การพยากรณ์ทิศทางราคาสัญญาฟิวเจอร์สข้าวโพดจากข้อความข่าวโดยใช้การประมวลภาษาธรรมชาติ

CORN FUTURES PRICE TREND PREDICTION FROM NEWS CONTENT USING NATURAL LANGUAGE PROCESSING

วันธนวัฒน์ พิมพ์สุข

สหกิจศึกษานี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
ปริญญาวิทยาศาสตรบัณฑิต (สถิติประยุกต์)
ภาควิชาสถิติ คณะวิทยาศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2565

CORN FUTURES PRICE TREND PREDICTION FROM NEWS CONTENT USING NATURAL LANGUAGE PROCESSING

WANTANAWAT PIMSUK

A COOPERATIVE EDUCATION SUBMITTED IN
PARTIAL FULFILLMENT OF THE REQUIREMENT FOR
THE DEGREE OF BACHELOR OF SCIENCE (APPLIED STATISTICS)
DEPARTMENT OF STATISTICS, SCHOOL OF SCIENCE
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG
ACADEMIC YEAR 2022

หัวข้อสหกิจศึกษา	การพยากรณ์ทิศทางราคาสัญญาฟิวเจอร์สข้าวโพดจากข้อความข่าวโดยใช้การประมวลภาษาธรรมชาติ Corn Futures Price Trend Prediction From News Content Using Natural Language Processing
ชื่อนักศึกษา	นายวันธนวัฒน์ พิมพ์สุข รหัสนักศึกษา 62050830
ปริญญา	วิทยาศาสตรบัณฑิต (สถิติประยุกต์)
ภาควิชา	สถิติ
ปีการศึกษา	2565
อาจารย์ที่ปรึกษา	ผศ.ดร.ยุวดี กล่อมวิเศษ

คณะวิทยาศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง (สจล.) อนุมัติให้สหกิจศึกษานี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรบัณฑิต (สถิติประยุกต์) ประจำปีการศึกษา 2565

คณะกรรมการสอบ	ลายมือชื่อ
ผศ.ดร.พรพิมล ชัยวุฒิศักดิ์ ประธานกรรมการ	
ดร.ภาณุ ทองจันทร์ กรรมการ	
คุณชญานิน บุญมานะ กรรมการ	
ผศ.ดร.ยุวดี กล่อมวิเศษ กรรมการและอาจารย์ที่ปรึกษา	

หัวข้อสหกิจศึกษา การพยากรณ์ทิศทางราคาสัญญาฟิวเจอร์สข้าวโพดจากข้อความข่าวโดยใช้การประมวลภาษาธรรมชาติ
ชื่อนักศึกษา นายวันธนวัฒน์ พิมพ์สุข รหัสนักศึกษา 62050830
ปริญญา วิทยาศาสตรบัณฑิต (สถิติประยุกต์)
ภาควิชา สถิติ
คณะ วิทยาศาสตร์
มหาวิทยาลัย สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง (สจล.)
ปีการศึกษา 2565
อาจารย์ที่ปรึกษา ผศ.ดร.ยุวดี กล่อมวิเศษ

บทคัดย่อ

การซื้อขายข้าวโพดในตลาดซื้อขายล่วงหน้านั้นมักจะนำปัจจัยทางเทคนิคและปัจจัยพื้นฐานมาใช้ในการวิเคราะห์ประกอบการลงทุน แต่ในปัจจุบันมีข่าวและเหตุการณ์ต่าง ๆ ที่เกิดขึ้นจากหลายประเทศทั่วโลก ล้วนเป็นอีกหนึ่งปัจจัยที่ส่งผลต่อความผันผวนของราคาสินค้าเกษตร ดังนั้นจึงต้องมีเครื่องมือช่วยในการนำข้อความข่าวในแต่ละวันมาพยากรณ์ราคาวัตถุดิบ เพื่อเพิ่มโอกาสในการสร้างผลตอบแทนที่คุ้มค่าต่อการลงทุน ผู้วิจัยจึงมีความสนใจในการศึกษาการพยากรณ์ทิศทางราคาสัญญาฟิวเจอร์สข้าวโพดจากข้อความข่าว โดยใช้วิธีการประมวลภาษาธรรมชาติ ใช้การแปลงเชิงปริมาณ 2 วิธี คือ ถุงคำ และ เทคนิคการคัดแยกคำตามความสำคัญ ใช้การแปลงเชิงคุณลักษณะด้วยวิธีเวิร์ดทูเวก เมื่อกำหนดคลังคำศัพท์และเวกเตอร์คุณลักษณะเท่ากับ 5,000 คำ และ 330 คำ เพื่อนำมาสร้างแบบจำลองการถดถอยลอจิสติก, ซัพพอร์ตเวกเตอร์แมชชีน และโครงข่ายประสาทเทียม ในการเปรียบเทียบประสิทธิภาพของแบบจำลองโดยพิจารณาจากค่า F1-Score เป็นอันดับแรกในการวัดผล ผู้วิจัยเก็บรวบรวมข้อมูลตั้งแต่เดือนมกราคม พ.ศ. 2556 ถึงเดือนธันวาคม พ.ศ. 2563 รวมทั้งสิ้น 2,065 วัน โดยแบ่งข้อมูลชุดเรียนรู้ 1,652 วัน และข้อมูลชุดทดสอบ 413 วัน เมื่อพิจารณาจากผลลัพธ์ของแบบจำลองในชุดข้อมูลทดสอบ พบว่าเมื่อใช้การแปลงคุณลักษณะด้วยวิธีเวิร์ดทูเวก และแบบจำลองการถดถอยลอจิสติก มีประสิทธิภาพสูงที่สุด และเป็นแบบจำลองที่เหมาะสมที่สุดสำหรับการพยากรณ์ทิศทางราคาสัญญาฟิวเจอร์สข้าวโพดจากข้อความข่าว

คำสำคัญ : ถุงคำ, เทคนิคการคัดแยกคำตามความสำคัญ, เวิร์ดทูเวก, การถดถอยลอจิสติก, ซัพพอร์ตเวกเตอร์แมชชีน, โครงข่ายประสาทเทียม

Title	Corn Futures Price Trend Prediction From News Content Using Natural Language Processing
Students	Mr. Wantanawat Pimsuk Student ID 62050830
Degree	Bachelor of Science (Applied Statistics)
Department	Statistics
School	Science
University	King Mongkut's Institute of Technology Ladkrabang (KMITL)
Academic Year	2565
Advisor	Asst. Prof. Dr. Yuwadee Klomwises

Abstract

Corn futures are often traded by using technical indicators and fundamental factors. However, news and situations around the world have had impacts on the volatility of agricultural commodity prices. Thus, it is recommended to create a tool for predicting corn price trends to increase the possibility of making worth returns that is worth the investment. Therefore, we are interested in developing a model for predicting corn futures price trends from news content using natural language processing. This study used two text representations, Bag of Words and TF-IDF. Feature vector used the Word2Vec method. When vocabulary size and feature vectors are given, 5,000 words and 330 words with Logistic Regression, Support Vector Machine and Artificial Neural Network. The model performance evaluation is based on the F1-Score value first in the result measure. We collected data from January 2013 to January 2020. There are a total of 2,065 days, which are divided into 1,652 days of training data and 413 days of testing data. As a result, model performance based on testing data we found that when using feature vector by Word2Vec method and a logistic regression model was the most efficient model and is the most suitable model for predicting corn futures price trend from news content.

Keywords : Bag of Words, TF-IDF, Word2Vec, Logistic Regression, Support Vector Machine, Artificial Neural Network

กิตติกรรมประกาศ

สหกิจศึกษานี้สำเร็จลุล่วงไปได้ด้วยดี เนื่องจากคณะผู้จัดทำได้รับความอนุเคราะห์และความกรุณาจากคณะอาจารย์และบุคคลผู้มีพระคุณหลายท่าน ดังรายนามต่อไปนี้

ขอขอบพระคุณ ผศ.ดร.ยุวดี กล่อมวิเศษ อาจารย์ภาควิชาสถิติ คณะวิทยาศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง อาจารย์ที่ปรึกษาสหกิจศึกษาที่ได้ช่วยแนะนำและให้คำปรึกษา รวมถึงเสนอแนะแนวทางการแก้ไขปัญหา ตลาดจนถ่ายทอดประสบการณ์ในการทำงานของท่านเพื่อเป็นประโยชน์ในการคิดวิเคราะห์ การวางแผนในโครงงานนี้ โดยท่านได้ให้คำปรึกษาตั้งแต่การค้นหาข้อมูลตลอดจนการทำงานวิจัยสำเร็จ รวมทั้งตรวจทานแก้ไขสหกิจศึกษาเล่มนี้ให้สมบูรณ์

ขอขอบพระคุณ ผศ.ดร.พรพิมล ชัยวุฒิศักดิ์ ที่กรุณาเป็นกรรมการในการสอบสหกิจศึกษา อีกทั้งยังให้ความรู้ คำแนะนำ และช่วยตรวจสอบแก้ไขให้สหกิจศึกษาเล่มนี้ออกมาสมบูรณ์

ขอขอบพระคุณ ดร.ภาณุ ทองจันทร์ ผู้ช่วยกรรมการผู้จัดการ หน่วยงานวิเคราะห์เชิงปริมาณ (Quantitative Model) ด้านเทคโนโลยีสารสนเทศ กลุ่มธุรกิจการค้าวัตถุดิบอาหารสัตว์ บริษัท กรุงเทพ โปรดิ๊วส จำกัด (มหาชน) ที่มอบโอกาส และให้การอนุเคราะห์ในการทำสหกิจครั้งนี้ และขอขอบพระคุณคุณชญานิน บุญมานะ คุณณรงค์พล วิชัยลักขณา คุณศุภพงศ์ คงเจริญ คุณฐิติมา ตโมหรณวงศ์ และพี่ ๆ ในหน่วยงานที่ดูแล เอาใจใส่ ให้คำปรึกษา ให้กำลังใจ และคำแนะนำ อีกทั้งยังช่วยแก้ไขข้อผิดพลาดต่าง ๆ ตลอดระยะเวลาการทำสหกิจศึกษาในครั้งนี้จนสามารถสำเร็จลุล่วงไปได้ด้วยดี

สุดท้ายนี้ผู้จัดทำขอขอบคุณบิดา มารดา และบุคคลในครอบครัว รวมทั้งเพื่อน ๆ พี่ ๆ และบุคคลที่ไม่ได้กล่าวถึงมา ณ ที่นี้ที่ให้ความช่วยเหลือ การสนับสนุน และกำลังใจตลอดการทำสหกิจศึกษานี้ให้สำเร็จไปได้ด้วยดี

วันธนวัฒน์ พิมพ์สุข

สารบัญ

สารบัญ (ต่อ)

สารบัญ (ต่อ)

หน้า

สารบัญ (ต่อ)

หน้า

สารบัญตาราง

ตารางที่ **หน้า**

สารบัญตาราง (ต่อ)

ตารางที่ **หน้า**

สารบัญตาราง (ต่อ)

ตารางที่ **หน้า**

สารบัญตาราง (ต่อ)

ตารางที่ **หน้า**

สารบัญรูป

รูปที่ **หน้า**

สารบัญรูป (ต่อ)

รูปที่ **หน้า**

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ข้าวโพด (Corn) เป็นธัญพืชเศรษฐกิจชนิดหนึ่งที่มีความสำคัญเป็นอันดับสามของโลก รองมาจากข้าวสาลี และข้าว สามารถปลูกได้ทั่วไปในเขตภูมิอากาศอบอุ่น (นพพร และคณะ, 2547) เขตกึ่งร้อนชื้น และพื้นที่ราบเขตร้อน โดยแหล่งปลูกมักกระจายอยู่ตามภูมิภาคต่างๆ ของโลก ได้แก่ ประเทศสหรัฐอเมริกา บราซิล เม็กซิโก จีน รวมทั้งในทวีปแอฟริกาใต้ สำหรับประเทศไทยข้าวโพดถือเป็นพืชเศรษฐกิจที่สำคัญ เนื่องจากมีพื้นที่เพาะปลูกครอบคลุมอยู่ทั่วทุกภาค (สำนักหอสมุดและศูนย์สารสนเทศวิทยาศาสตร์และเทคโนโลยี, 2561) ปัจจุบันมีความต้องการข้าวโพดเพิ่มขึ้นอย่างต่อเนื่อง ตามการขยายตัวของอุตสาหกรรมอาหารสัตว์ ในขณะที่ผลผลิตยังไม่เพียงพอกับความต้องการใช้ในประเทศ จึงต้องมีการนำเข้าข้าวโพดจากต่างประเทศ นอกจากการซื้อขายข้าวโพดจากตลาดธัญพืชแบบปรกติแล้ว ยังมีการซื้อขายข้าวโพดในรูปแบบสัญญาฟิวเจอร์ส (Futures Contract) เป็นหนึ่งในตลาดการลงทุนที่สามารถสร้างกำไรแก่ผู้ประกอบการ เกษตรกร และผู้ลงทุนได้อีกด้วย

สัญญาฟิวเจอร์ส (Futures Contract) หรือที่นิยมเรียกกันว่า "ฟิวเจอร์ส" ถือเป็นตราสารทางการเงินอย่างหนึ่งที่อยู่ในประเภทสัญญาซื้อขายล่วงหน้า ซึ่งจัดทำสัญญามาตรฐานขึ้นมาและนำมาซื้อขายโดยตรงผ่านทางศูนย์ซื้อขายที่จัดตั้งอย่างเป็นทางการ โดยลักษณะของสัญญามาตรฐานนี้จะครอบคลุมถึงคุณสมบัติต่าง ๆ อาทิเช่น สินค้าอ้างอิงมาตรฐาน วันและเดือนครบกำหนดอายุมาตรฐาน และลักษณะการส่งมอบหรือชำระราคามาตรฐาน เป็นต้น ด้วยเหตุนี้ผู้ซื้อและผู้ขายจึงมีเพียงราคาเป็นปัจจัยเดียวที่ใช้ในการต่อรอง โดยราคาที่ผู้ซื้อและผู้ขายตกลงในสัญญานี้เราเรียกว่า "ราคาฟิวเจอร์ส" ซึ่งหมายถึงราคาในอนาคตของสินค้าอ้างอิง ณ วันครบกำหนดอายุ (กวี, 2552) ข้อดีที่เด่นชัดของสัญญาฟิวเจอร์สคือสามารถทำการซื้อขายได้ทั้งในช่วงตลาดขาขึ้น (Bullish Market) และตลาดขาลง (Bearish Market) ถ้าหากมีแนวทางการลงทุนหรือการคาดการณ์ทิศทางราคาและกลยุทธ์การซื้อขายที่เหมาะสมและแม่นยำ จะสามารถสร้างกำไรแก่ผู้ลงทุนได้

ในอดีตการลงทุนในสัญญาฟิวเจอร์สมักจะมีการนำปัจจัยทางเทคนิค (Technical Indicator) ที่เป็นการนำข้อมูลราคาในอดีตมาวิเคราะห์พฤติกรรมของตลาดมาใช้ในการวิเคราะห์ประกอบการลงทุน และมีการนำปัจจัยพื้นฐาน (Fundamental Indicator) ที่สื่อถึงอุปสงค์และอุปทานในตลาดซึ่งมีผลต่อราคาของสินค้าในอนาคต แต่ในปัจจุบันอินเทอร์เน็ตและเทคโนโลยีสารสนเทศเข้ามามีบทบาทในชีวิตประจำวันของมนุษย์มากขึ้น ช่วยอำนวยความสะดวกในด้านการติดต่อซื้อขายสินค้า การทำธุรกรรมต่าง ๆ รวมไปถึงการนำเสนอข่าวซึ่งเป็นสื่อกลางในการนำเสนอข้อมูลจากหลายประเทศทั่วโลก และมีเนื้อหาหลากหลายด้าน เช่น เศรษฐกิจ การเมือง การพยากรณ์อากาศ เป็นต้น ซึ่งข่าวและเหตุการณ์ต่าง ๆ ที่เกิดขึ้นรอบโลกล้วนเป็นอีกหนึ่งปัจจัยที่ส่งผลต่อความผันผวนของราคา

สินค้าเกษตรทั้งทางตรงและทางอ้อม ดังนั้นจึงต้องมีเครื่องมือช่วยในการนำข้อความข่าวในแต่ละวันมาพยากรณ์ราคาวัตถุดิบที่มีความผันผวนตลอดเวลาเพื่อสร้างผลตอบแทนที่คุ้มค่าต่อการลงทุน

ผู้วิจัยจึงมีความสนใจในการศึกษาการพยากรณ์ทิศทางราคาสัญญาฟิวเจอร์สข้าวโพดจากข้อความข่าว โดยใช้วิธีการประมวลภาษาธรรมชาติ (Natural Language Processing; NLP) เพื่อสร้างแบบจำลองพยากรณ์ทิศทางด้วยแบบจำลองการถดถอยลอจิสติก (Logistic Regression) ซัพพอร์ตเวกเตอร์แมชชีน (Support Vector Machine; SVM) และโครงข่ายประสาทเทียม (Artificial Neural Network; ANN) รวมถึงศึกษาอิทธิพลของข้อความข่าว และเปรียบเทียบประสิทธิภาพของแบบจำลอง สามารถนำการพยากรณ์จากแบบจำลองไปประกอบการตัดสินใจลงทุนเพื่อให้ได้ผลตอบแทนที่คุ้มค่าต่อนักลงทุน

1.2 วัตถุประสงค์ของงานวิจัย

1.2.1 เพื่อศึกษาอิทธิพลของข้อความข่าวในการพยากรณ์ทิศทางราคาสัญญาฟิวเจอร์สข้าวโพด

1.2.2 เพื่อสร้างแบบจำลองการพยากรณ์ทิศทางราคาสัญญาฟิวเจอร์สข้าวโพด ด้วยแบบจำลองวิธีการถดถอยลอจิสติก (Logistic Regression) ซัพพอร์ตเวกเตอร์แมชชีน (Support Vector Machine; SVM) และโครงข่ายประสาทเทียม (Artificial Neural Network; ANN)

1.2.3 เพื่อเปรียบเทียบประสิทธิภาพของแบบจำลองในการพยากรณ์ทิศทางราคาสัญญาฟิวเจอร์สข้าวโพด

1.3 ขอบเขตของงานวิจัย

1.3.1 ขอบเขตด้านข้อมูล

1) สืบค้นข้อมูลข่าวรายวัน จากสำนักข่าวรอยเตอร์ (Reuters) ตั้งแต่วันที่ 1 มกราคม พ.ศ. 2556 ถึงวันที่ 31 ธันวาคม พ.ศ. 2563

2) สืบค้นข้อมูลราคาข้าวโพด จากตลาดหอการค้าแห่งนครชิคาโก หรือ Chicago Board of Trade (CBOT) ตั้งแต่วันที่ 1 มกราคม พ.ศ. 2556 ถึงวันที่ 31 ธันวาคม พ.ศ. 2563

1.3.2 ขอบเขตด้านเครื่องมือ

เครื่องมือที่ใช้สำหรับการวิเคราะห์ข้อมูลและสร้างแบบจำลอง

- โปรแกรมภาษาไพธอน (Python 3) Colab Notebooks
- โปรแกรมภาษาไพธอน (Python 3) Jupyter Notebooks
- โปรแกรม Microsoft Office Excel

1.4 ประโยชน์ที่คาดว่าจะได้รับ

1.4.1 สามารถนำแบบจำลองที่ได้ไปใช้ในการวางแผน และกำหนดกลยุทธ์ซื้อขายสัญญาฟิวเจอร์สข้าวโพด

1.4.2 สามารถนำแบบจำลองไปประยุกต์ใช้กับสินค้าทางการเกษตรอื่น ๆ

1.5 นิยามศัพท์เฉพาะ

1.5.1 สัญญาฟิวเจอร์ส (Futures Contract) หมายถึง การซื้อขายสินค้าในตลาดสินค้าโภคภัณฑ์รูปแบบหนึ่งที่ผู้ซื้อกับผู้ขายตกลงจะซื้อขายสินค้าโดยกำหนดราคากันตั้งแต่วันนี้ แต่จะส่งมอบและชำระเงินในอนาคตตามราคาที่ตกลงไว้ไม่ว่าราคาในขณะนั้นจะเป็นเท่าไหร่ก็ตาม ข้อดีที่เด่นชัดของสัญญาฟิวเจอร์สคือ โอกาสในการสร้างกำไร ได้ทั้งในตลาดกระทิง (Bullish Market) และตลาดหมี (Bearish Market)

1.5.2 สถานะซื้อ (Long Position) หมายถึง สถานะของผู้ซื้อสัญญาฟิวเจอร์ส หรือเรียกว่า "ฐานะซื้อ" เช่น นักลงทุนคาดการณ์ว่าราคาจะปรับขึ้นในอีก 2 เดือนข้างหน้า นักลงทุนจึงซื้อสัญญา

1.5.3 สถานะขาย (Short Position) หมายถึง สถานะของผู้ขายสัญญาฟิวเจอร์ส หรือเรียกว่า "ฐานะขาย" เช่น นักลงทุนคาดการณ์ว่าราคาจะปรับลงในอีก 2 เดือนข้างหน้า นักลงทุนจึงขายสัญญา

1.5.4 ผลตอบแทน (Return) คือ ผลตอบแทนที่ได้จากการการซื้อขายสัญญาฟิวเจอร์ส

1.5.5 ตลาดกระทิง (Bullish Market) หมายความว่า ราคาเป็นขาขึ้นหรือราคามากกว่าราคาก่อนหน้า นักลงทุนมีความเชื่อมั่น มีการเข้าซื้อปริมาณมาก ราคาจึงขึ้น ดังนั้น ตลาดกระทิง ก็คือการที่ราคาขึ้นอย่างต่อเนื่องหลายวัน

1.5.6 ตลาดหมี (Bearish Market) หมายความว่า ราคาเป็นขาลงหรือน้อยกว่าราคาก่อนหน้า นักลงทุนมีความกลัว ไม่มั่นใจ จึงขายออกเยอะ หรือขายเพื่อทำกำไร ราคาจึงตกลง ดังนั้น ตลาดหมีคือการที่ราคาลดลงต่อเนื่องหลายวัน

1.5.7 ตลาดหอการค้าแห่งนครชิคาโก หรือ Chicago Board of Trade (CBOT) เป็นตลาดสินค้าล่วงหน้าแห่งแรกในสหรัฐอเมริกา

1.5.8 ราคาข้าวโพด ในงานวิจัยนี้หมายถึง ราคาข้าวโพดในรัฐชิคาโก สหรัฐอเมริกา

บทที่ 2
ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

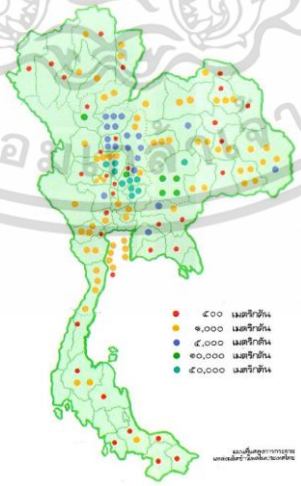
การศึกษาวิจัยเรื่อง “การพยากรณ์ทิศทางราคาสัญญาฟิวเจอร์สข้าวโพดจากข้อความข่าวโดยใช้การประมวลภาษาธรรมชาติ” ผู้วิจัยได้รวบรวมแนวคิด ทฤษฎี และหลักการต่าง ๆ จากเอกสารและงานวิจัยที่เกี่ยวข้อง โดยมีรายละเอียดของเนื้อหาดังนี้

1. ข้าวโพด (Corn)
2. การประมวลภาษาธรรมชาติ (Natural Language Processing; NLP)
    - 2.1 วิวัฒนาการของการประมวลภาษาธรรมชาติ
    - 2.2 ความสำคัญของการประมวลภาษาธรรมชาติ
    - 2.3 กระบวนการทำงานของการประมวลภาษาธรรมชาติ
        - 2.3.1 การเตรียมข้อมูล (Data Preprocessing)
        - 2.3.2 การแปลงเชิงปริมาณ (Text Representation)
        - 2.3.3 คำฝังตัว (Word Embedding)
        - 2.3.4 เวิร์ดทูเวก (Word2Vec)
3. การเรียนรู้ของเครื่อง (Machine Learning)
    - 3.1 การเรียนรู้แบบมีผู้สอน (Supervised Learning)
    - 3.2 การเรียนรู้แบบไม่มีผู้สอน (Unsupervised Learning)
    - 3.3 การเรียนรู้แบบเสริมกำลัง (Reinforcement Learning)
4. การจำแนกประเภท (Classification)
    - 4.1 การถดถอยแบบลอจิสติก (Logistic Regression)
    - 4.2 ซัพพอร์ตเวกเตอร์แมชชีน (Support Vector Machine; SVM)
    - 4.3 โครงข่ายประสาทเทียม (Artificial Neural Network; ANN)
5. การวัดประสิทธิภาพของแบบจำลอง (Model Performance Evaluation)
6. งานวิจัยที่เกี่ยวข้อง (Related Research)

2.1 ข้าวโพด (Corn)

ข้าวโพด (ภาษาอังกฤษ คือ Corn หรือ Maize) มีชื่อวิทยาศาสตร์ว่า Zea mays จัดอยู่ในวงศ์ Gramineae เป็นพืชล้มลุกใบเดี่ยวตระกูลเดียวกับหญ้า โดยข้าวโพดเป็นหนึ่งในธัญพืชที่มีความสำคัญรองมาจากข้าวสาลี และข้าว มนุษย์รู้จักข้าวโพดมานานกว่า 4,500 ปี สามารถปลูกได้ทั่วไปในเขตภูมิอากาศอบอุ่น เขตกึ่งร้อนชื้น และพื้นที่ราบเขตร้อน (นพพร และคณะ, 2547) ซึ่งอาจมีถิ่นกำเนิดดั้งเดิมอยู่ 2 แหล่ง โดยอาศัยหลักฐานของการเพาะปลูก คือ 1. พื้นที่แถบที่ราบสูงซึ่งเป็นที่ตั้งของประเทศเปรู โบลิเวีย เอกวาดอร์ ชิลี อาร์เจนตินา และบราซิล ในทวีปอเมริกาใต้ 2. พื้นที่ทางตอนใต้ของทวีปอเมริกา แถบอเมริกากลาง ประเทศแม็กซิโก กัวเตมาลา โคลัมโบ และเวเนซูเอลา ได้มีการแพร่กระจายของข้าวโพดไปยังส่วนต่างๆ ของโลก ได้แก่ ทวีปอเมริกาและหมู่เกาะแคริบเบียน และแพร่กระจายไปยังประเทศสเปน ทวีปยุโรป ใน พ.ศ. 2035 และ พ.ศ. 2036 ตามลำดับ หลังจากนั้นจึงแพร่กระจายไปสู่ส่วนอื่น ๆ ของทวีปแอฟริกา เอเชีย และออสเตรเลีย (สถาบันวิจัยและพัฒนาแห่งมหาวิทยาลัยเกษตรศาสตร์, 2560)

ในประเทศไทยข้าวโพดถือเป็นพืชเศรษฐกิจที่สำคัญ เนื่องจากมีพื้นที่เพาะปลูกครอบคลุมอยู่ทั่วทุกภาค สร้างรายได้จำนวนมากให้กับประเทศ ข้าวโพดที่ปลูกในประเทศไทยแบ่งออกเป็น 2 กลุ่ม คือ ข้าวโพดฝักสด และข้าวโพดเลี้ยงสัตว์ โดยข้าวโพดฝักสดปลูกเพื่อใช้สำหรับบริโภคและส่งออก ส่วนข้าวโพดเลี้ยงสัตว์เป็นพืชที่มีความสำคัญต่ออุตสาหกรรมอาหารสัตว์ เนื่องจากใช้เป็นวัตถุดิบในการผลิตอาหารสัตว์ นอกจากนี้สามารถนำมาแปรรูปเป็นผลิตภัณฑ์ได้หลายชนิด ทั้งในระดับครัวเรือนและในระดับอุตสาหกรรม เพื่อช่วยถนอมอาหาร เพิ่มผลผลิตทางการเกษตร ส่งเสริมการใช้ทรัพยากรธรรมชาติให้เกิดประโยชน์สูงสุด ซึ่งจังหวัดที่เป็นแหล่งปลูกข้าวโพดที่สำคัญของประเทศไทย ได้แก่ เพชรบูรณ์ นครราชสีมา ตาก น่าน นครสวรรค์ (โชคชัย และเกตุอร, 2561)



รูปที่ 2.1 แผนที่แสดงการกระจายแหล่งผลิตข้าวโพดในประเทศไทย
(ที่มา : https://saranukromthai.or.th/sub/book/book.php?book=3&chap=2&page=t3-2-infodetail03.html)

2.1.1 ลักษณะทางพฤกษศาสตร์ของข้าวโพด

ข้าวโพดเป็นพืชล้มลุก ปลูกง่าย อายุสั้น สามารถเจริญเติบโตได้ดีในทุกภาคของประเทศไทย โดยข้าวโพดมีลักษณะทางพฤกษศาสตร์ที่สำคัญ (นพพร และคณะ, 2547) ดังนี้

1) ราก เป็นระบบรากฝอย มีลักษณะกลมเล็กๆ แทงลงในดิน มีรากออกที่ข้อลำต้นที่อยู่ใต้ดินรอบ ๆ ลำต้น มีรัศมีประมาณ 1 เมตร และหยั่งลึกลงไปในดินได้ 2.1-2.4 เมตร มีหน้าที่ปกป้องพืชไม่ให้ร่วงหล่นและให้สารอาหารเพิ่มเติม

รูปที่ 2.2 รากข้าวโพด (ที่มา : https://shorturl.asia/XQNYk)

2) ลำต้น เป็นพืชล้มลุกขนาดเล็ก ลำต้นตั้งตรงและค่อนข้างกลม มีข้อและปล้อง มีแก่นเนื้อคล้ายฟองน้ำ ความสูงตั้งแต่ 30 เซนติเมตรขึ้นไป ขนาดเส้นผ่าศูนย์กลางประมาณ 2.5–5.0 เซนติเมตร มีขนหยาบ ๆ ปกคลุมลำต้นสีเขียว แต่บางสายพันธุ์มีลำต้นเป็นสีม่วง

รูปที่ 2.3 ลำต้นข้าวโพด (ที่มา : https://shorturl.asia/Vt0fd)

3) ใบ เป็นใบเดี่ยว มีลักษณะยาวรี เป็นเส้นตรงปลายแหลม ยาวประมาณ 30–100 เซนติเมตร เส้นกลางใบจะเห็นได้ชัด ตรงขอบใบมีขนอ่อน ๆ มีเขี้ยวใบ ลักษณะและสีของใบจะแตกต่างกันไปตามชนิดของสายพันธุ์ บางสายพันธุ์ใบสีเขียว บางสายพันธุ์ใบสีม่วง เป็นต้น

รูปที่ 2.4 ใบข้าวโพด (ที่มา : https://shorturl.asia/mSTbO)

4) ดอก มีช่อดอกตัวผู้และตัวเมียอยู่บนต้นเดียวกันแต่แยกกันอยู่คนละตำแหน่ง ช่อดอกตัวผู้อยู่ที่ส่วนยอดของลำต้น มีดอกย่อยเล็ก ๆ มีเกสรสีเหลือง และช่อดอกตัวเมียอยู่รวมกันเป็นช่อ ลักษณะทรงกรวยยาว มีกาบบาง ๆ สีเขียว หลายชั้นล้อมรอบ มีเส้นคล้ายเส้นไหมยาว มีสีน้ำตาล สีม่วงอ่อน หรือสีเหลืองส้ม อยู่ด้านบนเป็นกระจุก ก้านช่อดอกสั้น ดอกออกตามกาบของใบและลำต้น

(A) (B)

รูปที่ 2.5 ช่อดอกตัวผู้ (A) และช่อดอกตัวเมีย (B) ของข้าวโพด
(ที่มา : https://shorturl.asia/Vt0fd และ https://shorturl.asia/oxlmP)

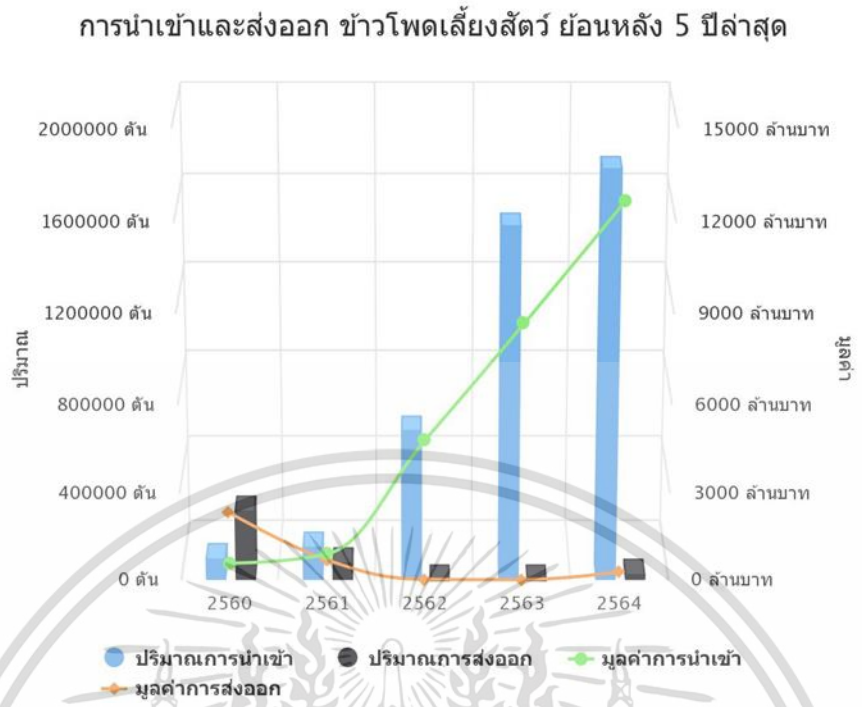
5) ผลและเมล็ด มีลักษณะทรงกระบอก หุ้มด้วยกาบบาง ๆ หลายชั้นรอบฝัก ฝักอ่อนมีสีเขียว ฝักแก่กาบจะแห้ง มีสีน้ำตาล และมีเมล็ดเรียงอยู่สม่ำเสมอรอบแกนกลางของฝัก มีเยื่อหุ้มเมล็ดผิวเรียบบางใส มีสีนวล สีเหลือง สีขาว หรือสีม่วงดำ ตามสายพันธุ์

รูปที่ 2.6 ผลและเมล็ดของข้าวโพด
(ที่มา : https://www.baanjomyut.com/library_3/extension-2/corn/index.html)

2.1.2 สถานการณ์การผลิตข้าวโพดในประเทศไทย

ข้าวโพดเป็นพืชเศรษฐกิจที่สำคัญของไทยที่มีการใช้บริโภคภายในประเทศและส่งออก ซึ่งช่วยสร้างรายได้ให้แก่เกษตรกร โดยเฉพาะอุตสาหกรรมอาหารสัตว์ โดยแต่ละปีประเทศไทยมีการส่งออกอาหารสัตว์เพิ่มขึ้น นอกจากนี้ ข้าวโพดยังเป็นวัตถุดิบสำคัญของอุตสาหกรรมที่เป็นมิตรกับสภาพแวดล้อม เช่น พลาสติกชีวภาพ และเอทานอล และสามารถนำไปแปรรูปเป็นผลิตภัณฑ์ต่าง ๆ เพื่อสร้างมูลค่าเพิ่มในอุตสาหกรรมอื่น ๆ เช่น อุตสาหกรรมแป้งข้าวโพด น้ำมันข้าวโพด (กรมเจรจาการค้าระหว่างประเทศ, 2565)

จากการสำรวจของสำนักงานเศรษฐกิจการเกษตรพบว่า ปี พ.ศ. 2564 ทั่วประเทศไทยมีพื้นที่การปลูกข้าวโพดโพดประมาณ 13,680,000 ไร่ โดยแบ่งเป็นปลูกในภาคเหนือ 9,400,00 ไร่ ภาคตะวันออกเฉียงเหนือ 2,440,00 ไร่ และภาคกลาง 1,840,000 ไร่ รวมผลผลิตในประเทศได้ประมาณ 9,800,000 ตัน จากสถิติภาวะการนำเข้าและส่งออกข้าวโพดเลี้ยงสัตว์ย้อนหลัง 5 ปี มีแนวโน้มเพิ่มขึ้นอย่างต่อเนื่อง แสดงให้เห็นว่าการปลูกข้าวโพดเพียงอย่างเดียวไม่เพียงพอต่อความต้องการภายในประเทศ จึงจำเป็นต้องมีการนำเข้าข้าวโพดเพื่อใช้เลี้ยงสัตว์เพิ่มเติม (สำนักเศรษฐกิจการเกษตร, 2565)



รูปที่ 2.7 สถิติการนำเข้าและส่งออก ข้าวโพดเลี้ยงสัตว์ของประเทศไทย ตั้งแต่ปี พ.ศ. 2561-2564 (ที่มา : https://mis-app.oae.go.th/product/ข้าวโพดเลี้ยงสัตว์)

2.2 การประมวลภาษาธรรมชาติ (Natural Language Processing; NLP)

การประมวลภาษาธรรมชาติ เป็นวิทยาการแขนงหนึ่งในหมวดหมู่ของเทคโนโลยีปัญญาประดิษฐ์หรือ Artificial Intelligence ซึ่งช่วยให้คอมพิวเตอร์สามารถเข้าใจภาษามนุษย์ตลอดจนตีความและใช้งานภาษาปกติที่มนุษย์ใช้สื่อสารได้ (SAS, 2563)

