด 12 คำ ได้แก่ คุณครู, น่ารัก, สอน, ดี, มาก, เนื้อหา, บทเรียน, ค่อนข้างยาก, สนุก, ห้องเรียน, ร้อน, สะอาด คำที่พบในคลาส yes จำนวน 9 คำ ได้แก่ คุณครู, น่ารัก, สอน, ดี, มาก, เนื้อหา, บทเรียน, ค่อนข้างยาก, สนุก คำที่พบในคลาส no จำนวน 3 คำ ห้องเรียน, ร้อน, สะอาด โดยคำทั้งหมดได้มีการให้น้ำหนักคำแบบ $tf - idf$ ไว้แล้ว สามารถคำนวณได้จากสมการ

$$\hat{P}(w_i|c_j) = \frac{\sum(tf - idf(w_i), c_j) + 1}{\sum_{weV}(tf - idf(w)c_j) + |V|}$$

เมื่อ $\sum(tf - idf(w_i), c_j)$ คือผลรวมของค่าน้ำหนักคำแบบ $tf - idf$ ของคำ w_i ในคลาส c_j ในขณะที่ $\sum_{weV}(tf - idf(w)c_j)$ คือผลรวมของค่าน้ำหนักแบบ $tf - idf$ ของคำทั้งหมดที่พบในคลาส c_j จากสมการข้างต้น คำนวณหาความน่าจะเป็นของแต่ละ “คำ” ที่พบในแต่ละคลาส สามารถคำนวณได้ดังนี้

ตารางที่ 3.15 การคำนวณค่าความน่าจะเป็นของแต่ละ “คำ” ในคลาส c_j

$\hat{P}(w_i\|c_{yes})$		$\hat{P}(w_i\|c_{no})$	
$P(\text{คุณครู}\|c_{yes})$	$= \frac{1+0}{12+4.168} = 0.061$	$P(\text{ห้องเรียน}\|c_{no})$	$= \frac{1}{12+1.43} = 0.074$
$P(\text{น่ารัก}\|c_{yes})$	$= \frac{1+0}{12+4.168} = 0.061$	$P(\text{ร้อน}\|c_{no})$	$= \frac{1}{12+1.43} = 0.074$
$P(\text{สอน}\|c_{yes})$	$= \frac{1+0}{12+4.168} = 0.061$	$P(\text{สะอาด}\|c_{no})$	$= \frac{1}{12+1.43} = 0.074$
$P(\text{ดี}\|c_{yes})$	$= \frac{1+1}{12+4.168} = 0.123$		
$P(\text{มาก}\|c_{yes})$	$= \frac{1+0}{12+4.168} = 0.061$		
$P(\text{เนื้อหา}\|c_{yes})$	$= \frac{0+1}{12+4.168} = 0.061$		
$P(\text{บทเรียน}\|c_{yes})$	$= \frac{0+1}{12+4.168} = 0.061$		

$P(\text{ค่อนข้างยาก}\mid c_{yes}) = \frac{0+1}{12+4.168} = 0.061$	
$P(\text{สนุก}\mid c_{yes}) = \frac{0+1}{12+4.168} = 0.061$	

3) ป่าสุ่ม (Random Forest: RF)

RF เป็นอัลกอริธึมการเรียนรู้ของเครื่องที่อยู่ในประเภทของ Ensemble Learning ซึ่งทำงานโดยการสร้างหลายๆ ต้นไม้การตัดสินใจ (Decision Trees) และผสมผสานผลลัพธ์จากต้นไม้เหล่านั้นเพื่อทำการทำนายที่ดีขึ้น โมเดลนี้สามารถใช้สำหรับงานทั้งการจำแนกประเภท (ในการวิเคราะห์ความรู้สึก RF สามารถใช้เพื่อจำแนกความรู้สึกในข้อความว่าเป็นบวก ลบ หรือกลาง โดยขึ้นอยู่กับเนื้อหาของข้อความ RF เป็นอัลกอริธึมที่ทรงพลังและมีความสามารถในการจัดการกับชุดข้อมูลที่มีความซับซ้อนและคุณลักษณะที่มีมิติสูง ทำให้เหมาะสำหรับงานการวิเคราะห์ความรู้สึก นอกจากนี้ยังมีความสามารถในการป้องกัน overfitting ได้ดีเมื่อเทียบกับการใช้ต้นไม้การตัดสินใจเพียงต้นเดียว ในการศึกษานี้ต้นไม้ตัดสินใจที่เลือกใช้คือ CART (Classification And Regression Trees) เนื่องจากเป็นต้นไม้การตัดสินใจที่ใช้กันอย่างแพร่หลายในการสร้าง RF

เมื่อใช้ Random Forest สำหรับสำหรับการสร้างโมเดลเพื่อการวิเคราะห์ความรู้สึกแบบ 2 กลุ่ม (เช่น บวกและลบ), การปรับพารามิเตอร์อย่างเหมาะสมสามารถช่วยให้โมเดลทำงานได้ดีขึ้นและมีความแม่นยำสูงขึ้น บางส่วนของพารามิเตอร์หลักในโมเดล RF ที่ต้องการพิจารณาสามารถแสดงได้ดังตารางที่ 3.16

ตารางที่ 3.16 พารามิเตอร์หลักในโมเดล RF ที่ต้องการพิจารณาสำหรับการสร้างโมเดลเพื่อการวิเคราะห์ความรู้สึกแบบ 2 กลุ่มด้วยข้อมูลขนาดเล็ก

พารามิเตอร์	ความหมาย	ค่าโดยทั่วไปที่เหมาะสม	ค่าที่ใช้
n_estimators	จำนวนต้นไม้ในป่า (Forest) ค่านี้สูงขึ้นอาจช่วยเพิ่มความแม่นยำของโมเดล แต่ก็ทำให้เวลาการฝึกอบรมและความต้องการหน่วยความจำเพิ่มขึ้นเช่นกัน	100-200	100
max_depth	ความลึกสูงสุดของแต่ละต้นไม้ การจำกัดความลึกของต้นไม้สามารถช่วยป้องกันปัญหา Overfitting	3-10	3
min_samples_split	จำนวนเอกสารขั้นต่ำที่จำเป็นสำหรับการแยกโหนด (Node) การเพิ่มค่านี้สามารถช่วยป้องกันการฝึกต้นไม้ที่ซับซ้อนเกินไป	2-10	2
min_samples_leaf	จำนวนตัวอย่างขั้นต่ำที่จำเป็นใน Leaf Node การเพิ่มค่านี้สามารถช่วยให้โมเดลมีความทนทานมากขึ้นและลดการ overfitting	1-6	2

max_features	จำนวนคุณลักษณะสูงสุดที่จะพิจารณาเมื่อแยกที่แต่ละโหนด ค่านี้สามารถส่งผลต่อความแม่นยำและความเร็วในการฝึกอบรมของโมเดล	sqrt(จำนวนคุณลักษณะ)	sqrt(108) ≈ 10
bootstrap	การใช้ Bootstrap Bamples ในการฝึกต้นไม้ โดยปกติแล้วค่านี้จะเป็น “True” ซึ่งหมายถึงการใช้การสุ่มตัวอย่างเพื่อฝึกต้นไม้แต่ละต้น	“True”	“True”

การปรับพารามิเตอร์เหล่านี้ต้องอาศัยการทดลองและการประเมินผลเพื่อหาค่าที่เหมาะสมสำหรับชุดข้อมูลและงานวิเคราะห์ความรู้สึกเฉพาะของคุณ เครื่องมืออย่าง GridSearchCV หรือ RandomizedSearchCV ในไลบรารี Scikit-learn สามารถช่วยให้ค้นหาค่าพารามิเตอร์ที่ดีที่สุดได้โดยอัตโนมัติ

หาก max_depth ตั้งไว้สูงเกินไป โมเดลอาจเรียนรู้รายละเอียดและความผิดปกติในข้อมูลการฝึกอบรมจนเกินไป ซึ่งอาจนำไปสู่ปัญหาการทำนายข้อมูลใหม่ที่ไม่แม่นยำ ในทางกลับกัน ถ้า max_depth ตั้งไว้ต่ำเกินไป โมเดลอาจไม่สามารถเรียนรู้ความซับซ้อนของข้อมูลได้เพียงพอนำไปสู่ปัญหา underfitting โดยทั่วไปจึงตั้งค่านี้ระหว่าง 3-10 ในกรณีที่ข้อมูลมีขนาดเล็ก

ค่า min_samples_split สำหรับข้อมูลขนาดเล็ก ที่แนะนำอาจอยู่ระหว่าง 2 ถึง 10 ค่านี้ไม่ควรสูงเกินไปเพราะอาจป้องกันไม่ให้ต้นไม้เติบโตได้เพียงพอที่จะตรวจจับความซับซ้อนของข้อมูล

ค่า min_samples_leaf ที่เหมาะสมสำหรับ RF เพื่อการวิเคราะห์ความรู้สึกแบบ 2 กลุ่ม ด้วยข้อมูลขนาดเล็ก ที่แนะนำอาจอยู่ในช่วง 1 ถึง 6 การตั้งค่านี้เพิ่มขึ้นอาจช่วยให้ต้นไม้มีความทนทานต่อการ overfitting โดยการป้องกันไม่ให้มีการเรียนรู้ข้อมูลฝึกอบรมเฉพาะเจาะจงเกินไป

การเลือกค่า max_features สำหรับข้อมูลขนาดเล็ก มักอยู่ที่ sqrt (จำนวนคุณลักษณะทั้งหมด) เป็นจุดตั้งต้น อย่างไรก็ตาม หากข้อมูลมีขนาดเล็กมากอาจพิจารณาทดลองด้วยค่าต่ำกว่า sqrt (จำนวนคุณลักษณะ) เพื่อลดความซับซ้อนของแต่ละต้นไม้ การตั้งค่าพารามิเตอร์นี้อย่างเหมาะสมสามารถช่วยปรับปรุงประสิทธิภาพของโมเดลและป้องกันการ overfitting โดยเฉพาะอย่างยิ่งในข้อมูลขนาดเล็ก

4) ซัพพอร์ตเวกเตอร์แมชชีน (Support Vector Machine: SVM)

SVM เป็นอัลกอริธึมการเรียนรู้ของเครื่องที่มีประสิทธิภาพสูงและเหมาะสำหรับงานการจำแนกประเภท, รวมถึงการวิเคราะห์ความรู้สึกแบบ 2 กลุ่ม (บวกและลบ) แม้ว่า SVM จะสามารถทำงานได้ดีกับชุดข้อมูลที่มีขนาดใหญ่ แต่ก็สามารถทำงานกับชุดข้อมูลขนาดเล็กเนื่องจากความสามารถในการหา Hyperplane ที่ดีที่สุดเพื่อแยกข้อมูลในคลาสต่างๆ การเลือกเคอร์เนล (Kernel) ที่เหมาะสม (เช่น linear, polynomial, RBF) มีความสำคัญต่อประสิทธิภาพของโมเดล SVM ในการจำแนกความรู้สึก เคอร์เนลแบบ Linear อาจเหมาะสำหรับชุดข้อมูลที่ความซับซ้อนต่ำและข้อมูลที่เป็นข้อความ ดังนั้นในการศึกษานี้จึงเลือกใช้เคอร์เนลแบบ Linear พารามิเตอร์ที่สำคัญของ SVM ที่จะช่วยให้ได้โมเดลที่มีประสิทธิภาพได้แก่