2.2.1 วิวัฒนาการของการประมวลภาษาธรรมชาติ

วิทยาการด้านการประมวลภาษาธรรมชาตินั้นไม่ใช่ศาสตร์ที่เพิ่งเกิดขึ้นใหม่ อย่างไรก็ตามความก้าวหน้าและนวัตกรรมใหม่ ๆ ก็กำลังเกิดขึ้นในสาขานี้อย่างต่อเนื่อง อันเป็นผลมาจากความสนใจด้านปฏิสัมพันธ์ระหว่างมนุษย์และอุปกรณ์ทางคอมพิวเตอร์ รวมไปถึงความก้าวหน้าของข้อมูลมหัต (Big Data) ตลอดจนความสามารถในการประมวลผลและอัลกอริทึมที่มีความทันสมัย (Copestake, 2004) มนุษย์มีภาษาเป็นของตนเอง เช่น ภาษาอังกฤษ ภาษาจีน หรือภาษาไทย เป็นต้น แต่ภาษาที่คอมพิวเตอร์ใช้ในการทำงานต่าง ๆ นั้น แตกต่างออกไปจากภาษาของมนุษย์ ซึ่งเป็นภาษาที่เรียกว่า รหัสเครื่อง (Machine code) หรือภาษาเครื่อง (Machine language) ซึ่งเป็นภาษาที่มนุษย์ส่วนมากไม่สามารถตีความได้ การทำงานทุกอย่างของอุปกรณ์นั้นล้วนแต่ประกอบขึ้นจากกระบวนการในรูปรหัส 0 และ 1 จำนวนนับล้าน ๆ รายการ ที่ถูกตีความและแปลงผลให้กลายเป็นการตอบสนองที่มีเหตุผล

ปัจจุบันนี้การออกคำสั่งแก่อุปกรณ์คอมพิวเตอร์เป็นเรื่องที่ง่ายดายอย่างยิ่ง เช่น สามารถบอกอุปกรณ์ว่า "Alexa ฉันชอบเพลงนี้" จากนั้นอุปกรณ์ที่สามารถเล่นเพลงจะตอบสนองความต้องการได้ เช่น การลดระดับเสียงลง และตอบด้วยคำพูดและน้ำเสียงที่เหมือนมนุษย์ว่า "โอเค บันทึกการจัดอันดับของคุณไว้แล้ว" จากนั้น มันจะปรับอัลกอริทึมในตัวของมันเองเพื่อเล่นเพลง ๆ นั้น และเพลงอื่น ๆ ที่อาจคล้ายคลึงกันในครั้งต่อ ๆ ไปที่คุณฟังเพลงจากช่องที่เล่นดนตรีช่องดังกล่าวอีก

เมื่อพิจารณาการมีปฏิสัมพันธ์ระหว่างมนุษย์และระบบคอมพิวเตอร์ให้ละเอียดยิ่งขึ้นนั้น จะเห็นว่าอุปกรณ์ทำงานเมื่อได้ยินเสียงของคุณและถ้อยคำที่คุณพูด และเข้าใจถึงเจตนาในการพูดของคุณแม้ว่าคุณจะไม่ได้พูดถึงเจตนานั้นโดยตรง จากนั้นมันจึงทำงานบางอย่างและตอบสนองกลับมาแก่คุณเป็นภาษาอังกฤษที่สละสลวย ซึ่งกระบวนการทั้งหมดนี้กินเวลาเพียงประมาณห้าวินาทีเท่านั้น ซึ่งการทำงานของอุปกรณ์ทั้งหมดที่กล่าวมานี้ เกิดขึ้นได้ด้วยการประมวลภาษาธรรมชาติ (Natural Language Processing; NLP) รวมถึงขีดความสามารถอื่น ๆ ของ AI เช่น การเรียนรู้ของเครื่อง (Machine Learning) และการเรียนรู้เชิงลึก (Deep Learning) เป็นต้น (SAS, 2563)

2.2.2 ความสำคัญของการประมวลภาษาธรรมชาติ

การประมวลภาษาธรรมชาตินั้นช่วยให้อุปกรณ์คอมพิวเตอร์ สามารถสื่อสารกับมนุษย์ได้ด้วยการใช้งานภาษาของเครื่องเอง และดำเนินการทำงานต่าง ๆ ที่เกี่ยวข้องกับภาษาได้ ยกตัวอย่างเช่น ช่วยให้อุปกรณ์และคอมพิวเตอร์สามารถอ่านอักขระภาษาปกติ หรือทำความเข้าใจและตีความคำพูดของมนุษย์ ไปจนถึงการวัดอารมณ์ ความรู้สึกที่แฝงอยู่ในข้อความเหล่านั้นและกลั่นกรองใจความหรือนัยยะที่สำคัญออกมาเพื่อใช้งาน

ระบบที่ทันสมัยในปัจจุบันสามารถวิเคราะห์ข้อมูลในปริมาณมหาศาลเกินกว่าขีดความสามารถของมนุษย์ โดยตัดข้อจำกัดเรื่องความเหน็ดเหนื่อยออกไป และสามารถทำงานด้วยความแม่นยำ คงเส้นคงวา และปราศจากอคติ การทำงานในปัจจุบัน มักต้องรับมือกับข้อมูลดิบจำนวนมหาศาล ซึ่งเกิดขึ้นอย่างต่อเนื่องในแต่ละวัน ไม่ว่าจะเป็นการทำงานในด้านประวัติคนไข้และทางการแพทย์ ไปจนถึงข้อมูลจากโซเชียลมีเดีย ซึ่งการทำงานโดยอัตโนมัติจาก AI จะเป็นกุญแจสำคัญในการวิเคราะห์ข้อมูลเหล่านี้ได้ ไม่ว่าจะเป็นข้อมูลในรูปแบบข้อความหรือคำพูด

เนื่องจากภาษาที่มนุษย์ใช้นั้น มีความซับซ้อนและหลากหลายอย่างยิ่ง เพราะมนุษย์มีวิธีการแสดงออกมากมายนับไม่ถ้วน ทั้งในด้านการสื่อสารด้วยคำพูดหรือข้อความที่เกิดขึ้นด้วยการเขียน นอกจากการมีภาษานับร้อย ๆ พัน ๆ ภาษา ซึ่งต่างมีภาษาถิ่นแยกย่อยลงไปอีกนั้น ทุกภาษายังทวีความซับซ้อนยิ่งขึ้นไปอีกด้วยการมีชุดไวยากรณ์และโครงสร้างทางภาษาเฉพาะตัวของตนเอง รวมถึงคำ กลุ่มคำ และแม้แต่ศัพท์แสลงต่าง ๆ และเมื่อมนุษย์ใช้ภาษาในการสื่อสารกันนั้น มนุษย์ยังมักนิยมเขียนข้อความในรูปแบบย่อ ละเครื่องหมายวรรคตอนออกไป หรือแม้แต่การสะกดคำผิด

แม้ว่าเทคนิคการทำงานทั้งแบบ การเรียนรู้แบบมีผู้สอน (Supervised Learning) และการเรียนรู้แบบไม่มีผู้สอน (Unsupervised Learning) โดยเฉพาะอย่างยิ่งกระบวนการทำงานแบบการ

เรียนรู้เชิงลึก (Deep Learning) จะได้ถูกนำมาใช้งานอย่างแพร่หลายในการสร้างแบบจำลองวิเคราะห์ภาษาของมนุษย์แล้วก็ตาม ก็ยังคงมีความจำเป็นในการสร้างความเข้าใจทางภาษาศาสตร์ที่ลึกและซับซ้อนยิ่งขึ้น รวมถึงความรู้ความเข้าใจเฉพาะด้าน ซึ่งแตกแขนงความชำนาญย่อยออกไปจากเทคนิคการเรียนรู้ของเครื่อง (Machine Learning) ตามปกติอีกด้วย ด้วยเหตุนี้การประมวลภาษาธรรมชาติจึงมีความสำคัญในการลดความสับสนทางการวิเคราะห์ภาษาลง และเพิ่มมิติให้แก่ข้อมูลในรูปของตัวเลข เพื่อการนำไปใช้งานต่าง ๆ ต่อไป

2.2.3 กระบวนการทำงานของการประมวลภาษาธรรมชาติ

การประมวลภาษาธรรมชาติ ประกอบด้วยหลากหลายวิธีการประมวลผลและแปลความหมายของภาษาปกติของมนุษย์ โดยมีขั้นตอนดังต่อไปนี้

2.2.3.1 การเตรียมข้อมูล (Data Preprocessing)

ตัวอย่างเครื่องมือที่ใช้ในการเตรียมข้อมูลสำหรับการทำงานของการประมวลภาษาธรรมชาติ (ณัฐโชติ, 2563) ได้แก่

1) การตัดคำ (Tokenization) คือการแบ่งคำออกเป็นคำ ๆ อย่างถูกต้องตามหลักภาษา เช่น ภาษาอังกฤษจะใช้ช่องว่างระหว่างคำ ในการแบ่งคำออกเป็นคำ ๆ

2) การกำจัดคำฟุ่มเฟือย (Stop Words) คือ การกำจัดคำที่ไม่สำคัญ เช่น คำเชื่อม

3) การลดความซ้ำซ้อนของคำ (Lemmatization/Stemming) คือ การแปลงคำให้อยู่ในรูปแบบดั้งเดิม (Lemmatization) เช่น is, am, are, is, was เปลี่ยนเป็น be และการตัดส่วนขยาย (Stemming) ของคำจะทำการตัดบางส่วนของคำทิ้ง เช่น s, es, ing หรือ ed ตัวอย่างเช่น hopes, hoping, hoped จะเปลี่ยนเป็น hope

4) การกำหนดรูปแบบหรือกลุ่มคำ (Regular Expression) การตัดตัวอักษรพิเศษที่ไม่ใช่ข้อความทิ้งตามรูปแบบหรือกลุ่มคำที่กำหนด เช่น “Give 100%!” เป็น “Give 100”

5) การลบช่องว่างระหว่างคำ (White space) เช่น “\t Hello NLP \t” เป็น “Hello NLP”

2.2.3.2 การแปลงเชิงปริมาณ (Text Representation)

คือ การแปลงข้อความ (Text) ให้กลายเป็นตัวเลข (Numerical) ที่อยู่ในรูปแบบของเวกเตอร์ (Vector) ที่เหมาะกับการนำไปเข้าการวิเคราะห์ตัวแบบ ซึ่งการแปลงเชิงปริมาณที่ใช้ในงานวิจัยนี้มี 2 วิธี ดังนี้

1) ถุงคำ (Bag of Words) คือ เป็นวิธีการในการสร้างคุณลักษณะของข้อความขึ้นมาโดยใช้หลักการของการเข้ารหัสแบบวันฮอท (One-Hot Encoding) (Zhang et al., 2010) โดยแทนค่า 1 เมื่อคำเหล่านั้นปรากฏขึ้นในชุดข้อมูล และแทนค่า 0 เมื่อคำเหล่านั้นไม่ปรากฏในชุดข้อมูล

	corn	export	fall	futures	hopes	on	profit	rise	taking
corn futures fall on profit taking	1	0	1	1	0	1	1	0	1
corn futures rise on export hopes	1	1	0	1	1	1	0	1	0

รูปที่ 2.8 ตัวอย่างการสร้างคุณลักษณะของข้อความของ Bag of Words

การใช้งาน Bag of Words นั้นยังติดปัญหาในส่วนของการนับจำนวนคำศัพท์อย่างเดียว แต่ไม่ได้นับในส่วนของจำนวนเอกสารที่คำนั้นปรากฏ ซึ่งทำให้ติดปัญหาในส่วนของคำที่ปรากฏเป็นจำนวนมากในเอกสารเพียงแค่ชุดเดียวหรือจำนวนน้อย ๆ ทำให้การวิเคราะห์ ข้อมูลออกมาแล้วสูญเสียความรู้บางส่วนในชุดข้อมูล จึงได้มีการคิดค้นอัลกอริทึมขึ้นมาใช้ในการแก้ปัญหาของ Bag of Words โดยมีชื่อว่า Term Frequency - Invert Document Frequency

2) เทคนิคการคัดแยกคำตามความสำคัญ (Term Frequency - Inverse Document Frequency; TF-IDF) เป็นเทคนิคที่ได้มีการพัฒนามาจาก Bag of Words มีจุดเด่นที่เหนือกว่าในส่วนที่ TF-IDF ไม่เพียงแต่นับจำนวนคำศัพท์ที่สามารถพบมากที่สุดในชุดข้อมูลเท่านั้น แต่ยังมีการวิเคราะห์ไปถึงความสำคัญของคำที่ปรากฏอยู่ในชุดข้อมูลอีกด้วย (Ramos, 2003) ซึ่งเทคนิคนี้จะมีองค์ประกอบอยู่สองส่วนด้วยกัน (บุษบงก์ และคณะ, 2564)

- **Term Frequency (TF)** มีแนวคิดว่าหากคำไหนถูกกล่าวถึงบ่อย ๆ ในเอกสารนั้น ๆ มีความเป็นไปได้สูงที่จะเกี่ยวข้องกับความสำคัญของเอกสารนั้นมาก ๆ มีสมการ ดังนี้

$$TF_{ij} = \frac{f_{ij}}{n_{ij}} \tag{2.1}$$

เมื่อ f_{ij} แทนจำนวนความถี่ของคำ i ในข้อความ j

n_{ij} แทนจำนวนคำทั้งหมดในข้อความ j

- **Inverse Document Frequency (IDF)** เป็นการคำนวณค่าน้ำหนักความสำคัญของแต่ละคำที่พบในเอกสารหากพบคำนั้นบ่อยจะมีค่า IDF ต่ำโดยมีสูตรการคำนวณ ดังนี้

$$IDF_{ij} = 1 + \log\frac{N}{c_i} \tag{2.2}$$

เมื่อ N แทนจำนวนข้อความทั้งหมด

c_i แทนจำนวนข้อความที่มีคำ i ปรากฏอยู่ในข้อความ

จะได้การคำนวณเทคนิคการคัดแยกคำตามความสำคัญ (TF-IDF) ดังนี้

$$TFIDF = TF \times IDF \tag{2.3}$$

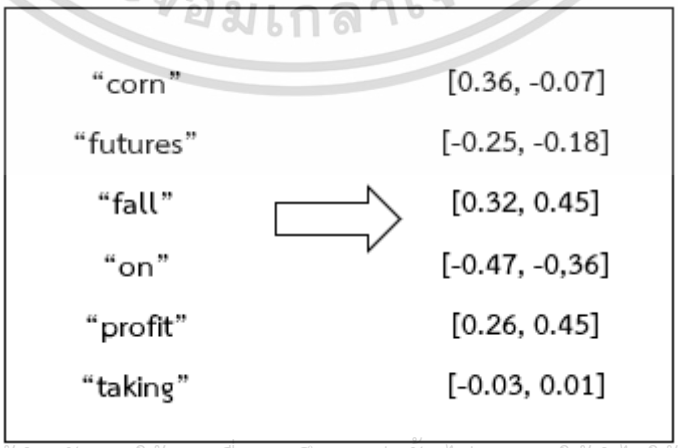
	corn	export	fall	futures	hopes	on	profit	rise	taking
corn futures fall on profit taking	0.3347	0.0000	0.4704	0.3347	0.0000	0.3347	0.4704	0.0000	0.4704
corn futures rise on export hopes	0.3347	0.4704	0.0000	0.3347	0.4704	0.3347	0.0000	0.4704	0.0000

รูปที่ 2.9 ตัวอย่างการสร้างคุณลักษณะของข้อความของ TF-IDF

2.2.3.3 คำฝังตัว (Word Embedding)

คำฝังตัว (Word Embedding) คือ การสร้างเวกเตอร์คุณลักษณะ (Feature Vector) ขึ้นมาจาก Token ที่เราได้ทำการสร้างไว้ โดยทำการสร้างเวกเตอร์คุณลักษณะขึ้นมาจากประโยคหรือเอกสารที่มีอยู่ในข้อมูลของเราเพื่อทำการสร้างคุณลักษณะที่อยู่ในรูปของตัวเลขที่สามารถนำไปใช้คำนวณความคล้ายคลึงกับคำอื่น ๆ ในบริบทของคำที่แตกต่างกันได้ (Ganguly et al., 2015)

คำฝังตัว (Word Embedding) จะทำการสร้างเวกเตอร์คุณลักษณะโดยเริ่มจากการเข้ารหัสคำแต่ละคำให้อยู่ในรูปที่คอมพิวเตอร์สามารถทำความเข้าใจได้ก่อนด้วยวิธีการเข้ารหัสแบบวันฮอท (One-Hot Encoding) จะทำงานโดยการนำจำนวนคำที่ปรากฏขึ้นมาในชุดข้อมูล และนำประโยคในชุดข้อมูลหรือเอกสารที่เรา ได้ทำการกำหนดไว้ในชุดข้อมูลมาเข้ารหัส ทำให้ได้เวกเตอร์ตามจำนวนคำในประโยคที่กำหนดให้อยู่ในรูปของบิต จากนั้นทำการรวมเวกเตอร์ที่ได้จากการเข้ารหัสแบบวันฮอท (One-Hot Encoding) ซึ่งสามารถกำหนดจำนวนมิติ (Dimension) หรือคุณลักษณะ (Features) ของเวกเตอร์ (Vector) ได้ดังนี้



รูปที่ 2.10 ตัวอย่างการสร้างเวกเตอร์คุณลักษณะ ในรูปแบบ 2 มิติ

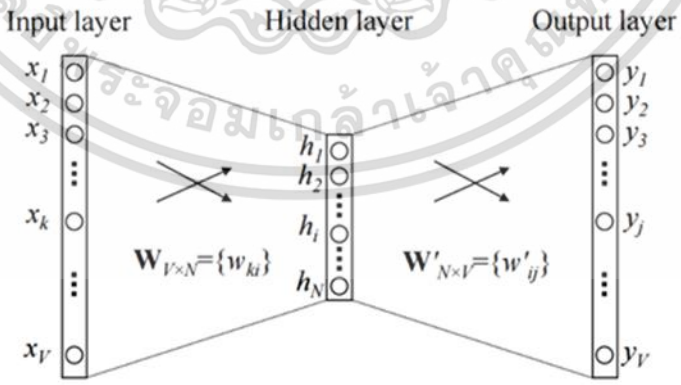
ในปัจจุบันได้มีการพัฒนาแบบจำลองคำฝังตัวต่าง ๆ ขึ้นมาให้เลือกใช้งานมากมาย ซึ่งแต่ละแบบจำลองจะมีจุดเด่นที่ไม่เหมือนกัน โดยผู้วิจัยได้เลือกแบบจำลอง เวิร์ดทูเวก (Word2Vec) มาใช้ในการวิจัยครั้งนี้

2.2.3.4 เวิร์ดทูเวก (Word2Vec)

เป็นวิธีคำฝังตัว (Word Embedding) แบบแรกที่ได้ทำการพัฒนาขึ้นโดย Google โดยมี Mikolov เป็นหัวหน้าในการพัฒนา (Mikolov et al, 2013) โดย Word2Vec ได้รับการออกแบบโครงสร้างให้มี โครงข่ายประสาทเทียม 2 ชั้น โดยมีการนำข้อมูลจำนวนมากมาใช้ในการฝึกสอนให้ได้เวกเตอร์คุณลักษณะออกมา โดยถูกปล่อยออกมาให้ได้ใช้งานครั้งแรกในปี ค.ศ. 2013 Word2Vec ส่วนมากถูกนำไปใช้กับงานทางด้านแบบจำลองทางภาษา (Language Modeling) ซึ่งเป็นแบบจำลองที่ใช้ในการทำนายคำถัดไปของประ โยคว่าควรที่จะเป็นคำใด โดยอาศัยหลักการของ Continuous Bag of Words (CBOW) สำหรับการใช้คำหลาย ๆ คำต่อกัน เพื่อทำนายคำที่อยู่ถัดไป และ Continuous Skip-Gram สำหรับการใช้คำหนึ่งคำในการทำนายคำอื่น ๆ ที่มีโอกาสเป็นคำถัดไปจากคำนี้

- **หลักการของ Continuous Bag of Words (CBOW)**

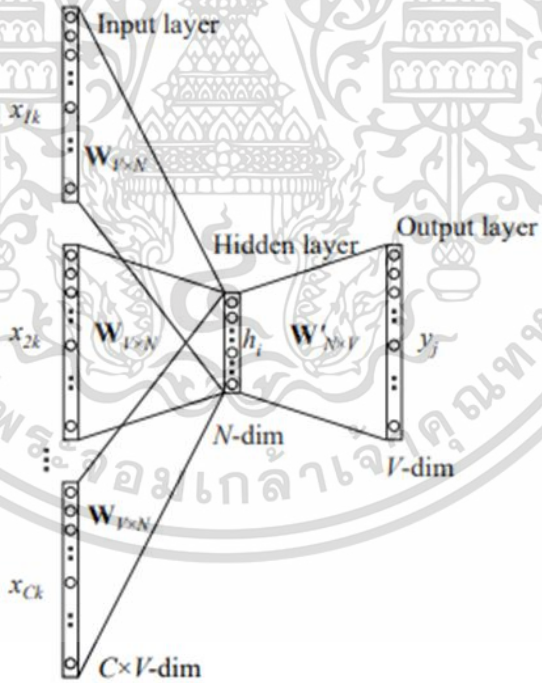
การใช้งานโมเดล Continuous Bag of Words (CBOW) เป็นการสร้างโครงข่ายสมองแบบตื้น (Shallow Neural Network) โดยใช้เวกเตอร์ของคำศัพท์ (Word Vector) เป็นชั้นข้อมูลนำเข้า (Input Layer) และเชื่อมต่อกับชั้นซ่อน (Hidden Layer) 1 ชั้น จำนวนโหนดสามารถกำหนดได้ตามต้องการ โดยทั่วไปจะมีจำนวนน้อยกว่าจำนวนมิติของเวกเตอร์ของข้อมูลนำเข้า จากนั้นนำไปเชื่อมต่อกับชั้นผลลัพธ์ (Output Layer) ที่มีจำนวนโหนดเท่ากับชั้นข้อมูลนำเข้า (Input Layer) และฝึกฝนแบบจำลองด้วยการจำแนกประเภท (Classification)



รูปที่ 2.11 ลักษณะของโครงสร้างของแบบจำลอง CBOW เบื้องต้น (ในรูปใช้คำเพียงคำเดียวเป็นบริบท) (ที่มา : Rong, 2014)

ในการฝึกฝนแบบจำลอง CBOW จะนำเวกเตอร์ของคำศัพท์ (Word Vector) ของคำที่อยู่รอบ ๆ คำที่กำลังพิจารณาซึ่งมีตำแหน่งอยู่ที่ตรงกลาง (Center Word) ของบริบทภายในระยะของบริบท (Context Size) ที่กำหนดมาเป็นข้อมูลนำเข้าในการทำการจำแนกประเภท (Classification) และใช้เวกเตอร์ของคำศัพท์ (Word Vector) ของคำที่กำลังพิจารณาเป็นเป้าหมายในการทำนายผลลัพธ์ สำหรับแต่ละคำในประโยคหรือข้อความที่จะวิเคราะห์

รูปที่ 2.12 ตัวอย่างการทำงานของแบบจำลอง Continuous Bag of Words (CBOW) (ที่มา : https://bigdata.go.th/big-data-101/word2vec/)



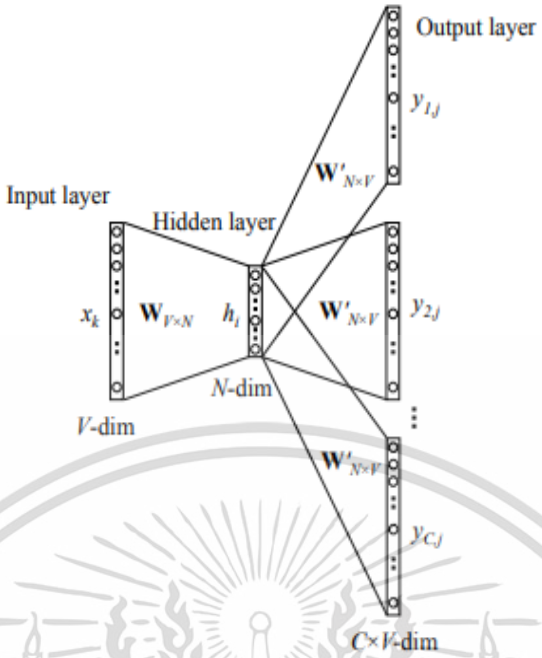
รูปที่ 2.13 แบบจำลอง Continuous Bag of Words (CBOW) (ที่มา : Rong, 2014)

- **หลักการของ Continuous Skip-Gram**

สำหรับการฝึกฝนแบบจำลอง Skip-gram นั้น จะสร้างโครงข่ายสมองแบบตื้น (Shallow Neural Network) ที่มีชั้นข้อมูลนำเข้า (Input Layer) และชั้นผลลัพธ์ (Output Layer) ขนาดเท่ากัน และมีชั้นซ่อน (Hidden Layer) เพียงชั้นเดียวเหมือนกับแบบจำลอง Continuous Bag of Words (CBOW) โดยการฝึกฝนจะใช้คำศัพท์ (Word) เดียวเพื่อทำนายคำที่เป็นบริบทของคำนั้น ๆ ในขณะที่แบบจำลอง CBOW ใช้เวกเตอร์ของคำศัพท์ (Word Vector) ของคำที่อยู่รอบ ๆ คำนั้น ๆ มาใช้ในการทำนายผลลัพธ์ของคำนั้น ๆ

รูปที่ 2.14 ตัวอย่างการทำงานของแบบจำลอง Continuous Skip-Gram
(ที่มา : https://bigdata.go.th/big-data-101/word2vec/)

การฝึกฝนแบบจำลองนี้จะใช้เวกเตอร์ของคำศัพท์ (Word Vector) ของคำในการทำนายคำที่เป็นบริบทของคำนั้น โดยเลือกใช้คำกลาง (Center Word) เพื่อทำนายการกระจายตัว (Probability distribution) ของคำที่น่าจะเป็นบริบทของคำนี้ นั่นคือการทำนายความน่าจะเป็นของคำต่างๆ ที่อยู่ในบริบทเดียวกันกับคำกลาง โดยใช้แบบจำลองโครงข่ายประสาทเทียม (Neural Network) ที่มีชั้นซ่อน (Hidden Layer) เพียงหนึ่งชั้น โดยชั้นข้อมูลนำเข้า (Input Layer) และชั้นผลลัพธ์ (Output Layer) จะมีขนาดเท่ากัน และจะนำเวกเตอร์ของคำศัพท์ (Word Vector) ของคำนั้นมาใช้ในการฝึกฝน โดยทำการปรับค่าถ่วงน้ำหนัก (Weight) ของโมเดลเพื่อให้โมเดลสามารถทำนายคำต่างๆ ในบริบทของคำกลางได้ถูกต้อง (ปฏิภาณ และพีรดล, 2564)



รูปที่ 2.15 แบบจำลอง Continuous Skip-Gram (ที่มา : Rong, 2014)

2.3 การเรียนรู้ของเครื่อง (Machine Learning)

เป็นส่วนหนึ่งของปัญญาประดิษฐ์ (Artificial Intelligence; AI) ที่ช่วยให้ระบบสารสนเทศรู้จักอัลกอริทึมและชุดข้อมูลต่าง ๆ เพื่อการเรียนรู้แบบอัตโนมัติผ่านข้อมูล และประสบการณ์ด้วยตนเอง เพื่อทำการค้นหา จำแนก สรุปผล พยากรณ์ และพัฒนากระบวนการแก้ไขปัญหาได้เหมาะสม โดยการเรียนรู้ของเครื่องมีหลากหลายวิธีเรียนรู้ (มีหลายเทคนิค) แต่ทั้งนี้สามารถจำแนกได้เป็น 3 หมวดหมู่ใหญ่ (จตุรพัชร์, 2562) ดังนี้

2.3.1 การเรียนรู้แบบมีผู้สอน (Supervised Learning)

วิธีนี้จะใช้แบบจำลองที่สร้างจากคณิตศาสตร์มาเรียนรู้ปัญหาจากชุดข้อมูลในการฝึกฝน (Train data) ที่รู้ผลลัพธ์ล่วงหน้าของชุดข้อมูลนั้น ๆ อยู่แล้ว ซึ่งการเรียนรู้แบบมีผู้สอน (Supervised Learning) สามารถแยกออกมาได้ 2 วิธีเรียนรู้ย่อย ได้แก่

1) การวิเคราะห์การถดถอย (Regression) วิธีนี้จะใช้ทำนายผลลัพธ์ที่เป็นข้อมูลเชิงตัวเลข (Numerical) หมายถึงสามารถเป็นตัวเลขได้ ทั้งจำนวนเต็มหรือจำนวนจริง ซึ่งตัวเลขที่ว่านี้จะมีผลในทางคณิตศาสตร์ ซึ่งสามารถบวกลบคูณหารกันได้ โดยจะเป็นเลขที่มีค่าต่อเนื่อง (Continuous) เช่น เลขจำนวนเต็มต่อเนื่อง -8, -3, 1, 3, 5, ... เป็นต้น

2) การจำแนกประเภท (Classification) วิธีนี้จะใช้ทำนายผลลัพธ์ที่เป็นข้อมูลเชิงกลุ่ม (Categorical) หมายถึงข้อมูลที่จัดเป็นหมวดหมู่หรือกลุ่มก้อนแยกประเภทชัดเจน (จำนวนข้อมูลที่แน่นอน ไม่ใช่ตัวเลขที่มีความต่อเนื่อง) ซึ่งจะไม่มีผลในทางคณิตศาสตร์ ไม่มีความหมายในการคำนวณ

ไม่มีผลเวลาบวกลบคูณหารกับตัวเลข เช่น สีขาว, สีดำ, สีแดง, สีน้ำเงิน มีแค่ 4 สีนี้เท่านั้น และนำไปใช้คำนวณทางคณิตศาสตร์ไม่ได้

2.3.2 การเรียนรู้แบบไม่มีผู้สอน (Unsupervised Learning)

วิธีนี้จะใช้แบบจำลองที่สร้างจากคณิตศาสตร์มาเรียนรู้ปัญหาจากชุดข้อมูลในการฝึกฝน (Train data) ที่ไม่มีผลลัพธ์ของชุดข้อมูลนั้น ๆ ด้วยตัวเอง ซึ่งการเรียนรู้แบบไม่มีผู้สอน (Unsupervised Learning) สามารถแยกออกมาได้ 3 วิธีเรียนรู้ย่อย ได้แก่ 1) การจับกลุ่ม (Clustering) เช่น การแบ่งกลุ่มลูกค้า ระบบแนะนำผลิตภัณฑ์ เป็นต้น 2) การค้นหารูปแบบ (Pattern Search) 3) การลดมิติของข้อมูล (Dimension Reduction)

2.3.3 การเรียนรู้แบบเสริมแรง (Reinforcement Learning)

วิธีนี้ได้แนวคิดมาจากพฤติกรรมทางจิตวิทยา (Behavioral Psychology) ซึ่งคำว่า “Reinforcement” ในทางจิตวิทยาแปลว่า เสริมแรง หรือ แรงจูงใจ โดยตามทฤษฎีจะมองการเสริมแรงออกเป็น 2 ด้าน ได้แก่

1) การเสริมแรงทางบวก (Positive reinforcement) จะให้ผลตอบแทนในสิ่งที่คนอยากได้ เช่น เพิ่มเงินเดือน ให้โบนัส เลื่อนตำแหน่ง เป็นต้น

2) การเสริมแรงทางลบ (Negative reinforcement) จะให้ผลตอบแทนที่คนไม่อยากได้ เช่น ตัดเงินเดือน งดให้โบนัส ลดตำแหน่ง เป็นต้น

ซึ่งการเสริมแรงจะมีแรงจูงใจให้คนเกิดพฤติกรรมใหม่ ๆ หรือทำให้เขามีพฤติกรรมซ้ำ ๆ หรือเลิกพฤติกรรมที่ไม่ต้องการนั้นก็คือ การเสริมแรงสามารถปรับพฤติกรรมของคนได้ และในการเรียนรู้แบบเสริมแรง (Reinforcement Learning) ก็ได้นำแนวคิดเสริมแรงในทางจิตวิทยามาสอนให้คอมพิวเตอร์ฉลาด เช่น ปล่อยให้คอมพิวเตอร์ได้ลองผิดลองถูก พร้อมให้รางวัลเมื่อทำถูก (เสริมแรงทางบวก) หรืออาจมีบทลงโทษเมื่อทำผิด (เสริมแรงทางลบ) เพื่อจูงใจให้คอมพิวเตอร์ทำงานบางอย่างให้กับเรา

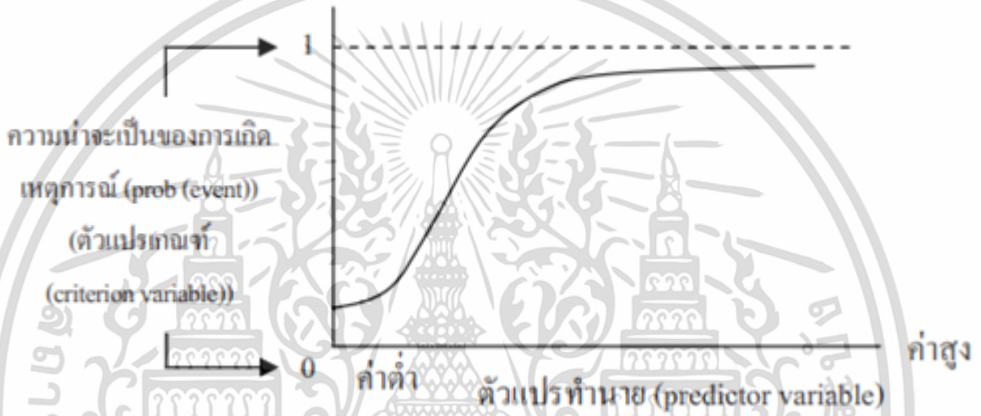
2.4 การจำแนกประเภท (Classification)

เป็นแบบจำลองประเภทการเรียนรู้แบบมีผู้สอน (Supervised Learning) หมายถึงแบบจำลองที่ต้องมีตัวแปรตามเป็นตัวตั้งต้นให้เรียนรู้ โดยเป้าหมายของการจำแนกประเภทจะมีข้อมูลเชิงกลุ่ม (Categorical) หมายถึงข้อมูลที่จัดเป็นหมวดหมู่หรือกลุ่มก้อนแยกประเภทชัดเจน (จำนวนข้อมูลที่แน่นอน ไม่ใช่ตัวเลขที่มีความต่อเนื่อง) ซึ่งจะไม่มีผลในทางคณิตศาสตร์ ไม่มีความหมายในการคำนวณ ไม่มีผลเวลาบวกลบคูณหารกับตัวเลข เช่น ใช่หรือไม่ เพศชายหรือหญิง เป็นต้น และนำไปใช้คำนวณทางคณิตศาสตร์ไม่ได้ ซึ่งสามารถประเมินผลที่ได้จากแบบจำลองการจำแนกประเภทข้อมูล (Classification Model) โดยการวัดค่าความถ่วงดุล (F1 - Score) ค่าความแม่นยำ (Accuracy) ค่าความเที่ยงตรง (Precision) ค่าความไว หรือค่าระลึก (Recall) เป็นต้น จากการใช้เมทริกซ์ความสับสน (Confusion Matrix) ในงานวิจัยครั้งนี้ผู้วิจัยเปรียบเทียบประสิทธิภาพของแบบจำลองในการ

จำแนกประเภทข้อมูลทั้งหมด 3 แบบจำลอง ดังนี้ 1) การถดถอยแบบลอจิสติก (Logistic Regression) 2) ซัพพอร์ตเวกเตอร์แมชชีน (Support Vector Machine; SVM) 3) โครงข่ายประสาทเทียม (Artificial Neural Network; ANN)

2.4.1 การถดถอยแบบลอจิสติก (Logistic Regression)

การถดถอยแบบลอจิสติก (Logistic Regression) เป็นวิธีการอธิบายความสัมพันธ์ของตัวแปรทำนาย (Predictor variable) กับตัวแปรตอบสนอง (Response variable) ที่เป็นข้อมูลเชิงกลุ่ม (Categorical) มีมาตรวัดแบบนามบัญญัติ (Nominal Scale) หรือมาตรวัดแบบเรียงอันดับ (Ordinal Scale) แตกต่างจากการถดถอยแบบเดิม (Traditional Regression) ในเบื้องต้น (ยุวดี, 2564) ดังนี้



รูปที่ 2.16 ฟังก์ชันลอจิสติก (Logistic Function) (ที่มา : ยุทธ, 2555)

1) Model Logistic Regression คือ Logit Model ในรูป

$$\log\left(\frac{p}{1-p}\right) = w = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_j X_j \qquad (2.4)$$

โดย Log เป็น Natural Log หรือเขียน ln หรือเขียนเป็น Model Logistic Regression คือ

$$p = \frac{1}{1 + e^{-\left(\beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_j X_j\right)}} \qquad (2.5)$$

หรือ

$$p = \frac{1}{1 + e^{-\left(\beta_0 + \sum \beta_j X_j\right)}} \qquad (2.6)$$

เมื่อ ค่า p เป็นค่าความน่าจะเป็น (Probability) ซึ่ง $0 \le p \le 1$ ซึ่งไม่สามารถใช้วิธีกำลังสองน้อยที่สุด (Least Square) ได้

2) ไม่สามารถใช้ F-test และ t-test เพราะไม่สามารถหาความแปรปรวน (Variance) ได้ เนื่องจากไม่เป็นเส้นตรง

3) การทำนายตัวแปรตอบสนอง (Response variable) เมื่อมีค่าใหม่ของตัวแปรทำนาย (Predictor variable) ต้องใช้ค่า เป็นเกณฑ์เพื่อจัดเข้ากลุ่ม (Categories) ของตัวแปรตอบสนอง (Response variable)

กลุ่มซื้อ แทนด้วย 1
และ
กลุ่มไม่ซื้อ แทนด้วย 0
} Y = ตัวแปรตอบสนอง (Response variable)

สมมติมีตัวแปรทำนาย (Predictor variable) คือ X1 = รายได้ (บาท), X2 = อายุ (ปี) ผลลัพธ์ของการวิเคราะห์การถดถอยแบบลอจิสติก (Logistic Regression) ได้

$$p = \frac{1}{1+e^{-(\beta_0+\beta_1 X_1+\beta_2 X_2)}} \tag{2.7}$$

เมื่อแทนค่า X1, X2 ค่าใหม่ เพื่อทำนาย Y จะแทนค่าได้ p โดยอาจตั้งเกณฑ์ว่า ถ้า $0 \le p \le 0.5$ ได้ Y = 0 เป็นกลุ่มไม่ซื้อ ถ้า $0.5 < p \le 1$ ได้ Y= 1 เป็นกลุ่มซื้อ เป็นต้น

2.4.1.1 ประเภทของการถดถอยแบบลอจิสติก (Types of Logistic Regression)

การถดถอยแบบลอจิสติกแบ่งประเภทตามจำนวนกลุ่ม (Categories) และชนิดของตัวแปรตอบสนอง (Response variable) ได้ 2 ประเภทใหญ่ ๆ คือ

1) การถดถอยแบบลอจิสติกทวิ (Binary Logistic Regression) ได้แก่ การถดถอยแบบลอจิสติก (Logistic Regression) ที่ตัวแปรตอบสนอง (Response variable) มีเพียง 2 กลุ่ม เช่น ซื้อหรือไม่ซื้อ ชนะหรือแพ้ รักษาหายหรือไม่หาย ควรลงทุนหรือไม่ควรลงทุน เป็นต้น และจำแนกย่อยได้อีกตาม จำนวนของตัวแปรทำนาย (Predictor variable) ดังนี้

1.1) การถดถอยแบบลอจิสติกทวิ (Binary Logistic Regression) แบบการถดถอยอย่างง่าย (Simple Regression) หมายถึง การถดถอยแบบลอจิสติกที่ตัวแปรตอบสนอง (Response variable) มี 2 กลุ่มและมีตัวแปรทำนาย (Predictor variable) ตัวเดียว

$$p = \frac{1}{1+e^{-(\beta_0+\beta_1 X)}} \tag{2.8}$$

1.2) การถดถอยแบบลอจิสติกทวิ (Binary Logistic Regression) แบบการถดถอยพหุคูณ (Multiple Regression) หมายถึง การถดถอยแบบลอจิสติกที่ตัวแปรตอบสนอง (Response variable) มี 2 กลุ่มและมีตัวแปรทำนาย (Predictor variable) หลายตัวได้

$$p = \frac{1}{1+e^{-\left(\beta_0+\beta_1 X_1+\beta_2 X_2+\ldots+\beta_j X_j\right)}} \tag{2.9}$$

หรือ

$$p = \frac{1}{1+e^{-\left(\beta_0+\sum \beta_j X_j\right)}} \tag{2.10}$$

2) การถดถอยแบบลอจิสติกพหุกลุ่ม (Multinomial Logistic Regression) ได้แก่ การถดถอยแบบลอจิสติกที่ตัวแปรตอบสนอง (Response variable) มีจำนวนมากกว่า 2 กลุ่ม (ตั้งแต่ 3 กลุ่ม ขึ้นไป) ซึ่งจำแนกย่อยไปตามชนิดของตัวแปรทำนาย (Predictor variable)

2.1) การถดถอยแบบลอจิสติกแบบนามบัญญัติ (Nominal Logistic Regression) หมายถึง การถดถอยแบบลอจิสติกที่ตัวแปรตอบสนอง (Response variable) มีจำนวนตั้งแต่ 3 กลุ่มขึ้นไป และ มีมาตรวัดแบบนามบัญญัติ (Nominal Scale) เช่น กลุ่มคนไข้กรุ๊ปเลือด O, A, B, AB กลุ่มผู้รับประทาน อาหารมังสวิรัติ, ไม่ผักเลย, อาหารทั่วไป กลุ่มผู้เรียนคณะวิทยาศาสตร์, คณะศิลปศาสตร์, คณะเกษตรศาสตร์, คณะวิศวกรรมศาสตร์ เป็นต้น

2.2) การถดถอยแบบลอจิสติกแบบเรียงอันดับ (Ordinal Logistic Regression) หมายถึง การถดถอยแบบลอจิสติกที่ตัวแปรตอบสนอง (Response variable) มีจำนวนตั้งแต่ 3 กลุ่มขึ้นไป และมีมาตรวัดแบบเรียงอันดับ (Ordinal Scale) เช่น กลุ่ม อายุมากกว่า 60 ปี, 40-59 ปี, 20-39 ปี, ต่ำกว่า 20 ปี กลุ่มผู้มีความคิดเห็น มาก, ปานกลาง, น้อย เป็นต้น ซึ่งแต่ละกลุ่มเปรียบเทียบคู่กันได้ว่ากลุ่มใดระดับสูงกว่ากัน ทั้งแบบ การถดถอยแบบลอจิสติกแบบนามบัญญัติ (Nominal Logistic Regression) และ การถดถอยแบบลอจิสติกแบบเรียงอันดับ (Ordinal Logistic Regression) มีสมการแบบ Simple หรือ Multiple ขึ้นกับจำนวนตัวแปรทำนาย (Predictor variable) เหมือนกับการถดถอยแบบลอจิสติกทวิ (Binary Logistic Regression)

ความสัมพันธ์ระหว่างตัวแปรอิสระกับตัวแปรตามของการถดถอยลอจิสติกไม่เป็นรูปแบบเชิงเส้น จึงต้องมีการปรับให้อยู่ในรูปของเชิงเส้น ในรูปแบบของ ออดส์ (odds) ซึ่งหมายถึง อัตราส่วนระหว่างโอกาสที่จะเกิดเหตุการณ์ที่สนใจกับโอกาสจะไม่เกิดเหตุการณ์ที่สนใจ จะได้ดังสมการ 2.11

$$Odds = \frac{p}{1-p} \tag{2.11}$$

เมื่อ p คือ โอกาสที่จะเกิดเหตุการณ์ที่สนใจ $p(y=1)$

โดยค่าของ Odds จะเป็นการบอกว่าโอกาสที่จะเกิดเหตุการณ์ที่สนใจเป็นกี่เท่าของโอกาสจะไม่เกิดเหตุการณ์ที่สนใจ การเขียนแบบจำลองลอจิสติก จะอยู่ในรูป Log ของ Odds ซึ่งเรียกว่า Logit หรือ Logistic Response Function โดยจะเขียนอยู่ในรูปดังสมการ 2.12

$$logit = \ln\left(\frac{p}{1-p}\right) \tag{2.12}$$

หรือ

$$logit = b_0 + b_1x_1 + b_2x_2 + ... + b_jx_j \tag{2.13}$$

เมื่อ b_i คือ สัมประสิทธิ์การถดถอย

x_i คือ ตัวแปรอิสระ

เมื่อได้ Logit แล้ว รูปแบบของตัวแปรตามจะสามารถทำนายได้ด้วยแบบจำลองเชิงเส้นตรง และสามารถอธิบายความสัมพันธ์ระหว่างตัวแปรอิสระกับตัวแปรตามได้ว่า เมื่อตัวแปร b_i เพิ่มขึ้น 1 หน่วย หาก b เป็นบวก หมายความว่าค่าออดส์ (Odds) จะเพิ่มขึ้น หาก b_i เป็นลบ หมายความว่าค่าออดส์ (Odds) จะลดลง และค่าหาก b_i เป็น 0 หมายความว่าค่าออดส์ (Odds) ไม่เปลี่ยนแปลง ซึ่งสามารถคำนวณค่าออดส์ที่เปลี่ยนแปลงไปได้ดังสมการต่อไปนี้

ร้อยละค่าออดส์ที่เปลี่ยนแปลงไป $= \left(e^{b_i} - 1\right) \times 100$ (2.14)

2.4.1.2 ข้อตกลงเบื้องต้นที่จำเป็นของการถดถอยแบบลอจิสติก (Assumptions of Logistic Regression)

ข้อกำหนดที่จำเป็นของการถดถอยแบบลอจิสติก ดังนี้

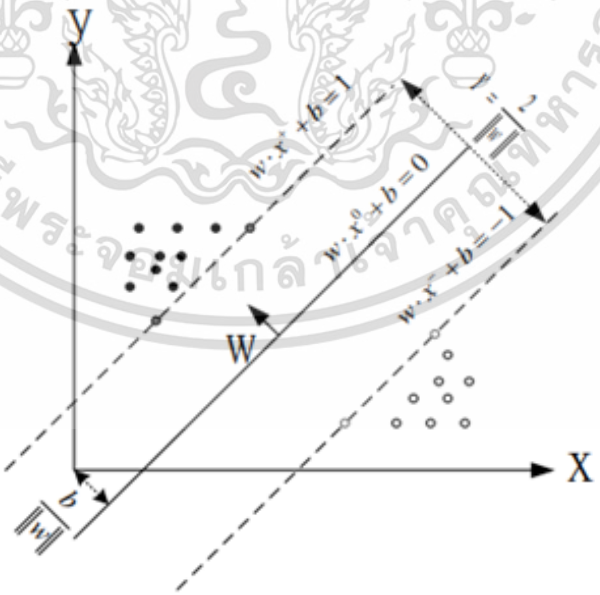
1) ตัวแปรตาม หรือตัวแปรตอบสนอง (Respond variable) เป็นข้อมูลเชิงกลุ่ม (Categories) มีมาตรวัดแบบนามบัญญัติ (Nominal Scale) หรือมาตรวัดแบบเรียงอันดับ (Ordinal Scale) อาจเป็น Binary หรือ Multinomial ก็ได้

2) ตัวแปรอิสระ หรือตัวแปรทำนาย (Predictor variable) ไม่ควรมีความสัมพันธ์ (Correlation) กันสูงเกินไป (ส่วนใหญ่กำหนดว่าไม่ควรเกิน 0.80) ถ้ามีความสัมพันธ์กันต้องหาความสัมพันธ์ร่วม (Covariance)

3) การถดถอยแบบลอจิสติกควรใช้ขนาดตัวอย่าง (Sample sizes) ขนาดใหญ่พอ (ส่วนใหญ่พบว่า Sample sizes ควรมีประมาณ 30 เท่าของจำนวนตัวแปรอิสระ หรือตัวแปรทำนาย (Predictor variable))

2.4.2 ซัพพอร์ตเวกเตอร์แมชชีน (Support Vector Machine; SVM)

ซัพพอร์ตเวกเตอร์แมชชีน (Support Vector Machine; SVM) เป็นอัลกอริทึมการเรียนรู้แบบมีผู้สอน (Supervised Learning) ซึ่งสามารถจัดให้อยู่ในประเภทของอัลกอริทึมเชิงตัวอย่าง อัลกอริทึมนี้ถูกนำเสนอโดย วราดิเมีย แวพนิค (Vadimir Vapnik) ในปี ค.ศ. 1963 (Cortes and Vapnik, 1995) ซัพพอร์ตเวกเตอร์แมชชีนนั้นมีประสิทธิภาพที่สูง และมีการถูกบันทึกไว้เป็นอย่างดีใน (Cristianini and Shawe-Taylor, 2000) อีกทั้งยังถูกนำมาประยุกต์ใช้ในงานหลาย ๆ ประเภท (Pasupa et al. 2019, 2016; Usachokcharoen et al, 2015; Pasupa et al, 2013) โดยหลักของซัพพอร์ตเวกเตอร์แมชชีน คือ การจำแนกข้อมูลออกเป็น 2 ประเภท คือ +1 และ-1 ด้วยระนาบเกิน (Hyperplane) ซึ่งระนาบเกินนั้นจะประกอบไปด้วยขอบที่ทำหน้าที่กั้นข้อมูลแต่ละประเภทออกจากกัน โดยซัพพอร์ตเวกเตอร์แมชชีนจะพยายามที่ทำให้ระยะห่างระหว่างขอบ (Margin) นั้นมีค่ามากที่สุด โดยขอบนั้นจะถูกสร้างจากตัวอย่าง ซึ่งตัวอย่างที่อยู่บนขอบของระนาบจะถูกเรียกว่า "เวกเตอร์สนับสนุน" (Support Vector) (กิติ์สุชาต, 2564) ซึ่งได้ถูกแสดงอยู่ในรูปที่ 2.17



รูปที่ 2.17 ซัพพอร์ตเวกเตอร์แมชชีน (Support Vector Machine; SVM)
(ที่มา : เทอดศักดิ์ และคณะ, 2560)

ซึ่งก่อนที่จะจำแนกข้อมูลจะต้องทำการสอน (Train) ให้เกิดการจดจำข้อมูลของกลุ่มตัวอย่างที่ต้องการจำแนก จากนั้นนำข้อมูลที่ต้องการจำแนกป้อนเข้าสู่ซัพพอร์เวกเตอร์แมชชีน เพื่อให้จำแนกกลุ่มข้อมูลออกมา โดยโครงสร้างข้อมูลสำหรับสอน และผลลัพธ์ที่ออกมาจะทำให้ระบบเกิดการจดจำ ดังสมการ 2.15

$$(x_i, y_i),...,(x_n, y_n) \text{ เมื่อ } x \in R^m, y \in \{+1, -1\} \tag{2.15}$$

เมื่อ $(x_i, y_i),...,(x_n, y_n)$ เป็นคุณลักษณะสำหรับใช้ในการสอน

n คือ จำนวนข้อมูลตัวอย่าง

m คือ จำนวนมิติของข้อมูล

y คือ ผลลัพธ์มีค่าเป็น +1 หรือ -1

ดังนั้นข้อมูลจะถูกจำแนกออกเป็นสองกลุ่ม ดังสมการ 2.16 และ 2.17

$$(w \cdot x) + b > 0 \text{ ถ้า } y_i = +1 \tag{2.16}$$

และ

$$(w \cdot x) + b < 0 \text{ ถ้า } y_i = -1 \tag{2.17}$$

เมื่อ w คือ ค่าถ่วงน้ำหนัก (weight)

b คือ ค่าความเอนเอียง (bias)

y คือ ผลลัพธ์มีค่าเป็น +1 หรือ -1

โดยมีเส้นแบ่งหรือระนาบการตัดสินใจ ซึ่งสามารถคำนวณได้จากสมการที่ 2.18

$$(w \cdot x) + b = 0 \tag{2.18}$$

เวกเตอร์ของข้อมูลที่ป้อนสู่ระบบการสอน เพื่อให้ระบบเรียนรู้ และข้อมูลทั้งสองด้านแบ่งเป็นบวกและลบ ข้อมูลถูกแทนด้วย y ซึ่งประกอบด้วย 2 ค่า คือ y = 1 และ y = -1 แต่ยังตัดสินไม่ได้ว่าเส้นแบ่งใดดีที่สุด ซึ่งวิธีการหาเส้นแบ่งที่ดีที่สุดคือการเพิ่มขอบให้กับเส้นแบ่งทั้งสองด้าน ทำให้ได้เส้นขอบ (Margin) เส้นใหม่ ซึ่งถือว่าเป็นขอบของข้อมูลแต่ละด้าน เส้นของทั้งสองเส้นจะถูกแทนด้วยสมการที่ 2.19 และ 2.20

$$(w \cdot x) + b \geq y \geq 1 \text{ ถ้าอยู่ด้าน } y = +1 \tag{2.19}$$

และ

$$(w \cdot x) + b \leq y \leq 1 \text{ ถ้าอยู่ด้าน } y = -1 \tag{2.20}$$

ถ้าเส้นขอบของเส้นแบ่งใด ๆ มีความกว้างมากที่สุด แสดงว่าข้อมูลทั้ง 2 ชุด มีการแบ่งออกกันอย่างชัดเจน จึงบอกได้ว่าเส้นแบ่งนั้นเป็นระนาบการตัดสินใจที่ดีที่สุด ซึ่งสามารถหาความกว้างของเส้นขอบ (Maximization of margin) ได้จากสมการที่ 2.21 ค่าของ หาได้จากสมการที่ 2.22

$$\begin{aligned} &Maximize \ \ \gamma = \frac{2}{\|w\|} \\ &subject\ to\ y_i\left(w^T x_i + b\right) \geq 1, \forall i \end{aligned} \tag{2.21}$$

โดย

$$w = \sum_{i=1}^{N} \alpha_i y_i x_i \tag{2.22}$$

เมื่อ α คือ สัมประสิทธิ์คงที่ $\alpha_i \geq 0 \ ; i = 1, 2, 3, \ldots, N$

เพื่อความสะดวกในการแก้ปัญหามักนิยมหาค่าน้อยที่สุดมากกว่าการหาค่ามากที่สุด ซึ่งพิจารณาได้จากความสัมพันธ์ต่อไปนี้

$$\gamma = \frac{2}{\|w\|} = \frac{2}{\sqrt{w^T w}} \propto \frac{2}{w^T w} \tag{2.23}$$

ดังนั้นการหาค่าที่เหมาะสมที่สุดเป็นต่อไปนี้

$$\begin{aligned} &Maximize \ \ \frac{1}{2} w^T w \\ &subject\ to\ y_i\left(w^T x_i + b\right) \geq 1, \forall i \end{aligned} \tag{2.24}$$

การหาขอบที่กว้างที่สุดระหว่างข้อมูล 2 กลุ่มจะทำได้ก็ต่อเมื่อสามารถหาระนาบที่สามารถแบ่งข้อมูลทั้ง 2 กลุ่มออกจากกันได้ถูกต้องทั้งหมด เรียกว่า ขอบที่แข็ง (Hard Margin) แต่ความเป็นจริงข้อมูลอาจไม่เป็นเช่นนั้น จึงทำการเพิ่มตัวแปรช่วย (Slack variable) (ξ) เข้าไปเพื่อเพิ่มประสิทธิภาพให้แบบจำลอง และยอมรับค่าสูญเสีย (Loss) ได้ในระดับหนึ่ง ซึ่งเรียกว่า ขอบที่อ่อน (Soft Margin) ซึ่งสามารถสร้างพจน์เพื่อกำหนดปริมาณความผิดพลาดได้โดยใช้ผลรวมของตัวแปรช่วย (Slack variable) (ξ) ดังนี้

$$C\sum_{i=1}^{n}\xi_i \tag{2.25}$$

เมื่อ C คือ ค่าคงที่ซึ่งเป็นพารามิเตอร์ในการกำหนดปริมาณความผิดพลาด

หากมีค่ามาก หมายถึง ยอมให้ความผิดพลาดเกิดได้น้อย ซึ่งหากมีค่ามากอาจจะเกิดปัญหาพอดีเกินไป (Overfitting) ของแบบจำลองได้

หากมีค่าน้อย หมายถึง ยอมให้ความผิดพลาดเกิดได้มาก แต่จะลดปัญหา Overfitting ทำให้สามารถใช้งานกับข้อมูลทั่วไปได้มากกว่า แต่หากน้อยเกินย่อมมีค่าผิดพลาดมากเกินที่จะยอมรับได้

ดังนั้นการเลือกค่า C จะมีผลต่อประสิทธิภาพของแบบจำลอง ซึ่งการเลือกค่าที่เหมาะสมนั้นทำได้ยาก ส่วนใหญ่ผู้ใช้มักเป็นผู้กำหนด เมื่อนำค่าความหย่อน (Slack) มารวมกับปัญหาเดิม จะได้ปัญหาใหม่สำหรับซัพพอร์ตเวเตอร์แมชชีน กรณี ขอบที่อ่อน (Soft Margin) เป็นดังนี้

$$\begin{aligned} &Maximize\ \ \frac{1}{2}w^T w + C\sum_{i=1}^{n}\xi_i \\ &subject\ to\ y_i\left(w^T x_i + b\right) \geq 1, \forall i \end{aligned} \tag{2.26}$$

ฟังก์ชันเคอร์เนล (Kernel Function)

เป็นฟังก์ชันการส่งชนิดหนึ่งที่เกิดจากผลคูณภายในทั้งหมดที่เป็นไปได้ของเวกเตอร์เซ็ตหนึ่ง สามารถเปลี่ยนข้อมูลที่มีมิติต่ำกว่าให้มีมิติสูงขึ้นเพื่อการแบ่งข้อมูล โดยจะอยู่ในรูปดังนี้

$$K(u,v) = \Phi(u)^T \Phi(v) \tag{2.27}$$

โดยฟังก์ชันเคอร์เนลที่นำมาใช้จะต้องสอดคล้องกับเงื่อนไขของเมอร์เซอร์ (Mercer's Condition) ซึ่งจะมีสมบัติต่อเนื่อง (Continuous), สมมาตร (Symmetric) และกึ่งบวกแน่นอน (Positive Semi-Definite) ซึ่งหมายความว่า เมทริกซ์นี้จะไม่มีค่าลักษณะเฉพาะ (Eigenvalue) ที่เป็นลบ โดยฟังก์ชันเคอร์เนลที่นิยมใช้ (ชิตพงษ์, 2563) มีดังต่อไปนี้

1) เส้นตรง (Linear)

$$K(a,b) = a^T b \tag{2.28}$$

2) พหุนาม (Polynomial)

$$K(a,b) = \left(\gamma a^T b + r\right)^b \tag{2.29}$$

3) เกาส์เซียน เรเดียลเบสิสฟังก์ชัน (Gaussian RBF)

$$K(a,b) = e^{\left(-\gamma\|a-b\|^2\right)} \tag{2.30}$$

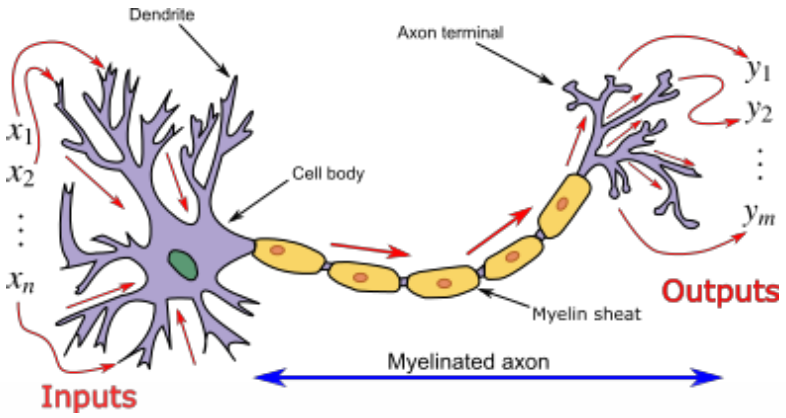
4) ซิกมอยด์ (Sigmoid)

$$K(a,b) = tanh\left(\gamma a^T b + r\right) \tag{2.31}$$

เมื่อ d, γ, a และ b เป็นพารามิเตอร์ของฟังก์ชันเคอร์เนล โดยมีค่าคงที่และขึ้นอยู่กับความเหมาะสม ซึ่งจะนิยมปรับด้วยมือ

2.4.3 โครงข่ายประสาทเทียม (Artificial Neural Network; ANN)

กระบวนการทางคณิตศาสตร์ที่จำลองมาจากการทำงานของเซลล์ประสาทในระบบประสาทของมนุษย์ เพื่อนำมาใช้ในการตัดสินใจ การจำแนก การทำนาย และอื่น ๆ (กริช และคณะ, 2556) โดยโครงข่ายประสาทเทียมประกอบขึ้นจากโหนด (Node) จำนวนมากที่มาเชื่อมต่อกันกลายเป็นโครงข่ายซึ่งบางโหนดสามารถเชื่อมต่อกับสิ่งแวดล้อมภายนอก อีกทั้งยังสามารถทำหน้าที่เป็นข้อมูลเข้า (Input) หรือข้อมูลออก (Output) ได้ ภายในโครงข่ายประสาทเทียมจะมีค่าน้ำหนัก (Weight) เก็บไว้เป็นค่าประจำตัวเปรียบเสมือนความรู้ของโครงข่ายประสาทเทียม ซึ่งการเรียนรู้ของโครงข่ายประสาทเทียมสามารถเกิดขึ้นได้จากการปรับเปลี่ยนค่าน้ำหนักเหล่านั้น

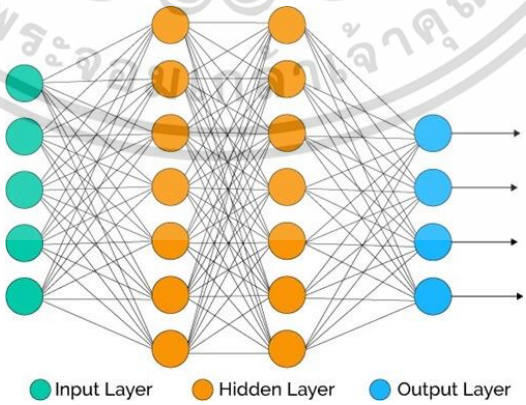


รูปที่ 2.18 เซลล์ประสาทในระบบประสาทของมนุษย์
(ที่มา : https://medium.com/@sarankhotsathian/machine-learning-ann-คืออะไร-3527a9aa0c8c)

โครงข่ายประสาทเทียมนิยมนำมาใช้กับการสร้างแบบจำลองประเภทจำแนกข้อมูล (Classification) ข้อได้เปรียบคือ สามารถพยากรณ์ตัวแปรตามได้ค่อนข้างแม่นยำจากข้อมูลที่มีลักษณะไม่ชัดเจน ไม่เชิงเส้น (Nonlinear) หรือปัญหาที่เกิดขึ้นจริงในธรรมชาติ แม้โครงข่ายประสาทเทียมมีความสามารถในการจำแนกรูปแบบได้ดี แต่เมื่อสร้างแบบจำลองโครงข่ายประสาทเทียมขึ้นมาแล้ว จำเป็นที่จะต้องได้รับการสอน (Train) เพื่อให้แบบจำลองจดจำรูปแบบต่าง ๆ ของข้อมูลตัวอย่างให้ได้ก่อนจึงจะสามารถนำไปใช้งานได้

โครงข่ายประสาทเทียมแบบป้อนไปข้างหน้า (Feed Forward Networks)

โครงข่ายประสาทเทียมที่ใช้ในการจำแนกประเภทข้อมูลเป็นโครงข่ายแบบแบบป้อนไปข้างหน้า ประกอบด้วยเซลล์ประสาทหลายชั้นที่เชื่อมต่อกับแบบสมบูรณ์ (Fully Connect) โดยข้อมูลจะไหลไปในทิศทางเดียว ดังรูปที่ 2.19



รูปที่ 2.19 โครงข่ายประสาทเทียม (Artificial Neural Network; ANN)
(ที่มา : https://www.mindphp.com/บทความ/240-ai-machine-learning/5659-artificial-neural-networks.html)

ประกอบด้วย 3 ส่วนหลักคือ

1) ชั้นนำเข้า หรือชั้นข้อมูลนำเข้า (Input Layer) ทำหน้าที่นำเข้าข้อมูล โดยข้อมูลนี้จะนำประมวลในชั้นถัด ๆ ไป

2) ชั้นซ่อน (Hidden Layer) ทำหน้าที่เพิ่มประสิทธิภาพในการจำแนกประเภท

3) ชั้นส่งออก หรือชั้นผลลัพธ์ (Output Layer) ทำหน้าที่ส่งออกข้อมูลที่ผ่านการประมวลผลจากฟังก์ชันการรวมผล (Summation Function) ดังสมการ 2.32

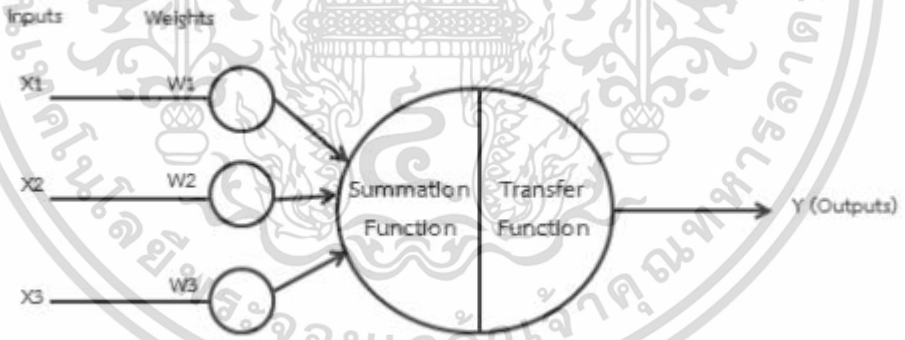
$$S = \sum_{i=1}^{n} w_i x_i + b \qquad (2.32)$$

เมื่อ w คือ ค่าถ่วงน้ำหนัก (Weight)

x คือ ผลลัพธ์จากการคำนวณจากโหนดก่อนหน้า

b คือ ค่าความเอนเอียง (Bias)

โดยโครงข่ายประสาทเทียมแต่ละโหนดประกอบด้วย 5 สิ่ง (ณัฏฐา, 2558) ดังรูปที่ 2.20 ซึ่งประกอบด้วย



รูปที่ 2.20 องค์ประกอบของโครงข่ายประสาทเทียม (ที่มา : ณัฏฐา, 2558)

1) ข้อมูลนำเข้า (Input) เป็นข้อมูลที่ระบบจะนำไปประมวลผล

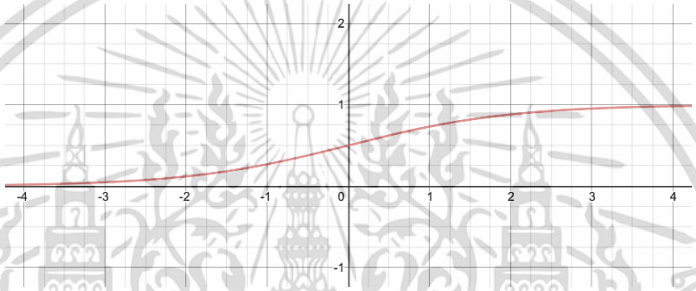
2) ค่าถ่วงน้ำหนัก (Weight) เป็นค่าเฉพาะที่กำหนดให้ข้อมูลนำเข้าแต่ละตัวเพื่อใช้แยกความแตกต่างของข้อมูลนำเข้า

3) ฟังก์ชันการรวมผล (Summation Function) เป็นผลรวมของข้อมูลนำเข้าและค่าถ่วงน้ำหนักในแต่ละชั้นเพื่อสรุปความสัมพันธ์ระหว่างข้อมูลนำเข้าทั้งหมด

4) ฟังก์ชันการแปลง (Transfer Function) เป็นการคำนวณการจำลองของแบบจำลอง โดยแปลงเพื่อใช้สำหรับการแสดงผล ซึ่งเรียกอีกอย่างว่าฟังก์ชันกระตุ้น (Activation Function) แบ่งได้ 3 ประเภท (ชิตพงษ์, 2563)

- ฟังก์ชันซิกมอยด์ (Sigmoid Function) คือฟังก์ชันที่เป็นโค้งรูปตัวเอส โดยข้อมูลส่งออก (Output) จะมีค่า 0 และ1 โดยจุดตัดสินใจจะอยู่ที่ 0.5 มีสมการดังต่อไปนี้

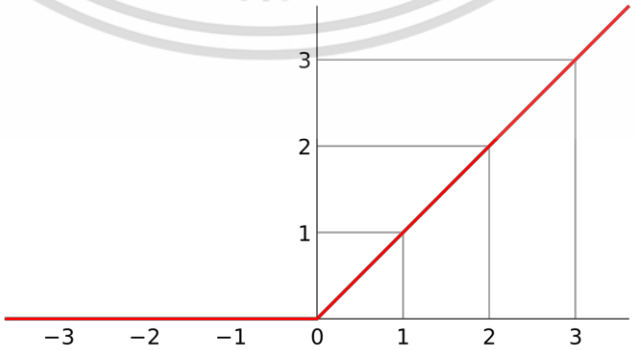
$$f(x)=\frac{1}{1+e^{x}} \tag{2.33}$$



รูปที่ 2.21 ฟังก์ชันซิกมอยด์ (Sigmoid Function)
(ที่มา : https://guopai.github.io/ml-blog16.html)

- เรลู (Rectified Linear Unit; ReLU) คือฟังก์ชันเส้นตรงที่ถูกปรับไม่ได้เป็นตัวเอสเหมือน ซิกมอยด์ (Sigmoid) มีสมการดังต่อไปนี้

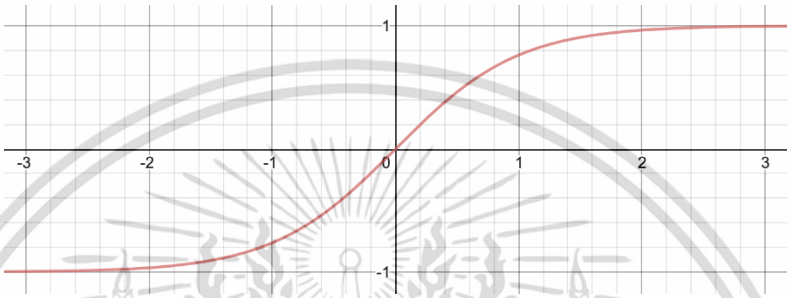
$$f(x)=\max(0,x)=\begin{cases}0 & for\ x\le 0\\ x & for\ x>0\end{cases} \tag{2.34}$$



รูปที่ 2.22 ฟังก์ชันเรลู (Rectified Linear Unit; ReLU)
(ที่มา : https://guopai.github.io/ml-blog16.html)

- ฟังก์ชันแทนเฮช (Hyperbolic Tangent Activation Function; Tanh Function) คือฟังก์ชันที่เป็นโค้งรูปตัวเอสคล้ายกับ ซิกมอยด์ (Sigmoid Function) โดยข้อมูลส่งออก (Output) จะมีค่า -1 และ 1 โดยจุดตัดสินใจจะอยู่ที่ 0 มีสมการดังต่อไปนี้

$$f(x)=\frac{1-e^{-2x}}{1+e^{-2x}} \tag{2.35}$$



รูปที่ 2.23 ฟังก์ชันแทนเฮช (Tanh Function)
(ที่มา : https://guopai.github.io/ml-blog16.html)

5) ข้อมูลที่ส่งออก (Output) เป็นผลลัพธ์ที่แท้จริงที่เกิดขึ้นจากแบบจำลอง

วิธีแพร่กระจายย้อนกลับ (Back Propagation)

ในแบบจำลองโครงข่ายประสาทเทียมที่ใช้หลักการป้อนไปข้างหน้าจะมีการนำข้อผิดพลาดจากการสอน (Training Error) ที่ได้จากการคำนวณฟังก์ชันการสูญเสีย (Loss Function) ซึ่งใช้เป็นฟังก์ชันในการวัดผลว่าแบบจำลองทำงานได้ดีแค่ไหน โดยเปรียบเทียบผลลัพธ์ที่ได้จากการทำนายของแบบจำลองกับค่าที่แท้จริง ซึ่งในงานวิจัยชิ้นนี้เป็นงานวิจัยการจำแนกประเภทแบบ 2 กลุ่มจึงเลือกใช้ฟังก์ชันการสูญเสีย Binary Cross Entropy (Log Loss) (Perlato, 2019) ดังสมการ 2.36

$$L=+-\frac{1}{N}\sum_{n=1}^{N}\left[y_n \log \hat{y}_n+(1-y_n)\log(1-\hat{y}_n)\right] \tag{2.36}$$

เมื่อ N คือ จำนวนข้อมูลทั้งหมดที่ใช้ในการสอน

y_n คือ ค่าที่แท้จริง

$\hat{y}$ คือ ค่าที่ได้จากการทำนายนของแบบจำลอง

จากนั้นนำค่าความผิดพลาดส่งย้อนกลับผ่านเครือข่ายไปในทิศทางตรงกันข้าม (Backpropagated) เพื่อปรับค่าถ่วงน้ำหนัก (Weight) ของโหนดต่าง ๆ ซึ่งแนวคิดพื้นฐานของ Back Propagation คือการคาดการณ์ว่าค่าถ่วงน้ำหนักของชั้นซ่อนเป็นอย่างไรการปรับค่าถ่วงน้ำหนักนั้นจะเป็นการปรับครั้งละเล็กน้อย โดยปริมาณที่ทำการปรับระหว่างการสอนนั้นมักจะใช้ผลคูณของอัตราการเรียนรู้ (Learning rate) กับค่าความผิดพลาด ซึ่งอัตราการเรียนรู้นั้นถือเป็นหนึ่งในไฮเพอร์พารามิเตอร์ (Hyperparameter) ที่สามารถปรับได้ในกระบวนการการสอนแบบจำลองโครงข่ายประสาทเทียม มักมีค่าอยู่ระหว่าง 0 ถึง 1 โดยทั่วไปอัตราการเรียนรู้ที่สูงจะทำให้แบบจำลองเรียนรู้ได้เร็วขึ้น โดยอาจทำให้ค่าถ่วงน้ำหนักที่ได้ไม่ใช่ค่าที่ดีที่สุดและหากใช้อัตราการเรียนรู้ที่น้อยกว่าอาจทำให้แบบจำลองสามารถพบค่าถ่วงน้ำหนักที่เหมาะสมมากกว่า หรือเหมาะสมที่สุด แต่อาจใช้เวลาในการสอนนานกว่า

2.5 การวัดประสิทธิภาพของแบบจำลอง (Model Performance Evaluation)

2.5.1 เมทริกซ์ความสับสน (Confusion Matrix)

ตารางในการวัดความสามารถของการเรียนรู้ของเครื่อง (Machine learning) ในการแก้ปัญหาการจำแนกประเภท (Classification) โดยมีอักษรย่อดังแสดงในตารางที่ 2.1 และมีการคำนวณค่าวัดผลต่าง ๆ ดังสมการที่ 2.37 ถึง 2.40

- True Positive (TP) คือ ทำนายออกมาว่า "บวก" และ มีค่าเป็น "บวก"
- True Negative (TN) คือ ทำนายออกมาว่า "ลบ" และ มีค่าเป็น "ลบ"
- False Positive (FP) คือ ทำนายออกมาว่า "บวก" และ มีค่าเป็น "ลบ"
- False Negative (FN) คือ ทำนายออกมาว่า "ลบ" และ มีค่าเป็น "บวก"

ตารางที่ 2.1 เมทริกซ์ความสับสน (Confusion Matrix) ขนาด 2x2

		Predicted Values	
		Positive	Negative
Actual Values	Positive	True Positive (TP)	False Negative (FN)
	Negative	False Positive (FP)	True Negative (TN)

2.5.2 ค่าความแม่นยำ (Accuracy)

การหาความถูกต้องของแบบจำลองซึ่งพิจารณารวมทุกค่า y ที่เกี่ยวข้อง

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{2.37}$$

2.5.3 ค่าความเที่ยง (Precision)

การหาความแม่นยำของแบบจำลองโดยพิจารณาแยกทีละค่าของ y

$$Precision = \frac{TP}{TP + FP} \tag{2.38}$$

2.5.4 ค่าความไว หรือค่าระลึก (Recall)

การวัดความถูกต้องของแบบจำลองโดยพิจารณาแยกทีละค่าของ y

$$Recall = \frac{TP}{TP + FN} \tag{2.39}$$

2.5.5 ค่าความถ่วงดุล (F1 - Score)