เคอร์เนล RBF เป็นตัวเลือกที่นิยมสำหรับหลายๆ ปัญหาการจำแนกประเภทเนื่องจากความสามารถในการจับความสัมพันธ์ไม่เชิงเส้นในข้อมูล อย่างไรก็ตาม เคอร์เนล Linear อาจเป็นตัวเลือกที่ดีสำหรับชุดข้อมูลขนาดเล็กที่ความซับซ้อนไม่สูงมาก

พารามิเตอร์ C (Regularization parameter) การเริ่มต้นด้วยค่า C ที่ต่ำ (เช่น, 0.01, 0.1) สามารถช่วยให้เข้าใจว่าโมเดลที่ได้มีการ overfit หรือ underfit หรือไม่ เมื่อ C ต่ำ โมเดลจะ Penalize ต่อการจำแนกที่ผิดพลาดน้อยลง ทำให้โมเดลมีความเรียบง่ายมากขึ้นและมีโอกาส underfit มากกว่า อย่างไรก็ตามการเพิ่มค่า C ควรทำอย่างค่อยเป็นค่อยไป (เช่น, 1, 10, 100) และสังเกตการเปลี่ยนแปลงของประสิทธิภาพโมเดล ค่า C ที่สูงขึ้นจะให้ความสำคัญกับการจำแนกข้อมูลฝึกอบรมอย่างถูกต้องมากขึ้น แต่ก็อาจนำไปสู่การ overfitting โดยทั่วไป ค่าเริ่มต้นที่เป็นที่นิยมสำหรับค่า C ที่มักถูกใช้เป็นจุดเริ่มต้นในการทดลองคือ 1.0 เนื่องจากให้การ trade-off ระหว่างความแม่นยำของการจำแนกและความเรียบง่ายของโมเดลที่สมดุล

ค่า Gamma ในโมเดล SVM มีบทบาทสำคัญในการกำหนดขอบเขตของการตัดสินใจ โดยค่า gamma ที่สูงจะทำให้ “ขอบเขต” ของการตัดสินใจมีความแคบ ซึ่งอาจนำไปสู่การ overfitting ขณะที่ค่า Gamma ที่ต่ำจะทำให้ “ขอบเขต” ของการตัดสินใจกว้างขึ้น ซึ่งอาจนำไปสู่การ underfitting ค่าเริ่มต้นที่บางครั้งถูกใช้คือ 1/(จำนวนคุณลักษณะในข้อมูล) ซึ่งเป็นจุดเริ่มต้นที่ดีสำหรับการทดลอง

ตารางที่ 3.17 พารามิเตอร์หลักในโมเดล SVM ที่ต้องการพิจารณาสำหรับการสร้างโมเดลเพื่อการวิเคราะห์ความรู้สึกแบบ 2 กลุ่มด้วยข้อมูลขนาดเล็ก

พารามิเตอร์	ความสำคัญ	ค่าโดยทั่วไปที่เหมาะสม	ค่าที่ใช้
Kernel	เคอร์เนลทำหน้าที่เปลี่ยนพื้นที่คุณลักษณะ (feature space) เดิมไปสู่พื้นที่คุณลักษณะใหม่ที่การจำแนกประเภทอาจทำได้ง่ายขึ้นหรือมีประสิทธิภาพดีกว่า เคอร์เนลช่วยให้ SVM สามารถแยกข้อมูลที่ไม่สามารถแยกได้โดยเส้นตรงในพื้นที่คุณลักษณะเดิมได้ ซึ่งเป็นการขยายความสามารถของ SVM อย่างมาก	Linear หรือ RBF	RBF
C (Regularization parameter)	เป็นค่าที่มีบทบาทสำคัญในการควบคุม trade-off ระหว่างการสร้างแบบจำลองที่เรียบง่าย (และทั่วไปได้ดีกับข้อมูลที่ไม่เคยเห็นมาก่อน) และการให้ความสำคัญกับการจำแนกข้อมูลฝึกอบรมอย่างถูกต้องมากที่สุด	-	1.0

Gamma (γ)	เป็นค่าที่มีบทบาทสำคัญในการกำหนดขอบเขตของการตัดสินใจ	1/(จำนวนคุณลักษณะในข้อมูล)	1/108 = 0.0092593

3.6 การวัดประสิทธิภาพของตัวจัดกลุ่มเอกสาร (Model Evaluation)

การวัดประสิทธิภาพของตัวจัดกลุ่มเอกสาร ประเมินการวิเคราะห์ความรู้สึกในแบบ 2 ขั้ว คือ ขั้วบวก (Positive) และขั้วลบ (Negative) ประเมินโมเดลเพื่อใช้ในการจัดกลุ่มเอกสารก่อนการนำไปใช้งานจริงที่โดยทั่วไป จะใช้เทคนิคมาตรฐานที่นิยมใช้กันอย่างแพร่หลาย ที่เรียกว่าการวัด ค่าความถูกต้อง (Accuracy) ค่าความระลึก $(\boldsymbol{Recall})$ การวัดค่าความแม่นยำ $(\boldsymbol{Precision})$ และการวัดค่า $\boldsymbol{F\text{-}measure}$ สมมุติว่าข้อมูลชุดทดสอบมีเอกสารทั้งสิ้น 200 เอกสาร โดยมีผลการทำนายกลุ่มของข้อมูลดังที่แสดงในรูปที่ 3.7

		Predicted results	
		Yes	no
Actual results	yes	$\boldsymbol{tp = 120}$	$\boldsymbol{fn = 10}$
	no	$\boldsymbol{fp = 20}$	$\boldsymbol{tn = 50}$

รูปที่ 3.7 ตัวอย่างผลการทำนายกลุ่มของข้อมูลชุดทดสอบที่แสดงด้วย Confusion Matrix

จากตัวอย่างผลการทำนายกลุ่มของข้อมูลชุดทดสอบที่แสดงด้วย Confusion Matrix จะสามารถยกตัวอย่างการคำนวณได้ดังต่อไปนี้

3.6.1 การวัดค่าความถูกต้อง $(\boldsymbol{Accuracy})$

สมมุติให้ ชุดทดสอบมีเอกสารทั้งหมดจำนวน 200 เอกสาร จากตาราง Confusion Matrix ที่แสดงดังรูปที่ 3.8 สามารถคำนวณได้จากสมการที่

$$Acc = \frac{TP + TN}{TP + FP + TN + FN}$$

$$Acc = \frac{120 + 50}{120 + 20 + 50 + 10}$$

$$Acc = 0.85$$

จากตัวอย่างการคำนวณจะพบว่า ค่าความถูกต้องอยู่ที่ 0.85 หมายความว่าประสิทธิภาพโดยรวมของโมเดลเพื่อการจำแนกเอกสารทำนายชุดทดสอบได้ถูกต้องอยู่ที่ร้อยละ 85 ของข้อมูลชุดทดสอบทั้งหมด

3.6.2 การวัดค่าความระลึก $(Recall)$

สมมุติให้ แต่ละคลาสมีเอกสารจำนวน 100 เอกสาร ซึ่งรวมทั้งสิ้น 200 เอกสาร และในการจำแนกเอกสารอัตโนมัติทำนายได้ถูกต้องตามความจริง (tp) และทำนายผิด (fn) จะสามารถคำนวณได้จากสมการที่ 5 ดังต่อไปนี้

$$Recall = \frac{tp}{tp + fn}$$

$$Recall = \frac{120}{120 + 10}$$

$$Recall = 0.9230$$

จากตัวอย่างการคำนวณจะพบว่า ค่าความระลึกอยู่ที่ 0.9230 หมายความว่า เอกสารที่ควรอยู่ในคลาส yes จริงๆ นั้น โมเดลเพื่อการจำแนกเอกสารสามารถทำนายว่าอยู่ในคลาส yes ได้ถูกต้องและตรงกับความเป็นจริงอยู่ที่ร้อยละ 92.30 ซึ่งแสดงว่าโมเดลในการจำแนกเอกสารนั้นมีประสิทธิภาพ

3.6.3 การวัดค่าความแม่นยำ $(Precision)$

สมมุติให้ แต่ละคลาสมีเอกสารจำนวน 100 เอกสาร ซึ่งรวมทั้งสิ้น 200 เอกสาร และในการจำแนกเอกสารอัตโนมัติทำนายได้ถูกต้องตามความจริง (tp) และทำนายผิด (fp) จะสามารถคำนวณได้จากสมการที่ 6 ดังต่อไปนี้

$$Precision = \frac{tp}{tp + fp}$$

$$Precision = \frac{120}{120 + 20}$$

$$Precision = 0.8571$$

จากตัวอย่างการคำนวณจะพบว่า ค่าความแม่นยำอยู่ที่ 0.8571 หมายความว่า โมเดลจำแนกเอกสารสามารถทำนายกลุ่มเป็นคลาส yes มีความถูกต้องอยู่ที่ร้อยละ 85.71 แสดงว่าโมเดลดังกล่าวให้ความแม่นยำในการจำแนกข้อมูลในคลาส yes สูง