ค่าเฉลี่ยแบบฮาร์โมนิกของค่าความเที่ยง (Precision) และค่าความไว หรือค่าระลึก (Recall) โดยพิจารณาทีละค่าของ y

$$F1 - Score = 2 \times \left[\frac{Precision \times Recall}{Precision + Recall} \right] \tag{2.40}$$

ซึ่งการวัดผลทั้ง 4 ค่าที่กล่าวมาข้างต้น หากมีค่ามาก หมายความว่ามีประสิทธิภาพดีและสามารถอธิบายเป็นค่าร้อยละได้ (Ninenox Developer, 2020)

2.6 งานวิจัยที่เกี่ยวข้อง (Related Research)

Velay and Daniel (2018) ศึกษาการใช้การประมวลภาษาธรรมชาติเพื่อทำนายแนวโน้มของดัชนี DJIA ใช้การสร้างตัวแทนเชิงความหมายของคำและข้อความ (Word embedding) ด้วยวิธี Word2Vec และใช้แบบจำลอง Logistic Regression, Linear Discriminant Analysis, K-Nearest Neighbors, Decision Tree Classifier, Support Vector Machine, Random Forrest, Extreme Gradient Boosting, Naive Bayes, Long Short Term Memory (LSTM), Multi-Layer Perceptron (MLP) พบว่าแบบจำลอง Logistic Regression มีความแม่นยำ 57% ซึ่งมากกว่าแบบจำลองประเภทอื่นๆ

Soni (2018) ศึกษาการทำนายความเป็นอัจฉริยะของข่าวโดยใช้การเรียนรู้ของเครื่องขั้นสูงและอัลกอริทึมการประมวลภาษาธรรมชาติ เปรียบเทียบ Feature Set 3 แบบคือ 1. เนื้อหาข่าวและพาดหัวข่าวรวมกัน 2. เนื้อหาข่าว 3. พาดหัวข่าว และใช้แบบจำลอง Naïve Bayes with Lidstone smoothing, Support Vector Machine, Logistic Regression พบว่า Feature Set แบบเนื้อหาข่าวและพาดหัวข่าวรวมกันความแม่นยำของแบบจำลอง Naïve Bayes with Lidstone smoothing อยู่ที่ 83.16% ซึ่งดีกว่าแบบจำลอง Support Vector Machine ที่มีความแม่นยำ 81.66%

Yildirim et al. (2018) ศึกษาการจำแนก “ข่าวด่วน” เพื่อพยากรณ์การเงินด้วยเทคนิค NLP ใช้ Text Representation ด้วยวิธี Bag of Words และ TF-IDF พบว่าแบบจำลอง Support Vector Machine (SVM) สามารถจำแนก “ข่าวด่วน” เพื่อพยากรณ์การเงินโดยมีความแม่นยำที่ 91.4% ได้ดีกว่าเมื่อเปรียบเทียบกับแบบจำลอง k-Nearest Neighbor (KNN), Logistic Regression, linear kernel and multinomial Naïve Bayes (m-NB)

Kara et al. (2011) ศึกษาการทำนายทิศทางการเคลื่อนไหวของราคาดัชนีหุ้นโดยใช้โครงข่ายประสาทเทียม และซัพพอร์ตเวกเตอร์แมชชีน ตัวอย่างตลาดหลักทรัพย์อิสตันบูล พบว่าความแม่นยำของแบบจำลองโครงข่ายประสาทเทียม (Artificial Neural Network; ANN) อยู่ที่ 75.74% ซึ่งดีกว่าแบบจำลองซัพพอร์ตเวกเตอร์แมชชีน (SVM) ที่มีความแม่นยำอยู่ที่ 71.52%

บุษบงก์ และคณะ (2564) ศึกษาการสร้างระบบคัดกรองข้อความการเกลียดกลัวคนต่างชาติบนทวิตเตอร์ในช่วงการแพร่ระบาดของโรคติดเชื้อไวรัสโคโรนา 2019 โดยใช้การแปลงเชิงปริมาณทั้งหมด 3 วิธีคือ TF-IDF, Word2Vec และ GloVe และใช้ตัวแบบจำลองแรนดอมฟอเรสต์ (Random Forest) และแบบจำลองซัพพอร์ตเวกเตอร์แมชชีน (Support Vector Machine; SVM) พบว่าแบบจำลองแรนดอมฟอเรสต์ (Random Forest) เมื่อใช้การแปลงเชิงปริมาณด้วยวิธี TF-IDF ให้ค่าความแม่นยำ, F1-Score, Recall ไม่แตกต่างกัน เมื่อเทียบกับวิธี Word2Vec และ GloVe ส่วนแบบจำลองซัพพอร์ตเวกเตอร์แมชชีน (Support Vector Machine; SVM) เมื่อใช้การแปลงเชิงปริมาณด้วยวิธี Word2Vec ให้ค่า F1-Score และ Precision สูงที่สุดคือ 0.43 และ 0.28 ดังนั้น

แบบจำลองแรนดอมฟอเรสต์ (Random Forest) เมื่อใช้ TF-IDF และแบบจำลองซัพพอร์ตเวกเตอร์แมชชีน (Support Vector Machine; SVM) เมื่อใช้ Word2Vec เป็นแบบจำลองที่เหมาะสมที่สุด

ภูมิรพี (2562) ศึกษาเทคนิคการเรียนรู้เชิงลึกเพื่อวิเคราะห์ความรู้สึกจากผู้ใช้ผลิตภัณฑ์ ใช้การแปลงข้อมูลข้อความทั้งหมด 3 วิธี คือ TF-IDF, Word Embedding, Doc2Vec โดยกำหนดคลังคำศัพท์เท่ากับ 5,000 คำ และใช้อัลกอริทึม Deep Neural Network (DNN), อัลกอริทึม Convolutional Neural Network แบบ 1 มิติ (CNN1D), อัลกอริทึม Convolutional Neural Network แบบ 2 มิติ (CNN2D), อัลกอริทึม Long Short-Term Memory (LSTM) และ อัลกอริทึม Gated Recurrent Unit (GRU) พบว่า อัลกอริทึม Deep Neural Network (DNN) เมื่อใช้การแปลงข้อมูลข้อความแบบ TF-IDF ให้ค่าความแม่นยำ (Accuracy) ที่ 84% ค่าความเที่ยง (Precision) ที่ 84% ค่าความไว หรือค่าระลึก (Recall) ที่ 83% ซึ่งมากกว่าแบบจำลองประเภทอื่น ๆ

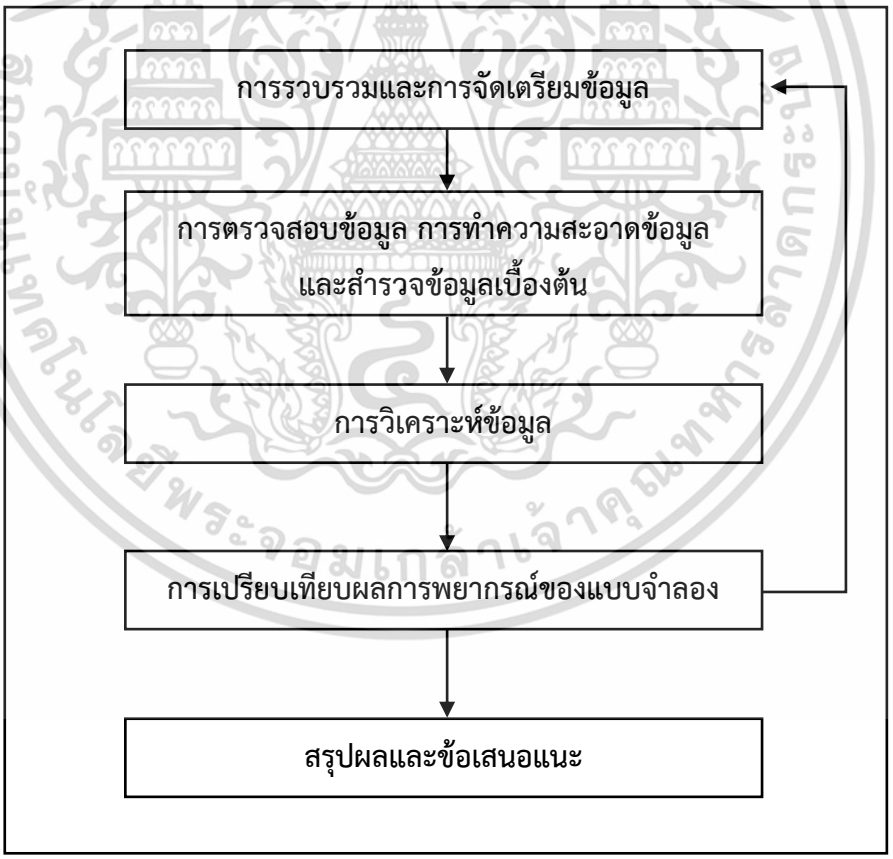
บทที่ 3

วิธีการดำเนินงานวิจัย

งานวิจัยนี้สนใจศึกษาเกี่ยวกับการพยากรณ์ทิศทางราคาสัญญาฟิวเจอร์สข้าวโพดจากข้อความข่าวโดยใช้การประมวลภาษาธรรมชาติ ผู้วิจัยได้นำทฤษฎี แนวคิด และงานวิจัยที่เกี่ยวข้องมากำหนดขั้นตอนในการศึกษาดังนี้

3.1 ขั้นตอนการดำเนินงาน

การพยากรณ์ทิศทางราคาสัญญาฟิวเจอร์สข้าวโพดจากข้อความข่าวโดยใช้การประมวลภาษาธรรมชาติ เพื่อให้เกิดความเข้าใจในข้อมูล และการจัดเตรียมข้อมูลในการวิเคราะห์ สามารถแบ่งออกเป็นขั้นตอนต่าง ๆ ดังนี้



รูปที่ 3.1 ขั้นตอนการดำเนินงาน

3.2 การรวบรวมข้อมูล

3.2.1 ตัวแปรตาม

รวบรวมราคาสัญญาฟิวเจอร์สข้าวโพดรายวันจาก ตลาดหอการค้าแห่งนครชิคาโก Chicago Board of Trade (CBOT) ตั้งแต่วันที่ 1 มกราคม พ.ศ. 2556 ถึงวันที่ 31 ธันวาคม พ.ศ. 2563

3.2.2 ตัวแปรอิสระ

รวบรวมข้อมูลข่าวรายวันจาก สำนักข่าวรอยเตอร์ (Reuters) ตั้งแต่วันที่ 1 มกราคม พ.ศ. 2556 ถึงวันที่ 31 ธันวาคม พ.ศ. 2563

งานวิจัยนี้สนใจศึกษาเกี่ยวกับการพยากรณ์ทิศทางราคาสัญญาฟิวเจอร์สข้าวโพดจากข้อความข่าวโดยใช้การประมวลภาษาธรรมชาติ เนื่องจากมีข้อมูลข่าวรายวันจาก สำนักข่าวรอยเตอร์ (Reuters) เป็นจำนวนมาก ทั้งที่เป็นข่าวที่เกี่ยวข้องและข่าวที่ไม่เกี่ยวข้องกับการพยากรณ์ทิศทางราคาสัญญาฟิวเจอร์สข้าวโพด ทางผู้วิจัยจึงมีวิธีการคัดเลือกข่าวเพื่อลดความซับซ้อนของข้อมูลข่าวที่ไม่เกี่ยวข้อง โดยใช้วิธีกำหนดคำสำคัญ (Keyword) ที่ปรากฏในแต่ละข่าว โดยเลือกใช้ข่าวที่มีคำว่า “corn” หรือ คำว่า “maize” ปรากฏอยู่ในหัวข้อข่าวหรือเนื้อหาข่าวมาใช้ในการพยากรณ์ทิศทางราคาสัญญาฟิวเจอร์สข้าวโพด

3.3 การจัดเตรียมข้อมูล

3.3.1 แปลงราคาสัญญาฟิวเจอร์สข้าวโพดก่อนนำไปสร้างแบบจำลอง

เนื่องจากงานวิจัยนี้เป็นการทำนายทิศทางราคาสัญญาฟิวเจอร์สของข้าวโพด ดังนั้นจึงจำเป็นที่จะต้องแปลงค่าของราคาข้าวโพดให้อยู่ในรูปของทิศทางราคาก่อนโดยการคำนวณร้อยละของผลตอบแทนที่เปลี่ยนไปของราคาเฉลี่ยในแต่ละวันจากสูตรคำนวณจากสมการที่ 3.1

$$R_t = \begin{cases} 0, LAST_{t+1} < LAST_t \\ 1, LAST_{t+1} \geq LAST_t \end{cases} \quad (3.1)$$

เมื่อ R_t คือ ร้อยละของผลตอบแทนที่เปลี่ยนไปในแต่ละวัน

t คือ เวลา

$LAST_t$ คือ ราคาปิดของข้าวโพด ณ วันที่ t

$LAST_{t+1}$ คือ ราคาปิดของข้าวโพด ณ วันที่ t+1

ทิศทางของราคาในวันถัดไปแบ่งออกเป็นสองประเภท: '0' หรือ '1' โดยที่ '0' หมายถึงราคาปิดของวันถัดไปต่ำกว่าราคาปิดของวันนี้ และ '1' หมายถึงราคาปิดของวันถัดไปสูงกว่าหรือเท่ากับราคาปิดของวันนี้ (Zhai et al., 2007)

ตารางที่ 3.1 การแปลงร้อยละของผลตอบแทนที่เปลี่ยนไปของราคาในแต่ละวัน

ก่อนการแปลงข้อมูล	หลังการปลงข้อมูล
$R_t < 0$	0 (Bearish)
$R_t \geq 0$	1 (Bullish)

3.3.2 แปลงข้อความข่าวก่อนนำไปสร้างแบบจำลอง

ชุดข้อมูลข่าวรายวันจาก สำนักข่าวรอยเตอร์ ที่นำมาใช้ คือ คอลัมน์ “Dates” แทนวันที่ข่าวเผยแพร่ คอลัมน์ “News Topic” แทนหัวข้อข่าว และคอลัมน์ “News Content” แทนเนื้อหาข่าว ตัวอย่างข้อมูลแสดงดังตารางที่ 3.2

ตารางที่ 3.2 ชุดข้อมูลข่าวรายวันจาก สำนักข่าวรอยเตอร์ (Reuters)

Dates	News Topic	News Content
2013-01-01	2013 outlook: farm bill, crop production, land values, livestock, biofuels.	the breakeven cost of producing corn at trend line yields will likely be close to $5/bu. or higher ...
2013-01-01	beans, corn down but above lows.	corn was lower on technical and commercial selling, in addition to spillover from beans ...
2013-01-01	zambia lifts restriction on maize exports.	lusaka (reuters) - zambia has lifted its restriction on maize exports due to inadequate storage capacity ...

จากตารางที่ 3.2 ทำการรวมคอลัมน์ “News Topic” หรือหัวข้อข่าว และคอลัมน์ “News Content” หรือเนื้อหาข่าว จะได้คอลัมน์ “News” คือการรวมหัวข้อข่าวและเนื้อหาข่าวไว้ด้วยกัน เพื่อแสดงถึงข้อความข่าวทั้งหมดในข่าวเรื่องนั้น ๆ ดังแสดงในตารางที่ 3.3

ตารางที่ 3.3 ชุดข้อมูลข้อความข่าวทั้งหมดในข่าวเรื่องนั้น ๆ

Dates	News
2013-01-01	2013 outlook: farm bill, crop production, land values, livestock, biofuels. the breakeven cost of producing corn at trend line yields will likely be close to $5/bu. or higher ...
2013-01-01	beans, corn down but above lows. corn was lower on technical and commercial selling, in addition to spillover from beans ...
2013-01-01	zambia lifts restriction on maize exports. lusaka (reuters) - zambia has lifted its restriction on maize exports due to inadequate storage capacity ...

จากตารางที่ 3.3 ในคอลัมน์ “Dates” จะสังเกตได้ว่า ข้อความข่าวแต่ละเรื่อง มีวันที่เดียวกัน เพราะว่าในหนึ่งวันมีข่าวที่เกิดขึ้นมากมาย จึงทำการรวมข่าวโดยใช้วันที่เดียวกัน เพื่อให้ได้ภาพรวม

ของข่าวในวันนั้น ๆ และสามารถกำหนดสถานะผลตอบแทนที่เปลี่ยนไปของราคาสัญญาฟิวเจอร์สข้าวโพดในแต่ละวันได้ ดังแสดงในตารางที่ 3.4

ตารางที่ 3.4 ภาพรวมของชุดข้อมูลข้อความในวันนั้น ๆ

Dates	News
2013-01-01	2013 outlook: farm bill, crop production, land values, livestock, biofuels. the breakeven cost of producing corn at trend line yields will likely be close to $5/bu. or higher ... beans, corn down but above lows. corn was lower on technical and commercial selling, in addition to spillover from beans ... zambia lifts restriction on maize exports. lusaka (reuters) - zambia has lifted its restriction on maize exports due to inadequate storage capacity ...

ขั้นตอนที่ 1 การแปลงข้อมูลให้อยู่ในรูปแบบที่คอมพิวเตอร์สามารถประมวลผลได้

1) การทำความสะอาดข้อมูล (Text Cleaning) เป็นขั้นตอนในการทำความสะอาดข้อความต่าง ๆ สำหรับข้อมูลที่อยู่ในรูปแบบข้อความนั้น จะเป็นการแก้ไขรายละเอียดคำในส่วนที่สะกดผิดเป็นส่วนใหญ่ หรือการปรับตัวอักษรของคำให้อยู่ในรูปแบบเดียวกัน เช่นตัวพิมพ์เล็กในภาษาอังกฤษ เนื่องจากคอมพิวเตอร์จะเข้าใจตัวอักษรพิมพ์ใหญ่ และตัวอักษรพิมพ์เล็กว่าแตกต่างกัน ไม่ใช่ตัวเดียวกัน เช่น 'A' กับ 'a' เป็นต้น และการลบสัญลักษณ์ต่าง ๆ ที่อยู่ในชุดข้อมูลออกไป ในการวิจัยนี้ใช้วิธีการทำความสะอาดข้อมูล ดังต่อไปนี้

1) การปรับตัวอักษรเป็นตัวพิมพ์เล็ก (Lowercase)

2) การตัดคำ (Tokenization) คือ การแยกคำออกจากกันในประโยต ให้อยู่ในรูปของคำเดี่ยวหรือกลุ่มคำที่มีจำนวนตามที่สนใจ

3) การกำจัดคำฟุ่มเฟือย (Stop Words) คือ การกำจัดคำที่ไม่สำคัญ

4) การลดความซ้ำซ้อนของคำ (Lemmatization/Stemming) คือ การแปลงคำให้อยู่ในรูปแบบดังเดิม (Lemmatization และการตัดส่วนขยาย (Stemming) ของคำจะทำการตัดบางส่วนของคำทิ้ง เช่น s, es, ing หรือ ed

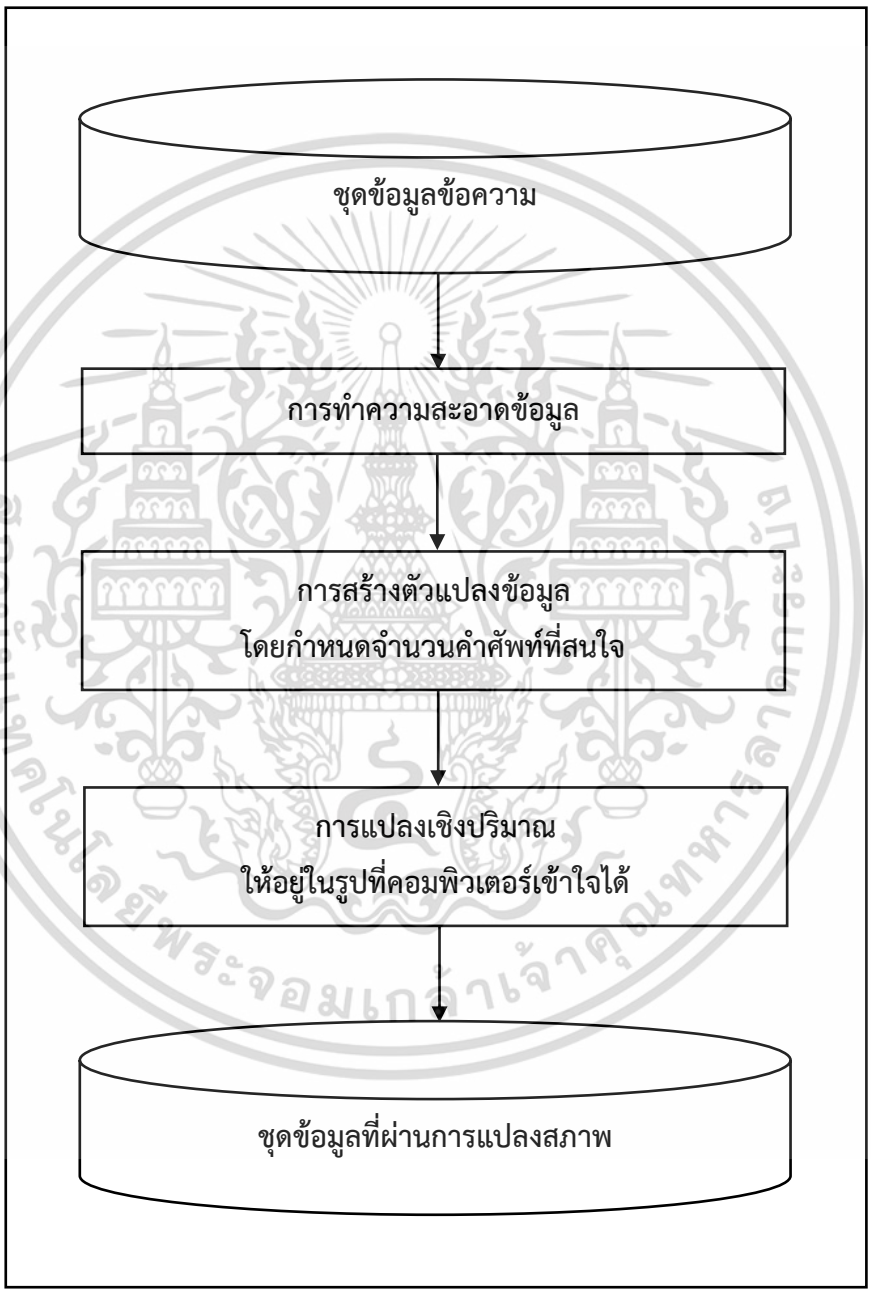
5) การกำหนดรูปแบบหรือกลุ่มคำ (Regular Expression) การตัดตัวอักษรพิเศษที่ไม่ใช่ข้อความทิ้งตามรูปแบบหรือกลุ่มคำที่กำหนด

6) การลบช่องว่างระหว่างคำ (White space)

2) การสร้างคลังคำศัพท์ (Vocabulary size) เป็นขั้นตอนในการสร้างคลังคำศัพท์สำหรับใช้ไนการวิเคราะห์ โดยทำการรวบรวมคำที่มีอยู่ในประ โยคและทำการสร้างคลังคำศัพท์โดยทำการนำคำที่แตกต่างกันมาใช้ในการสร้าง ซึ่งสามารถกำหนดขอบเขตของคำที่ใช้ในการศึกษาได้โดยการกำหนดจำนวนคำที่แสดงผลมากที่สุดในชุดข้อมูล เพื่อลดข้อมูลในส่วนที่ไม่จำเป็นในการวิเคราะห์ออกไป โดย

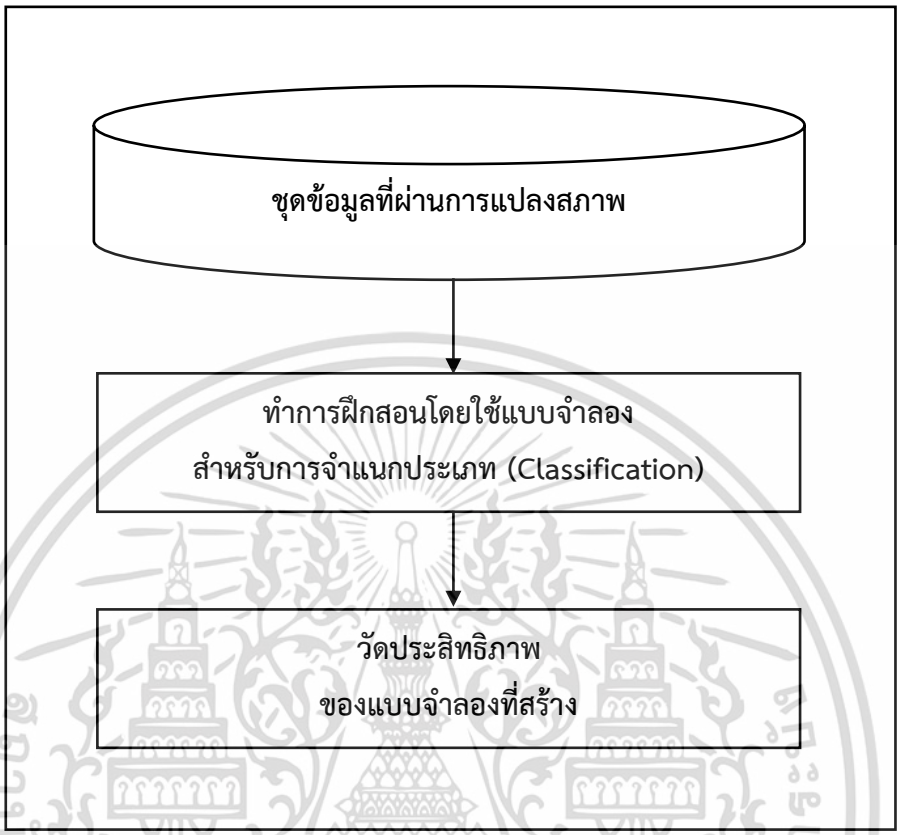
ในงานวิจัยนี้กำหนดคลังคำศัพท์ (Vocabulary size) ไว้ที่ 5,000 คำ และกำหนดเวกเตอร์คุณลักษณะ (Feature vector) จำนวน 5,000 ตัวเช่นกัน (ภูมิรพี, 2562)

3) การแปลงเชิงปริมาณ (Text Representation) เป็นขั้นตอนที่ใช้ในการแปลงคำต่าง ๆ ให้อยู่ในรูปที่คอมพิวเตอร์สามารถนำไปใช้ในการประมวลผลต่อได้ ซึ่งจะต้องทำการแปลงข้อมูลก่อนที่จะนำมาใช้วิเคราะห์ด้วยแบบจำลองต่าง ๆ



รูปที่ 3.2 กรอบแนวคิดวิธีการแปลงข้อมูล

ขั้นตอนที่ 2 การสร้างแบ บจำลองการพยากรณ์ทิศทางราคาสัญญาฟิวเจอร์สข้าวโพด



รูปที่ 3.3 กรอบแนวคิดการสร้างแบบจำลองสำหรับทำนายผล

จากรูปที่ 3.3 กรอบแนวคิดการใช้ การพยากรณ์ทิศทางราคาสัญญาฟิวเจอร์สข้าวโพด จะประกอบไปด้วย 3 ขั้นตอน ดังนี้

1) นำข้อมูลที่ผ่านการแปลงสภาพแล้ว มาทำการเตรียมข้อมูลเพื่อที่จะนำไปใช้ในการวิเคราะห์และสร้างแบบจำลอง

2) ทำการสร้างแบบจำลองการพยากรณ์ทิศทางราคาสัญญาฟิวเจอร์สข้าวโพด

3) นำผลลัพธ์ที่ได้จากแบบจำลอง มาทำการวัดประสิทธิภาพการทำนาย พิจารณาจากค่าความถ่วงดุล (F1 – Score) เป็นอันดับแรกในการวัดผลและจะคำนึงถึงค่าความแม่นยำ (Accuracy) รองลงมา

3.4 ชุดข้อมูล

งานวิจัยนี้ได้ทำการเลือกข้อมูลข่าวรายวันจาก สำนักข่าวรอยเตอร์ ตั้งแต่วันที่ 1 มกราคม พ.ศ. 2556 ถึงวันที่ 31 ธันวาคม พ.ศ. 2563 และราคาสัญญาฟิวเจอร์สข้าวโพดรายวันจาก ตลาดหอการค้าแห่งนครชิคาโก หรือ Chicago Board of Trade (CBOT) ตั้งแต่วันที่ 1 มกราคม พ.ศ. 2556 ถึงวันที่

31 ธันวาคม พ.ศ. 2563 รวมทั้งสิ้น 8 ปี (2,065 วัน) ภายในชุดข้อมูลประกอบด้วย ชุดข้อมูลราคาในรูปแบบเพิ่มขึ้นจำนวน 984 ระเบียน และด้านลดลงจำนวน 1081 ระเบียน โดยแสดงออกมาเป็นตารางที่มีจำนวน 3 คอลัมน์ โดยคอลัมน์ “Dates” แทนวันที่ข่าวเผยแพร่ คอลัมน์ “News” แทนข้อความข่าวทั้งหมดในหนึ่งวัน และคอลัมน์ “Class” แสดงการจำแนกประเภทของข้อความโดยที่เลข 0 (ตลาดหมีหรือตลาดขาลง (Bearish; 0)) แทนข้อความที่ส่งผลให้ราคาสัญญาฟิวเจอร์สข้าวโพดไปในทางลดลง และเลข 1 (ตลาดกระทิงหรือตลาดขาขึ้น (Bullish)) แทนข้อความที่ส่งผลให้ราคาสัญญาฟิวเจอร์สข้าวโพดไปในทางเพิ่มขึ้น ตัวอย่างข้อมูลแสดงดังตารางที่ 3.5

ตารางที่ 3.5 ตัวอย่างชุดข้อมูล

Dates	News	Class
2013-01-01	2013 outlook: farm bill, crop production, land values, livestock, biofuels commodities-wheat, soy top 2012 gains; coffee, juice lead losses bee threat at planting northeastern brazil continues to be impacted by dry weather commodities-wheat, soy top 2012 gains; coffee, juice lead losses livestock-u.s. live cattle up nearly 6 pct in 2012 grains-wheat posts largest 2012 gain among commodities ...	0
2013-01-02	grains-prices tumble after u.s. fiscal deal euphoria fades commodities-oil and metals start year strongly after us fiscal deal one dead at adm iowa corn processing plant grains-wheat rebounds after losing 3 pct in previous session update 3-boehner sets house votes on sandy aid after republican attacks grains-strong dollar hits markets ...	0
2013-01-03	livestock-u.s. live cattle futures surge to record high the grain and soy markets seem set to open weakly thursday drought persists in u.s. plains; slight improvement in midwest grain, livestock markets mixed thursday grains - wheat, corn tumble to 6-month lows on demand fears commodities-fed doubts about stimulus pressure gold, oil commodities-fed doubts about stimulus pressure gold, oil grain futures reversed to the upside ...	1

3.5 สถิติที่ใช้ในการวิเคราะห์

3.5.1 สถิติเชิงพรรณนา (Descriptive Statistics)

1) ความถี่ (Frequency) ของคุณลักษณะ Bag of Words สำหรับการแปลงเชิงปริมาณ

2) ค่าเฉลี่ย (Mean) แทนด้วยสัญลักษณ์ $\overline{x}$

$$\overline{x} = \frac{\sum_{i=1}^{n} x_i}{n} \tag{3.2}$$

เมื่อ $\sum_{i=1}^{n} x_i$ คือ ผลรวมของข้อมูล

n คือ จำนวนข้อมูลทั้งหมด

3) ส่วนเบี่ยงเบนมาตรฐาน (Standard Deviation) แทนด้วยสัญลักษณ์ sd

$$sd = \sqrt{\frac{\sum_{i=1}^{n}\left(x_i - \bar{x}\right)^2}{n-1}} \tag{3.3}$$

เมื่อ x_i คือ ข้อมูลตัวที่ i เมื่อ i = 1,2,3,...,n

$\bar{x}$ คือ ค่าเฉลี่ยข้อมูล

n คือ จำนวนข้อมูลทั้งหมด

4) ร้อยละ (Percentage)

$$Percentage = \left(\frac{X}{N}\right) \times 100 \tag{3.4}$$

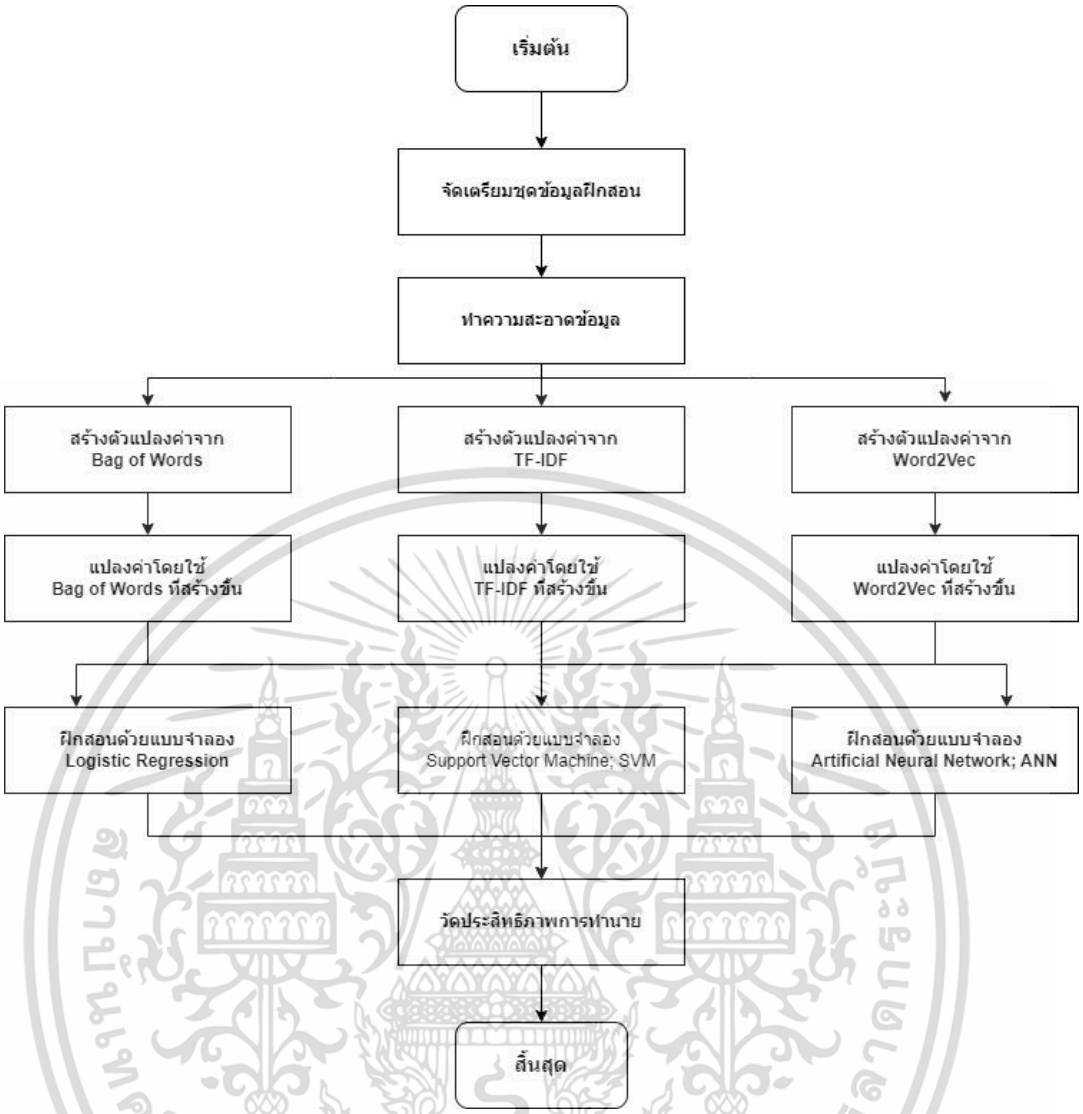
เมื่อ X คือ จำนวนข้อมูลที่ต้องการนำมาหาร้อยละ (ความถี่)

N คือ จำนวนข้อมูลทั้งหมด

3.6 การออกแบบคุณลักษณะและแบบจำลอง

ในการวิจัยนี้ ผู้วิจัยได้ทำการแบ่งการทดลองออกเป็น 9 รูปแบบ โดยแบ่งออกเป็นการสร้างตัวแปลงเชิงปริมาณให้อยู่ในรูปของคุณลักษณะที่ใช้ในการประมวลผลได้ ได้แก่ Bag of Words, TF-IDF และ Word2Vec จากนั้นทำการฝึกสอนโดยใช้แบบจำลองสำหรับการจำแนกประเภท (Classification) ได้แก่ แบบจำลองการถดถอยลอจิสติก (Logistic Regression), ซัพพอร์ตเวกเตอร์แมชชีน (Support Vector Machine; SVM) และ โครงข่ายประสาทเทียม (Artificial Neural Network; ANN)

โดยแบ่งข้อมูลทดสอบ (Split Testing) ในงานวิจัยนี้ได้แบ่งข้อมูลออกเป็น 2 ชุด โดยข้อมูลชุดแรกคือ ข้อมูลชุดเรียนรู้ (Training Dataset) จากข้อมูลที่ผ่านการแปลงสภาพแล้ว 80% คิดเป็น 1,652 วัน และข้อมูลชุดทดสอบ (Testing Dataset) อีก 20% คิดเป็น 413 วัน รวมทั้งสิ้น 2,065 วัน เพื่อนำไปใช้ทดสอบประสิทธิภาพของตัวแบบ โดยมีขั้นตอนในการทดลองแบ่งออกเป็น ดังนี้



รูปที่ 3.4 การออกแบบคุณลักษณะและแบบจำลอง

จากรูปที่ 3.4 แสดงขั้นตอนการออกแบบคุณลักษณะและแบบจำลอง มีขั้นตอน ดังนี้

1) เริ่มทำการทดลองโดยการจัดเตรียมข้อมูลสำหรับการวิเคราะห์และสร้างแบบจำลอง

2) ทำความสะอาดชุดข้อมูลและสำรวจข้อมูลเบื้องต้น

3) สร้างตัวแปลงข้อมูลทั้ง 3 วิธีได้แก่ Bag of Words, TF-IDF และ Word2Vec ตามลำดับ

4) แปลงสภาพชุดข้อมูลโดยใช้ตัวแปลงข้อมูลทั้ง 3 วิธีโดยที่ Bag of Words เมื่อกำหนดคลังคำศัพท์ 5,000 คำ, TF-IDF เมื่อกำหนดคลังคำศัพท์ 5,000 คำ และ Word2Vec เมื่อกำหนดหลักการเป็นแบบ Continuous Bag of Words (CBOW) และ เวกเตอร์คุณลักษณะจำนวน 5,000 ตัว

5) นำข้อมูลที่ผ่านการแปลงสภาพแล้วมาทำการฝึกสอนด้วยแบบจำลองการถดถอยลอจิสติก, ซัพพอร์ตเวกเตอร์แมชชีน เมื่อกำหนดค่าพารามิเตอร์ C=1.0, kernel = 'rbf' และโครงข่ายประสาทเทียม เมื่อกำหนดค่าพารามิเตอร์ learning rate = 0.001, activation function = ReLU และ Sigmoid ตามลำดับ

6) บันทึกผลการทดลองและนำไปคำนวณประสิทธิภาพการทำนาย

7) สรุปผลการทดลอง

3.7 การเปรียบเทียบผลการพยากรณ์ของแบบจำลอง

การเปรียบเทียบประสิทธิภาพของแบบจำลองจะใช้ค่า F1-Score เป็นอันดับแรกในการวัดผล ซึ่งจะคำนึงถึงค่า Accuracy รองลงมา ดังนั้นผลลัพธ์ที่ได้จะไม่ได้มีค่าความแม่นยำสูงสุดสำหรับ Positive Class (ตลาดกระทิงหรือตลาดขาขึ้น (Bullish; 1)) และ Negative Class (ตลาดหมีหรือตลาดขาลง (Bearish; 0)) แต่จะสามารถรวบรวมจำนวน Positive Class (ตลาดกระทิงหรือตลาดขาขึ้น (Bullish; 1)) ได้มากและแม่นยำที่สุด (บุษบงก์ และคณะ, 2564)

3.8 เครื่องมือที่ใช้ในการวิจัย

3.8.1 โปรแกรมภาษาไพธอน (Python 3)

การจัดเตรียมชุดข้อมูล (Dataset) ให้พร้อมสำหรับการสร้างแบบจำลอง วัดประสิทธิภาพของแบบจำลอง ดำเนินการด้วยการเขียนโปรแกรมภาษาไพธอน (Python 3) บน Jupyter Notebook และ Colab Notebook และใช้ไลบรารี (Library) ที่จำเป็นต่อการวิเคราะห์ดังตารางที่ 3.6

ตารางที่ 3.6 ไลบรารี (Library) ที่จำเป็นต่อการวิเคราะห์

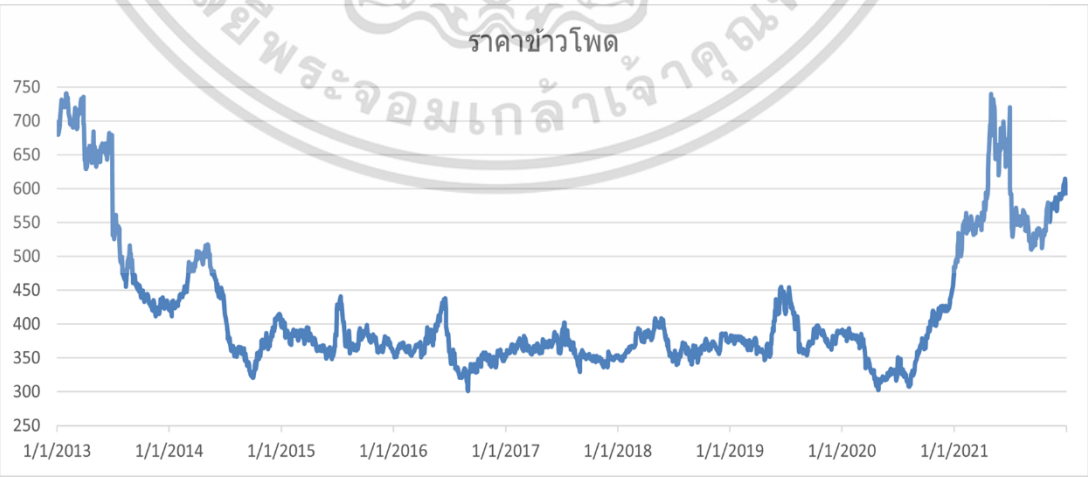
ไลบรารี (Library)	คำอธิบาย (Description)
Pandas	ใช้สำหรับการจัดการข้อมูล (Data Manipulation)
Numpy	ใช้สำหรับการคำนวณทางคณิตศาสตร์และสถิติ (Mathematics and Statistics)
NLTK	ใช้สำหรับการประมวลผลภาษาทางธรรมชาติ (Natural Language Processing)
Gensim	ใช้สำหรับคุณลักษณะ Word2Vec สำหรับการแปลงคุณลักษณะ
Scikit-Learn	ใช้สำหรับคุณลักษณะ Bag of Words และ TF-IDF สำหรับการแปลงเชิงปริมาณ ใช้สำหรับการสร้างแบบจำลองแบบการเรียนรู้ของเครื่อง (Machine Learning) เช่น แบบจำลองการถดถอยลอจิสติก (Logistic Regression) และแบบจำลองซัพพอร์ตเวกเตอร์แมชชีน (Support Vector Machine; SVM) เป็นต้น
Tensorflow	ใช้สำหรับการสร้างแบบจำลองแบบการเรียนรู้ของเครื่อง (Machine Learning) เช่น แบบจำลองโครงข่ายประสาทเทียม (Artificial Neural Network; ANN)
Matplotlib	ใช้สำหรับแสดงผลข้อมูล (Data Visualization)

บทที่ 4
ผลการวิจัยและการอภิปรายผล

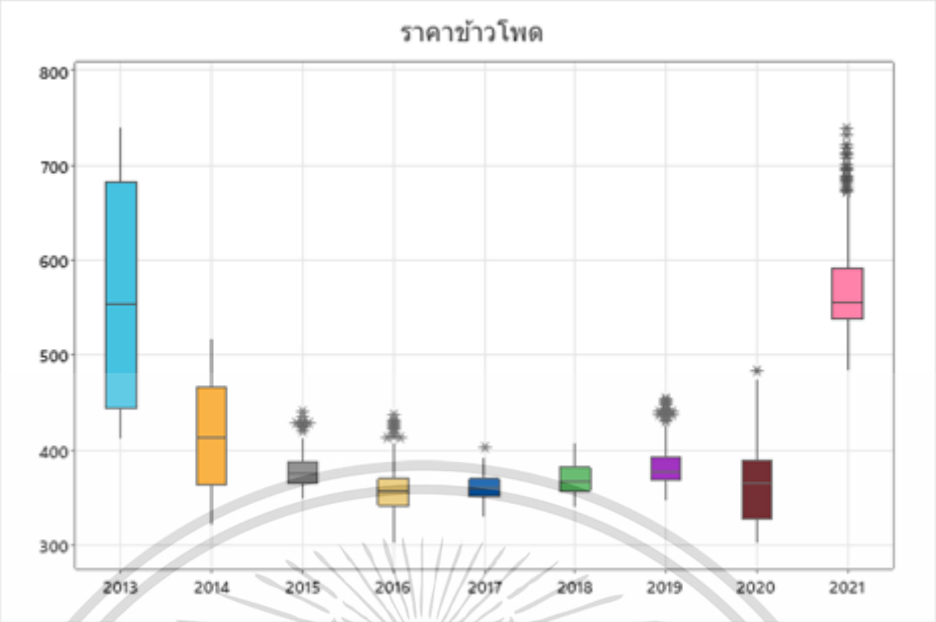
ในบทนี้ผู้วิจัยจะกล่าวถึงผลการวิเคราะห์การพยากรณ์ทิศทางราคาสัญญาฟิวเจอร์สข้าวโพดจากข้อความข่าวโดยใช้การประมวลภาษาธรรมชาติ โดยใช้การสร้างตัวแปลงเชิงปริมาณให้อยู่ในรูปของคุณลักษณะที่ใช้ในการประมวลผลได้ ได้แก่ Bag of Words, TF-IDF และ Word2Vec จากนั้นทำการฝึกสอนโดยใช้ แบบจำลองการถดถอยลอจิสติก (Logistic Regression), ซัพพอร์ตเวกเตอร์แมชชีน (Support Vector Machine; SVM) และโครงข่ายประสาทเทียม (Artificial Neural Network; ANN) สำหรับเนื้อหาในบทนี้จะประกอบไปด้วย ผลการทดสอบประสิทธิภาพของแบบจำลองต่าง ๆ และการอภิปรายผล

ตารางที่ 4.1 จำนวนข้อมูลชุดเรียนรู้ และข้อมูลชุดทดสอบ

	ข้อมูลชุดเรียนรู้ (80% จากข้อมูลทั้งหมด)	**ข้อมูลชุดทดสอบ (20% จากข้อมูลทั้งหมด)**
ตลาดหมีหรือตลาดขาลง (Bearish)	865	216
ตลาดกระทิงหรือตลาดขาขึ้น (Bullish)	787	197
รวม	1,652	413



รูปที่ 4.1 การเคลื่อนไหวของราคาข้าวโพด



รูปที่ 4.2 แผนภาพกล่อง (Box Plot) ของราคาข้าวโพด

จากการเคลื่อนไหวของราคาข้าวโพดจะเห็นว่าราคาข้าวโพดลดลงอย่างต่อเนื่องตั้งแต่ปี 2013 ถึงปี 2014 อาจมีสาเหตุมาจากเกษตรกรส่วนใหญ่ในสหรัฐอเมริกาและต่างประเทศได้มีการปลูกข้าวโพดเพิ่มขึ้นเป็นอย่างมาก เนื่องจากราคาข้าวโพดในปีก่อนหน้านั้นมีราคาที่สูง การเพิ่มผลผลิตทำให้มีของมากขึ้นและส่งผลให้ราคาลดลง ส่งผลให้ราคาข้าวโพดในปี 2015 ถึงปี 2020 เกิดภาวะราคาทรงตัวในช่วงราคา 300 ถึง 450 ดอลลาร์สหรัฐ นอกจากนี้ช่วงปลายปี 2020 จะเห็นว่าราคาข้าวโพดเริ่มกลับมาปรับตัวสูงขึ้น อาจมีสาเหตุมาจากหลายประเทศส่วนใหญ่เริ่มเข้าสู่ขั้นตอนการฟื้นตัวหลังจากการระบาดของโรคติดเชื้อไวรัสโคโรนา (COVID-19) และสถานการณ์สงครามรัสเซีย–ยูเครนก็อาจจะเป็นอีกหนึ่งสาเหตุสำคัญที่ส่งผลให้ราคาข้าวโพดปรับตัวสูงขึ้น เนื่องจากประเทศยูเครนเป็นผู้ผลิตข้าวโพดรายใหญ่อันดับ 5 ของโลก ซึ่งอาจส่งผลให้ความต้องการและกำลังซื้อข้าวโพดเพิ่มขึ้น

รูปที่ 4.3 กราฟความถี่ของ 20 คำศัพท์ที่พบบ่อยที่สุดในข้อมูลชุดเรียนรู้

ตารางที่ 4.2 ลำดับความถี่ของ 20 คำศัพท์ที่พบบ่อยที่สุดในข้อมูลชุดเรียนรู้

ลำดับ	คำศัพท์	ความถี่
1	corn	60,369
2	said	50,100
3	us	40,556
4	soybean	40,408
5	year	38,522
6	wheat	38,426
7	price	37,053
8	crop	35,548
9	percent	32,350
10	market	30,917
11	week	28,419
12	report	23,884
13	plant	22,731
14	future	22,255
15	trade	22,017
16	million	21,144
17	bushel	21,074
18	last	20,663
19	grain	18,619
20	farmer	17,653

จากรูปที่ 4.3 แสดงความถี่ของ 20 คำศัพท์ที่พบบ่อยที่สุดในข้อมูลชุดเรียนรู้จากคลังคำศัพท์ 5,000 คำ โดยเรียงลำดับความถี่คำศัพท์ที่พบบ่อยที่สุดดังตารางที่ 4.2 สามารถอธิบายได้ดังนี้ คำศัพท์ที่พบบ่อยที่สุดในข้อมูลชุดเรียนรู้ของงานวิจัยนี้ได้แก่ “corn” มีความถี่มากที่สุดคือ 60,369 รองลงมาคือ “said” และ “us” มีความถี่เท่ากับ 50,100 และ 40,556 ตามลำดับ

4.1 ผลการทดสอบประสิทธิภาพ คลังคำศัพท์ 5,000 คำ

4.1.1 ผลการทดสอบโดยใช้การแปลงเชิงปริมาณ Bag of Words และแบบจำลองการถดถอยลอจิสติก คลังคำศัพท์ 5,000 คำ

การทดสอบการพยากรณ์ทิศทางราคาโดยใช้การแปลงเชิงปริมาณ Bag of Words และแบบจำลองการถดถอยลอจิสติก คลังคำศัพท์ 5,000 คำ ซึ่งแบบจำลองสามารถทำนายชุดทดสอบออกมาว่าเป็นตลาดกระทิงหรือตลาดขาขึ้น (Bullish) 217 วัน และตลาดหมีหรือตลาดขาลง (Bearish) 196 วัน ซึ่งสามารถสรุปการทำนายตามตารางที่ 4.3 และมีประสิทธิภาพการทำนายดังตารางที่ 4.4

ตารางที่ 4.3 เมทริกซ์ความสับสน (Confusion Matrix) ของการแปลงเชิงปริมาณ Bag of Words และแบบจำลองการถดถอยลอจิสติก คลังคำศัพท์ 5,000 คำ

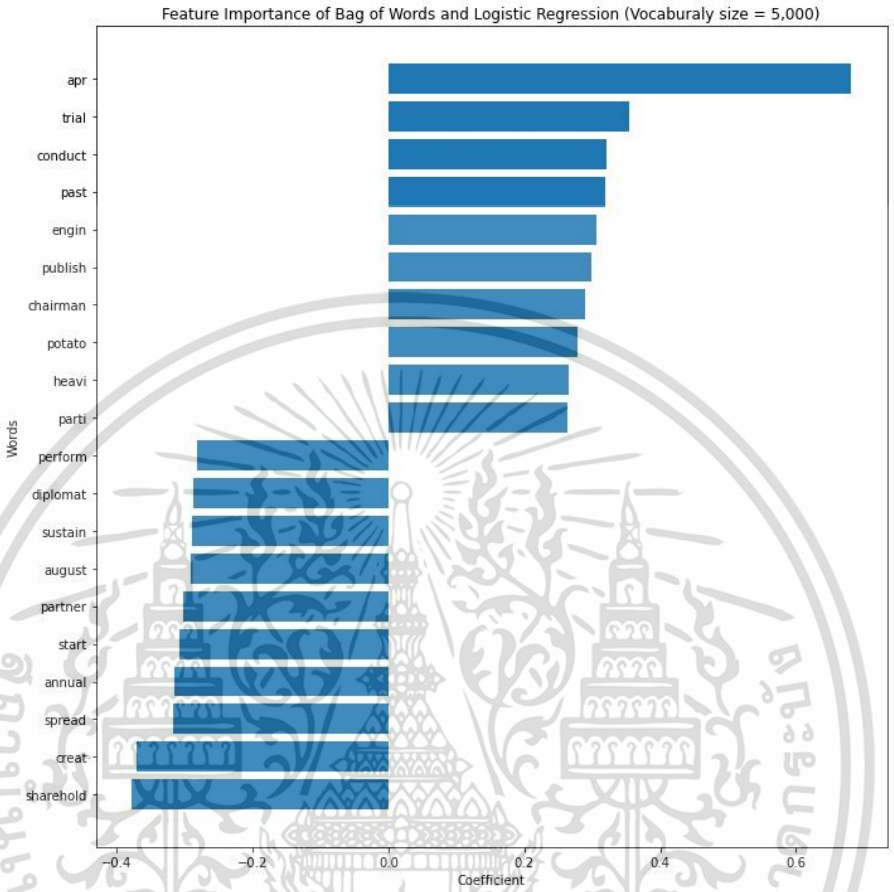
		ทิศทางที่แบบจำลองทำนาย		รวม
		0 (Bearish)	1 (Bullish)	
ทิศทางจากชุดทดสอบ	0 (Bearish)	96	120	216
	1 (Bullish)	100	97	197
รวม		196	217	413

ตารางที่ 4.4 ประสิทธิภาพการทำนายจากการแปลงเชิงปริมาณ Bag of Words และแบบจำลองการถดถอยลอจิสติก คลังคำศัพท์ 5,000 คำ

ทิศทางราคา	ประสิทธิภาพการทำนายของแบบจำลอง			
	Precision	Recall	F1-Score	Accuracy
0 (Bearish)	0.4898	0.4444	0.4666	0.4673
1 (Bullish)	0.4470	0.4924	0.4686	
Average	0.4684	0.4684	0.4673	

จากตารางที่ 4.4 แสดงประสิทธิภาพการทำนายจากการแปลงเชิงปริมาณ Bag of Words และแบบจำลองการถดถอยลอจิสติก คลังคำศัพท์ 5,000 คำ จะเห็นได้ว่ามีค่าเฉลี่ย F1-Score เป็น 46.73% โดยตลาดขาลงมีค่า F1-Score เท่ากับ 44.44% และตลาดขาขึ้นมีค่า F1-Score เท่ากับ

49.24% และมีความแม่นยำอยู่ที่ 46.73% หมายความว่า แบบจำลองสามารถทำนายตลาดขาขึ้นได้ดีกว่าเล็กน้อย



รูปที่ 4.4 Feature Importance 20 อันดับแรกของการแปลงเชิงปริมาณ Bag of Words และแบบจำลองการถดถอยลอจิสติก คลังคำศัพท์ 5,000 คำ

จากภาพที่ 4.4 คือ Feature Importance 20 อันดับแรกของการแปลงเชิงปริมาณ Bag of Words และแบบจำลองการถดถอยลอจิสติก คลังคำศัพท์ 5,000 คำ วัดจากค่า Coefficient โดยคำศัพท์ที่มีค่า Coefficient สูงที่สุดทางด้านบวก คือ “apr” มีค่า Coefficient เท่ากับ 0.6809 หมายความว่ามีอิทธิพลที่เพิ่มขึ้นต่อการทำนายผลลัพธ์ในเชิงบวก ดังนั้นเมื่อ “apr” มีค่าเพิ่มขึ้นหนึ่งหน่วยโอกาสที่ผลลัพธ์จะเป็นคลาสที่สนใจจะเพิ่มขึ้น 0.6809 ในทางกลับกันคำศัพท์ที่มีค่า Coefficient สูงที่สุดทางด้านลบ คือ “sharehold” มีค่า Coefficient เท่ากับ -0.3782 หมายความว่ามีอิทธิพลที่ลดลงต่อการทำนายผลลัพธ์ในเชิงบวก ดังนั้นเมื่อ “sharehold” มีค่าเพิ่มขึ้นหนึ่งหน่วยโอกาสที่ผลลัพธ์จะเป็นคลาสที่สนใจจะลดลง -0.3782

4.1.2 ผลการทดสอบโดยใช้การแปลงเชิงปริมาณ Bag of Words และแบบจำลองซัพพอร์ตเวกเตอร์แมชชีน คลังคำศัพท์ 5,000 คำ

การทดสอบการพยากรณ์ทิศทางราคาโดยใช้การแปลงเชิงปริมาณ Bag of Words และแบบจำลองซัพพอร์ตเวกเตอร์แมชชีน คลังคำศัพท์ 5,000 คำ ซึ่งแบบจำลองสามารถทำนายชุดทดสอบออกมาว่าเป็นตลาดกระทิงหรือตลาดขาขึ้น (Bullish) 208 วัน และตลาดหมีหรือตลาดขาลง (Bearish) 205 วัน ซึ่งสามารถสรุปการทำนายตามตารางที่ 4.5 และมีประสิทธิภาพการทำนายดังตารางที่ 4.6

ตารางที่ 4.5 เมทริกซ์ความสับสน (Confusion Matrix) ของการแปลงเชิงปริมาณ Bag of Words และแบบจำลองซัพพอร์ตเวกเตอร์แมชชีน คลังคำศัพท์ 5,000 คำ

		ทิศทางที่แบบจำลองทำนาย		รวม
		0 (Bearish)	1 (Bullish)	
ทิศทางจากชุดทดสอบ	0 (Bearish)	109	107	216
	1 (Bullish)	96	101	197
รวม		205	208	413

ตารางที่ 4.6 ประสิทธิภาพการทำนายจากการแปลงเชิงปริมาณ Bag of Words และแบบจำลองซัพพอร์ตเวกเตอร์แมชชีน คลังคำศัพท์ 5,000 คำ

ทิศทางราคา	**ประสิทธิภาพการทำนายของแบบจำลอง**			
	Precision	Recall	F1-Score	Accuracy
0 (Bearish)	0.5317	0.5046	0.5178	0.5085
1 (Bullish)	0.4856	0.5127	0.4988	
Average	0.5086	0.5087	0.5083	

จากตารางที่ 4.6 แสดงประสิทธิภาพการทำนายจากการแปลงเชิงปริมาณ Bag of Words และแบบจำลองซัพพอร์ตเวกเตอร์แมชชีน คลังคำศัพท์ 5,000 คำ จะเห็นได้ว่ามีค่าเฉลี่ย F1-Score เป็น 50.83% โดยตลาดขาลงมีค่า F1-Score เท่ากับ 51.78% และตลาดขาขึ้นมีค่า F1-Score เท่ากับ

49.88% และมีความแม่นยำอยู่ที่ 50.85% หมายความว่า แบบจำลองสามารถทำนายตลาดขาลงได้ดีกว่า

4.1.3 ผลการทดสอบโดยใช้การแปลงเชิงปริมาณ Bag of Words และแบบจำลองโครงข่ายประสาทเทียม คลังคำศัพท์ 5,000 คำ

การทดสอบการพยากรณ์ทิศทางราคาโดยใช้การแปลงเชิงปริมาณ Bag of Words และแบบจำลองโครงข่ายประสาทเทียม คลังคำศัพท์ 5,000 คำ ซึ่งแบบจำลองสามารถทำนายชุดทดสอบออกมาว่าเป็นตลาดกระทิงหรือตลาดขาขึ้น (Bullish) 384 วัน และตลาดหมีหรือตลาดขาลง (Bearish) 29 วัน ซึ่งสามารถสรุปการทำนายตามตารางที่ 4.7 และมีประสิทธิภาพการทำนายดังตารางที่ 4.8

ตารางที่ 4.7 เมทริกซ์ความสับสน (Confusion Matrix) ของการแปลงเชิงปริมาณ Bag of Words และแบบจำลองโครงข่ายประสาทเทียม คลังคำศัพท์ 5,000 คำ

		ทิศทางที่แบบจำลองทำนาย		รวม
		0 (Bearish)	1 (Bullish)	
ทิศทางจากชุดทดสอบ	0 (Bearish)	17	199	216
	1 (Bullish)	12	185	197
รวม		29	384	413

ตารางที่ 4.8 ประสิทธิภาพการทำนายจากการแปลงเชิงปริมาณ Bag of Words และแบบจำลองโครงข่ายประสาทเทียม คลังคำศัพท์ 5,000 คำ

ทิศทางราคา	**ประสิทธิภาพการทำนายของแบบจำลอง**			
	Precision	Recall	F1-Score	Accuracy
0 (Bearish)	0.5862	0.0787	0.1388	0.4891
1 (Bullish)	0.4818	0.9391	0.6368	
Average	0.5340	0.5089	0.3878	

จากตารางที่ 4.8 แสดงประสิทธิภาพการทำนายจากการแปลงเชิงปริมาณ Bag of Words และแบบจำลองโครงข่ายประสาทเทียม คลังคำศัพท์ 5,000 คำ จะเห็นได้ว่ามีค่าเฉลี่ย F1-Score เป็น

38.78% โดยตลาดขาลงมีค่า F1-Score เท่ากับ 13.88% และตลาดขาขึ้นมีค่า F1-Score เท่ากับ 63.68% และมีความแม่นยำอยู่ที่ 48.91% หมายความว่า แบบจำลองสามารถทำนายตลาดขาขึ้นได้ดีกว่า

4.1.4 ผลการทดสอบโดยใช้การแปลงเชิงปริมาณ TF-IDF และแบบจำลองการถดถอยลอจิสติก คลังคำศัพท์ 5,000 คำ

การทดสอบการพยากรณ์ทิศทางราคาโดยใช้การแปลงเชิงปริมาณ TF-IDF และแบบจำลองการถดถอยลอจิสติก คลังคำศัพท์ 5,000 คำ ซึ่งแบบจำลองสามารถทำนายชุดทดสอบออกมาว่าเป็นตลาดกระทิงหรือตลาดขาขึ้น (Bullish) 192 วัน และตลาดหมีหรือตลาดขาลง (Bearish) 221 วัน ซึ่งสามารถสรุปการทำนายตามตารางที่ 4.9 และมีประสิทธิภาพการทำนายดังตารางที่ 4.10

ตารางที่ 4.9 เมทริกซ์ความสับสน (Confusion Matrix) ของการแปลงเชิงปริมาณ TF-IDF และแบบจำลองการถดถอยลอจิสติก คลังคำศัพท์ 5,000 คำ

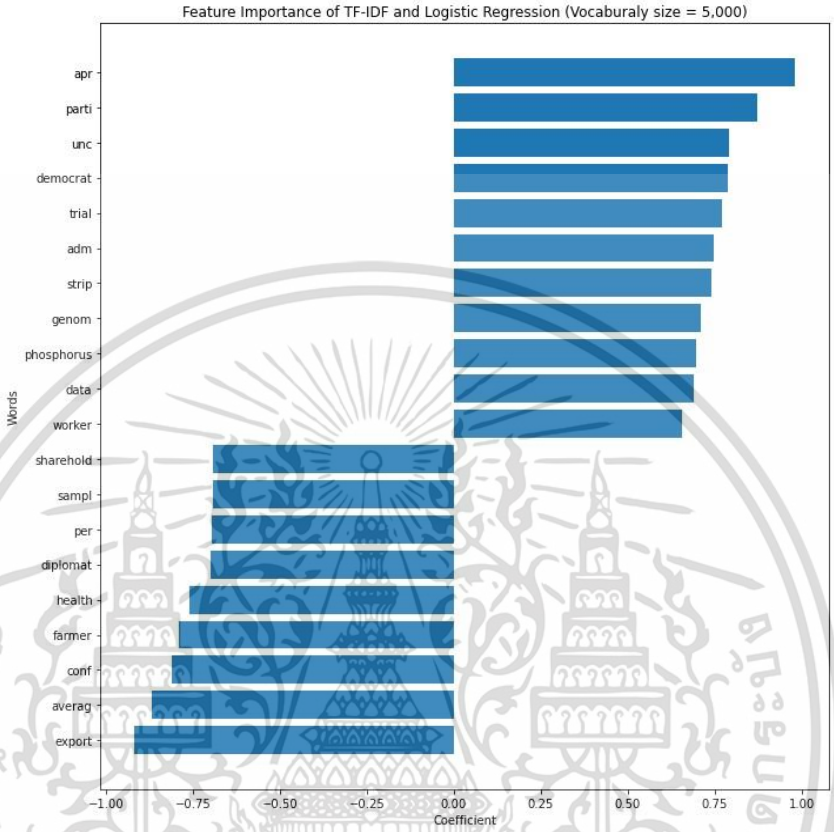
		ทิศทางที่แบบจำลองทำนาย		รวม
		0 (Bearish)	1 (Bullish)	
ทิศทางจากชุดทดสอบ	0 (Bearish)	113	103	216
	1 (Bullish)	108	89	197
รวม		221	192	413

ตารางที่ 4.10 ประสิทธิภาพการทำนายจากการแปลงเชิงปริมาณ TF-IDF และแบบจำลองการถดถอยลอจิสติก คลังคำศัพท์ 5,000 คำ

ทิศทางราคา	**ประสิทธิภาพการทำนายของแบบจำลอง**			
	Precision	Recall	F1-Score	Accuracy
0 (Bearish)	0.5113	0.5231	0.5172	0.4891
1 (Bullish)	0.4635	0.4518	0.4576	
Average	0.4874	0.4875	0.4874	

จากตารางที่ 4.10 แสดงประสิทธิภาพการทำนายจากการแปลงเชิงปริมาณ TF-IDF และแบบจำลองการถดถอยลอจิสติก คลังคำศัพท์ 5,000 คำ จะเห็นได้ว่ามีค่าเฉลี่ย F1-Score เป็น

48.74% โดยตลาดขาลงมีค่า F1-Score เท่ากับ 51.72% และตลาดขาขึ้นมีค่า F1-Score เท่ากับ 45.76% และมีความแม่นยำอยู่ที่ 48.91% หมายความว่า แบบจำลองสามารถทำนายตลาดขาลงได้ดีกว่าเล็กน้อย



รูปที่ 4.5 Feature Importance 20 อันดับแรกของการแปลงเชิงปริมาณ TF-IDF และแบบจำลองการถดถอยลอจิสติก คลังคำศัพท์ 5,000 คำ

จากภาพที่ 4.5 คือ Feature Importance 20 อันดับแรกขอการแปลงเชิงปริมาณ TF-IDF และแบบจำลองการถดถอยลอจิสติก คลังคำศัพท์ 5,000 คำ วัดจากค่า Coefficient โดยคำศัพท์ที่มีค่า Coefficient สูงที่สุดทางด้านบวก คือ “apr” มีค่า Coefficient เท่ากับ 0.9826 หมายความว่ามีอิทธิพลที่เพิ่มขึ้นต่อการทำนายผลลัพธ์ในเชิงบวก ดังนั้นเมื่อ “apr” มีค่าเพิ่มขึ้นหนึ่งหน่วยโอกาสที่ผลลัพธ์จะเป็นคลาสที่สนใจจะเพิ่มขึ้น 0.9826 ในทางกลับกันคำศัพท์ที่มีค่า Coefficient สูงที่สุดทางด้านลบ คือ “export” มีค่า Coefficient เท่ากับ -0.9212 หมายความว่ามีอิทธิพลที่ลดลงต่อการทำนายผลลัพธ์ในเชิงบวก ดังนั้นเมื่อ “export” มีค่าเพิ่มขึ้นหนึ่งหน่วยโอกาสที่ผลลัพธ์จะเป็นคลาสที่สนใจจะลดลง -0.9212

4.1.5 ผลการทดสอบโดยใช้การแปลงเชิงปริมาณ TF-IDF และแบบจำลองซัพพอร์ตเวกเตอร์แมชชีน คลังคำศัพท์ 5,000 คำ

การทดสอบการพยากรณ์ทิศทางราคาโดยใช้การแปลงเชิงปริมาณ TF-IDF และแบบจำลองซัพพอร์ตเวกเตอร์แมชชีน คลังคำศัพท์ 5,000 คำ ซึ่งแบบจำลองสามารถทำนายชุดทดสอบออกมาว่าเป็นตลาดกระทิงหรือตลาดขาขึ้น (Bullish) 166 วัน และตลาดหมีหรือตลาดขาลง (Bearish) 247 วัน ซึ่งสามารถสรุปการทำนายตามตารางที่ 4.11 และมีประสิทธิภาพการทำนายดังตารางที่ 4.12

ตารางที่ 4.11 เมทริกซ์ความสับสน (Confusion Matrix) ของการแปลงเชิงปริมาณ TF-IDF และแบบจำลองซัพพอร์ตเวกเตอร์แมชชีน คลังคำศัพท์ 5,000 คำ

		ทิศทางที่แบบจำลองทำนาย		รวม
		0 (Bearish)	1 (Bullish)	
ทิศทางจากชุดทดสอบ	0 (Bearish)	126	90	216
	1 (Bullish)	121	76	197
รวม		247	166	413

ตารางที่ 4.12 ประสิทธิภาพการทำนายจากการแปลงเชิงปริมาณ TF-IDF และแบบจำลองซัพพอร์ตเวกเตอร์แมชชีน คลังคำศัพท์ 5,000 คำ

ทิศทางราคา	ประสิทธิภาพการทำนายของแบบจำลอง			
	Precision	Recall	F1-Score	Accuracy
0 (Bearish)	0.5101	0.5833	0.5443	0.4891
1 (Bullish)	0.4578	0.3858	0.4187	
Average	0.4840	0.4846	0.4815	

จากตารางที่ 4.12 แสดงประสิทธิภาพการทำนายจากการแปลงเชิงปริมาณ TF-IDF และแบบจำลองซัพพอร์ตเวกเตอร์แมชชีน คลังคำศัพท์ 5,000 คำ จะเห็นได้ว่ามีค่าเฉลี่ย F1-Score เป็น 48.15% โดยตลาดขาลงมีค่า F1-Score เท่ากับ 54.43% และตลาดขาขึ้นมีค่า F1-Score เท่ากับ 41.87% และมีความแม่นยำอยู่ที่ 48.91% หมายความว่า แบบจำลองสามารถทำนายตลาดขาลงได้ดีกว่าเล็กน้อย

4.1.6 ผลการทดสอบโดยใช้การแปลงเชิงปริมาณ TF-IDF และแบบจำลองโครงข่ายประสาทเทียม คลังคำศัพท์ 5,000 คำ

การทดสอบการพยากรณ์ทิศทางราคาโดยใช้การแปลงเชิงปริมาณ TF-IDF และแบบจำลองโครงข่ายประสาทเทียม คลังคำศัพท์ 5,000 คำ ซึ่งแบบจำลองสามารถทำนายชุดทดสอบออกมาว่าเป็นตลาดกระทิงหรือตลาดขาขึ้น (Bullish) 191 วัน และตลาดหมีหรือตลาดขาลง (Bearish) 222 วัน ซึ่งสามารถสรุปการทำนายตามตารางที่ 4.13 และมีประสิทธิภาพการทำนายดังตารางที่ 4.14

ตารางที่ 4.13 เมทริกซ์ความสับสน (Confusion Matrix) ของการแปลงเชิงปริมาณ TF-IDF และแบบจำลองโครงข่ายประสาทเทียม คลังคำศัพท์ 5,000 คำ

		ทิศทางที่แบบจำลองทำนาย		รวม
		0 (Bearish)	1 (Bullish)	
ทิศทางจากชุดทดสอบ	0 (Bearish)	115	101	216
	1 (Bullish)	107	90	197
รวม		222	191	413

ตารางที่ 4.14 ประสิทธิภาพการทำนายจากการแปลงเชิงปริมาณ TF-IDF และแบบจำลองโครงข่ายประสาทเทียม คลังคำศัพท์ 5,000 คำ

ทิศทางราคา	ประสิทธิภาพการทำนายของแบบจำลอง			
	Precision	Recall	F1-Score	Accuracy
0 (Bearish)	0.5180	0.5324	0.5251	0.4964
1 (Bullish)	0.4712	0.4569	0.4639	
Average	0.4946	0.4946	0.4945	

จากตารางที่ 4.14 แสดงประสิทธิภาพการทำนายจากการแปลงเชิงปริมาณ TF-IDF และแบบจำลองโครงข่ายประสาทเทียม คลังคำศัพท์ 5,000 คำ จะเห็นได้ว่ามีค่าเฉลี่ย F1-Score เป็น 49.45% โดยตลาดขาลงมีค่า F1-Score เท่ากับ 52.51% และตลาดขาขึ้นมีค่า F1-Score เท่ากับ 46.39% และมีความแม่นยำอยู่ที่ 49.64% หมายความว่า แบบจำลองสามารถทำนายตลาดขาลง ได้ดีกว่าเล็กน้อย

4.1.7 ผลการทดสอบโดยใช้การแปลงคุณลักษณะ Word2Vec และแบบจำลองการถดถอยลอจิสติก เวกเตอร์คุณลักษณะจำนวน 5,000 ตัว

การทดสอบการพยากรณ์ทิศทางราคาโดยใช้การแปลงคุณลักษณะ Word2Vec และแบบจำลองการถดถอยลอจิสติก เวกเตอร์คุณลักษณะจำนวน 5,000 ตัว ซึ่งแบบจำลองสามารถทำนายชุดทดสอบออกมาว่าเป็นตลาดกระทิงหรือตลาดขาขึ้น (Bullish) 187 วัน และตลาดหมีหรือตลาดขาลง (Bearish) 226 วัน ซึ่งสามารถสรุปการทำนายตามตารางที่ 4.15 และมีประสิทธิภาพการทำนายดังตารางที่ 4.16

ตารางที่ 4.15 เมทริกซ์ความสับสน (Confusion Matrix) ของการแปลงคุณลักษณะ Word2Vec และแบบจำลองการถดถอยลอจิสติก เวกเตอร์คุณลักษณะจำนวน 5,000 ตัว

		ทิศทางที่แบบจำลองทำนาย		รวม
		0 (Bearish)	1 (Bullish)	
ทิศทางจากชุดทดสอบ	0 (Bearish)	120	96	216
	1 (Bullish)	106	91	197
รวม		226	187	413

ตารางที่ 4.16 ประสิทธิภาพการทำนายจากการแปลงคุณลักษณะ Word2Vec และแบบจำลองการถดถอยลอจิสติก เวกเตอร์คุณลักษณะจำนวน 5,000 ตัว

ทิศทางราคา	ประสิทธิภาพการทำนายของแบบจำลอง			
	Precision	Recall	F1-Score	Accuracy
0 (Bearish)	0.5310	0.5556	0.5430	0.5109
1 (Bullish)	0.4866	0.4619	0.4740	
Average	0.5088	0.5087	0.5085	

จากตารางที่ 4.16 แสดงประสิทธิภาพการทำนายจากการแปลงคุณลักษณะ Word2Vec และแบบจำลองการถดถอยลอจิสติก เวกเตอร์คุณลักษณะจำนวน 5,000 ตัว จะเห็นได้ว่ามีค่าเฉลี่ย F1-Score เป็น 50.85% โดยตลาดขาลงมีค่า F1-Score เท่ากับ 54.30% และตลาดขาขึ้นมีค่า F1-

Score เท่ากับ 47.40% และมีความแม่นยำอยู่ที่ 51.09% หมายความว่า แบบจำลองสามารถทำนายตลาดขาลงได้ดีกว่าเล็กน้อย