3.6.4 การวัดค่า $F\text{-}measure$

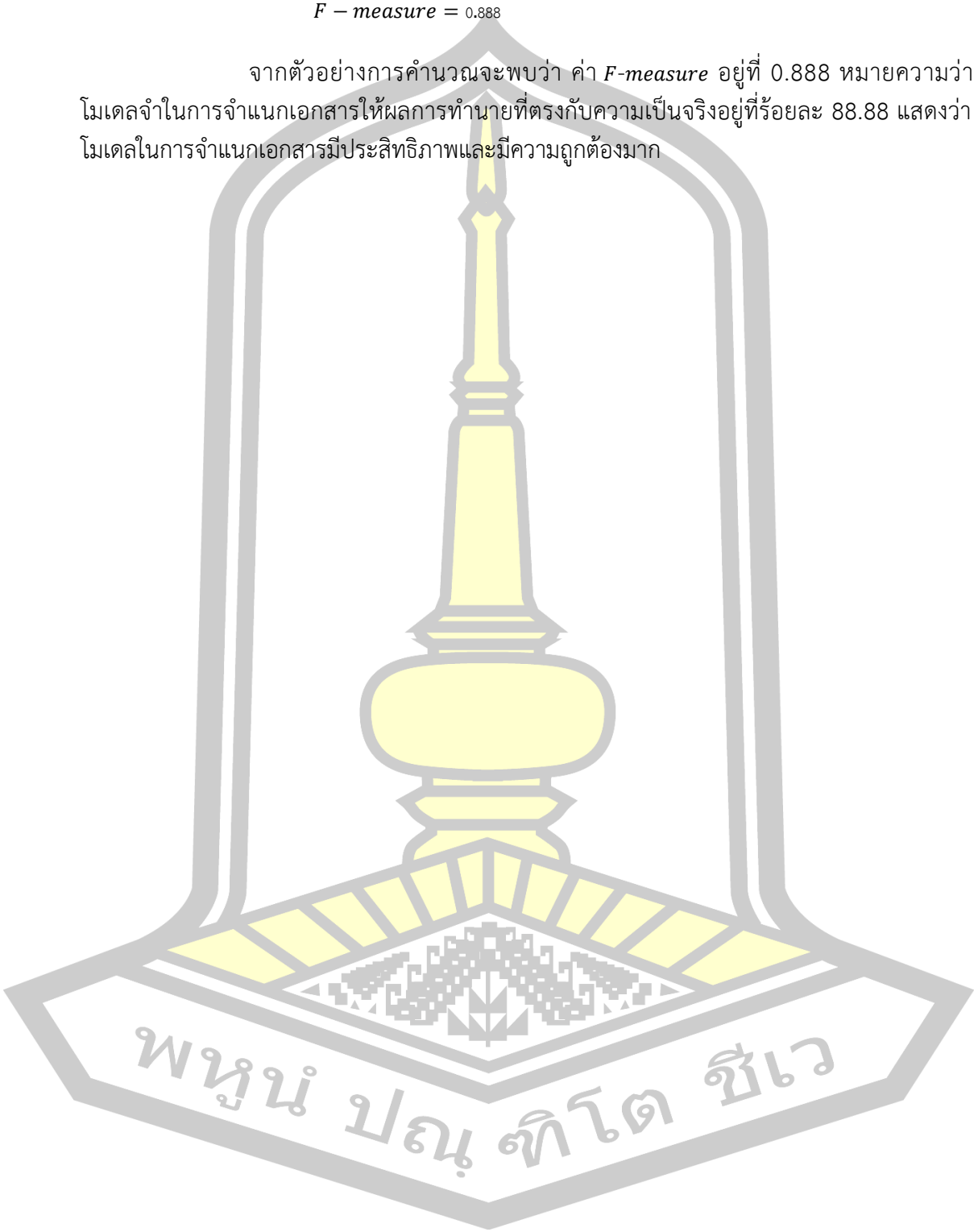
ค่า $F\text{-}measure$ คือ ผลเฉลี่ยระหว่างค่าความแม่นยำและค่าความระลึกสามารถคำนวณได้จาก ดังต่อไปนี้

$$F - measure = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

$$F - measure = 2 \times \frac{0.8571 \times 0.9230}{0.8571 + 0.9230}$$

$$F - measure = 0.888$$

จากตัวอย่างการคำนวณจะพบว่า ค่า $F\text{-}measure$ อยู่ที่ 0.888 หมายความว่าโมเดลจำในการจำแนกเอกสารให้ผลการทำนายที่ตรงกับความเป็นจริงอยู่ที่ร้อยละ 88.88 แสดงว่าโมเดลในการจำแนกเอกสารมีประสิทธิภาพและมีความถูกต้องมาก



บทที่ 4
ผลการทดลอง

ในบทนี้จะแสดงผลการทดสอบการจำแนกความรู้สึกระดับประโยคสำหรับความคิดเห็นของนักเรียนมัธยมศึกษาตอนต้น

4.1 ข้อมูลที่ใช้ในการทดสอบ

สำหรับข้อมูลความคิดเห็นในชุดที่ 3 จะถูกใช้เป็นชุดข้อมูลทดสอบ เป็นความคิดเห็นของนักเรียนทั้งสิ้น 200 ความคิดเห็น โดย “กลุ่มคำ” ที่ไม่แสดงหรือเกี่ยวข้องกับ Aspect เช่น คำเชื่อม คำสันธาน จะถูกตัดออกไปในขั้นตอนที่ผู้เชี่ยวชาญคัดแยกประโยคและระบุขั้วความรู้สึกของประโยค เช่น ประโยค “*คุณครูน่ารัก เนื้อหาสนุก แต่ห้องเรียนร้อน*” คำว่า “*แต่*” จะถูกตัดออกไป

ตารางที่ 4.1 ตัวอย่างข้อมูลชุดทดสอบ

ID	ความคิดเห็นจากนักเรียน	Aspect	ประโยคของแต่ละ Aspect	ขั้วความรู้สึก
1	คุณครูน่ารักและสอนดีมาก เนื้อหาในบทเรียนค่อนข้างยาก ห้องเรียนร้อน	ครู	คุณครูน่ารักและสอนดีมาก	Positive
		บทเรียน	เนื้อหาในบทเรียนค่อนข้างยาก	Negative
		สภาพแวดล้อม	ห้องเรียนร้อน	Negative
2	คุณครูน่ารัก เนื้อหาสนุก แต่ห้องเรียนร้อน	ครู	คุณครูน่ารัก	Positive
		บทเรียน	เนื้อหาสนุก	Positive
		สภาพแวดล้อม	ห้องเรียนร้อน	Negative
3	ครูดุมาก เนื้อหายาก	ครู	ครูดุมาก	Negative
		บทเรียน	เนื้อหายาก	Negative
		สภาพแวดล้อม	-	-

ตารางที่ 4.2 จำนวนประโยคในการทดสอบ

Aspect	จำนวนประโยค
ครู	200
บทเรียน	103
สภาพแวดล้อม	98

จากตารางที่ 4.2 สามารถนำมาสร้างตาราง confusion matrix ในแต่ละ Aspect ได้ดังตารางที่ 4.3

ตารางที่ 4.3 ตาราง Confusion Matrix สำหรับ Aspect ครู

	Predicted Positive	Predicted Negative
Actual Positive	TN (87)	FP (13)
Actual Negative	FN (15)	TP (85)

การวัดค่าความถูกต้อง $(Accuracy)$

$$Acc = \frac{TP + TN}{TP + FP + TN + FN}$$

$$Acc = \frac{87 + 85}{87 + 15 + 13 + 85}$$

$$Acc = 0.86$$

ดังนั้นค่า Accuracy คือ 0.86 หรือ 86%

การวัดค่าความระลึก $(Recall)$

$$Recall = \frac{tp}{tp + fn}$$

$$Recall = \frac{85}{85 + 15}$$

$$Recall = 0.85$$

ดังนั้น ค่า Recall คือ 0.85 หรือ 85%

การวัดค่าความแม่นยำ $(Precision)$

$$Precision = \frac{tp}{tp + fp}$$

$$Precision = \frac{85}{85 + 13}$$

$$Precision = 0.8673$$

ดังนั้นค่า Precision คือ 0.8673 หรือ 86.73%

การวัดค่า *F-measure*

$$F - measure = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

$$F - measure = 2 \times \frac{0.8673 \times 0.85}{0.8673 + 0.85}$$

$$F - measure = 0.8586$$

ดังนั้นค่า F-measure (F1 score) คือ 0.8586 หรือ 85.86%

ตารางที่ 4.4 ตาราง Confusion Matrix สำหรับ Aspect บทเรียน

	Predicted Positive	Predicted Negative
Actual Positive	TN (44)	FP (8)
Actual Negative	FN (10)	TP (41)

การวัดค่าความถูกต้อง $(Accuracy)$

$$Acc = \frac{TP + TN}{TP + FP + TN + FN}$$

$$Acc = \frac{44 + 41}{44 + 10 + 8 + 41}$$

$$Acc = 0.825$$

ดังนั้นค่า Accuracy คือ 0.825 หรือ 82.5%

การวัดค่าความระลึก $(Recall)$

$$Recall = \frac{tp}{tp + fn}$$

$$Recall = \frac{41}{41 + 10}$$

$$Recall = 0.804$$

ดังนั้น ค่า Recall คือ 0.804 หรือ 80.4%

การวัดค่าความแม่นยำ $(Precision)$

$$Precision = \frac{tp}{tp + fp}$$

$$Precision = \frac{44}{44 + 8}$$

$$Precision = 0.837$$

ดังนั้นค่า Precision คือ 0.837 หรือ 83.7%

การวัดค่า $F\text{-}measure$

$$F - measure = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

$$F - measure = 2 \times \frac{0.837 \times 0.804}{0.837 + 0.804}$$

$$F - measure = 0.818$$

ดังนั้นค่า F1-measure คือ 0.818 หรือ 81.8%

ตารางที่ 4.5 ตาราง Confusion Matrix สำหรับ Aspect สภาพแวดล้อม

	Predicted Positive	Predicted Negative
Actual Positive	TN (44)	FP (8)
Actual Negative	FN (11)	TP (35)

การวัดค่าความถูกต้อง $(Accuracy)$

$$Acc = \frac{TP + TN}{TP + FP + TN + FN}$$

$$Acc = \frac{44 + 35}{44 + 11 + 8 + 35}$$

$$Acc = 0.8061$$

ดังนั้นค่า Accuracy คือ 0.8061 หรือ 80.61%

การวัดค่าความระลึก $(Recall)$

$$Recall = \frac{tp}{tp + fn}$$

$$Recall = \frac{44}{44 + 8}$$

$$Recall = 0.8462$$

ดังนั้น ค่า Recall คือ 0.8462 หรือ 84.62%

การวัดค่าความแม่นยำ $(Precision)$

$$Precision = \frac{tp}{tp + fp}$$

$$Precision = \frac{44}{44 + 11}$$

$$Precision = 0.8$$

ดังนั้นค่า Precision คือ 0.8 หรือ 80%

การวัดค่า $F\text{-}measure$

$$F - measure = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

$$F - measure = 2 \times \frac{0.8 \times 0.8462}{0.8 + 0.8462}$$

$$F - measure = 0.8214$$

ดังนั้นค่า F1-measure คือ 0.8214หรือ 82.14%

4.2 ผลการประเมินการจัดกลุ่มประโยคที่สอดคล้องกับ Aspect ด้วย Similarity Analysis

จากคลังคำของแต่ละ Aspect ที่ได้จากชุดข้อมูลที่ 1 จะถูกใช้เป็น คำขอ (Query) เพื่อใช้ในการวิเคราะห์เพื่อจัดกลุ่มประโยคที่สอดคล้องกับ Aspect แต่ละตัวมากที่สุด ซึ่งภายหลังจากการประเมินด้วย Recall, Precision, F1 และ Accuracy ได้ผลการทดลองดังนี้

ตารางที่ 4.6 ผลการจัดกลุ่มประโยคที่สอดคล้องกับ Aspect ต่างๆ ด้วย Similarity Analysis