4.1.8 ผลการทดสอบโดยใช้การแปลงคุณลักษณะ Word2Vec และแบบจำลองซัพพอร์ตเวกเตอร์แมชชีน เวกเตอร์คุณลักษณะจำนวน 5,000 ตัว

การทดสอบการพยากรณ์ทิศทางราคาโดยใช้การแปลงคุณลักษณะ Word2Vec และแบบจำลองซัพพอร์ตเวกเตอร์แมชชีน เวกเตอร์คุณลักษณะจำนวน 5,000 ตัว แบบจำลองสามารถทำนายชุดทดสอบออกมาว่าเป็นตลาดกระทิงหรือตลาดขาขึ้น (Bullish) 2 วัน และตลาดหมีหรือตลาดขาลง (Bearish) 412 วัน ซึ่งสามารถสรุปการทำนายตามตารางที่ 4.17 และมีประสิทธิภาพการทำนายดังตารางที่ 4.18

ตารางที่ 4.17 เมทริกซ์ความสับสน (Confusion Matrix) ของการแปลงคุณลักษณะ Word2Vec และแบบจำลองซัพพอร์ตเวกเตอร์แมชชีน เวกเตอร์คุณลักษณะจำนวน 5,000 ตัว

		ทิศทางที่แบบจำลองทำนาย		รวม
		0 (Bearish)	1 (Bullish)	
ทิศทางจากชุดทดสอบ	0 (Bearish)	215	1	216
	1 (Bullish)	197	0	197
รวม		412	1	413

ตารางที่ 4.18 ประสิทธิภาพการทำนายจากการแปลงคุณลักษณะ Word2Vec และแบบจำลองซัพพอร์ตเวกเตอร์แมชชีน เวกเตอร์คุณลักษณะจำนวน 5,000 ตัว

ทิศทางราคา	**ประสิทธิภาพการทำนายของแบบจำลอง**			
	Precision	Recall	F1-Score	Accuracy
0 (Bearish)	0.5218	0.9954	0.6847	0.5206
1 (Bullish)	0.0000	0.0000	0.0000	
Average	0.2609	0.4977	0.3424	

จากตารางที่ 4.18 แสดงประสิทธิภาพการทำนายจากการแปลงคุณลักษณะ Word2Vec และแบบจำลองซัพพอร์ตเวกเตอร์แมชชีน เวกเตอร์คุณลักษณะจำนวน 5,000 ตัว มีค่าเฉลี่ย

F1-Score เป็น 34.24% โดยตลาดขาลงมีค่า F1-Score เท่ากับ 68.47% และตลาดขาขึ้นมีค่า F1-Score เท่ากับ 0.00% และมีความแม่นยำอยู่ที่ 52.06% หมายความว่า แบบจำลองสามารถทำนายตลาดขาลงได้ดีกว่า

4.1.9 ผลการทดสอบโดยใช้การแปลงคุณลักษณะ Word2Vec และแบบจำลองโครงข่ายประสาทเทียม เวกเตอร์คุณลักษณะจำนวน 5,000 ตัว

การทดสอบการพยากรณ์ทิศทางราคาโดยใช้การแปลงคุณลักษณะ Word2Vec และแบบจำลองโครงข่ายประสาทเทียม เวกเตอร์คุณลักษณะจำนวน 5,000 ตัว แบบจำลองสามารถทำนายชุดทดสอบออกมาว่าเป็นตลาดกระทิงหรือตลาดขาขึ้น (Bullish) 8 วัน และตลาดหมีหรือตลาดขาลง (Bearish) 405 วัน ซึ่งสามารถสรุปการทำนายตามตารางที่ 4.19 และมีประสิทธิภาพการทำนายดังตารางที่ 4.20

ตารางที่ 4.19 เมทริกซ์ความสับสน (Confusion Matrix) ของการแปลงคุณลักษณะ Word2Vec และแบบจำลองโครงข่ายประสาทเทียม เวกเตอร์คุณลักษณะจำนวน 5,000 ตัว

		ทิศทางที่แบบจำลองทำนาย		รวม
		0 (Bearish)	1 (Bullish)	
ทิศทางจากชุดทดสอบ	0 (Bearish)	213	3	216
	1 (Bullish)	192	5	197
รวม		405	8	413

ตารางที่ 4.20 ประสิทธิภาพการทำนายจากการแปลงคุณลักษณะ Word2Vec และแบบจำลองโครงข่ายประสาทเทียม เวกเตอร์คุณลักษณะจำนวน 5,000 ตัว

ทิศทางราคา	**ประสิทธิภาพการทำนายของแบบจำลอง**			
	Precision	Recall	F1-Score	Accuracy
0 (Bearish)	0.5259	0.9861	0.6860	0.5278
1 (Bullish)	0.6250	0.0254	0.0488	
Average	0.5755	0.5057	0.3674	

จากตารางที่ 4.20 แสดงประสิทธิภาพการทำนายจากการแปลงคุณลักษณะ Word2Vec และแบบจำลองโครงข่ายประสาทเทียม เวกเตอร์คุณลักษณะจำนวน 5,000 ตัว จะเห็นได้ว่ามีค่าเฉลี่ย F1-Score เป็น 36.74% โดยตลาดขาลงมีค่า F1-Score เท่ากับ 68.60% และตลาดขาขึ้นมีค่า F1-Score เท่ากับ 4.88% และมีความแม่นยำอยู่ที่ 52.78% หมายความว่า แบบจำลองสามารถทำนายตลาดขาลงได้ดีกว่า

4.2 การเปรียบเทียบประสิทธิภาพ คลังคำศัพท์ 5,000 คำ

4.2.1 การเปรียบเทียบประสิทธิภาพของแบบจำลอง คลังคำศัพท์ 5,000 คำ

ค่าความถ่วงดุล (F1-Score) และค่าความแม่นยำ (Accuracy) ของแบบจำลองต่าง ๆ ในการการพยากรณ์ทิศทางราคาสัญญาฟิวเจอร์สข้าวโพดจากข้อความข่าวโดยใช้การประมวลภาษาธรรมชาติ จากชุดข้อมูลทดสอบ ได้ผลดังตารางที่ 4.21, ตารางที่ 4.22 และตารางที่ 4.23

ตารางที่ 4.21 เปรียบเทียบประสิทธิภาพของแบบจำลองทั้ง 9 แบบจำลอง จากร้อยละของค่าความแม่นยำ Accuracy (ลำดับของค่าความแม่นยำ) คลังคำศัพท์ 5,000 คำ

การแปลงคุณลักษณะ	แบบจำลอง	ร้อยละของค่าความแม่นยำ Accuracy (ลำดับที่)	
		ข้อมูลชุดเรียนรู้	ข้อมูลชุดทดสอบ
Bag of Words	Logistic Regression	100.00	46.73 (7)
	SVM	83.54	50.85 (4)
	ANN	57.87	48.91 (6)
TF-IDF	Logistic Regression	84.26	48.91 (6)
	SVM	96.43	48.91 (6)
	ANN	99.94	49.64 (5)
Word2Vec	Logistic Regression	61.86	51.09 (3)
	SVM	52.54	52.06 (2)
	ANN	52.60	52.78 (1)

จากการวัดประสิทธิภาพของแบบจำลองหากพิจารณาจากร้อยละของค่าความแม่นยำ Accuracy คลังคำศัพท์ 5,000 คำ พบว่าการแปลงคุณลักษณะ Word2Vec และแบบจำลองโครงข่ายประสาทเทียม มีความแม่นยำในการพยากรณ์ดีที่สุดคือ 52.78% รองลงมาคือการแปลงคุณลักษณะ

Word2Vec และแบบจำลองซัพพอร์ตเวกเตอร์แมชชีน และการแปลงคุณลักษณะ Word2Vec และแบบจำลองการถดถอยลอจิสติก มีความแม่นยำในการพยากรณ์เท่ากับ 52.06% และ 51.09% ตามลำดับ ในทางกลับกันการแปลงคุณลักษณะ Bag of Words และแบบจำลองการถดถอยลอจิสติก มีความแม่นยำในการพยากรณ์ต่ำที่สุดคือ 46.73%

ตารางที่ 4.22 เปรียบเทียบประสิทธิภาพของแบบจำลองทั้ง 9 แบบจำลอง จากร้อยละของค่าความถ่วงดุล F1-Score (ลำดับของ F1-Score) คลังคำศัพท์ 5,000 คำ

การแปลงคุณลักษณะ	แบบจำลอง	ร้อยละของค่าความถ่วงดุล F1-Score (ลำดับที่)	
		ข้อมูลชุดเรียนรู้	ข้อมูลชุดทดสอบ
Bag of Words	Logistic Regression	100.00	46.73 (6)
	SVM	83.26	50.83 (2)
	ANN	51.71	38.78 (7)
TF-IDF	Logistic Regression	83.98	48.74 (4)
	SVM	96.41	48.15 (5)
	ANN	99.94	49.45 (3)
Word2Vec	Logistic Regression	61.01	50.85 (1)
	SVM	34.79	34.24 (9)
	ANN	35.92	36.74 (8)

จากการวัดประสิทธิภาพของแบบจำลองหากพิจารณาจากร้อยละของค่าความถ่วงดุล F1-Score คลังคำศัพท์ 5,000 คำ พบว่าการแปลงคุณลักษณะ Word2Vec และแบบจำลองการถดถอยลอจิสติก มีค่า F1-Score ดีที่สุดคือ 50.85% รองลงมาคือการแปลงคุณลักษณะ Bag of Words และแบบจำลองซัพพอร์ตเวกเตอร์แมชชีน และการแปลงคุณลักษณะ TF-IDF และแบบจำลองโครงข่ายประสาทเทียม มีค่า F1-Score เท่ากับ 50.83% และ 49.45% ตามลำดับ ในทางกลับกันการแปลงคุณลักษณะ Word2Vec และแบบจำลองซัพพอร์ตเวกเตอร์แมชชีน มีค่า F1-Score ต่ำที่สุดเท่ากับ 34.24%

ตารางที่ 4.23 เปรียบเทียบประสิทธิภาพของแบบจำลองทั้ง 9 แบบจำลอง จากร้อยละของค่าความถ่วงดุล F1-Score (ลำดับของ F1-Score) และร้อยละของค่าความแม่นยำ Accuracy (ลำดับของค่าความแม่นยำ) ร่วมกัน คลังคำศัพท์ 5,000 คำ

การแปลงคุณลักษณะ	แบบจำลอง	ข้อมูลชุดทดสอบ	
		ค่าความแม่นยำ Accuracy	ค่าความถ่วงดุล F1-Score
Bag of Words	Logistic Regression	46.73	46.73 (6)
	SVM	50.85	50.83 (2)
	ANN	48.91	38.78 (7)
TF-IDF	Logistic Regression	48.91	48.74 (4)
	SVM	48.91	48.15 (5)
	ANN	49.64	49.45 (3)
Word2Vec	Logistic Regression	51.09	50.85 (1)
	SVM	52.06	34.24 (9)
	ANN	52.78	36.74 (8)

หากพิจารณาโดยใช้ร้อยละของค่า F1-Score เป็นอันดับแรกในการวัดผล และใช้ค่า Accuracy ร่วมกัน คลังคำศัพท์ 5,000 คำ พบว่าการแปลงคุณลักษณะ Word2Vec และแบบจำลองการถดถอยลอจิสติก มีค่า F1-Score ดีที่สุดคือ 50.85% และมีค่าความแม่นยำในการพยากรณ์ดีที่สุดคือ 51.09% รองลงมาคือการแปลงคุณลักษณะ Bag of Words และแบบจำลองซัพพอร์ตเวกเตอร์แมชชีน และการแปลงคุณลักษณะ TF-IDF และแบบจำลองโครงข่ายประสาทเทียม มีค่า F1-Score เท่ากับ 50.83% และ 49.45% ตามลำดับ และมีค่าความแม่นยำในการพยากรณ์เท่ากับ 50.85% และ 49.64% ตามลำดับ

เนื่องจาก ตารางที่ 4.21 เปรียบเทียบประสิทธิภาพของแบบจำลองทั้ง 9 แบบจำลอง จากร้อยละของค่าความแม่นยำ Accuracy (ลำดับของค่าความแม่นยำ) คลังคำศัพท์ 5,000 คำ และตารางที่ 4.22 เปรียบเทียบประสิทธิภาพของแบบจำลองทั้ง 9 แบบจำลอง จากร้อยละของค่าความถ่วงดุล F1-Score (ลำดับของ F1-Score) คลังคำศัพท์ 5,000 คำ จะสังเกตได้ว่าชุดข้อมูลการเรียนรู้มีค่าสูงมากกว่าชุดข้อมูลทดสอบแบบผิดปรกติ ดังแสดงในตารางที่ 4.24 และ 4.25

ตารางที่ 4.24 เปรียบเทียบปัญหา Overfitting จากค่า Accuracy คลังคำศัพท์ 5,000 คำ

การแปลงคุณลักษณะ	แบบจำลอง	ร้อยละของค่าความแม่นยำ Accuracy (ลำดับที่)	
		ข้อมูลชุดเรียนรู้	ข้อมูลชุดทดสอบ
Bag of Words	Logistic Regression	100.00	46.73 (7)
	SVM	83.54	50.85 (4)
	ANN	57.87	48.91 (6)
TF-IDF	Logistic Regression	84.26	48.91 (6)
	SVM	96.43	48.91 (6)
	ANN	99.94	49.64 (5)
Word2Vec	Logistic Regression	61.86	51.09 (3)
	SVM	52.54	52.06 (2)
	ANN	52.60	52.78 (1)

ตารางที่ 4.25 เปรียบเทียบปัญหา Overfitting จากค่า F1-Score คลังคำศัพท์ 5,000 คำ

การแปลงคุณลักษณะ	แบบจำลอง	ร้อยละของค่าความถ่วงดุล F1-Score (ลำดับที่)	
		ข้อมูลชุดเรียนรู้	ข้อมูลชุดทดสอบ
Bag of Words	Logistic Regression	100.00	46.73 (6)
	SVM	83.26	50.83 (2)
	ANN	51.71	38.78 (7)
TF-IDF	Logistic Regression	83.98	48.74 (4)
	SVM	96.41	48.15 (5)
	ANN	99.94	49.45 (3)
Word2Vec	Logistic Regression	61.01	50.85 (1)
	SVM	34.79	34.24 (9)
	ANN	35.92	36.74 (8)

จากตารางที่ 4.24 และ 4.25 พบว่าชุดข้อมูลการเรียนรู้มีค่าสูงมากกว่าชุดข้อมูลทดสอบแบบผิดปรกติใน 5 แบบจำลอง ทำให้สันนิษฐานได้ว่าอาจจะเกิดปัญหาพอดีเกินไป (Overfitting) ในแบบจำลองดังนี้ 1) การแปลงคุณลักษณะ Bag of Words และแบบจำลองการถดถอยลอจิสติก

2) การแปลงคุณลักษณะ Bag of Words และแบบจำลองซัพพอร์ตเวกเตอร์แมชชีน 3) การแปลงคุณลักษณะ TF-IDF และแบบจำลองการถดถอยลอจิสติก 4) การแปลงคุณลักษณะ TF-IDF และแบบจำลองซัพพอร์ตเวกเตอร์แมชชีน 5) การแปลงคุณลักษณะ TF-IDF และแบบจำลองโครงข่ายประสาทเทียม

ผู้วิจัยจึงแก้ไขปัญหาพอดีเกินไป (Overfitting) ที่เกิดขึ้นด้วยวิธี เลือกฟีเจอร์ที่จำเป็น (Feature Selection) วิธีนี้เป็นวิธีที่จะตัดฟีเจอร์ตัวที่ไม่จำเป็นออกไปเพื่อลดความซับซ้อนของแบบจำลองลงไป โดยในงานวิจัยนี้ทำการเลือกฟีเจอร์ที่จำเป็น (Feature Selection) โดยการลดขนาดคลังคำศัพท์จาก 5,000 คำ เป็น 330 คำ (ภาคผนวก ก การทำ Grid Search เพื่อหา Feature Selection ที่เหมาะสม)

4.3 ผลการทดสอบประสิทธิภาพ คลังคำศัพท์ 330 คำ

4.3.1 ผลการทดสอบโดยใช้การแปลงเชิงปริมาณ Bag of Words และแบบจำลองการถดถอยลอจิสติก คลังคำศัพท์ 330 คำ

การทดสอบการพยากรณ์ทิศทางราคาโดยใช้การแปลงเชิงปริมาณ Bag of Words และแบบจำลองการถดถอยลอจิสติก คลังคำศัพท์ 330 คำ ซึ่งแบบจำลองสามารถทำนายชุดทดสอบออกมาว่าเป็นตลาดกระทิงหรือตลาดขาขึ้น (Bullish) 212 วัน และตลาดหมีหรือตลาดขาลง (Bearish) 201 วัน ซึ่งสามารถสรุปการทำนายตามตารางที่ 4.26 และมีประสิทธิภาพการทำนายดังตารางที่ 4.27

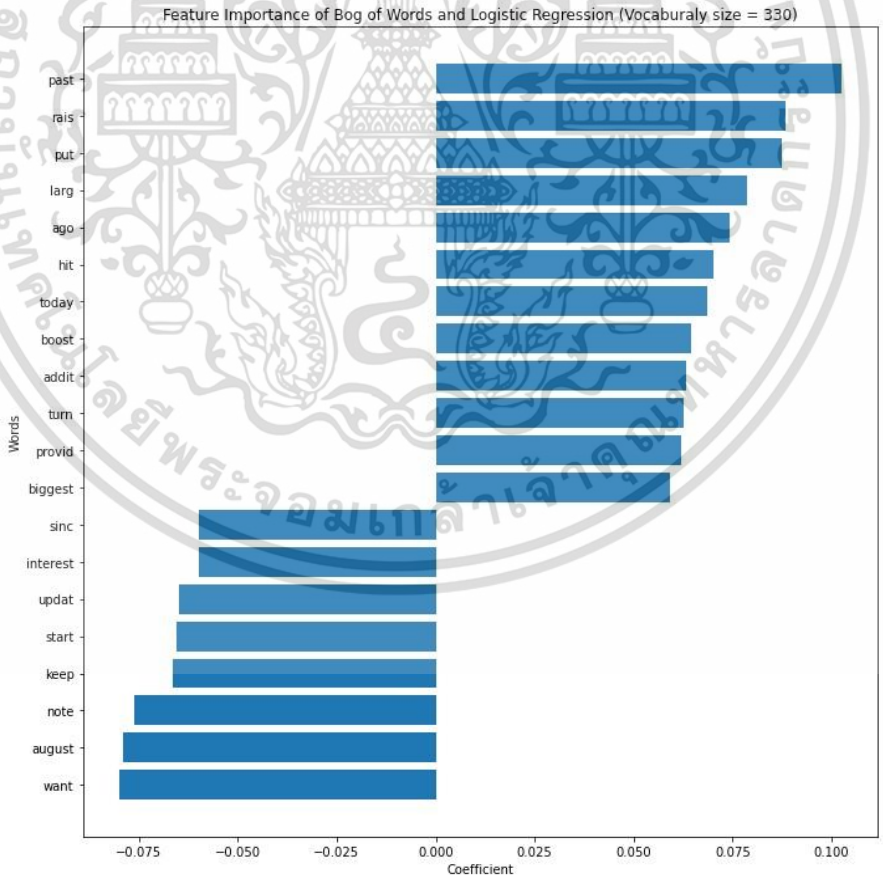
ตารางที่ 4.26 เมทริกซ์ความสับสน (Confusion Matrix) ของการแปลงเชิงปริมาณ Bag of Words และแบบจำลองการถดถอยลอจิสติก คลังคำศัพท์ 330 คำ

		ทิศทางที่แบบจำลองทำนาย		รวม
		0 (Bearish)	1 (Bullish)	
ทิศทางจากชุดทดสอบ	0 (Bearish)	108	108	216
	1 (Bullish)	93	104	197
รวม		201	212	413

ตารางที่ 4.27 ประสิทธิภาพการทำนายจากการแปลงเชิงปริมาณ Bag of Words และแบบจำลองการถดถอยลอจิสติก คลังคำศัพท์ 330 คำ

ทิศทางราคา	ประสิทธิภาพการทำนายของแบบจำลอง			
	Precision	Recall	F1-Score	Accuracy
0 (Bearish)	0.5373	0.5000	0.5180	0.5133
1 (Bullish)	0.4906	0.5279	0.5086	
Average	0.5139	0.5140	0.5133	

จากตารางที่ 4.27 แสดงประสิทธิภาพการทำนายจากการแปลงเชิงปริมาณ Bag of Words และแบบจำลองการถดถอยลอจิสติก คลังคำศัพท์ 330 คำ จะเห็นได้ว่ามีค่าเฉลี่ย F1-Score เป็น 51.33% โดยตลาดขาลงมีค่า F1-Score เท่ากับ 51.80% และตลาดขาขึ้นมีค่า F1-Score เท่ากับ 50.86% และมีความแม่นยำอยู่ที่ 51.33% หมายความว่า แบบจำลองสามารถทำนายตลาดขาลงได้ดีกว่าเล็กน้อย



รูปที่ 4.6 Feature Importance 20 อันดับแรกของการแปลงเชิงปริมาณ Bag of Words และแบบจำลองการถดถอยลอจิสติก คลังคำศัพท์ 330 คำ

จากภาพที่ 4.6 คือ Feature Importance 20 อันดับแรกของการแปลงเชิงปริมาณ Bag of Words และแบบจำลองการถดถอยลอจิสติก คลังคำศัพท์ 330 คำ วัดจากค่า Coefficient โดยคำศัพท์ที่มีค่า Coefficient สูงที่สุดทางด้านบวก คือ “past” มีค่า Coefficient เท่ากับ 0.1024 หมายความว่ามีอิทธิพลที่เพิ่มขึ้นต่อการทำนายผลลัพธ์ในเชิงบวก ดังนั้นเมื่อ “past” มีค่าเพิ่มขึ้นหนึ่งหน่วยโอกาสที่ผลลัพธ์จะเป็นคลาสที่สนใจจะเพิ่มขึ้น 0.1024 ในทางกลับกันคำศัพท์ที่มีค่า Coefficient สูงที่สุดทางด้านลบ คือ “want” มีค่า Coefficient เท่ากับ -0.0800 หมายความว่ามีอิทธิพลที่ลดลงต่อการทำนายผลลัพธ์ในเชิงบวก ดังนั้นเมื่อ “want” มีค่าเพิ่มขึ้นหนึ่งหน่วยโอกาสที่ผลลัพธ์จะเป็นคลาสที่สนใจจะลดลง -0.0800

4.3.2 ผลการทดสอบโดยใช้การแปลงเชิงปริมาณ Bag of Words และแบบจำลองซัพพอร์ตเวกเตอร์แมชชีน คลังคำศัพท์ 330 คำ

การทดสอบการพยากรณ์ทิศทางราคาโดยใช้การแปลงเชิงปริมาณ Bag of Words และแบบจำลองซัพพอร์ตเวกเตอร์แมชชีน คลังคำศัพท์ 330 คำ ซึ่งแบบจำลองสามารถทำนายชุดทดสอบออกมาว่าเป็นตลาดกระทิงหรือตลาดขาขึ้น (Bullish) 219 วัน และตลาดหมีหรือตลาดขาลง (Bearish) 194 วัน ซึ่งสามารถสรุปการทำนายตามตารางที่ 4.28 และมีประสิทธิภาพการทำนายดังตารางที่ 4.29

ตารางที่ 4.28 เมทริกซ์ความสับสน (Confusion Matrix) ของการแปลงเชิงปริมาณ Bag of Words และแบบจำลองซัพพอร์ตเวกเตอร์แมชชีน คลังคำศัพท์ 330 คำ

		ทิศทางที่แบบจำลองทำนาย		รวม
		0 (Bearish)	1 (Bullish)	
ทิศทางจากชุดทดสอบ	0 (Bearish)	100	116	216
	1 (Bullish)	94	103	197
รวม		194	219	413

ตารางที่ 4.29 ประสิทธิภาพการทำนายจากการแปลงเชิงปริมาณ Bag of Words และแบบจำลองซัพพอร์ตเวกเตอร์แมชชีน คลังคำศัพท์ 330 คำ

ทิศทางราคา	ประสิทธิภาพการทำนายของแบบจำลอง			
	Precision	Recall	F1-Score	Accuracy
0 (Bearish)	0.5155	0.4630	0.4878	0.4915
1 (Bullish)	0.4703	0.5228	0.4952	
Average	0.4929	0.4929	0.4915	

จากตารางที่ 4.29 แสดงประสิทธิภาพการทำนายจากการแปลงเชิงปริมาณ Bag of Words และแบบจำลองซัพพอร์ตเวกเตอร์แมชชีน คลังคำศัพท์ 330 คำ มีค่าเฉลี่ย F1-Score เป็น 49.15% โดยตลาดขาลงมีค่า F1-Score เท่ากับ 49.29% และตลาดขาขึ้นมีค่า F1-Score เท่ากับ 49.29% และมีความแม่นยำอยู่ที่ 49.15% หมายความว่า แบบจำลองสามารถทำนายตลาดขาขึ้นได้ดีกว่าเล็กน้อย

4.3.3 ผลการทดสอบโดยใช้การแปลงเชิงปริมาณ Bag of Words และแบบจำลองโครงข่ายประสาทเทียม คลังคำศัพท์ 330 คำ

การทดสอบการพยากรณ์ทิศทางราคาโดยใช้การแปลงเชิงปริมาณ Bag of Words และแบบจำลองโครงข่ายประสาทเทียม คลังคำศัพท์ 330 คำ ซึ่งแบบจำลองสามารถทำนายชุดทดสอบออกมาว่าเป็นตลาดกระทิงหรือตลาดขาขึ้น (Bullish) 205 วัน และตลาดหมีหรือตลาดขาลง (Bearish) 208 วัน ซึ่งสามารถสรุปการทำนายตามตารางที่ 4.30 และมีประสิทธิภาพการทำนายดังตารางที่ 4.31

ตารางที่ 4.30 เมทริกซ์ความสับสน (Confusion Matrix) ของการแปลงเชิงปริมาณ Bag of Words และแบบจำลองโครงข่ายประสาทเทียม คลังคำศัพท์ 330 คำ

		ทิศทางที่แบบจำลองทำนาย		รวม
		0 (Bearish)	1 (Bullish)	
ทิศทางจากชุดทดสอบ	0 (Bearish)	107	109	216
	1 (Bullish)	101	96	197
รวม		208	205	413

ตารางที่ 4.31 ประสิทธิภาพการทำนายจากการแปลงเชิงปริมาณ Bag of Words และแบบจำลองโครงข่ายประสาทเทียม คลังคำศัพท์ 330 คำ

ทิศทางราคา	ประสิทธิภาพการทำนายของแบบจำลอง			
	Precision	Recall	F1-Score	Accuracy
0 (Bearish)	0.5144	0.4954	0.5047	0.4915
1 (Bullish)	0.4683	0.4873	0.4776	
Average	0.4914	0.4913	0.4912	

จากตารางที่ 4.31 แสดงประสิทธิภาพการทำนายจากการแปลงเชิงปริมาณ Bag of Words และแบบจำลองโครงข่ายประสาทเทียม คลังคำศัพท์ 330 คำ จะเห็นได้ว่ามีค่าเฉลี่ย F1-Score เป็น 49.12% โดยตลาดขาลงมีค่า F1-Score เท่ากับ 50.47% และตลาดขาขึ้นมีค่า F1-Score เท่ากับ 47.76% และมีความแม่นยำอยู่ที่ 49.15% หมายความว่า แบบจำลองสามารถทำนายตลาดขาขึ้นได้ดีกว่า

4.3.4 ผลการทดสอบโดยใช้การแปลงเชิงปริมาณ TF-IDF และแบบจำลองการถดถอยลอจิสติก คลังคำศัพท์ 330 คำ

การทดสอบการพยากรณ์ทิศทางราคาโดยใช้การแปลงเชิงปริมาณ TF-IDF และแบบจำลองการถดถอยลอจิสติก คลังคำศัพท์ 330 คำ ซึ่งแบบจำลองสามารถทำนายชุดทดสอบออกมาว่าเป็นตลาดกระทิงหรือตลาดขาขึ้น (Bullish) 177 วัน และตลาดหมีหรือตลาดขาลง (Bearish) 236 วัน ซึ่งสามารถสรุปการทำนายตามตารางที่ 4.32 และมีประสิทธิภาพการทำนายดังตารางที่ 4.33

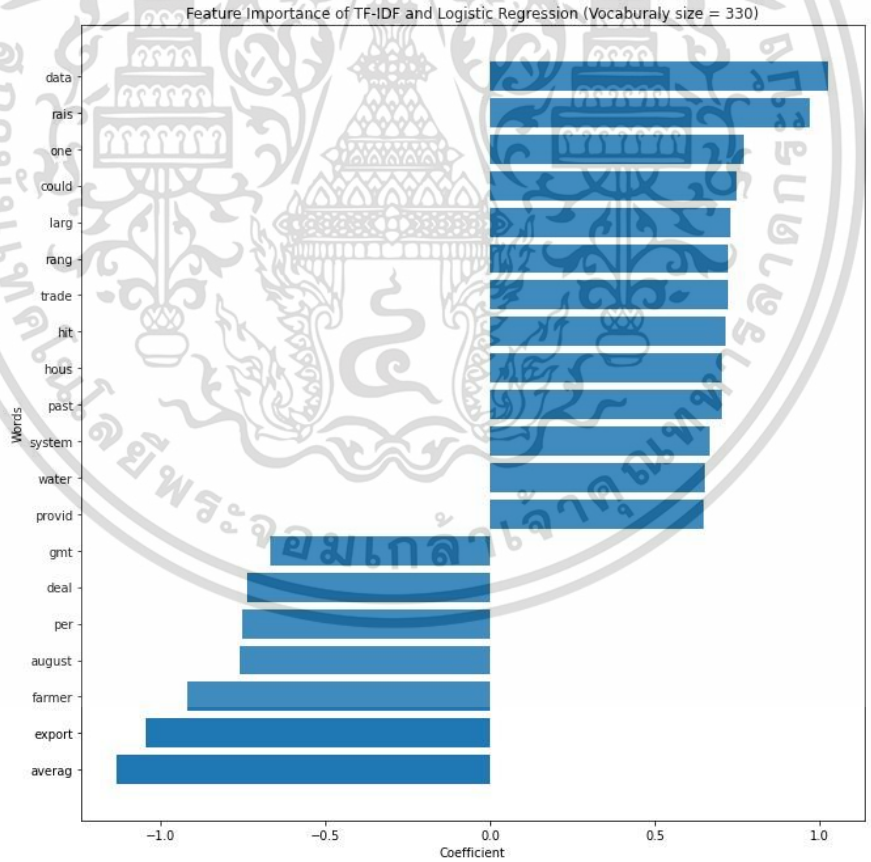
ตารางที่ 4.32 เมทริกซ์ความสับสน (Confusion Matrix) ของการแปลงเชิงปริมาณ TF-IDF และแบบจำลองการถดถอยลอจิสติก คลังคำศัพท์ 330 คำ

		ทิศทางที่แบบจำลองทำนาย		รวม
		0 (Bearish)	1 (Bullish)	
ทิศทางจากชุดทดสอบ	0 (Bearish)	122	94	216
	1 (Bullish)	114	83	197
รวม		236	177	413

ตารางที่ 4.33 ประสิทธิภาพการทำนายจากการแปลงเชิงปริมาณ TF-IDF และแบบจำลองการถดถอยลอจิสติก คลังคำศัพท์ 330 คำ

ทิศทางราคา	ประสิทธิภาพการทำนายของแบบจำลอง			
	Precision	Recall	F1-Score	Accuracy
0 (Bearish)	0.5169	0.5648	0.5398	0.4964
1 (Bullish)	0.4689	0.4213	0.4439	
Average	0.4929	0.4931	0.4918	

จากตารางที่ 4.33 แสดงประสิทธิภาพการทำนายจากการแปลงเชิงปริมาณ TF-IDF และแบบจำลองการถดถอยลอจิสติก คลังคำศัพท์ 330 คำ จะเห็นได้ว่ามีค่าเฉลี่ย F1-Score เป็น 49.18% โดยตลาดขาลงมีค่า F1-Score เท่ากับ 53.98% และตลาดขาขึ้นมีค่า F1-Score เท่ากับ 44.39% และมีความแม่นยำอยู่ที่ 49.64% หมายความว่า แบบจำลองสามารถทำนายตลาดขาลงได้ดีกว่า



รูปที่ 4.7 Feature Importance 20 อันดับแรกของการแปลงเชิงปริมาณ TF-IDF และแบบจำลองการถดถอยลอจิสติก คลังคำศัพท์ 330 คำ

จากภาพที่ 4.7 คือ Feature Importance 20 อันดับแรกของการแปลงคุณลักษณะ TF-IDF และแบบจำลองการถดถอยลอจิสติก คลังคำศัพท์ 330 คำ วัดจากค่า Coefficient โดยคำศัพท์ที่มีค่า Coefficient สูงที่สุดทางด้านบวก คือ “data” มีค่า Coefficient เท่ากับ 1.0276 หมายความว่ามีอิทธิพลที่เพิ่มขึ้นต่อการทำนายผลลัพธ์ในเชิงบวก ดังนั้นเมื่อ “data” มีค่าเพิ่มขึ้นหนึ่งหน่วยโอกาสที่ผลลัพธ์จะเป็นคลาสที่สนใจจะเพิ่มขึ้น 1.0276 ในทางกลับกันคำศัพท์ที่มีค่า Coefficient สูงที่สุดทางด้านลบ คือ “gmt” มีค่า Coefficient เท่ากับ -0.6673 หมายความว่ามีอิทธิพลที่ลดลงต่อการทำนายผลลัพธ์ในเชิงบวก ดังนั้นเมื่อ “gmt” มีค่าเพิ่มขึ้นหนึ่งหน่วยโอกาสที่ผลลัพธ์จะเป็นคลาสที่สนใจจะลดลง -0.6673

4.3.5 ผลการทดสอบโดยใช้การแปลงเชิงปริมาณ TF-IDF และแบบจำลองซัพพอร์ตเวกเตอร์แมชชีน คลังคำศัพท์ 330 คำ

การทดสอบการพยากรณ์ทิศทางราคาโดยใช้การแปลงเชิงปริมาณ TF-IDF และแบบจำลองซัพพอร์ตเวกเตอร์แมชชีน คลังคำศัพท์ 330 คำ ซึ่งแบบจำลองสามารถทำนายชุดทดสอบออกมาว่าเป็นตลาดกระทิงหรือตลาดขาขึ้น (Bullish) 197 วัน และตลาดหมีหรือตลาดขาลง (Bearish) 216 วัน ซึ่งสามารถสรุปการทำนายตามตารางที่ 4.34 และมีประสิทธิภาพการทำนายดังตารางที่ 4.35

ตารางที่ 4.34 เมทริกซ์ความสับสน (Confusion Matrix) ของการแปลงเชิงปริมาณ TF-IDF และแบบจำลองซัพพอร์ตเวกเตอร์แมชชีน คลังคำศัพท์ 330 คำ

		ทิศทางที่แบบจำลองทำนาย		รวม
		0 (Bearish)	1 (Bullish)	
ทิศทางจากชุดทดสอบ	0 (Bearish)	112	104	216
	1 (Bullish)	104	93	197
รวม		216	197	413

ตารางที่ 4.35 ประสิทธิภาพการทำนายจากการแปลงเชิงปริมาณ TF-IDF และแบบจำลองซัพพอร์ตเวกเตอร์แมชชีน คลังคำศัพท์ 330 คำ

ทิศทางราคา	ประสิทธิภาพการทำนายของแบบจำลอง			
	Precision	Recall	F1-Score	Accuracy
0 (Bearish)	0.5185	0.5185	0.5185	0.4964
1 (Bullish)	0.4721	0.4721	0.4721	
Average	0.4953	0.4953	0.4953	

จากตารางที่ 4.35 แสดงประสิทธิภาพการทำนายจากการแปลงเชิงปริมาณ TF-IDF และแบบจำลองซัพพอร์ตเวกเตอร์แมชชีน คลังคำศัพท์ 330 คำ จะเห็นได้ว่ามีค่าเฉลี่ย F1-Score เป็น 49.53% โดยตลาดขาลงมีค่า F1-Score เท่ากับ 51.85% และตลาดขาขึ้นมีค่า F1-Score เท่ากับ 47.21% และมีความแม่นยำอยู่ที่ 49.64% หมายความว่า แบบจำลองสามารถทำนายตลาดขาลงได้ดีกว่า

4.3.6 ผลการทดสอบโดยใช้การแปลงเชิงปริมาณ TF-IDF และแบบจำลองโครงข่ายประสาทเทียม คลังคำศัพท์ 330 คำ

การทดสอบการพยากรณ์ทิศทางราคาโดยใช้การแปลงเชิงปริมาณ TF-IDF และแบบจำลองโครงข่ายประสาทเทียม คลังคำศัพท์ 330 คำซึ่งแบบจำลองสามารถทำนายชุดทดสอบออกมาว่าเป็นตลาดกระทิงหรือตลาดขาขึ้น (Bullish) 173 วัน และตลาดหมีหรือตลาดขาลง (Bearish) 240 วัน ซึ่งสามารถสรุปการทำนายตามตารางที่ 4.36 และมีประสิทธิภาพการทำนายดังตารางที่ 4.37

ตารางที่ 4.36 เมทริกซ์ความสับสน (Confusion Matrix) ของการแปลงเชิงปริมาณ TF-IDF และแบบจำลองโครงข่ายประสาทเทียม คลังคำศัพท์ 330 คำ

		ทิศทางที่แบบจำลองทำนาย		รวม
		0 (Bearish)	1 (Bullish)	
ทิศทางจากชุดทดสอบ	0 (Bearish)	125	91	216
	1 (Bullish)	115	82	197
รวม		240	173	413

ตารางที่ 4.37 ประสิทธิภาพการทำนายจากการแปลงเชิงปริมาณ TF-IDF และแบบจำลองโครงข่ายประสาทเทียม คลังคำศัพท์ 330 คำ

ทิศทางราคา	ประสิทธิภาพการทำนายของแบบจำลอง			
	Precision	Recall	F1-Score	Accuracy
0 (Bearish)	0.5208	0.5787	0.5482	0.5012
1 (Bullish)	0.4740	0.4162	0.4432	
Average	0.4974	0.4975	0.4957	