Aspect	Recall	Precision	F1	Accuracy
ครู	0.85	0.87	0.86	0.86
บทเรียน	0.80	0.83	0.82	0.83
สภาพแวดล้อม	0.85	0.80	0.82	0.81

ตารางที่ 4.6 แสดงให้เห็นว่าคลังคำของแต่ละ Aspect ที่ประสิทธิภาพในการจำแนกประโยคที่สอดคคล้องกับแต่ละ Aspect ได้อย่างน่าพอใจ อย่างไรก็ตามความผิดพลาดที่เกิดขึ้นอาจจะเนื่องมากจากบางประโยคยากต่อการพิจารณาเพราะเป็นประโยคที่สอดคล้องกับ Aspect มากกว่า 1 Aspect เช่น “*เวลาครูสอนอยู่ในห้องเรียนที่อากาศอบอ้าวจะดุมาก*” หรือ บางประโยคเป็นประโยคที่ชัดเจนเนื่องจากการเขียนของนักเรียนบางคนจะเป็นเว้นวรรคไม่เป็นประโยค เช่น “ครู ก็ดี” เนื่องจากว่า ในการศึกษานี้ใช้ white space ในการแยกประโยค ดังนั้นก็จะตัดประโยคได้เป็น “ครู” และ “ก็ดี” ซึ่งหากต้องการให้เกิดความถูกต้องมากขึ้นอาจจะต้องพิจารณาโครงสร้างและขอบเขตของประโยค

4.3 ผลการประเมินโมเดลการจำแนกความรู้สึกสำหรับความคิดเห็นของนักเรียนของแต่ละอัลกอริทึมด้วย 10-fold cross validation

4.3.1 K-nearest Neighbor

ผลการทดลองเมื่อ K=3, K= 5 และ K=7 สามารถแสดงได้ดังตารางที่ 4.7

ตารางที่ 4.7 ผลการประเมินโมเดลการจำแนกความรู้สึกสำหรับความคิดเห็นของนักเรียนด้วย KNN โดย K=3, K= 5 และ K=7

K value	Iteration	Aspect Analyzer (tf-idf)				Aspect Analyzer (tf-igm)			
		Acc.	Recall	Prec.	F1	Acc.	Recall	Prec.	F1
K=3	1	0.5765	0.4565	0.7805	0.5761	0.6526	0.5685	0.7658	0.6526
	2	0.5490	0.4286	0.7625	0.5487	0.6775	0.6201	0.7465	0.6775
	3	0.6045	0.5011	0.7600	0.6040	0.6598	0.6232	0.7011	0.6598
	4	0.6210	0.5234	0.7600	0.6200	0.6566	0.6587	0.6545	0.6566
	5	0.6949	0.6501	0.7465	0.6949	0.6982	0.6857	0.7111	0.6982
	6	0.6611	0.6698	0.6505	0.6600	0.7085	0.6953	0.7222	0.7085
	7	0.6938	0.6812	0.7056	0.6932	0.7074	0.6778	0.7396	0.7074
	8	0.6635	0.6122	0.7236	0.6633	0.7093	0.6989	0.7201	0.7093
	9	0.6320	0.5423	0.7564	0.6317	0.6994	0.7001	0.6987	0.6994
	10	0.6260	0.5306	0.7628	0.6259	0.7038	0.7121	0.6956	0.7038

ตารางที่ 4.7 (ต่อ)

K value	Iteration	Aspect Analyzer (tf-idf)				Aspect Analyzer (tf-igm)			
		Acc.	Recall	Prec.	F1	Acc.	Recall	Prec.	F1
K=5	1	0.6778	0.4694	0.8846	0.6133	0.7667	0.7667	0.7798	0.7705
	2	0.6667	0.4286	0.9130	0.5833	0.8222	0.8222	0.8324	0.8219
	3	0.7333	0.5714	0.9032	0.7000	0.7667	0.7667	0.7846	0.7720
	4	0.7000	0.5510	0.8438	0.6667	0.7778	0.7778	0.7955	0.7841
	5	0.8111	0.7551	0.8850	0.8132	0.7556	0.7556	0.8018	0.7600
	6	0.7778	0.7143	0.5837	0.7778	0.7333	0.7333	0.7560	0.7313
	7	0.7444	0.5510	0.9643	0.7013	0.6778	0.6778	0.7396	0.6810
	8	0.7333	0.6122	0.8571	0.7143	0.7889	0.7889	0.8190	0.7902
	9	0.6667	0.4490	0.8800	0.5946	0.7778	0.7778	0.8156	0.7858
	10	0.6854	0.5306	0.8387	0.6500	0.8090	0.8090	0.8155	0.7950
K=7	1	0.5765	0.4565	0.7805	0.5761	0.6526	0.5685	0.7658	0.6526
	2	0.5490	0.4286	0.7625	0.5487	0.6775	0.6201	0.7465	0.6775
	3	0.6045	0.5011	0.7600	0.6040	0.6598	0.6232	0.7011	0.6598
	4	0.6210	0.5234	0.7600	0.6200	0.6566	0.6587	0.6545	0.6566
	5	0.6949	0.6501	0.7465	0.6949	0.6982	0.6857	0.7111	0.6982
	6	0.6611	0.6698	0.6505	0.6600	0.7085	0.6953	0.7222	0.7085
	7	0.6938	0.6812	0.7056	0.6932	0.7074	0.6778	0.7396	0.7074
	8	0.6635	0.6122	0.7236	0.6633	0.7093	0.6989	0.7201	0.7093
	9	0.6320	0.5423	0.7564	0.6317	0.6994	0.7001	0.6987	0.6994
	10	0.6260	0.5306	0.7628	0.6259	0.7038	0.7121	0.6956	0.7038

KNN ในชุดข้อมูลขนาดเล็กอาจให้ผลลัพธ์ที่แตกต่างกันเมื่อปรับค่า K ซึ่งเป็นพารามิเตอร์ที่กำหนดจำนวนเพื่อนบ้านที่ใกล้ที่สุดที่จะพิจารณาสำหรับการตัดสินใจจำแนกประเภท การที่ K=5 ให้ผลลัพธ์ที่ดีกว่า K=3 หรือ K=7 ในบางชุดข้อมูลขนาดเล็กอาจเกิดจากหลายปัจจัย:

การลดผลกระทบของ “Nosie”: ในชุดข้อมูลขนาดเล็กที่มีความแปรปรวนสูงหรือ “Nosie” การเลือกค่า K ที่เหมาะสมเป็นสิ่งสำคัญเพื่อหลีกเลี่ยงการถูกอิทธิพลจากข้อมูลที่เป็นผิดปกติ (outliers) หรือข้อมูลที่ไม่เกี่ยวข้อง การใช้ K=5 อาจช่วยให้ระบบมีความทนทานต่อ “Nosie” มากกว่า K=3 ซึ่งอาจทำให้ระบบไวต่อการถูกอิทธิพลจากข้อมูลเหล่านั้นได้ง่ายกว่า

การสมดุลระหว่างความแม่นยำและความทั่วไป: การเลือก K ที่เหมาะสมช่วยให้ระบบสามารถสมดุลระหว่างความแม่นยำ (สามารถจำแนกข้อมูลได้ดีในชุดฝึกหัด) กับความทั่วไป (สามารถจำแนกข้อมูลใหม่ที่ไม่เคยเห็นมาก่อนได้ดี) K=5 อาจเป็นค่าที่ช่วยให้ระบบมีความสมดุลนี้ได้ดีกว่า K=3 หรือ K=7 โดยเฉพาะอย่างยิ่งในชุดข้อมูลที่มีขนาดเล็กและความหลากหลายของข้อมูลจำกัด

การหลีกเลี่ยงการให้น้ำหนักมากเกินไปต่อข้อมูลจำนวนน้อย: การใช้ค่า K ที่มากเกินไปในชุดข้อมูลขนาดเล็กอาจทำให้ระบบให้น้ำหนักมากเกินไปต่อข้อมูลจำนวนน้อย ซึ่งอาจนำไปสู่การตัดสินใจที่ไม่แม่นยำ ในทางตรงกันข้าม ค่า K ที่น้อยเกินไปอาจทำให้ระบบไวต่อข้อมูลผิดปกติ การเลือก K=5 อาจช่วยลดปัญหาเหล่านี้ได้

การตัดสินใจที่ครอบคลุมมากขึ้น: การเลือก K=5 สามารถให้มุมมองที่ครอบคลุมมากขึ้นเกี่ยวกับข้อมูลที่ใกล้เคียง ช่วยให้การตัดสินใจจำแนกประเภทมีความน่าเชื่อถือมากขึ้นในบางสถานการณ์ โดยไม่ต้องพึ่งพาข้อมูลจำนวนน้อยจนเกินไปที่อาจไม่เป็นตัวแทนของประเภทที่แท้จริง

4.3.2 Multinomial Naïve Bayes

ผลการทดลองสามารถแสดงได้ดังตารางที่ 4.8

ตารางที่ 4.8 ผลการประเมินโมเดลการจำแนกความรู้สึกสำหรับความคิดเห็นของนักเรียนด้วย Multinomial Naïve Bayes

Algorithm	Iteration	Aspect Analyzer (tf-idf)				Aspect Analyzer (tf-igm)			
		Acc.	Recall	Prec.	F1	Acc.	Recall	Prec.	F1
MNB	1	0.7222	0.5306	0.9286	0.6753	0.6444	0.6444	0.7528	0.6475
	2	0.6444	0.5306	0.7429	0.6190	0.6222	0.6222	0.6587	0.6269
	3	0.6556	0.5102	0.7812	0.6173	0.6333	0.6333	0.7299	0.6467
	4	0.7111	0.6327	0.7949	0.7045	0.6556	0.6556	0.7493	0.6598
	5	0.7000	0.5102	0.8929	0.6494	0.5778	0.5778	0.6820	0.5884
	6	0.7222	0.6531	0.8000	0.7191	0.7111	0.7111	0.8100	0.7258
	7	0.7444	0.6918	0.9362	0.7260	0.6333	0.6333	0.7090	0.6490
	8	0.7111	0.6122	0.8108	0.6977	0.5444	0.5444	0.6911	0.5602
	9	0.6778	0.5714	0.7778	0.6588	0.6556	0.6556	0.7732	0.6704
	10	0.7303	0.6327	0.8378	0.7209	0.6517	0.6517	0.7610	0.6665

4.3.3 Support Vector Machine

ผลการทดลองสามารถแสดงได้ดังตารางที่ 4.9

ตารางที่ 4.9 ผลการประเมินโมเดลการจำแนกความรู้สึกสำหรับความคิดเห็นของนักเรียนด้วย Support Vector Machine