จากตารางที่ 4.37 แสดงประสิทธิภาพการทำนายจากกการแปลงเชิงปริมาณ TF-IDF และแบบจำลองโครงข่ายประสาทเทียม คลังคำศัพท์ 330 คำ จะเห็นได้ว่ามีค่าเฉลี่ย F1-Score เป็น 49.57% โดยตลาดขาลงมีค่า F1-Score เท่ากับ 54.82% และตลาดขาขึ้นมีค่า F1-Score เท่ากับ 44.32% และมีความแม่นยำอยู่ที่ 49.64% หมายความว่า แบบจำลองสามารถทำนายตลาดขาลงได้ดีกว่า

4.3.7 ผลการทดสอบโดยใช้การแปลงคุณลักษณะ Word2Vec และแบบจำลองการถดถอยลอจิสติก เวกเตอร์คุณลักษณะจำนวน 330 ตัว

การทดสอบการพยากรณ์ทิศทางราคาโดยใช้การแปลงคุณลักษณะ Word2Vec และแบบจำลองการถดถอยลอจิสติก เวกเตอร์คุณลักษณะจำนวน 330 ตัว ซึ่งแบบจำลองสามารถทำนายชุดทดสอบออกมาว่าเป็นตลาดกระทิงหรือตลาดขาขึ้น (Bullish) 180 วัน และตลาดหมีหรือตลาดขาลง (Bearish) 233 วัน ซึ่งสามารถสรุปการทำนายตามตารางที่ 4.38 และมีประสิทธิภาพการทำนายดังตารางที่ 4.39

ตารางที่ 4.38 เมทริกซ์ความสับสน (Confusion Matrix) ของการแปลงคุณลักษณะ Word2Vec และแบบจำลองการถดถอยลอจิสติก เวกเตอร์คุณลักษณะจำนวน 330 ตัว

		ทิศทางที่แบบจำลองทำนาย		รวม
		0 (Bearish)	1 (Bullish)	
ทิศทางจากชุดทดสอบ	0 (Bearish)	125	91	216
	1 (Bullish)	108	89	197
รวม		233	180	413

ตารางที่ 4.39 ประสิทธิภาพการทำนายจากการแปลงคุณลักษณะ Word2Vec และแบบจำลองการถดถอยลอจิสติก เวกเตอร์คุณลักษณะจำนวน 330 ตัว

ทิศทางราคา	ประสิทธิภาพการทำนายของแบบจำลอง			
	Precision	Recall	F1-Score	Accuracy
0 (Bearish)	0.5365	0.5787	0.5568	0.5182
1 (Bullish)	0.4944	0.4518	0.4721	
Average	0.5155	0.5152	0.5145	

จากตารางที่ 4.39 แสดงประสิทธิภาพการทำนายจากการแปลงคุณลักษณะ Word2Vec และแบบจำลองการถดถอยลอจิสติก เวกเตอร์คุณลักษณะจำนวน 330 ตัว จะเห็นได้ว่ามีค่าเฉลี่ย F1-Score เป็น 51.45% โดยตลาดขาลงมีค่า F1-Score เท่ากับ 55.68% และตลาดขาขึ้นมีค่า F1-Score เท่ากับ 47.21% และมีความแม่นยำอยู่ที่ 51.82% หมายความว่า แบบจำลองสามารถทำนายตลาดขาลงได้ดีกว่า

4.3.8 ผลการทดสอบโดยใช้การแปลงคุณลักษณะ Word2Vec และแบบจำลองซัพพอร์ตเวกเตอร์แมชชีน เวกเตอร์คุณลักษณะจำนวน 330 ตัว

การทดสอบการพยากรณ์ทิศทางราคาโดยใช้การแปลงคุณลักษณะ Word2Vec และแบบจำลองซัพพอร์ตเวกเตอร์แมชชีน เวกเตอร์คุณลักษณะจำนวน 330 ตัว ซึ่งแบบจำลองสามารถทำนายชุดทดสอบออกมาว่าเป็นตลาดกระทิงหรือตลาดขาขึ้น (Bullish) 2 วัน และตลาดหมีหรือตลาดขาลง (Bearish) 411 วัน ซึ่งสามารถสรุปการทำนายตามตารางที่ 4.40 และมีประสิทธิภาพการทำนายดังตารางที่ 4.41

ตารางที่ 4.40 เมทริกซ์ความสับสน (Confusion Matrix) ของการแปลงคุณลักษณะ Word2Vec และแบบจำลองซัพพอร์ตเวกเตอร์แมชชีน เวกเตอร์คุณลักษณะจำนวน 330 ตัว

		ทิศทางที่แบบจำลองทำนาย		รวม
		0 (Bearish)	1 (Bullish)	
ทิศทางจากชุดทดสอบ	0 (Bearish)	214	2	216
	1 (Bullish)	197	0	197
รวม		411	2	413

ตารางที่ 4.41 ประสิทธิภาพการทำนายจากการแปลงคุณลักษณะ Word2Vec และแบบจำลองซัพพอร์ตเวกเตอร์แมชชีน เวกเตอร์คุณลักษณะจำนวน 330 ตัว

ทิศทางราคา	ประสิทธิภาพการทำนายของแบบจำลอง			
	Precision	Recall	F1-Score	Accuracy
0 (Bearish)	0.5207	0.9907	0.6826	0.5182
1 (Bullish)	0.0000	0.0000	0.0000	
Average	0.2603	0.4954	0.3413	

จากตารางที่ 4.41 แสดงประสิทธิภาพการทำนายจากการแปลงคุณลักษณะ Word2Vec และแบบจำลองซัพพอร์ตเวกเตอร์แมชชีน เวกเตอร์คุณลักษณะจำนวน 330 ตัว มีค่าเฉลี่ย F1-Score เป็น 34.13% โดยตลาดขาลงมีค่า F1-Score เท่ากับ 68.26% และตลาดขาขึ้นมีค่า F1-Score เท่ากับ 0.00% และมีความแม่นยำอยู่ที่ 51.82% หมายความว่า แบบจำลองสามารถทำนายตลาดขาลงได้ดีกว่า

4.3.9 ผลการทดสอบโดยใช้การแปลงคุณลักษณะ Word2Vec และแบบจำลองโครงข่ายประสาทเทียม เวกเตอร์คุณลักษณะจำนวน 330 ตัว

การทดสอบการพยากรณ์ทิศทางราคาโดยใช้การแปลงคุณลักษณะ Word2Vec และแบบจำลองโครงข่ายประสาทเทียม เวกเตอร์คุณลักษณะจำนวน 330 ตัว ซึ่งแบบจำลองสามารถทำนายชุดทดสอบออกมาว่าเป็นตลาดกระทิงหรือตลาดขาขึ้น (Bullish) 176 วัน และตลาดหมีหรือตลาดขาลง (Bearish) 237 วัน ซึ่งสามารถสรุปการทำนายตามตารางที่ 4.42 และมีประสิทธิภาพการทำนายดังตารางที่ 4.43

ตารางที่ 4.42 เมทริกซ์ความสับสน (Confusion Matrix) ของการแปลงคุณลักษณะ Word2Vec และแบบจำลองโครงข่ายประสาทเทียม เวกเตอร์คุณลักษณะจำนวน 330 ตัว

		ทิศทางที่แบบจำลองทำนาย		รวม
		0 (Bearish)	1 (Bullish)	
ทิศทางจากชุดทดสอบ	0 (Bearish)	126	90	216
	1 (Bullish)	111	86	197
รวม		237	176	413

ตารางที่ 4.43 ประสิทธิภาพการทำนายจากการแปลงคุณลักษณะ Word2Vec และแบบจำลองโครงข่ายประสาทเทียม เวกเตอร์คุณลักษณะจำนวน 330 ตัว

ทิศทางราคา	ประสิทธิภาพการทำนายของแบบจำลอง			
	Precision	Recall	F1-Score	Accuracy
0 (Bearish)	0.5316	0.5833	0.5563	0.5133
1 (Bullish)	0.4886	0.4365	0.4611	
Average	0.5101	0.5099	0.5087	

จากตารางที่ 4.43 แสดงประสิทธิภาพการทำนายจากการแปลงคุณลักษณะ Word2Vec และแบบจำลองโครงข่ายประสาทเทียม เวกเตอร์คุณลักษณะจำนวน 330 ตัว จะเห็นได้ว่ามีค่าเฉลี่ย F1-Score เป็น 50.87% โดยตลาดขาลงมีค่า F1-Score เท่ากับ 55.63% และตลาดขาขึ้นมีค่า F1-Score เท่ากับ 46.11% และมีความแม่นยำอยู่ที่ 51.33% หมายความว่า แบบจำลองสามารถทำนายตลาดขาลงได้ดีกว่า

4.4 การเปรียบเทียบประสิทธิภาพ คลังคำศัพท์ 330 คำ

4.4.1 การเปรียบเทียบประสิทธิภาพของแบบจำลอง คลังคำศัพท์ 330 คำ

ค่าความถ่วงดุล (F1-Score) และค่าความแม่นยำ (Accuracy) ของแบบจำลองต่าง ๆ ในการการพยากรณ์ทิศทางราคาสัญญาฟิวเจอร์สข้าวโพดจากข้อความข่าวโดยใช้การประมวลภาษาธรรมชาติ จากชุดข้อมูลทดสอบ ได้ผลดังตารางที่ 4.44, ตารางที่ 4.45 และตารางทรา 4.46

ตารางที่ 4.44 เปรียบเทียบประสิทธิภาพของแบบจำลองทั้ง 9 แบบจำลอง จากร้อยละของค่าความแม่นยำ Accuracy (ลำดับของค่าความแม่นยำ) คลังคำศัพท์ 330 คำ

การแปลงคุณลักษณะ	แบบจำลอง	ร้อยละของค่าความแม่นยำ Accuracy (ลำดับที่)	
		ข้อมูลชุดเรียนรู้	ข้อมูลชุดทดสอบ
Bag of Words	Logistic Regression	68.28	51.33 (2)
	SVM	76.21	49.15 (5)
	ANN	76.76	49.15 (5)

ตารางที่ 4.44 เปรียบเทียบประสิทธิภาพของแบบจำลองทั้ง 9 แบบจำลอง จากร้อยละของค่าความแม่นยำ Accuracy (ลำดับของค่าความแม่นยำ) คลังคำศัพท์ 330 คำ (ต่อ)

การแปลงคุณลักษณะ	แบบจำลอง	ร้อยละของค่าความแม่นยำ Accuracy (ลำดับที่)	
		ข้อมูลชุดเรียนรู้	ข้อมูลชุดทดสอบ
TF-IDF	Logistic Regression	63.08	49.64 (4)
	SVM	83.35	49.64 (4)
	ANN	79.06	50.12 (3)
Word2Vec	Logistic Regression	58.29	51.82 (1)
	SVM	52.66	51.82 (1)
	ANN	54.24	51.33 (2)

จากการวัดประสิทธิภาพของแบบจำลองหากพิจารณาจากร้อยละของค่าความแม่นยำ Accuracy คลังคำศัพท์ 330 คำ พบว่าการแปลงคุณลักษณะ Word2Vec และแบบจำลองการถดถอยลอจิสติก และการแปลงคุณลักษณะ Word2Vec และแบบจำลองซัพพอร์ตเวกเตอร์แมชชีน มีความแม่นยำในการพยากรณ์ดีที่สุดเท่ากัน คือ 51.82% ในทางกลับกันการแปลงเชิงปริมาณ Bag of Words และแบบจำลองซัพพอร์ตเวกเตอร์แมชชีน และการแปลงเชิงปริมาณ Bag of Words และแบบจำลองโครงข่ายประสาทเทียม มีความแม่นยำในการพยากรณ์ต่ำที่สุดเท่ากันคือ 49.15%

ตารางที่ 4.45 เปรียบเทียบประสิทธิภาพของแบบจำลองทั้ง 9 แบบจำลอง จากร้อยละของค่าความถ่วงดุล F1-Score (ลำดับของ F1-Score) คลังคำศัพท์ 330 คำ

การแปลงคุณลักษณะ	แบบจำลอง	ร้อยละของค่าความถ่วงดุล F1-Score (ลำดับที่)	
		ข้อมูลชุดเรียนรู้	ข้อมูลชุดทดสอบ
Bag of Words	Logistic Regression	68.16	51.33 (2)
	SVM	75.90	49.15 (7)
	ANN	76.64	49.12 (8)
TF-IDF	Logistic Regression	62.01	49.18 (6)
	SVM	83.14	49.53 (5)
	ANN	78.54	49.57 (4)

ตารางที่ 4.45 เปรียบเทียบประสิทธิภาพของแบบจำลองทั้ง 9 แบบจำลอง จากร้อยละของค่าความถ่วงดุล F1-Score (ลำดับของ F1-Score) คลังคำศัพท์ 330 คำ (ต่อ)

การแปลงคุณลักษณะ	แบบจำลอง	ร้อยละของค่าความถ่วงดุล F1-Score (ลำดับที่)	
		ข้อมูลชุดเรียนรู้	ข้อมูลชุดทดสอบ
Word2Vec	Logistic Regression	57.12	51.45 (1)
	SVM	35.07	34.13 (9)
	ANN	52.31	50.87 (3)

จากการวัดประสิทธิภาพของแบบจำลองหากพิจารณาจากร้อยละของค่าความถ่วงดุล F1-Score คลังคำศัพท์ 330 คำ พบว่าการแปลงคุณลักษณะ Word2Vec และแบบจำลองการถดถอยลอจิสติก มีค่า F1-Score ดีที่สุดคือ 51.45% รองลงมาคือการแปลงเชิงปริมาณ Bag of Words และแบบจำลองการถดถอยลอจิสติก และการแปลงคุณลักษณะ Word2Vec และแบบจำลองโครงข่ายประสาทเทียม มีค่า F1-Score เท่ากับ 51.33% และ 50.87% ตามลำดับ ในทางกลับกันการแปลงคุณลักษณะ Word2Vec และแบบจำลองซัพพอร์ตเวกเตอร์แมชชีน มีค่า F1-Score ต่ำที่สุดเท่ากับ 34.13%

ตารางที่ 4.46 เปรียบเทียบประสิทธิภาพของแบบจำลองทั้ง 9 แบบจำลอง จากร้อยละของค่าความถ่วงดุล F1-Score (ลำดับของ F1–Score) และร้อยละของค่าความแม่นยำ Accuracy (ลำดับของค่าความแม่นยำ) ร่วมกัน คลังคำศัพท์ 330 คำ

การแปลงคุณลักษณะ	แบบจำลอง	ข้อมูลชุดทดสอบ	
		ค่าความแม่นยำ Accuracy	ค่าความถ่วงดุล F1-Score
Bag of Words	Logistic Regression	51.33	51.33 (2)
	SVM	49.15	49.15 (7)
	ANN	49.15	49.12 (8)
TF-IDF	Logistic Regression	49.64	49.18 (6)
	SVM	49.64	49.53 (5)
	ANN	50.12	49.57 (4)

ตารางที่ 4.46 เปรียบเทียบประสิทธิภาพของแบบจำลองทั้ง 9 แบบจำลอง จากร้อยละของค่าความถ่วงดุล F1-Score (ลำดับของ F1–Score) และร้อยละของค่าความแม่นยำ Accuracy (ลำดับของค่าความแม่นยำ) ร่วมกัน คลังคำศัพท์ 330 คำ (ต่อ)

การแปลงคุณลักษณะ	แบบจำลอง	ข้อมูลชุดทดสอบ	
		ค่าความแม่นยำ Accuracy	ค่าความถ่วงดุล F1-Score
Word2Vec	Logistic Regression	51.82	51.45 (1)
	SVM	51.82	34.13 (9)
	ANN	51.33	50.87 (3)

หากพิจารณาโดยใช้ร้อยละของค่า F1-Score เป็นอันดับแรกในการวัดผล และใช้ค่า Accuracy ร่วมกัน คลังคำศัพท์ 330 คำ พบว่าการแปลงคุณลักษณะ Word2Vec และแบบจำลองการถดถอยลอจิสติก มีค่า F1-Score ดีที่สุดคือ 51.45% และมีค่าความแม่นยำในการพยากรณ์ดีที่สุดคือ 51.82% รองลงมาคือการแปลงเชิงปริมาณ Bag of Words และแบบจำลองการถดถอยลอจิสติก และการแปลงคุณลักษณะ Word2Vec และแบบจำลองโครงข่ายประสาท มีค่า F1-Score เท่ากับ 51.33% และ 50..87% ตามลำดับ และมีค่าความแม่นยำในการพยากรณ์เท่ากับ 51.33% และ 51.33% ตามลำดับ

ตารางที่ 4.47 เปรียบเทียบปัญหา Overfitting จากค่า Accuracy คลังคำศัพท์ 330 คำ

การแปลงคุณลักษณะ	แบบจำลอง	ร้อยละของค่าความแม่นยำ Accuracy (ลำดับที่)	
		ข้อมูลชุดเรียนรู้	ข้อมูลชุดทดสอบ
Bag of Words	Logistic Regression	68.28	51.33 (2)
	SVM	76.21	49.15 (5)
	ANN	76.76	49.15 (5)
TF-IDF	Logistic Regression	63.08	49.64 (4)
	SVM	83.35	49.64 (4)
	ANN	79.06	50.12 (3)
Word2Vec	Logistic Regression	58.29	51.82 (1)
	SVM	52.66	51.82 (1)
	ANN	54.24	51.33 (2)

ตารางที่ 4.48 เปรียบเทียบปัญหา Overfitting จากค่า F1-Score คลังคำศัพท์ 330 คำ

การแปลงคุณลักษณะ	แบบจำลอง	ร้อยละของค่าความถ่วงดุล F1-Score (ลำดับที่)	
		ข้อมูลชุดเรียนรู้	ข้อมูลชุดทดสอบ
Bag of Words	Logistic Regression	68.16	51.33 (2)
	SVM	75.90	49.15 (7)
	ANN	76.64	49.12 (8)
TF-IDF	Logistic Regression	62.01	49.18 (6)
	SVM	83.14	49.53 (5)
	ANN	78.54	49.57 (4)
Word2Vec	Logistic Regression	57.12	51.45 (1)
	SVM	35.07	34.13 (9)
	ANN	52.31	50.87 (3)

จากตารางที่ 4.48 และ 4.49 จะเห็นว่าแบบจำลองที่เกิดปัญหา Overfitting ขึ้นที่คลังคำศัพท์ 5,000 คำ แต่หลังจากการเลือกฟีเจอร์ที่จำเป็น (Feature Selection) โดยลดขนาดคลังคำศัพท์เป็น 330 คำ พบว่าข้อมูลชุดการเรียนรู้มีค่าลดลง และมีค่าใกล้เคียงกับข้อมูลชุดทดสอบ ดังนี้

1) การแปลงเชิงปริมาณ Bag of Words และแบบจำลองการถดถอยลอจิสติก ค่าความแม่นยำ ในข้อมูลชุดเรียนรู้ที่คลังคำศัพท์ 5,000 คำ เท่ากับ 100% ในข้อมูลชุดทดสอบเท่ากับ 46.73% เทียบกับ ข้อมูลชุดเรียนรู้ที่คลังคำศัพท์ 330 คำ เท่ากับ 68.28% ในข้อมูลชุดทดสอบเท่ากับ 51.33% ดังนั้น ค่าความแม่นยำ ในข้อมูลชุดเรียนรู้ที่คลังคำศัพท์ 330 ลดลง และในข้อมูลชุดทดสอบเพิ่มขึ้น และในด้านของค่า F1-Score ในข้อมูลชุดเรียนรู้ที่คลังคำศัพท์ 5,000 คำ เท่ากับ 100% ในข้อมูลชุดทดสอบเท่ากับ 46.73% เทียบกับ ข้อมูลชุดเรียนรู้ที่คลังคำศัพท์ 330 คำ เท่ากับ 68.16% ในข้อมูลชุดทดสอบเท่ากับ 51.33% ดังนั้น ค่า F1-Score ในข้อมูลชุดเรียนรู้ที่คลังคำศัพท์ 330 ลดลง และในข้อมูลชุดทดสอบเพิ่มขึ้น

2) การแปลงเชิงปริมาณ Bag of Words และแบบจำลองซัพพอร์ตเวกเตอร์แมชชีน ค่าความแม่นยำ ในข้อมูลชุดเรียนรู้ที่คลังคำศัพท์ 5,000 คำ เท่ากับ 83.54% ในข้อมูลชุดทดสอบเท่ากับ 50.85% เทียบกับ ข้อมูลชุดเรียนรู้ที่คลังคำศัพท์ 330 คำ เท่ากับ 76.21% ในข้อมูลชุดทดสอบเท่ากับ 49.15% ดังนั้น ค่าความแม่นยำ ในข้อมูลชุดเรียนรู้ที่คลังคำศัพท์ 330 ลดลง แต่ในข้อมูลชุดทดสอบลดลงก็ลดลงเช่นกัน และในด้านของค่า F1-Score ในข้อมูลชุดเรียนรู้ที่คลังคำศัพท์ 5,000 คำ เท่ากับ 83.26% ในข้อมูลชุดทดสอบเท่ากับ 50.83% เทียบกับ ข้อมูลชุดเรียนรู้ที่คลังคำศัพท์ 330 คำ เท่ากับ

75.90% ในข้อมูลชุดทดสอบเท่ากับ 49.15% ดังนั้น ค่า F1-Score ในข้อมูลชุดเรียนรู้ที่คลังคำศัพท์ 330 ลดลง แต่ในข้อมูลชุดทดสอบก็ลดลงเช่นกัน

3) การแปลงเชิงปริมาณ TF-IDF และแบบจำลองการถดถอยลอจิสติก ค่าความแม่นยำในข้อมูลชุดเรียนรู้ที่คลังคำศัพท์ 5,000 คำ เท่ากับ 84.26% ในข้อมูลชุดทดสอบเท่ากับ 48.91% เทียบกับ ข้อมูลชุดเรียนรู้ที่คลังคำศัพท์ 330 คำ เท่ากับ 63.08% ในข้อมูลชุดทดสอบเท่ากับ 49.64% ดังนั้น ค่าความแม่นยำ ในข้อมูลชุดเรียนรู้ที่คลังคำศัพท์ 330 ลดลง และในข้อมูลชุดทดสอบเพิ่มขึ้น และในด้านของค่า F1-Score ในข้อมูลชุดเรียนรู้ที่คลังคำศัพท์ 5,000 คำ เท่ากับ 83.98% ในข้อมูลชุดทดสอบเท่ากับ 48.74% เทียบกับ ข้อมูลชุดเรียนรู้ที่คลังคำศัพท์ 330 คำ เท่ากับ 62.01% ในข้อมูลชุดทดสอบเท่ากับ 49.18% ดังนั้น ค่า F1-Score ในข้อมูลชุดเรียนรู้ที่คลังคำศัพท์ 330 ลดลง และในข้อมูลชุดทดสอบเพิ่มขึ้น

4) การแปลงเชิงปริมาณ TF-IDF และแบบจำลองซัพพอร์ตเวกเตอร์แมชชีน ค่าความแม่นยำในข้อมูลชุดเรียนรู้ที่คลังคำศัพท์ 5,000 คำ เท่ากับ 96.43% ในข้อมูลชุดทดสอบเท่ากับ 48.91% เทียบกับ ข้อมูลชุดเรียนรู้ที่คลังคำศัพท์ 330 คำ เท่ากับ 83.35% ในข้อมูลชุดทดสอบเท่ากับ 49.64% ดังนั้น ค่าความแม่นยำ ในข้อมูลชุดเรียนรู้ที่คลังคำศัพท์ 330 ลดลง และในข้อมูลชุดทดสอบเพิ่มขึ้น และในด้านของค่า F1-Score ในข้อมูลชุดเรียนรู้ที่คลังคำศัพท์ 5,000 คำ เท่ากับ 96.41% ในข้อมูลชุดทดสอบเท่ากับ 48.15% เทียบกับ ข้อมูลชุดเรียนรู้ที่คลังคำศัพท์ 330 คำ เท่ากับ 83.14% ในข้อมูลชุดทดสอบเท่ากับ 49.53% ดังนั้น ค่า F1-Score ในข้อมูลชุดเรียนรู้ที่คลังคำศัพท์ 330 ลดลง และในข้อมูลชุดทดสอบเพิ่มขึ้น

5) การแปลงเชิงปริมาณ TF-IDF และแบบจำลองโครงข่ายประสาท ค่าความแม่นยำในข้อมูลชุดเรียนรู้ที่คลังคำศัพท์ 5,000 คำ เท่ากับ 99.94% ในข้อมูลชุดทดสอบเท่ากับ 49.67% เทียบกับ ข้อมูลชุดเรียนรู้ที่คลังคำศัพท์ 330 คำ เท่ากับ 79.06% ในข้อมูลชุดทดสอบเท่ากับ 50.12% ดังนั้น ค่าความแม่นยำ ในข้อมูลชุดเรียนรู้ที่คลังคำศัพท์ 330 ลดลง และในข้อมูลชุดทดสอบเพิ่มขึ้น และในด้านของค่า F1-Score ในข้อมูลชุดเรียนรู้ที่คลังคำศัพท์ 5,000 คำ เท่ากับ 99.94% ในข้อมูลชุดทดสอบเท่ากับ 49.45% เทียบกับ ข้อมูลชุดเรียนรู้ที่คลังคำศัพท์ 330 คำ เท่ากับ 78.54% ในข้อมูลชุดทดสอบเท่ากับ 49.57% ดังนั้น ค่า F1-Score ในข้อมูลชุดเรียนรู้ที่คลังคำศัพท์ 330 ลดลง และในข้อมูลชุดทดสอบเพิ่มขึ้น

สรุปว่า 4 จาก 5 แบบจำลองพบว่านอกจากช่วยลดการเกิดปัญหาพอดีเกินไป (Overfitting) แล้วยังส่งผลช่วยให้ค่าความแม่นยำ (Accuracy) และค่า F1-Score เพิ่มขึ้นด้วย ดังนั้นการเลือกฟีเจอร์ที่จำเป็น (Feature Selection) โดยการลดขนาดคลังคำศัพท์จาก 5,000 คำ เป็น 330 คำ จึงช่วยลดปัญหาพอดีเกินไป (Overfitting) ที่เกิดขึ้นได้

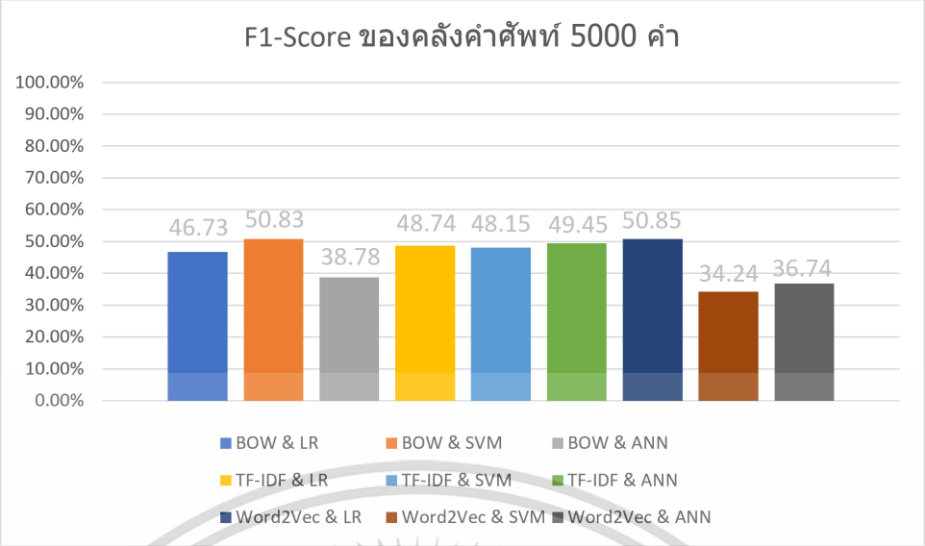
บทที่ 5
สรุปผลการวิจัยและข้อเสนอแนะ

งานวิจัยนี้ได้พัฒนาแบบจำลองในการพยากรณ์ทิศทางราคาสัญญาฟิวเจอร์สข้าวโพดจากข้อความข่าวโดยใช้การประมวลภาษาธรรมชาติ โดยใช้การสร้างตัวแปลงเชิงปริมาณให้อยู่ในรูปของคุณลักษณะที่ใช้ในการประมวลผลได้ ได้แก่ Bag of Words, TF-IDF และ Word2Vec จากนั้นทำการฝึกสอนโดยใช้ แบบจำลองการถดถอยลอจิสติก (Logistic Regression), ซัพพอร์ตเวกเตอร์แมชชีน (Support Vector Machine; SVM) และ โครงข่ายประสาทเทียม (Artificial Neural Network; ANN) สามารถสรุปผลได้ดังนี้

5.1 สรุปผลการวิจัย

ตาราง 5.1 สรุปผลการเปรียบเทียบประสิทธิภาพของแบบจำลองทั้ง 9 แบบจำลอง จากร้อยละของค่าความถ่วงดุล F1-Score (ลำดับของ F1–Score) คลังคำศัพท์ 5,000 คำ

การแปลงคุณลักษณะ	**แบบจำลอง**	**ค่าความถ่วงดุล F1-Score**	**ตัวบบที่เหมาะสมที่สุด**
Bag of Words	Logistic Regression	46.73 (6)	การแปลงคุณลักษณะ Word2Vec และแบบจำลองการถดถอยลอจิสติก
	SVM	50.83 (2)	
	ANN	38.78 (7)	
TF-IDF	Logistic Regression	48.74 (4)	
	SVM	48.15 (5)	
	ANN	49.45 (3)	
Word2Vec	Logistic Regression	50.85 (1)	
	SVM	34.24 (9)	
	ANN	36.74 (8)	

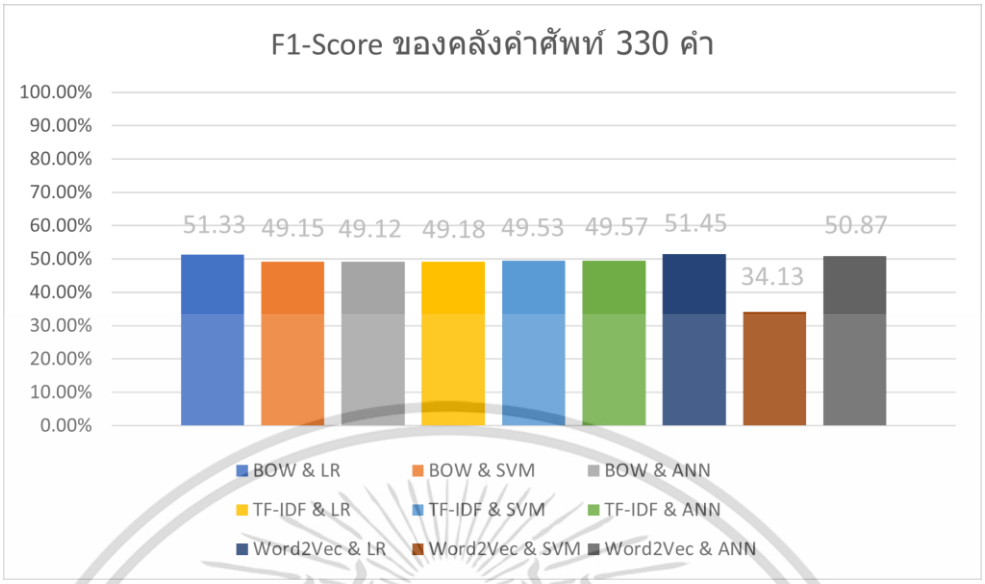


รูปที่ 5.1 กราฟเปรียบเทียบประสิทธิภาพของแบบจำลองจากค่า F1-Score คลังคำศัพท์ 5,000 คำ

จากตาราง 5.1 และรูปที่ 5.1 พบว่า แบบจำลองที่ดีที่สุดสำหรับการกำหนดคลังคำศัพท์ไว้ที่ 5,000 คำ โดยใช้ค่า F1-Score ในการวัดผลคือ การแปลงคุณลักษณะ Word2Vec และแบบจำลองการถดถอยลอจิสติก

ตาราง 5.2 สรุปผลการเปรียบเทียบประสิทธิภาพของแบบจำลองทั้ง 9 แบบจำลอง จากร้อยละของค่าความถ่วงดุล F1-Score (ลำดับของ F1–Score) คลังคำศัพท์ 330 คำ

การแปลงคุณลักษณะ	**แบบจำลอง**	**ค่าความถ่วงดุล F1-Score**	**ตัวบบที่เหมาะสมที่สุด**
Bag of Words	Logistic Regression	51.33 (2)	การแปลงคุณลักษณะ Word2Vec และแบบจำลองการถดถอยลอจิสติก
	SVM	49.15 (7)	
	ANN	49.12 (8)	
TF-IDF	Logistic Regression	49.18 (6)	
	SVM	49.53 (5)	
	ANN	49.57 (4)	
Word2Vec	Logistic Regression	51.45 (1)	
	SVM	34.13 (9)	
	ANN	50.87 (3)	



รูปที่ 5.2 กราฟเปรียบเทียบประสิทธิภาพของแบบจำลองจากค่า F1-Score คลังคำศัพท์ 330 คำ

จากตารางที่ 5.2 รูปที่ 5.2 พบว่า แบบจำลองที่ดีที่สุดสำหรับการกำหนดคลังคำศัพท์ไว้ที่ 330 คำ โดยใช้ค่า F1-Score ในการวัดผลคือ การแปลงคุณลักษณะ Word2Vec และแบบจำลองการถดถอยลอจิสติก

สำหรับงานวิจัยนี้ในการเลือกฟีเจอร์ที่จำเป็น (Feature Selection) โดยการกำหนดขนาดคลังคำศัพท์ (Vocabulary size) และกำหนดเวกเตอร์คุณลักษณะ (Feature vector) เท่ากับ 5,000 คำ และ 330 คำ ทั้ง 2 แบบ ให้ผลลัพธ์ที่สอดคล้องกัน เนื่องจากการกำหนดคลังคำศัพท์ทั้ง 2 แบบ ได้แบบจำลองที่ดีที่สุดเหมือนกันคือ การแปลงคุณลักษณะ Word2Vec และแบบจำลองการถดถอยลอจิสติก ดังนั้นการแปลงคุณลักษณะ Word2Vec และแบบจำลองการถดถอยลอจิสติก เป็นแบบจำลองที่มีประสิทธิภาพดีที่สุด และเป็นแบบจำลองที่เหมาะสมที่สุดสำหรับการพยากรณ์ทิศทางราคาสัญญาฟิวเจอร์สข้าวโพดจากข้อความข่าว

5.2 ข้อเสนอแนะ

5.2.1 ข้อเสนอแนะที่ได้จากงานวิจัย

1) งานวิจัยนี้ใช้การกำหนดขนาดคลังคำศัพท์ (Vocabulary size) และเวกเตอร์คุณลักษณะ (Feature vector) เท่ากับ 5,000 คำ และ 330 คำ ถึงแม้ว่าทั้ง 2 แบบให้ผลลัพธ์ที่เหมือนกัน แต่ทั้ง 2 แบบ ก็มีข้อดี ข้อเสียที่ต่างกันออกไป เช่น

คลังคำศัพท์ 5,000 คำ ข้อดี คือ มีคลังคำศัพท์ที่ใหญ่กว่าสามารถเรียนรู้รูปแบบได้หลากหลาย และสามารถรับมือกับชุดข้อมูลทดสอบที่ไม่เคยเจอได้ดีกว่า เป็นต้น ข้อเสีย คือ

เกิดปัญหาพอดีเกินไป (Overfitting) ขึ้น ซึ่งทำให้แบบจำลองยึดติดกับข้อมูลชุดเรียนรู้มากเกินไป พอไปเจอข้อมูลชุดทดสอบที่ไม่เคยเห็นมาก่อน ก็จะให้ผลทำนายที่แย่มาก เป็นต้น

คลังคำศัพท์ 330 คำ ข้อดี คือ คลังคำศัพท์มีแต่คำศัพท์ที่ปรากฏบ่อย และตัดคำศัพท์ที่ปรากฏไม่บ่อยออกไป จึงเป็นเหตุสำคัญที่ช่วยลดปัญหาพอดีเกินไป (Overfitting) เป็นต้น ข้อเสีย คือ สูญเสียข้อมูลบางส่วนที่สำคัญและความต่างที่มีอยู่ในข้อวาม แบบจำลองไม่ครอบคลุมถึงความซับซ้อนของภาษา และอาจรับมือกับชุดข้อมูลทดสอบที่ไม่เคยเจอได้ไม่ดี เป็นต้น

5.2.2 ข้อเสนอแนะในการทำวิจัยในอนาคต

1) การปรับค่าพารามิเตอร์ (Hyperparameter) ที่ใช้ในการสร้างตัวแบบโครงข่ายประสาทเทียม (Artificial Neural Network; ANN) ให้เหมาะสมกับข้อมูลเพื่อให้ตัวแบบมีประสิทธิภาพมากขึ้นในการพยากรณ์ราคาข้าวโพด เนื่องจากงานวิจัยนี้ใช้พารามิเตอร์เป็นค่าเริ่มต้น (Default) ทั้งหมด

2) งานวิจัยชิ้นนี้ใช้การแปลงคุณลักษณะเพียง 3 แบบ ซึ่งยังมีการแปลงคุณลักษณะที่น่าสนใจอีกมากมาย เช่น Doc2Vec, GloVe, FastText และ BERT เป็นต้น นอกจากนี้ยังมีแบบจำลองอื่นๆ ตัวอย่างเช่น Recurrent Neural Networks (RNN), Convolutional Neural Networks (CNN), Transformer เป็นต้น ซึ่งสามารถนำแบบจำลองข้างต้นมาใช้ในการพยากรณ์ทิศทางราคา และเปรียบเทียบความแม่นยำของแบบจำลองต่าง ๆ ได้

3) งานวิจัยชิ้นนี้เลือกใช้สัญญาฟิวเจอร์สข้าวโพดจากตลาด Chicago Board of Trade (CBOT) ของประเทศสหรัฐอเมริกา โดยมีการรวบรวมข้อมูลข่าวจากสำนักข่าวรอยเตอร์ (Reuter) ซึ่งสามารถนำงานวิจัยชิ้นนี้ไปต่อยอดเพื่อประยุกต์ใช้กับสัญญาฟิวเจอร์สอื่น ๆ เช่น สัญญาฟิวเจอร์สถั่วเหลือง, สัญญาฟิวเจอร์สกากถั่วเหลืองและสัญญาฟิวเจอร์สข้าวสาลี เป็นต้น

เอกสารอ้างอิง

กรมเจรจาการค้าระหว่างประเทศ. 2565. **สินค้าข้าวโพดเลี้ยงสัตว์.** [Online]. เข้าถึงได้จาก https://www.dtn.go.th/th/content/page/index/id/1048

กริช สมกันธา, วิไลพร กุลตังวัฒนา และวรวิทย์ กุลตังวัฒนา. 2556. **การพัฒนาระบบประเมินบุคลากรโดยใช้โครงข่ายประสาทเทียม.** วารสารเทคโนโลยีสารสนเทศ. 9(1) : 58-66.