Algorithm	Iteration	Aspect Analyzer (tf-idf)				Aspect Analyzer (tf-igm)			
		Acc.	Recall	Prec.	F1	Acc.	Recall	Prec.	F1
SVM	1	0.8444	0.8163	0.8889	0.8511	0.8111	0.8111	0.8098	0.8103
	2	0.8000	0.7143	0.8974	0.7955	0.8333	0.8333	0.8556	0.8382
	3	0.8222	0.8367	0.8367	0.8367	0.8111	0.8111	0.8069	0.8073
	4	0.8222	0.7347	0.9231	0.8182	0.8333	0.8333	0.8539	0.8378
	5	0.8667	0.8163	0.9302	0.8696	0.7889	0.7889	0.8042	0.7927
	6	0.8111	0.8163	0.8333	0.8247	0.8000	0.8000	0.7976	0.7959

ตารางที่ 4.9 (ต่อ)

	7	0.8889	0.8987	0.9349	0.8958	0.7222	0.7222	0.7503	0.7276
	8	0.8444	0.8980	0.8302	0.8627	0.8000	0.8000	0.7982	0.7920
	9	0.8111	0.7959	0.8478	0.8211	0.8222	0.8222	0.8337	0.8261
	10	0.7865	0.7347	0.8571	0.7912	0.7528	0.7528	0.7496	0.7484

4.3.4 Random Forest

ผลการทดลองสามารถแสดงได้ดังตารางที่ 4.10

ตารางที่ 4.10 ผลการประเมินโมเดลการจำแนกความรู้สึกสำหรับความคิดเห็นของนักเรียนด้วย Random Forest

Algorithm	Iteration	Aspect Analyzer (tf-idf)				Aspect Analyzer (tf-igm)			
		Acc.	Recall	Prec.	F1	Acc.	Recall	Prec.	F1
RF	1	0.9000	0.8871	0.9545	0.9032	0.8000	0.8000	0.7965	0.7960
	2	0.8111	0.6735	0.9706	0.7952	0.8556	0.8556	0.8606	0.8466
	3	0.8333	0.8367	0.8454	0.8454	0.8222	0.8222	0.8162	0.8135
	4	0.8000	0.8163	0.8163	0.8163	0.8222	0.8222	0.8183	0.8167
	5	0.8556	0.7959	0.8571	0.8571	0.8333	0.8333	0.8377	0.8233
	6	0.8556	0.8776	0.8687	0.8687	0.8444	0.8444	0.8456	0.8350
	7	0.7667	0.7511	0.7789	0.7789	0.7667	0.7667	0.7790	0.7672
	8	0.8889	0.8776	0.8985	0.8958	0.8444	0.8444	0.8453	0.8446
	9	0.8222	0.8163	0.8333	0.8333	0.8444	0.8444	0.8439	0.8419
	10	0.7978	0.7755	0.8085	0.8085	0.8202	0.8202	0.8354	0.8056

4.4 ผลการประเมินโมเดลการจำแนกความรู้สึกระดับประโยคสำหรับความคิดเห็นของนักเรียน

ในการประเมินผลการจำแนกความรู้สึกระดับประโยคสำหรับความคิดเห็นของนักเรียนด้วยโมเดลเพื่อการวิเคราะห์ความรู้สึก จะประเมินด้วย Recall, Precision, F1 และ Accuracy โดยแต่ละอัลกอริทึมให้ผลดังแสดงในตารางที่ 4.11

ตารางที่ 4.11 ผลประเมินผลการจำแนกความรู้สึกระดับประโยคสำหรับความคิดเห็นของนักเรียนด้วยโมเดลเพื่อการวิเคราะห์ความรู้สึก

อัลกอริทึม	Term Weighting	Average Accuracy	Average Recall	Average Precision	Average F1
KNN	tf-idf	0.7564	0.6959	0.8339	0.7565
	tf-igm	0.7374	0.5633	0.9286	0.6956
MNB	tf-idf	0.7119	0.5796	0.8436	0.6859
	tf-igm	0.7242	0.6061	0.8470	0.7043

SVM with RBF	tf-idf	0.8431	0.8633	0.8590	0.8576
	tf-igm	0.8231	0.8082	0.8611	0.8328
RF	**tf-idf**	**0.8503**	**0.8704**	**0.8906**	**0.8794**
	tf-igm	0.8423	0.8563	0.8900	0.8720

จากตารางที่ 4.11 จะเห็นว่าการจำแนกความรู้สึกระดับประโยคสำหรับความคิดเห็นของนักเรียนด้วยโมเดลเพื่อการวิเคราะห์ความรู้สึกที่สร้างจาก RF ให้ประสิทธิภาพที่ดีที่สุดในการจำแนกความรู้สึกระดับประโยคสำหรับความคิดเห็น เนื่องจาก RF เป็นหนึ่งในอัลกอริทึมการเรียนรู้ของเครื่องที่มีประสิทธิภาพสูงสำหรับงานการจำแนกประเภทและการทำนายค่าต่อเนื่อง และสามารถทำงานได้ดีกับชุดข้อมูลขนาดเล็ก โดยเฉพาะอย่างยิ่งในงานการวิเคราะห์ความรู้สึกจากข้อความ การที่ RF ให้ประสิทธิภาพที่ดีอาจมีสาเหตุมาจากหลายปัจจัยเมื่อเปรียบเทียบกับ KNN, MNB และ SVM:

1. การลด Variance และ Bias - Random Forest สร้างขึ้นจากต้นไม้การตัดสินใจหลายต้น (Decision Trees) ซึ่งทำงานร่วมกันเพื่อลด variance และเพิ่มความแม่นยำของโมเดลโดยรวม โดยแต่ละต้นไม้จะถูกฝึกอบรมด้วยชุดข้อมูลย่อยที่สุ่มขึ้น ทำให้ลดความเสี่ยงของการ overfitting ซึ่งเป็นปัญหาสำคัญในชุดข้อมูลขนาดเล็ก

2. การจัดการคุณลักษณะที่ไม่เชิงเส้น - Random Forest สามารถจัดการกับความสัมพันธ์ที่ไม่เชิงเส้นระหว่างคุณลักษณะและคลาสเป้าหมายได้ดี ทำให้มีประสิทธิภาพสูงในการจำแนกความรู้สึกจากข้อความที่อาจมีความซับซ้อนและไม่เชิงเส้น

3. การจัดการกับข้อมูลที่หายไป - Random Forest สามารถจัดการกับข้อมูลที่หายไปได้โดยอัตโนมัติ ซึ่งช่วยให้สามารถใช้งานได้กับชุดข้อมูลที่ไม่สมบูรณ์โดยไม่ต้องมีการจัดการข้อมูลเพิ่มเติม

4. ความสามารถในการทำนายที่แม่นยำ - อัลกอริทึมอื่นๆ เช่น KNN อาจได้รับผลกระทบจาก “คำสั่งคำนวณระยะห่าง” ในข้อมูลที่มีมิติสูง ส่วน Multinomial Naïve Bayes อาจต้องเผชิญกับปัญหาความน่าจะเป็นที่เป็นอิสระระหว่างคุณลักษณะ (ซึ่งอาจไม่เป็นจริงในข้อความ) และ SVM อาจต้องการการปรับแต่งพารามิเตอร์อย่างละเอียดเพื่อป้องกันการ overfitting

5. ความสามารถในการจัดการกับคุณลักษณะจำนวนมาก - Random Forest มีประสิทธิภาพสูงในการจัดการกับชุดข้อมูลที่มีคุณลักษณะจำนวนมาก (เช่น คำศัพท์ใน TF-IDF) โดยไม่จำเป็นต้องลดมิติของข้อมูล

ดังนั้น ความสามารถของ RF ในการลดความเสี่ยงของการ overfitting จัดการกับความสัมพันธ์ที่ไม่เชิงเส้น และการจัดการกับคุณลักษณะจำนวนมากทำให้เป็นทางเลือกที่ดีสำหรับการวิเคราะห์ความรู้สึกจากข้อความ โดยเฉพาะอย่างยิ่งกับชุดข้อมูลขนาดเล็กที่อาจมีความซับซ้อนและต้องการโมเดลที่ทนทานและมีประสิทธิภาพสูง

อย่างไรก็ตามจะเห็นว่า RF ที่ใช้ร่วมกับการให้น้ำหนักคำแบบ tf-idf ให้ประสิทธิภาพที่ดีกว่า RF ดีกว่า ที่ใช้ร่วมกับการให้น้ำหนักคำแบบ tf-igm นั่นคือ

1. ความสามารถในการระบุคำที่สำคัญ - TF-IDF จะช่วยเน้นคำที่ปรากฏไม่บ่อยในเอกสารทั้งหมดแต่มีความสำคัญในเอกสารบางเอกสาร ซึ่งเป็นคุณสมบัติสำคัญในการจำแนกความรู้สึก เนื่องจากคำที่มีความสำคัญอาจมีบทบาทสำคัญในการกำหนดความรู้สึกหรือทัศนคติ ในขณะที่ TF-IGM โฟกัสที่การปรับน้ำหนักคำตามความสามารถในการจำแนกประเภทเอกสารตามหมวดหมู่ ซึ่งอาจไม่เฉพาะเจาะจงพอสำหรับการระบุความรู้สึกหรือทัศนคติที่แสดงในข้อความ

2. การแยกแยะข้อมูล - TF-IDF สามารถสร้างคุณสมบัติที่ช่วยให้โมเดลสามารถแยกแยะข้อความที่มีความรู้สึกหรือทัศนคติต่างกันได้ดีขึ้น เนื่องจากมันให้น้ำหนักกับคำที่มีความสำคัญในแง่ของการปรากฏไม่บ่อยแต่มีความสำคัญสูงในบางเอกสาร ในขณะที่ TF-IGM อาจไม่ให้น้ำหนักเพียงพอกับคำที่ปรากฏไม่บ่อยแต่มีความสำคัญในการแสดงความรู้สึกหรือทัศนคติในข้อความ

3. การปรับปรุงสำหรับความหลากหลายของเอกสาร - TF-IDF มีประสิทธิภาพสูงในการปรับปรุงคุณสมบัติให้เหมาะสมกับชุดข้อมูลที่มีเอกสารหลากหลาย ในประเด็นนี้จึงให้น้ำหนักกับคำที่ช่วยในการจำแนกเอกสารได้ดีกว่า TF-IGM

ในส่วนของ SVM ที่ทำงานร่วมกับเคอร์เนล RBF สามารถให้ประสิทธิภาพในการจำแนกความรู้สึกจากข้อความที่สร้างจากชุดข้อมูลขนาดเล็กได้ดีกว่า KNN และ MNB ด้วยเหตุผลหลายคือ

1. ความสามารถในการจัดการกับความสัมพันธ์ไม่เชิงเส้น - เคอร์เนล RBF ช่วยให้ SVM สามารถจับความสัมพันธ์ไม่เชิงเส้นในข้อมูลได้ดี เคอร์เนลนี้ทำการแปลงข้อมูลไปยังพื้นที่คุณลักษณะที่มีมิติสูงกว่า เพื่อทำให้ข้อมูลที่ไม่สามารถแยกด้วยเส้นตรงในพื้นที่เดิมสามารถแยกจากกันได้ในพื้นที่คุณลักษณะใหม่ ในขณะที่ KNN จะขึ้นอยู่กับการคำนวณระยะห่าง ซึ่งอาจไม่มีประสิทธิภาพในชุดข้อมูลที่มีความสัมพันธ์ไม่เชิงเส้น และมีประสิทธิภาพลดลงในชุดข้อมูลที่มีมิติสูง ส่วน Multinomial Naive Bayes มีสมมุติว่าคุณลักษณะเป็นอิสระต่อกัน ซึ่งอาจไม่เหมาะสมสำหรับข้อมูลที่มีความสัมพันธ์ระหว่างคำหรือคุณลักษณะ

2. การจัดการกับมิติของข้อมูล - SVM with RBF: สามารถจัดการได้ดีกับชุดข้อมูลที่มีมิติสูง ทำให้เหมาะกับการวิเคราะห์ข้อความที่อาจถูกแปลงเป็นเวกเตอร์คุณลักษณะขนาดใหญ่ผ่านวิธีการแปลง เช่น TF-IDF ในขณะที่ KNN และ Multinomial Naive Bayes อาจประสบปัญหาในการจัดการกับข้อมูลที่มีมิติสูง โดยเฉพาะ KNN ที่อาจต้องใช้เวลาและทรัพยากรคอมพิวเตอร์มากขึ้นในการคำนวณระยะห่าง

3. ความทนทานต่อ overfitting - SVM with RBF: ด้วยการปรับพารามิเตอร์เช่น C และ gamma อย่างเหมาะสม สามารถลดความเสี่ยงของการ overfitting ในชุดข้อมูลขนาดเล็กได้ ในขณะที่ KNN อาจเกิดการ overfitting ได้ง่าย โดยเฉพาะเมื่อเลือกค่า K ที่ไม่เหมาะสม ส่วน Multinomial Naive Bayes แม้ว่าจะมีความทนทานต่อการ overfitting ในบางสถานการณ์ แต่การสมมุติว่าคุณลักษณะเป็นอิสระต่อกันอาจไม่เหมาะสมกับข้อมูลบางชนิด

บทที่ 5
สรุปผล อภิปรายผล และข้อเสนอแนะ

5.1 สรุปผลการศึกษา

การวิจัยนี้เป็นการศึกษาการจำแนกความรู้สึกระดับประโยคจากความคิดเห็นของนักเรียนมัธยมศึกษาตอนต้น โดยมีวัตถุประสงค์หลักเพื่อพัฒนากระบวนการที่สามารถจำแนกความรู้สึกแบบ 2 ขั้ว คือ ขั้วบวกและขั้วลบ จากคำติชมหรือความคิดเห็นของนักเรียน งานวิจัยได้รวบรวมข้อมูลจากนักเรียน 600 คน จาก 3 โรงเรียนในจังหวัดร้อยเอ็ด ประกอบด้วยข้อมูล 3 ชุดจาก 10 วิชาในภาคเรียนที่ 1 และ 2 ปีการศึกษา 2564

ข้อมูลชุดที่ 1 จะถูกใช้สำหรับสร้างคลังคำที่เกี่ยวข้องในแต่ละด้าน (Aspect) ด้วย Skip gram ของ Word2Vec และคลังคำที่ได้นี้จะถูกใช้เป็น features ในการสร้างโมเดลเพื่อการจำแนกความรู้สึกระดับประโยคจากความคิดเห็น และใช้เป็นคำขอ (Query) ในการจำแนกประโยคในความคิดเห็นไปยังคลัสเตอร์ของแต่ละ Aspect โดยได้คำที่เกี่ยวข้องกับ Aspect ครู จำนวน 43 คำ คำที่เกี่ยวข้องกับ Aspect บทเรียน จำนวน 37 คำ และคำที่เกี่ยวข้องกับ Aspect สภาพแวดล้อม จำนวน 28 คำ

ข้ออมูลชุดที่ 2 จะถูกใช้สำหรับสร้างโมเดลเพื่อการจำแนกความรู้สึกระดับประโยคจากความคิดเห็นและทดสอบตัวจำแนกความรู้สึกแบบ 2 ขั้ว คือ ขั้วบวกและขั้วลบ และข้อมูลชุดที่ 2 จะถูกใช้สำหรับทดสอบโมเดลเพื่อการจำแนกความรู้สึกระดับประโยคจากความคิดเห็น สำหรับข้อมูลในชุดที่ 2 และ 3 จะมีผู้เชี่ยวชาญด้านภาษาเป็นผู้กำหนดขั้วความรู้สึกให้กับความคิดเห็น โดยเฉพาะในข้อมูลชุดที่ 3 ผู้เชี่ยวชาญจะระบุว่าประโยคที่อยู่ในแต่ละความคิดเห็นนั้น ประโยคใดอยู่ใน Aspect ใด และมีขั้วความรู้สึกของประโยคนั้นเป็นอย่างไร

ในการสร้างโมเดลเพื่อการจำแนกความรู้สึกแบบ 2 ขั้วในระดับประโยคจากความคิดเห็นนั้นจะเป็นการศึกษาเชิงเปรียบเทียบใน 4 อัลกอริทึมคือ Multinomial Naïve Bayes (MNB), k-nearest neighbors (K-NN), Support Vector Machines (SVM), และ Random Forest (RF) รวมถึงการใช้ K-Fold Cross Validation เพื่อประเมินและเปรียบเทียบประสิทธิภาพ

นอกจากการศึกษาเชิงเปรียบเทียบด้านอัลกอริทึม การเรียนรู้ของเครื่องในการสร้างโมเดลเพื่อการวิเคราะห์ความรู้สึกแล้ว งานวิจัยนี้ยังได้ศึกษาเชิงเปรียบเทียบเทคนิคในการให้น้ำหนักคำทั้งแบบไม่มีผู้สอน (ในที่นี้ใช้ tf-idf) และการให้น้ำหนักคำทั้งแบบมีผู้สอน (ในที่นี้ใช้ tf-igm)

ในขั้นตอนของการทดสอบด้วยข้อมูลชุดที่ 3 เราได้ใช้คลังคำที่ได้จากข้อมูลชุดที่ 1 มาเป็นคำขอ (Query) เพื่อวิเคราะห์ความคล้ายคลึง (Similarity Analysis) ในการคัดเลือกประโยคในความคิดเห็นไปยังคลัสเตอร์ของแต่ละ Aspect ที่สอดคล้องกับใจความในประโยค ภายหลังจากประเมินด้วยค่า Recall, Precision, F1 และ Accuracy พบว่าให้ผลลัพธ์ที่น่าพอใจ อย่างไรก็ตามความผิดพลาดที่เกิดขึ้นอาจจะเนื่องมากจากบางประโยคยากต่อการพิจารณาเพราะเป็นประโยคที่สอดคล้องกับ Aspect มากกว่า 1 Aspect เช่น “เวลาครูสอนอยู่ในห้องเรียนที่อากาศ

อบอ้าวจะดุมาก” หรือ บางประโยคเป็นประโยคที่ชัดเจนเนื่องจากการเขียนของนักเรียนบางคนจะเป็นเว้นวรรคไม่เป็นประโยค เช่น “ครู ก็ดี” เนื่องจากว่า ในการศึกษานี้ใช้ white space ในการแยกประโยค ดังนั้นก็จะตัดประโยคได้เป็น “ครู” และ “ก็ดี” ซึ่งหากต้องการให้เกิดความถูกต้องมากขึ้นอาจจะต้องพิจารณาโครงสร้างและขอบเขตของประโยค

ภายหลังจากการคัดแยกประโยคต่างๆ ไปยัง Aspect ที่สอดคล้องกัน ก็จะใช้โมเดลเพื่อการจำแนกความรู้สึกระดับประโยคในการจำแนกความรู้สึกของประโยคในแต่ละ Aspect โดยทำการประเมินด้วยค่า Accuracy, Recall, Precision และ F1 พบว่า Random Forest ที่ใช้งานร่วมกับ tf-idf ให้ประสิทธิภาพในการจำแนกความรู้สึกแบบ 2 ขั้วในระดับประโยคจากความคิดเห็นได้ดีที่สุด เนื่องจาก Random Forest มีความสามารถของ Random Forest ในการลดความเสี่ยงของการ overfitting, การจัดการกับความสัมพันธ์ที่ไม่เชิงเส้น, และการจัดการกับคุณลักษณะจำนวนมากทำให้เป็นทางเลือกที่ดีสำหรับการวิเคราะห์ความรู้สึกจากข้อความโดยเฉพาะอย่างยิ่งกับชุดข้อมูลขนาดเล็กที่อาจมีความซับซ้อนและต้องการโมเดลที่ทนทานและมีประสิทธิภาพสูง หากพิจารณาจากเทคนิคการให้น้ำหนักคำ พบว่า tf-idf ให้ประสิทธิภาพที่ดีกว่า tf-igm นั่นคือ tf-idf จะช่วยเน้นคำที่ปรากฏไม่บ่อยในเอกสารทั้งหมดแต่มีความสำคัญในเอกสารบางเอกสาร ซึ่งเป็นคุณสมบัติสำคัญในการจำแนกความรู้สึก เนื่องจากคำที่มีความสำคัญอาจมีบทบาทสำคัญในการกำหนดความรู้สึกหรือทัศนคติ ในขณะที่ tf-igm โฟกัสที่การปรับน้ำหนักคำตามความสามารถในการจำแนกประเภทเอกสารตามหมวดหมู่ ซึ่งอาจไม่เฉพาะเจาะจงพอสำหรับการระบุความรู้สึกหรือทัศนคติที่แสดงในข้อความ นอกจากนั้น tf-idf สามารถสร้างคุณสมบัติที่ช่วยให้โมเดลสามารถแยกแยะข้อความที่มีความรู้สึกหรือทัศนคติต่างกันได้ดีขึ้น เนื่องจากมันให้น้ำหนักกับคำที่มีความสำคัญในแง่ของการปรากฏไม่บ่อยแต่มีความสำคัญสูงในบางเอกสาร ในขณะที่ tf-igm อาจไม่ให้น้ำหนักเพียงพอกับคำที่ปรากฏไม่บ่อยแต่มีความสำคัญในการแสดงความรู้สึกหรือทัศนคติในข้อความ และเหตุผลสุดท้ายคือ tf-idf มีประสิทธิภาพสูงในการปรับปรุงคุณสมบัติให้เหมาะสมกับชุดข้อมูลที่มีเอกสารหลากหลาย ในประเด็นนี้จึงให้น้ำหนักกับคำที่ช่วยในการจำแนกเอกสารได้ดีกว่า tf-igm