กวี นำพาเจริญ. 2552. **เริ่มรู้จัก...ฟิวเจอร์ส.** [Online]. เข้าถึงได้จาก https://www01.bualuang.co.th/

กอบเกียรติ สระอุบล. 2563. **เรียนรู้ Data Science และ AI: Machine Learning ด้วย Python.** กรุงเทพฯ. มีเดีย เนทเวิร์ค.

กิติ์สุชาต พสุภา. 2564. **ระบบอัจฉริยะชั้นสูง: ทฤษฎี อัลกอริทึม และการประยุกต์ใช้.** กรุงเทพฯ. สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง.

จตุรพัชร์ พัฒนทรงศิวิไล. 2562. **AI ไม่ยาก (เล่ม 1+2) เข้าใจได้ด้วยเลข ม.ปลาย.** [Online]. เข้าถึงได้จาก https://www.mebmarket.com/ebook-108246-AI-ไม่ยาก-เล่ม-1-2-เข้าใจได้ด้วยเลขม--ปลาย

ชิตพงษ์ กิตตินราดร. 2563. **Neural Network Vanishing Gradients Problem.** [Online]. เข้าถึงได้จาก https://guopai.github.io/ml-blog16.html

ชิตพงษ์ กิตตินราดร. 2563. **Support Vector Machines.** [Online]. เข้าถึงได้จาก https://guopai.github.io/ml-blog08.html

โชคชัย และเกตุอร. 2561. **การปลูกข้าวโพด.** [Online]. เข้าถึงได้จาก http://eto.ku.ac.th/neweto/e-book/plant/herb_gar/corn2.pdf

ณัชภัคสรณ์ ปิยะบัณฑิตกุล. 2564. **การพยากรณ์ทิศทางราคาสัญญาฟิวเจอร์สถั่วเหลืองโดยใช้ปัจจัยข่าว ปัจจัยพื้นฐาน และปัจจัยเทคนิค.** วิทยาศาสตรบัณฑิต สาขาวิชาสถิติประยุกต์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง.

ณัฏฐา ผิวมา. 2558. การพัฒนาแบบจำลองพยากรณ์แนวโน้มการสมัครงานให้ตรงกับวุฒิการศึกษาสาขาคอมพิวเตอร์ โดยใช้โครงข่ายประสาทเทียม. วารสารปัญญาภิวัฒน์. 7(2) : 1-16.

ณัฐโชติ พรหมฤทธิ์. 2563. **Introduction to NLP.** [Online]. เข้าถึงได้จาก https://blog.pjjop.org/intro-to-nlp-for-air/

เทอดศักดิ์ เงินมูล, พิเชษฐ เหมยคำ, วิโรจน์ ปงลังกา และวิวัฒน์ ทิพจร. 2560. **การคัดแยกความสุกสตรอเบอรี่ด้วยซัพพอร์ตเวกเตอร์แมชชีน.** Naresuan University Engineering Journal. 12(2) : 55-62.

เอกสารอ้างอิง (ต่อ)

นพพร คล้ายพงษ์พันธุ์, เรวัต เลิศฤทัยโยธิน, รังสฤษดิ์ กาวีต๊ะ และสนธิชัย จันทร์เปรม. 2547. **พืชเศรษฐกิจ.** พิมพ์ครั้งที่ 2. กรุงเทพฯ. สำนักพิมพ์มหาวิทยาลัยเกษตรศาสตร์.

บุษบงก์ คชินทรโรจน์, เดือนเพ็ญ ธีรวรรณวิวัฒน์ และพาชิตชนัต สิริพานิช. 2564. **การสร้างระบบคัดกรองข้อความการเกลียดกลัวคนต่างชาติบนทวิตเตอร์ในช่วงการแพร่ระบาดของโรคติดเชื้อไวรัสโคโรนา 2019.** วารสารไทยการวิจัยดำเนินงาน. 9(1): 31-44.

ปฏิภาณ ประเสริฐสม และพีรดล สามะศิริ. 2564. **การค้นหาตัวแทนเชิงความหมายของข้อความ: Word2Vec Word Embedding, Part I.** [Online]. เข้าถึงได้จาก https://bigdata.go.th/big-data-101/word2vec/

ภูมิรพี ภูมิค้า. 2562. **เทคนิคการเรียนรู้เชิงลึกเพื่อวิเคราะห์ความรู้สึกจากผู้ใช้ผลิตภัณฑ์.** วิศวกรรมศาสตรมหาบัณฑิต สาขาวิชาวิสวกรรมโทรคมนาคมและคอมพิวเตอร์. มหาวิทยาลัยเทคโนโลยีสุรนารี.

ยุทธ ไกยวรรณ์. 2555. **หลักการและการใช้การวิเคราะห์การถดถอยโลจิสติกสำหรับการวิจัย.** วารสารวิจัยมหมาวิทยาลัยเทคโนโลยีราชมงคลศรีวิชัย. 4(1) : 1-12.

ยุวดี เปรมวิชัย. 2564. **Data Analytics: Prediction with Logistic Regression.** [Online]. เข้าถึงได้จาก https://www.mebmarket.com/ebook-153641-Data-Analytics-Prediction-with-Logistic-Regression

สถาบันวิจัยและพัฒนาแห่งมหาวิทยาลัยเกษตรศาสตร์. 2560. **ถิ่นฐานดั้งเดิมของข้าวโพด.** [Online]. เข้าถึงได้จาก https://www3.rdi.ku.ac.th/?p=8961

สำนักเศรษฐกิจการเกษตร. 2565. **ข้าวโพดเลี้ยงสัตว์**. [Online]. เข้าถึงได้จาก https://mis-app.oae.go.th/product/ข้าวโพดเลี้ยงสัตว์

สำนักหอสมุดและศูนย์สารสนเทศวิทยาศาสตร์และเทคโนโลยี. 2561. **ประมวลสารสนเทศพร้อมใช้ เรื่อง "ข้าวโพด (Corn)".** [Online]. เข้าถึงได้จาก http://siweb1.dss.go.th/repack/fulltext/IR47.pdf

Copestake, A. 2004. **Natural Language Processing.** [Online]. Available https://www.cl.cam.ac.uk/teaching/2002/NatLangProc/revised.pdf

Ganguly, D, Roy, D, Mitra, M. and Jones, G. J. 2015. **Word embedding based generalized language model for information retrieval.** In Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval. 795-798.

เอกสารอ้างอิง (ต่อ)

Kara, Y, Boyacioglu, M. A. and Baykan, O. K. 2011. **Predicting direction of stock price index movement using artificial neural networks and support vector machines: The sample of the Istanbul Stock Exchange.** Expert systems with Applications. 38(5) : 5311-5319.

Mikolov, T, Sutskever, I, Chen, K, Corrado, G. S. and Dean, J. 2013. **Distributed representations of words and phrases and their compositionality.** In Advances in Neural Information Processing Systems 26. 3111-3119.

Ninenox Developer. 2020. **ทำความเข้าใจ accuracy, precision, recall, f1-score.** [Online]. Available http://www.ninenox.com/2020/09/24/ทำความเข้าใจ-accuracyprecisionrecallf1-score/

Perlato, A. 2019. **Backpropagation Intuition.** [Online]. Available https://www.andreaperlato.com/aipost/backpropagation/

Ramos, J. 2003. **Using TF-IDF to Determine Word Relevance in Document Queries.** In Proceedings of the First Instructional Conference on Machine Learning. 29-48.

Rong, X. 2014. **word2vec Parameter Learning Explained.** [Online]. Available https://arxiv.org/pdf/1411.2738.pdf

SAS. 2563. **การประมวลภาษาธรรมชาติ.** [Online]. เข้าถึงได้จาก https://www.sas.com/th_th/insights/analytics/what-is-natural-language-processing-nlp.html

Soni, V. D. 2018. **Prediction of Geniunity of News using advanced Machine Learning and Natural Language processing Algorithms.** International Journal of Innovative Research in Science Engineering and Technology. 7(5) : 6349-6354.

Velay, M and Daniel, F. 2018. **Using NLP on news headlines to predict index trends.** [Online]. Available https://arxiv.org/abs/1806.09533

Yildirim, S, Jothimani, D, Kavakioglu, C. and Basar, A. 2018. **Classification of "Hot News" for Financial Forecast Using NLP Techniques.** In 2018 IEEE International Conference on Big Data (Big Data). 4719-4722

เอกสารอ้างอิง (ต่อ)

Zhai, Y, Hsu, A. and Halgamuge S. K. 2007. **Combining News and Technical Indicators in Daily Stock Price Trends Prediction.** In Advances in Neural Networks–ISNN 2007: 4th International Symposium on Neural Networks, ISNN 2007, Nanjing, China.

Zhang, Y, Jin, R. and Zhou, Z. H. 2010. **Understanding bag-of-words model: a statistical framework.** International Journal of Machine Learning and Cybernetics. 1: 43-52.

ภาคผนวก

ภาคผนวก ก

ตารางที่ ก การทำ Grid Search เพื่อหา Feature Selection ที่เหมาะสม

Data Set	Vocabulary size	Prob Level	Accuracy	Precision	Sensitivity (Recall)	Specificity	Absolute	False POS	False NEG	F1-Score	AUC
Train Set	5,000	0.5	100.00%	100.00%	100.00%	100.00%	0.00%	0.00%	0.00%	100.00%	46.73%
Test Set	5,000	0.5	46.73%	44.70%	49.24%	44.44%	4.80%	55.56%	50.76%	46.86%	
Train Set	3,000	0.5	100.00%	100.00%	100.00%	100.00%	0.00%	0.00%	0.00%	100.00%	46.67%
Test Set	3,000	0.5	46.73%	44.84%	50.76%	43.06%	7.70%	56.94%	49.24%	47.62%	
Train Set	2,000	0.5	100.00%	100.00%	100.00%	100.00%	0.00%	0.00%	0.00%	100.00%	48.42%
Test Set	2,000	0.5	49.88%	47.64%	51.27%	48.61%	2.66%	51.39%	48.73%	49.39%	
Train Set	1,000	0.5	96.79%	97.29%	95.39%	97.57%	2.18%	2.43%	4.07%	96.61%	45.74%
Test Set	1,000	0.5	45.04%	43.42%	50.25%	40.28%	9.97%	59.72%	49.75%	46.59%	
Train Set	500	0.5	74.27%	73.32%	72.30%	76.07%	3.77%	23.93%	27.70%	72.81%	45.66%
Test Set	500	0.5	47.46%	45.10%	46.70%	48.15%	1.45%	51.85%	53.30%	45.89%	
Train Set	400	0.5	70.76%	69.19%	69.63%	71.79%	2.16%	28.21%	30.37%	69.41%	47.31%
Test Set	400	0.5	46.97%	44.86%	48.73%	45.37%	3.36%	54.63%	51.27%	46.72%	
Train Set	350	0.5	69.01%	67.56%	67.22%	70.64%	3.42%	29.36%	32.78%	67.39%	50.04%
Test Set	350	0.5	50.85%	48.62%	53.81%	48.15%	5.66%	51.85%	46.19%	51.08%	
Train Set	330	0.5	68.34%	67.23%	65.44%	70.98%	5.54%	29.02%	34.56%	75.66%	50.20%
Test Set	330	0.5	51.33%	49.06%	52.79%	50.00%	2.79%	50.00%	47.21%	50.86%	
Train Set	300	0.5	67.80%	66.33%	65.82%	69.60%	3.78%	30.40%	34.18%	66.07%	49.06%
Test Set	300	0.5	46.49%	44.55%	49.75%	43.52%	6.23%	56.48%	50.25%	47.00%	
Train Set	200	0.5	63.32%	61.77%	60.36%	66.01%	5.65%	33.99%	39.64%	61.05%	49.63%
Test Set	200	0.5	51.33%	49.04%	51.78%	50.93%	0.85%	49.07%	48.22%	50.37%	
Train Set	100	0.5	59.93%	58.74%	53.37%	65.90%	12.53%	34.10%	46.63%	55.93%	49.87%
Test Set	100	0.5	52.30%	50.00%	46.19%	57.87%	11.68%	42.13%	53.81%	48.02%	

จากตาราง ก เมื่อใช้การแปลงคุณลักษณะ Bag of Words และแบบจำลองการถดถอยลอจิสติก (Logistic Regression) พบว่าเมื่อกำหนดคลังคำศัพท์ (Vocabulary size) เท่ากับ 330 และจุดตัด (Cutoff Probability) เท่ากับ 0.5 พบว่า ค่า AUC มีค่าสูงที่สุดคือ 50.20%

เมื่อพิจารณาจากค่า Sensitivity และ Specificity จากทั้งข้อมูลชุด Train Set และ Test Set (โดยจะพิจารณาเลือกตัวแบบที่ให้ค่า Sensitivity และ Specificity ที่ให้ค่าใกล้เคียงกันมากที่สุด (Balance between Sensitivity and Specificity)) เมื่อใช้การแปลงคุณลักษณะ Bag of Words และแบบจำลองการถดถอยลอจิสติก พบว่าเมื่อกำหนดกำหนดคลังคำศัพท์ (Vocabulary size) เท่ากับ 330 และจุดตัดเป็น 0.5 ให้ค่า Sensitivity และ Specificity ใกล้เคียงกันมากที่สุด โดยให้ค่า Sensitivity เท่ากับ 65.44% และ Specificity เท่ากับ 70.98% ในชุดข้อมูล Train set (ค่า Sensitivity และ Specificity แตกต่างกัน 5.54%) และ Sensitivity เท่ากับ 52.79% และ Specificity เท่ากับ 50.00% ในชุดข้อมูล Test set (ค่า Sensitivity และ Specificity แตกต่างกัน 2.79%) ตามลำดับ

ดังนั้นตัวแบบที่เหมาะสมที่สุดคือ เมื่อใช้การแปลงคุณลักษณะ Bag of Words และแบบจำลองการถดถอยลอจิสติก ที่กำหนดคลังคำศัพท์ (Vocabulary size) เท่ากับ 330 และจุดตัดเป็น 0.5 โดยตัวแบบข้างต้น ให้ค่า Accuracy และ F1 Score เท่ากับ 68.34% และ 75.66% ตามลำดับในข้อมูลชุด Train Set และให้ค่า Accuracy และ F1 Score เท่ากับ 51.33% และ 50.86% ในข้อมูลชุด Test Set

ภาคผนวก ข

การพยากรณ์ทิศทางราคาสัญญาฟิวเจอร์สข้าวโพดจากข้อความข่าวโดยใช้การประมวลภาษาธรรมชาติ โดยใช้ภาษาไพธอน (Python) มีคำสั่งดังต่อไปนี้

ชุดคำสั่งที่ใช้ในงานวิจัย

1. คำสั่งดึงข้อมูลข่าวรายวันจากฐานข้อมูล

```
# Import library
import os
import pandas as pd

folder_path = 'raw_data'

files = [f for f in os.listdir(folder_path) if f.endswith('.xlsx')]
files.sort()

counter = 0
dataframes = []
year = 2013

# Loop through each file in the list
for i, file in enumerate(files):
    file_path = os.path.join(folder_path, file)
    df = pd.read_excel(file_path)
    df = df.apply(lambda x: x.str.lower() if x.dtype == "object" else x)
    df['DatePublishSKey'] = pd.to_datetime(df['DatePublishSKey'],
format='%Y%m%d')
    df['DatePublishSKey'] = df['DatePublishSKey'].dt.strftime('%Y-%m-%d')
    dataframes.append(df)

    counter += 1
```

```
    if counter == 12 or i == len(files) - 1:
        result = pd.concat(dataframes, axis=0)
        filtered_df = result[(result['NewsTopic'].str.contains("corn|maize")) |
        (result['NewsContent'].str.contains("corn|maize"))]
        filtered_df.to_excel(f'corn_news_{year}.xlsx', index=False)

        dataframes = []
        counter = 0
        year += 1
```

2. คำสั่งสำหรับจัดเตรียมข้อมูล

2.1 แปลงราคาสัญญาฟิวเจอร์สข้าวโพดก่อนนำไปสร้างแบบจำลอง

```
import pandas as pd
import plotly.graph_objects as go

df = pd.read_excel('./Price.xlsx',sheet_name='Corn')
df['Dates'] = pd.to_datetime(df['Dates']).dt.date

start_date = pd.to_datetime("2013-01-01")
end_date = pd.to_datetime("2021-12-31")
filtered_df = df[(df["Dates"] >= start_date) & (df["Dates"] <=
end_date)].copy().reset_index(drop=True)

tmp = filtered_df[['Dates', 'LAST']].copy()
tmp['Last+1'] = filtered_df['LAST'].shift(-1)
tmp['pctDiff_Next'] = tmp['Last+1'] - tmp['LAST']

def dir(pct):
    if pct >= 0:
        return 1
    else:
        return 0
tmp['label'] = tmp['pctDiff_Next'].apply(lambda x: dir(x))
```

```
tmp['label'].value_counts()
value_counts = tmp['label'].value_counts()
percentages = 100 * value_counts / len(tmp)
print(percentages)

real = tmp[['Dates','label']].copy()
real.to_excel('file_label.xlsx', index=False)
```

2.2 การเตรียมข้อมูลข่าวรายวันที่ได้มาจากฐานข้อมูล

```
import os
import pandas as pd
import re

folder_path = 'corn_new_from_rowdata'

files = [f for f in os.listdir(folder_path) if f.endswith('.xlsx')]

dfs = []
for file in files:
    file_path = os.path.join(folder_path, file)
    df = pd.read_excel(file_path)
    dfs.append(df)

final_df = pd.concat(dfs, axis=0, ignore_index=True)
tmp = final_df[['DatePublishSKey','NewsTopic','NewsContent']].copy()

def clean_text(text):
    # Remove newline and carriage return characters
    text = re.sub(r'\n|\r', ' ', text)
    # Remove Unicode control characters
    text = re.sub(r'_x[0-9A-Fa-f]{4}_', '', text)
    # Remove extra spaces
    text = re.sub(r' +', ' ', text)
```

```
    return text.strip()
tmp['NewsContent'] = tmp['NewsContent'].apply(clean_text)
tmp['NewsTopic'] = tmp['NewsTopic'].apply(clean_text)
tmp['NewsContent'] = tmp['NewsContent'].apply(clean_text)

tmp['News'] = tmp.apply(lambda row: row['NewsTopic'] + ' ' + row['NewsContent'],
axis=1)

data = tmp.groupby('DatePublishSKey').agg({'NewsTopic': ' '.join, 'NewsContent': '
'.join, 'News': ' '.join}).reset_index()

# นำการแปลงราคาสัญญาฟิวเจอร์สข้าวโพดเข้ามาเพื่อกำหนด Label ให้ข้อความข่าวในแต่ละวัน
file_label = pd.read_excel('file_label.xlsx')

file_label = file_label.rename(columns={'Dates': 'DatePublishSKey'})

data['DatePublishSKey'] = pd.to_datetime(data['DatePublishSKey'])
file_label['DatePublishSKeys'] = pd.to_datetime(file_label['DatePublishSKey'])

merged_df = pd.merge(data, file_label, on='DatePublishSKey')
merged_df = merged_df.rename(columns={'label': 'Class'})

result_df = merged_df[['DatePublishSKey', 'NewsTopic', 'NewsContent', 'News',
'Class']]

result_df['Class'].value_counts()
value_counts = result_df['Class'].value_counts()
percentages = 100 * value_counts / len(result_df)
print(percentages)

result_df.to_excel('corn_news_data.xlsx', index=False)
```

2.3 การทำความสะอาดข้อมูล (Text Cleaning)

```
import pandas as pd
import numpy as np
import nltk
nltk.download('punkt')
nltk.download('stopwords')
import re

from nltk.tokenize import sent_tokenize, word_tokenize
from nltk.corpus import stopwords
from nltk.stem import WordNetLemmatizer, SnowballStemmer

df = pd.read_excel('corn_news_data.xlsx')

def preprocess_text(text):
    sentences = sent_tokenize(text)
    words = []

    # Tokenize text into sentences and words
    for sentence in sentences:
        words += word_tokenize(sentence)

    # Remove stop words
    stop_words = set(stopwords.words('english'))
    words = [word for word in words if word not in stop_words]

    # Apply lemmatization
    lemmatizer = WordNetLemmatizer()
    words = [lemmatizer.lemmatize(word) for word in words]

    # Apply Stemming
    stemmer = SnowballStemmer('english')
    words = [stemmer.stem(word) for word in words]
```

```
    # Remove numbers and special characters
    words = [re.sub(r'[^A-Za-z]+', '', word) for word in words]

    # Remove extra whitespaces
    words = [word.strip() for word in words if word.strip()]

    # Return preprocessed text
    return ' '.join(words)

df['preprocessed_News'] = df['News'].apply(preprocess_text)

preprocessed_df = df[['DatePublishSKey', 'preprocessed_News', 'Class']]
preprocessed_df['DatePublishSKey'] =
pd.to_datetime(preprocessed_df['DatePublishSKey']).dt.date

start_date = pd.to_datetime("2013-01-01")
end_date = pd.to_datetime("2020-12-31")
preprocessed_df_date = preprocessed_df[(preprocessed_df["DatePublishSKey"] >=
start_date) & (preprocessed_df["DatePublishSKey"] <=
end_date)].copy().reset_index(drop=True)

preprocessed_df_date.to_excel('preprocessed_News.xlsx', index=False)
```

3. การสร้างตัวแปลงข้อมูลด้วยวิธี Bag of Words (คลังคำศัพท์ 5,000 คำ)

```
import pandas as pd
import numpy as np
import tensorflow as tf
import random
from sklearn.model_selection import train_test_split
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.linear_model import LogisticRegression
```

```
from sklearn import svm
from sklearn.svm import SVC, LinearSVC
from sklearn.preprocessing import StandardScaler
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense, Dropout
from tensorflow.keras.callbacks import ModelCheckpoint
from sklearn.metrics import classification_report, f1_score, accuracy_score,
precision_score, recall_score, confusion_matrix, roc_curve, roc_auc_score

df = pd.read_excel('preprocessed_News.xlsx')

X_train, X_test, y_train, y_test = train_test_split(df['preprocessed_News'], df['Class'],
test_size=0.2, shuffle=False)

# กำหนดคลังคำศัพท์ 5,000 (max_features=5000)
vectorizer = CountVectorizer(max_features = 5000)
X_train_bow = vectorizer.fit_transform(X_train).toarray()
X_test_bow = vectorizer.transform(X_test).toarray()

# Get feature names as columns in a dataframe
fn = vectorizer.get_feature_names()
pd.DataFrame(X_train_bow, columns=fn)
```

3.1 การสร้างตัวแปลงข้อมูลด้วยวิธี Bag of Words และแบบจำลอง Logistic Regression

```
model = LogisticRegression()
model.fit(X_train_bow, y_train)

# Evaluate the model on the train data
print("Train Data Evaluation:")
print(confusion_matrix(y_train, y_pred_train))
print(classification_report(y_train, y_pred_train, digits=4))
print("Accuracy Train set :", accuracy_score(y_train, y_pred_train))
```

```
# Evaluate the model on the test data
y_pred = model.predict(X_test_bow)
pd.crosstab(y_test, y_pred, rownames=["Actual"], colnames=["Predicted"],
margins=True)
print("Test Data Evaluation:")
print(confusion_matrix(y_test, y_pred))
print(classification_report(y_test, y_pred, digits=4))
print("Accuracy Test set :",accuracy_score(y_test, y_pred))

# การหาค่า Coef.
feature_importance = pd.DataFrame({'Feature': fn, 'Coefficient': model.coef_[0]})
feature_importance =
feature_importance.reindex(feature_importance['Coefficient'].abs().sort_values(asce
nding=False).index)
print(feature_importance)
```

3.2 การสร้างตัวแปลงข้อมูลด้วยวิธี Bag of Words และแบบจำลอง SVM

```
model = svm.SVC(kernel='rbf')
model.fit(X_train_bow, y_train)

# Evaluate the model on the train data
print("Train Data Evaluation:")
print(confusion_matrix(y_train, y_pred_train))
print(classification_report(y_train, y_pred_train, digits=4))
print("Accuracy Train set :", accuracy_score(y_train, y_pred_train))

# Evaluate the model on the test data
y_pred = model.predict(X_test_bow)
pd.crosstab(y_test, y_pred, rownames=["Actual"], colnames=["Predicted"],
margins=True)
print("Test Data Evaluation:")
print(confusion_matrix(y_test, y_pred))
print(classification_report(y_test, y_pred, digits=4))
```

```
print("Accuracy Test set :",accuracy_score(y_test, y_pred))
```

3.3 การสร้างตัวแปลงข้อมูลด้วยวิธี Bag of Words และแบบจำลอง ANN

```
seed_value = 0
random.seed(seed_value)
np.random.seed(seed_value)
tf.random.set_seed(seed_value)

model = Sequential()
model.add(Dense(128, input_dim=X_train_bow.shape[1], activation='relu'))
model.add(Dense(64))
model.add(Dense(32))
model.add(Dense(16))
model.add(Dense(1, activation='sigmoid'))
model.summary()

model.compile(loss='binary_crossentropy', optimizer='adam', metrics=['accuracy']
,run_eagerly = True)
model.fit(X_train_bow, y_train, batch_size=10, epochs=10, shuffle=False)

# Evaluate the model on the train data
y_pred_train = (model.predict(X_train_bow) > 0.5).astype(int)
print("Train Data Evaluation:")
print(confusion_matrix(y_train, y_pred_train))
print(classification_report(y_train, y_pred_train, digits=4))
print("Accuracy Train set :", accuracy_score(y_train, y_pred_train))

# Evaluate the model on the test data
y_pred = (model.predict(X_test_bow) > 0.5).astype(int)
print("Test Data Evaluation:")
print(confusion_matrix(y_test, y_pred))
print(classification_report(y_test, y_pred, digits=4))
print("Accuracy Test set :",accuracy_score(y_test, y_pred))
```

```
loss, accuracy = model.evaluate(X_test_bow, y_test)
print("Test Loss: ", loss)
print("Test Accuracy: ", accuracy)
```

4. การสร้างตัวแปลงข้อมูลด้วยวิธี TF-IDF (คลังคำศัพท์ 5,000 คำ)

```
import pandas as pd
import numpy as np
import tensorflow as tf
import random
from sklearn.model_selection import train_test_split
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.linear_model import LogisticRegression
from sklearn import svm
from sklearn.svm import SVC, LinearSVC
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense
from tensorflow.keras.callbacks import ModelCheckpoint
from sklearn.metrics import classification_report, f1_score, accuracy_score,
precision_score, recall_score, confusion_matrix

df = pd.read_excel('preprocessed_News.xlsx')

X_train, X_test, y_train, y_test = train_test_split(df['preprocessed_News'], df['Class'],
test_size=0.2, shuffle=False)

# กำหนดคลังคำศัพท์ 5,000 (max_features=5000)
vectorizer = TfidfVectorizer(max_features=5000)
X_train_bow = vectorizer.fit_transform(X_train).toarray()
X_test_bow = vectorizer.transform(X_test).toarray()

# Get feature names as columns in a dataframe
fn = vectorizer.get_feature_names()
pd.DataFrame(X_train_bow, columns=fn)
```

4.1 การสร้างตัวแปลงข้อมูลด้วยวิธี TF-IDF และแบบจำลอง Logistic Regression

```
model = LogisticRegression()
model.fit(X_train_bow, y_train)

# Evaluate the model on the train data
print("Train Data Evaluation:")
print(confusion_matrix(y_train, y_pred_train))
print(classification_report(y_train, y_pred_train, digits=4))
print("Accuracy Train set :", accuracy_score(y_train, y_pred_train))

# Evaluate the model on the test data
y_pred = model.predict(X_test_bow)
pd.crosstab(y_test, y_pred, rownames=["Actual"], colnames=["Predicted"],
margins=True)
print("Test Data Evaluation:")
print(confusion_matrix(y_test, y_pred))
print(classification_report(y_test, y_pred, digits=4))
print("Accuracy Test set :",accuracy_score(y_test, y_pred))

# การหาค่า Coef.
feature_importance = pd.DataFrame({'Feature': fn, 'Coefficient': model.coef_[0]})
feature_importance =
feature_importance.reindex(feature_importance['Coefficient'].abs().sort_values(asce
nding=False).index)
print(feature_importance)
```

4.2 การสร้างตัวแปลงข้อมูลด้วยวิธี TF-IDF และแบบจำลอง SVM

```
model = svm.SVC(kernel='rbf')
model.fit(X_train_bow, y_train)

# Evaluate the model on the train data
print("Train Data Evaluation:")
print(confusion_matrix(y_train, y_pred_train))
```

```
print(classification_report(y_train, y_pred_train, digits=4))
print("Accuracy Train set :", accuracy_score(y_train, y_pred_train))

# Evaluate the model on the test data
y_pred = model.predict(X_test_bow)
pd.crosstab(y_test, y_pred, rownames=["Actual"], colnames=["Predicted"],
margins=True)
print("Test Data Evaluation:")
print(confusion_matrix(y_test, y_pred))
print(classification_report(y_test, y_pred, digits=4))
print("Accuracy Test set :",accuracy_score(y_test, y_pred))
```

4.3 การสร้างตัวแปลงข้อมูลด้วยวิธี TF-IDF และแบบจำลอง ANN

```
seed_value = 0
random.seed(seed_value)
np.random.seed(seed_value)
tf.random.set_seed(seed_value)

model = Sequential()
model.add(Dense(128, input_dim=X_train_bow.shape[1], activation='relu'))
model.add(Dense(64))
model.add(Dense(32))
model.add(Dense(16))
model.add(Dense(1, activation='sigmoid'))
model.summary()

model.compile(loss='binary_crossentropy', optimizer='adam', metrics=['accuracy']
,run_eagerly = True)
model.fit(X_train_bow, y_train, batch_size=10, epochs=10, shuffle=False)
# Evaluate the model on the train data
y_pred_train = (model.predict(X_train_bow) > 0.5).astype(int)
print("Train Data Evaluation:")
print(confusion_matrix(y_train, y_pred_train))
```

```
print(classification_report(y_train, y_pred_train, digits=4))
print("Accuracy Train set :", accuracy_score(y_train, y_pred_train))

# Evaluate the model on the test data
y_pred = (model.predict(X_test_bow) > 0.5).astype(int)
print("Test Data Evaluation:")
print(confusion_matrix(y_test, y_pred))
print(classification_report(y_test, y_pred, digits=4))
print("Accuracy Test set :",accuracy_score(y_test, y_pred))

loss, accuracy = model.evaluate(X_test_bow, y_test)
print("Test Loss: ", loss)
print("Test Accuracy: ", accuracy)
```

5. การสร้างตัวแปลงข้อมูลด้วยวิธี Word2Vec (เวกเตอร์คุณลักษณะจำนวน 5,000 ตัว)

```
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from gensim.models import Word2Vec
import tensorflow as tf
import random
from sklearn.linear_model import LogisticRegression
from sklearn import svm
from sklearn.svm import SVC, LinearSVC
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense
from sklearn.metrics import classification_report, f1_score, accuracy_score,
precision_score, recall_score, confusion_matrix

df = pd.read_excel('preprocessed_News.xlsx')

X_train, X_test, y_train, y_test = train_test_split(df['preprocessed_News'], df['Class'],
test_size=0.2, shuffle=False)
```

```
sentences = [sentence.split() for sentence in X_train]

# กำหนดเวกเตอร์คุณลักษณะ 5000 ตัว (vector_size=5000) ใช้โมเดล CBOW (sg=0)
w2v_model = Word2Vec(sentences, vector_size=5000, window=5, min_count=5,
workers=4, sg=0 , epochs=5)

# Vectorize the text data
def vectorize(sentence):
    words = sentence.split()
    words_vecs = [w2v_model.wv[word] for word in words if word in
w2v_model.wv]
    if len(words_vecs) == 0:
        return np.zeros(5000)
    words_vecs = np.array(words_vecs)
    return words_vecs.mean(axis=0)

X_train_w2v = np.array([vectorize(sentence) for sentence in X_train])
X_test_w2v = np.array([vectorize(sentence) for sentence in X_test])
```

5.1 การสร้างตัวแปลงข้อมูลด้วยวิธี Word2Vec และแบบจำลอง Logistic Regression

```
model = LogisticRegression()
model.fit(X_train_w2v, y_train)

# Evaluate the model on the train data
print("Train Data Evaluation:")
print(confusion_matrix(y_train, y_pred_train))
print(classification_report(y_train, y_pred_train, digits=4))
print("Accuracy Train set :", accuracy_score(y_train, y_pred_train))

# Evaluate the model on the test data
y_pred = model.predict(X_test_w2v)
```

```
pd.crosstab(y_test, y_pred, rownames=["Actual"], colnames=["Predicted"],
margins=True)
print("Test Data Evaluation:")
print(confusion_matrix(y_test, y_pred))
print(classification_report(y_test, y_pred, digits=4))
print("Accuracy Test set :",accuracy_score(y_test, y_pred))
```

5.2 การสร้างตัวแปลงข้อมูลด้วยวิธี Word2Vec และแบบจำลอง SVM

```
model = svm.SVC(kernel='rbf')
model.fit(X_train_w2v, y_train)

# Evaluate the model on the train data
print("Train Data Evaluation:")
print(confusion_matrix(y_train, y_pred_train))
print(classification_report(y_train, y_pred_train, digits=4))
print("Accuracy Train set :", accuracy_score(y_train, y_pred_train))

# Evaluate the model on the test data
y_pred = model.predict(X_test_w2v)
pd.crosstab(y_test, y_pred, rownames=["Actual"], colnames=["Predicted"],
margins=True)
print("Test Data Evaluation:")
print(confusion_matrix(y_test, y_pred))
print(classification_report(y_test, y_pred, digits=4))
print("Accuracy Test set :",accuracy_score(y_test, y_pred))
```

5.3 การสร้างตัวแปลงข้อมูลด้วยวิธี Word2Vec และแบบจำลอง ANN

```
seed_value = 0
random.seed(seed_value)
np.random.seed(seed_value)
tf.random.set_seed(seed_value)
```



```
model = Sequential()
```

```
model.add(Dense(128, input_dim=X_train_w2v.shape[1], activation='relu'))
model.add(Dense(64))
model.add(Dense(32))
model.add(Dense(16))
model.add(Dense(1, activation='sigmoid'))
model.summary()

model.compile(loss='binary_crossentropy', optimizer='adam', metrics=['accuracy']
,run_eagerly = True)
model.fit(X_train_w2v, y_train, batch_size=10, epochs=10, shuffle=False)

# Evaluate the model on the train data
print("Train Data Evaluation:")
print(confusion_matrix(y_train, y_pred_train))
print(classification_report(y_train, y_pred_train, digits=4))
print("Accuracy Train set :", accuracy_score(y_train, y_pred_train))

# Evaluate the model on the test data
y_pred = (model.predict(X_test_w2v) > 0.5).astype(int)
print("Test Data Evaluation:")
print(confusion_matrix(y_test, y_pred))
print(classification_report(y_test, y_pred, digits=4))
print("Accuracy Test set :",accuracy_score(y_test, y_pred))

loss, accuracy = model.evaluate(X_test_w2v, y_test)
print("Test Accuracy: ", accuracy)
```

งานทะเบียนคณะวิทยาศาสตร์
สภาบันเทคโนโลยีพนะจอมเกล้าเจ้าคุณทหารลาดกระบัง
คำรับรองเล่มสหกิจศึกษา

วันที่ 8 เดือน มิถุนายน พ.ศ 2566

ข้าพเจ้า นายวันธนวัฒน์ พิมพ์สุข รหัสประจำตัว 62050830
นักศึกษาหลักสูตรวิทยาศาสตรบัณฑิต สาขาวิชา สถิติประยุกต์ ภาควิชา สถิติ
ขอรับรองว่าสหกิจศึกษา เรื่อง

ชื่อภาษาไทย การพยากรณ์ทิศทางราคาสัญญาฟิวเจอร์สข้าวโพดจากข้อความข่าว โดยใช้การประมวลภาษาธรรมชาติ

ชื่อภาษาอังกฤษ CORN FUTURES PRICE TREND PREDICTION FROM NEWS CONTENT USING NATURAL LANGUAGE PROCESSING

ปีการศึกษา 2565

เป็นผลงานวิจัยที่มิได้คัดลอกหรือละเมิดลิขสิทธิ์ของผู้อื่นและได้ผ่านการตรวจสอบความซ้ำซ้อนเรียบร้อยแล้ว และได้แนบเอกสารการตรวจสอบการลอกเลียนงานวรรณกรรมที่ตรวจสอบจากเล่มสหกิจศึกษาฉบับสมบูรณ์แล้ว
โปรแกรมอักขราวิสุทธิ์ 1.66 %

ลงชื่อ วันธนวัฒน์ พิมพ์สุข
(นายวันธนวัฒน์ พิมพ์สุข)
นักศึกษา

ข้าพเจ้า ผศ.ดร.ยุวดี กล่อมวิเศษ อาจารย์ที่ปรึกษาสหกิจศึกษา ได้ตรวจสอบสหกิจศึกษาของนักศึกษาข้างต้นแล้ว ขอรับรองว่าเป็นผลงานวิจัยของนักศึกษาจริงและมีเนื้อหาสมบูรณ์ จึงลงชื่อไว้เป็นหลักฐาน

ลงชื่อ.....
อาจารย์ที่ปรึกษา