5.2 อภิปรายผล

การวิจัยนี้มุ่งเน้นไปที่การพัฒนากระบวนการจำแนกความรู้สึกระดับประโยคจากความคิดเห็นของนักเรียนมัธยมศึกษาตอนต้นในสองขั้วคือบวกและลบ การวิจัยใช้ชุดข้อมูลจากนักเรียน 600 คนจาก 3 โรงเรียนในจังหวัดร้อยเอ็ด เก็บรวบรวมข้อมูลความคิดเห็นจาก 10 วิชาในการวิเคราะห์คำติชมและความคิดเห็นที่ถูกแยกออกเป็นสามด้าน ได้แก่ ครูผู้สอน, บทเรียน, และสภาพแวดล้อมการเรียนรู้

การวิเคราะห์และจำแนกความรู้สึกจากความคิดเห็นถูกทำผ่านกระบวนการหลักที่ประกอบด้วยการสร้างคลังคำด้วย Skip gram ของ Word2Vec และการใช้โมเดลจำแนกความรู้สึกที่พัฒนาจากอัลกอริทึมต่างๆ อาทิ Multinomial Naïve Bayes, k-nearest

neighbors, Support Vector Machines, และ Random Forest นอกจากนี้ยังได้ทดลองเปรียบเทียบการให้น้ำหนักคำด้วย tf-idf และ tf-igm

ผลลัพธ์จากการวิจัยนี้ชี้ให้เห็นว่าการใช้ Random Forest ร่วมกับ tf-idf ให้ประสิทธิภาพในการจำแนกความรู้สึกได้ดีที่สุด เนื่องจาก Random Forest มีความสามารถในการลดความเสี่ยงของการ overfitting, จัดการกับความสัมพันธ์ที่ไม่เชิงเส้น, และจัดการกับคุณลักษณะจำนวนมากได้ดี ทำให้เหมาะกับการวิเคราะห์ความรู้สึกจากข้อความ โดยเฉพาะในชุดข้อมูลขนาดเล็กที่อาจมีความซับซ้อน

การให้น้ำหนักคำด้วย tf-idf ช่วยเน้นคำที่ปรากฏไม่บ่อยในเอกสารทั้งหมดแต่มีความสำคัญในเอกสารบางเอกสาร ซึ่งเป็นคุณสมบัติสำคัญในการจำแนกความรู้สึก เนื่องจากคำที่มีความสำคัญอาจมีบทบาทสำคัญในการกำหนดความรู้สึกหรือทัศนคติ ในทางตรงกันข้าม tf-igm โฟกัสที่การปรับน้ำหนักคำตามความสามารถในการจำแนกประเภทเอกสารตามหมวดหมู่ ซึ่งอาจไม่เฉพาะเจาะจงพอสำหรับการระบุความรู้สึกหรือทัศนคติที่แสดงในข้อความ

ข้อจำกัดของการศึกษานี้อาจรวมถึงความท้าทายในการตีความความคิดเห็นที่อาจเกี่ยวข้องกับหลายด้านหรือมีโครงสร้างประโยคที่ไม่ชัดเจน ซึ่งสามารถแก้ไขได้ด้วยการพัฒนาเทคนิคการตัดคำและการแยกประโยคที่แม่นยำยิ่งขึ้น และการใช้งานร่วมกับการเรียนรู้เชิงลึกเพื่อเพิ่มความสามารถในการจำแนกความรู้สึกและคุณลักษณะของข้อความ

การวิจัยนี้เป็นก้าวสำคัญในการพัฒนาเครื่องมือวิเคราะห์ความรู้สึกสำหรับข้อความจากนักเรียนมัธยมศึกษาตอนต้น และเป็นพื้นฐานสำหรับการศึกษาต่อยอดในอนาคต ทั้งในด้านการปรับปรุงเทคนิคการวิเคราะห์และการขยายไปยังบริบทการศึกษาอื่นๆ

5.3 ประโยชน์และการประยุกต์ใช้งาน

การจำแนกความรู้สึกหรืออารมณ์ในระดับประโยคสำหรับความคิดเห็นของนักเรียนมัธยมศึกษาตอนต้นตาม aspect ที่เกี่ยวข้อง เช่น ครูผู้สอน, บทเรียน, และสภาพแวดล้อม มีประโยชน์และการประยุกต์ใช้งานในหลายด้าน ดังนี้:

5.3.1 ประโยชน์

1. การปรับปรุงคุณภาพการสอน: โดยการวิเคราะห์ความคิดเห็นของนักเรียนเกี่ยวกับครูผู้สอน สามารถช่วยให้ครูได้รับทราบถึงจุดอ่อนและจุดแข็งในวิธีการสอนของตนเอง ทำให้สามารถปรับปรุงและพัฒนาเทคนิคการสอนให้ดียิ่งขึ้น

2. การพัฒนาเนื้อหาบทเรียน: การวิเคราะห์ความคิดเห็นต่อบทเรียนช่วยให้ผู้สอนสามารถทราบถึงความน่าสนใจและความเข้าใจของนักเรียนต่อเนื้อหาบทเรียน เปิดโอกาสในการปรับปรุงและพัฒนาบทเรียนให้เหมาะสมกับความต้องการของนักเรียนมากขึ้น

3. การปรับปรุงสภาพแวดล้อมการเรียนรู้: ความคิดเห็นเกี่ยวกับสภาพแวดล้อมการเรียนรู้ เช่น ความสะดวกสบายในห้องเรียน วัสดุอุปกรณ์การเรียน สามารถช่วยให้โรงเรียนและครูผู้สอนทราบถึงปัญหาและปรับปรุงสภาพแวดล้อมให้เหมาะสมยิ่งขึ้น

5.3.2 การประยุกต์ใช้งาน

1. การวิเคราะห์ความรู้สึกอัตโนมัติ: ใช้เทคนิคการเรียนรู้ของเครื่อง (Machine Learning) หรือการเรียนรู้เชิงลึก (Deep Learning) เพื่อจำแนกความรู้สึกหรืออารมณ์จากความคิดเห็น, ทำให้สามารถวิเคราะห์ข้อมูลจำนวนมากได้อย่างรวดเร็วและแม่นยำ.

2. การติดตามและการรายงาน: สร้างระบบที่สามารถติดตามความคิดเห็นและรายงานผลวิเคราะห์ไปยังผู้ที่เกี่ยวข้อง เช่น ครูผู้สอน หรือผู้บริหารโรงเรียน เพื่อทำให้สามารถตอบสนองและปรับปรุงได้อย่างรวดเร็ว

3. การพัฒนาเครื่องมือสนับสนุนการตัดสินใจ: ผลลัพธ์จากการวิเคราะห์สามารถนำไปใช้เป็นข้อมูลสนับสนุนการตัดสินใจในการพัฒนานโยบาย โครงการ หรือกิจกรรมที่เกี่ยวข้องกับการเรียนการสอน

5.4 ข้อเสนอแนะ

การวิจัยนี้นำเสนอกระบวนการที่ละเอียดและเป็นระบบในการวิเคราะห์ความรู้สึกจากข้อความของนักเรียนมัธยมศึกษาตอนต้น โดยใช้เทคนิคการเรียนรู้ของเครื่องหลากหลายรูปแบบและการให้น้ำหนักคำผ่านทั้ง tf-idf และ tf-igm งานวิจัยได้ให้ข้อมูลเชิงลึกเกี่ยวกับประสิทธิภาพของอัลกอริทึมต่างๆ ในการจำแนกความรู้สึกในระดับประโยคและได้แสดงถึงความสำคัญของการเลือกเทคนิคให้น้ำหนักคำที่เหมาะสมสำหรับงานวิเคราะห์ความรู้สึก อย่างไรก็ตาม มีข้อเสนอแนะสำหรับงานวิจัยในอนาคตคือ

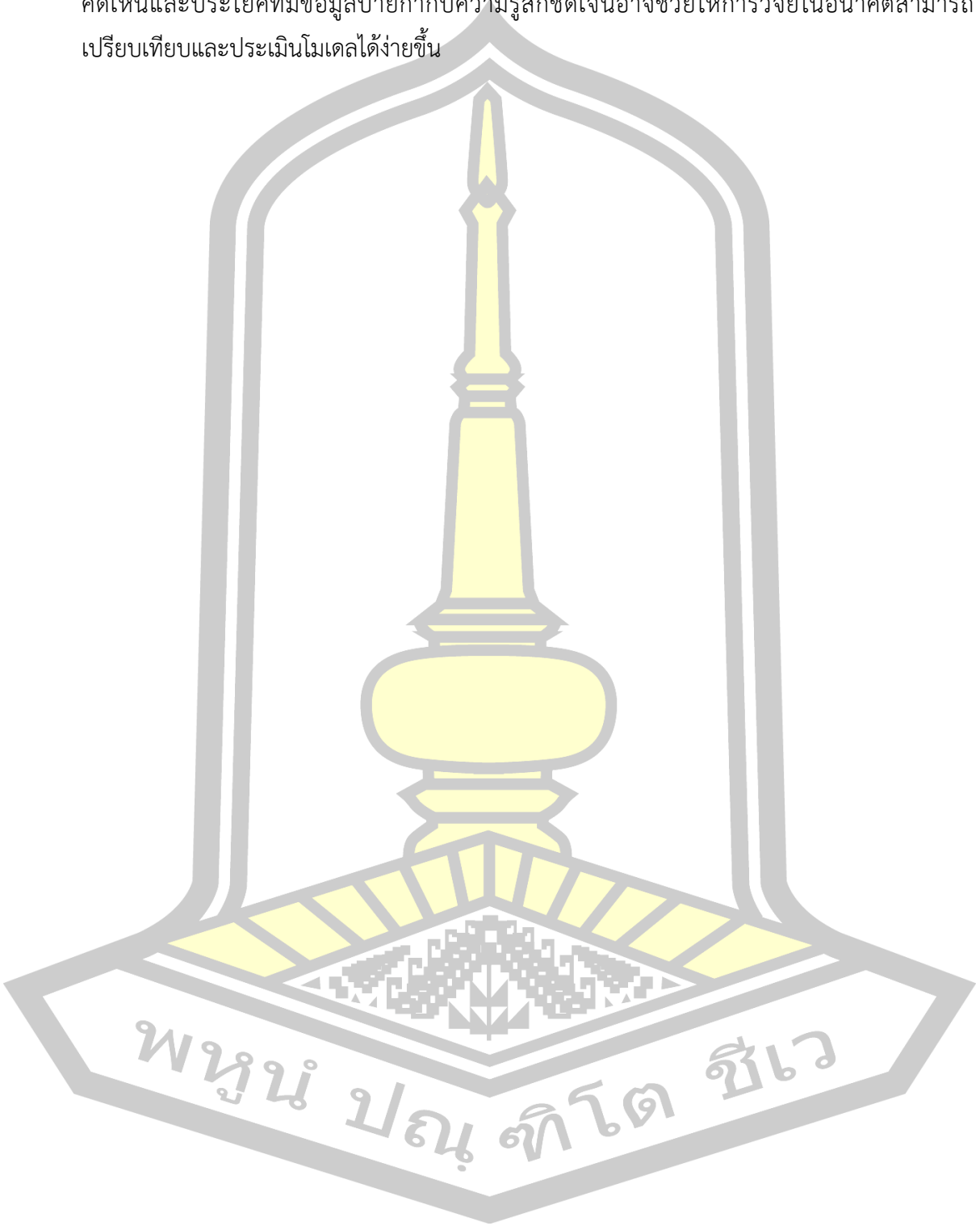
1) การปรับปรุงเทคนิคการตัดคำและแยกประโยค: เพื่อเพิ่มความแม่นยำในการจำแนกความรู้สึกและคุณลักษณะประโยค การพัฒนาหรือใช้เทคนิคการตัดคำและการแยกประโยคที่ซับซ้อนมากขึ้นอาจช่วยลดความผิดพลาดที่เกิดจากโครงสร้างประโยคที่ไม่ชัดเจนหรือมีหลาย Aspect

2) การใช้งานร่วมกับการเรียนรู้เชิงลึก: การผสมผสานเทคนิคการเรียนรู้ของเครื่องแบบดั้งเดิมกับการเรียนรู้เชิงลึกอาจนำไปสู่การปรับปรุงประสิทธิภาพในการวิเคราะห์ความรู้สึก โดยเฉพาะอย่างยิ่งการใช้เครือข่ายประสาทเทียมสำหรับการแยกคุณลักษณะ (feature extraction) และการจำแนกความรู้สึก

3) การพิจารณาบริบททางภาษาและวัฒนธรรม: การวิเคราะห์ความรู้สึกจากข้อความที่เขียนโดยนักเรียนอาจต้องพิจารณาถึงบริบททางภาษาและวัฒนธรรม การศึกษาเพิ่มเติมเกี่ยวกับการใช้ภาษาและการแสดงออกทางอารมณ์ในวัฒนธรรมต่างๆ อาจช่วยเพิ่มความเข้าใจและความแม่นยำในการวิเคราะห์ความรู้สึก

4) การศึกษาเชิงลึกเกี่ยวกับประสิทธิภาพของ tf-idf และ tf-igm: แม้ว่า tf-idf จะได้รับการยืนยันว่าให้ประสิทธิภาพที่ดีในการวิจัยนี้ แต่การศึกษาเชิงลึกเพิ่มเติมเกี่ยวกับการใช้ tf-igm และการปรับปรุงหรือพัฒนาเทคนิคให้น้ำหนักคำอาจนำไปสู่การค้นพบเทคนิคใหม่ๆ ที่เหมาะสมยิ่งขึ้นสำหรับงานวิเคราะห์ความรู้สึก

5) การพัฒนาชุดข้อมูลอ้างอิง: การสร้างชุดข้อมูลอ้างอิงที่มีความหลากหลายของความคิดเห็นและประโยคที่มีข้อมูลป้ายกำกับความรู้สึกชัดเจนอาจช่วยให้การวิจัยในอนาคตสามารถเปรียบเทียบและประเมินโมเดลได้ง่ายขึ้น



บรรณานุกรม

1. แนวคิด หลักการในการวัดและประเมินผลการศึกษา [ออนไลน์]. Available from: http://satun.nfe.go.th/satun/111/ภาค%20ข/1.8%20หลักการและแนวคิดในการวัดและประเมินผลการทางศึกษา.pdf.
2. Rowe, A.D., *Feelings about feedback: the role of emotions in assessment for learning* 2016: p. 159-172.
3. Thematic. *Sentiment Analysis: Comprehensive Beginners Guide* [ออนไลน์]. Available from: https://getthematic.com/sentiment-analysis/.
4. Colace, F., M.D. Santo, and L. Greco, *A Sentiment Analysis Framework for E-Learning.* 2014. 9.
5. Pang, B., L. Lee, and S. Vaithyanathan, *Thumbs up? Sentiment Classification using Machine Learning Techniques.* Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP 2002), 2002: p. 79-86.
6. Mite-Baidal, K., et al., *Sentiment Analysis in Education Domain: A Systematic Literature Review.* Technologies and Innovation. CITI 2018. Communications in Computer and Information Science(2018), 2018. 883.
7. Ortigosa, A., J.M. Martín, and R.M. Carro, *Sentiment analysis in Facebook and its application to e-learning.* 2014. 31: p. 527-541.
8. Pang, B. and L. Lee, *Opinion mining and sentiment analysis.* 2008: p. 1-135.
9. Priya, R.C.M. and J.G.R. Sathiaseelan, *An Explorative Study on Sentiment Analysis.* World Congress on Computing and Communication Technologies (WCCCT), 2017: p. 140-142.
10. Haruechaiyasak, C., S. Kongyoung, and M. Dailey, *A comparative study on Thai word segmentation approaches.* International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology, 2008: p. 125-128.
11. Indurkhya, N. and F.J. Damerau, *Handbook of Natural Language Processing*. 2010.
12. Salton, G., A. Wong, and C. Yang, *A vector space model for automatic indexing.* 1975. 18(11): p. 613-620.
13. Salton, G. and C. Buckley, *Term-weighting approaches in automatic text retrieval.* 1988. 24(5): p. 513-523.
14. Debole, F. and F. Sebastiani, *Supervised term weighting for automated text categorization.* 18th ACM Symposium on Applied Computing, 2004. 138: p. 81-

97.

15. Lan, M., et al., *Supervised and traditional term weighting methods for automatic text categorization.* IEEE Transactions on Pattern Analysis and Machine Intelligence 2009. 31: p. 721-735.
16. Mikolov, T., W.-t. Yih, and G. Zweig, *Linguistic Regularities in Continuous Space Word Representations.* The 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT-2013), 2013: p. 746-751.
17. Zhang, W., W. Xu, and G. Chen, *A Feature Extraction Method Based on Word Embedding for Word Similarity Computing.* International Conference on Natural Language Processing and Chinese Computing, 2014. 496: p. 160-167.
18. Jatnika, D., M.A. Bijaksana, and A.A. Suryani, *Word2Vec Model Analysis for Semantic Similarities in English Words.* The 4th International Conference on Computer Science and Computational Intelligence (ICCSCI 2019), 2019. 157: p. 160-167.
19. Kavlakoglu, E. *Classifying data using the Multinomial Naive Bayes algorithm [*ออนไลน์]. 2024; Available from: https://developer.ibm.com/tutorials/awb-classifying-data-multinomial-naive-bayes-algorithm/.
20. Hassanat, A.B., et al., *Solving the Problem of the K Parameter in the KNN Classifier Using an Ensemble Learning Approach.* International Journal of Computer Science and Information Security, 2014. 12.
21. Burges, C.J., *A Tutorial on Support Vector Machines for Pattern Recognition.* 1998. 2: p. 121-167.
22. Sruthi, E.R. *Understand Random Forest Algorithms With Examples [*ออนไลน์]. 2024; Available from: https://www.analyticsvidhya.com/blog/2021/06/understanding-random-forest/.
23. Yiu, T. *Understanding Random Forest [*ออนไลน์]. 2019; Available from: https://towardsdatascience.com/understanding-random-forest-58381e0602d2.
24. Anguita, D., S. Ridella, and F. Rivieccio, *K-fold generalization capability assessment for support vector classifiers.* IEEE International Joint Conference on Neural Networks, 2005. 2: p. 855-858.
25. Powers, D., *Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation.* Journal of Machine Learning Technologies, 2011. 2: p. 37-63.
26. Ulfa, S., et al., *Student Feedback on Online Learning by Using Sentiment Analysis: A Literature Review.* 2020 6th International Conference on Education

and Technology (ICET), 2020: p. 53-58.

27. Brennan, J. and R. Williams, *Collecting and using student feedback - a guide to good practice.* 2004.
28. Gottipati, S., V. Shankararaman, and S. Gan, *A conceptual framework for analyzing students' feedback.* IEEE Frontiers in Education Conference (FIE), 2017: p. 1-8.
29. Promla, P. and C. Sanrach, *The Comparison of Efficiency on The Analysis of Satisfaction on Teaching Performance using Sentiment Analysis by Ensemble Technique.* 2020. 20: p. 140-149.
30. Qi, C. and S. Liu, *Evaluating On-Line Courses via Reviews Mining.* 2021. 9: p. 35439-35451.
31. Wongkar, M. and A. Angdresey, *Sentiment Analysis Using Naive Bayes Algorithm Of The Data Crawler: Twitter.* International Conference on Informatics and Computing (ICIC), 2019: p. 1-5.
32. Onan, A., *Sentiment analysis on massive open online course evaluations: A text mining and deep learning approach.* 2020. 29(9): p. 572-589.
33. Osmanoğlu, U.Ö., et al., *Sentiment Analysis for Distance Education Course Materials: A Machine Learning Approach.* Journal of Educational Technology and Online Learning, 2020. 3(1): p. 31-48.
34. Kechaou, Z., M.B. Ammar, and A.M. Alimi, *Improving e-learning with sentiment analysis of users' opinions.* IEEE Global Engineering Education Conference (EDUCON), 2011: p. 1032-1038.
35. ปรีดาวรรณ เกษเมธีการุณ, พยุง มีสัจ, and สุมิตรา นวลมีสรี, ระบบการวิเคราะห์ความรู้สึกจากวิดีโอบนโซเชียลมีเดียด้วยซัพพอร์ตเวกเตอร์แมชชีน. 2016.

พหุน ปณฺ ฑิโต ชีเว

ประวัติผู้เขียน

ชื่อ	อรทัย จันท่าม่วง
วันเกิด	25 มกราคม 2526
สถานที่เกิด	จังหวัดร้อยเอ็ด
สถานที่อยู่ปัจจุบัน	82 หมู่ 2 ตำบลวังหลวง อำเภอเสลภูมิ จังหวัดร้อยเอ็ด 45120
ตำแหน่งหน้าที่การงาน	ครูผู้สอน
สถานที่ทำงานปัจจุบัน	โรงเรียนคำพระโคกก่งวิทยา อำเภอโพนทอง จังหวัดร้อยเอ็ด
ประวัติการศึกษา	พ.ศ. 2549 บธ.บ. (บริหารธุรกิจบัณฑิต) มหาวิทยาลัยราชภัฏมหาสารคาม พ.ศ. 2567 วท.ม (วิทยาศาสตรมหาบัณฑิต) มหาวิทยาลัยมหาสารคาม
ผลงานวิจัย	Chantamuang, O., Polpinij, J., Vorakitphan, V. and Luaphol, B. Sentence-Level Sentiment Analysis for Student Feedback Relevant to Teaching Process Assessment, The 15th Multi-Disciplinary International Conference on Artificial Intelligence 2022 (MIWAI2022).