โครงการ

การเรียนการสอนเพื่อเสริมประสบการณ์

ชื่อโครงการ	การพัฒนาโมเดลการจำแนกความรู้สึกของข้อความภาษาไทยโดยใช้เทคนิคการเรียนรู้ด้วยเครื่อง Development of a classification model for Thai statement sentiments
ชื่อนิสิต	นายนิติกร องค์ศิริมงคล 583 36374 23
ภาควิชา	คณิตศาสตร์และวิทยาการคอมพิวเตอร์ สาขาวิชาวิทยาการคอมพิวเตอร์
ปีการศึกษา	2561

การพัฒนาโมเดลการจำแนกความรู้สึกของข้อความภาษาไทยโดยใช้เทคนิคการเรียนรู้ด้วยเครื่อง

นายนิติกร องค์ศิริมงคล

โครงงานนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิทยาศาสตร์บัณฑิต
สาขาวิชาวิทยาการคอมพิวเตอร์ ภาควิชาคณิตศาสตร์และวิทยาการคอมพิวเตอร์
คณะวิทยาศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2561

Development of a classification model for Thai statement sentiments

Nitikorn Ongsirimongkol

A Project Submitted in Partial Fulfillment of the Requirements
for the Degree of Bachelor of Science Program in Computer Science
Department of Mathematics and Computer Science
Faculty of Science
Chulalongkorn University
Academic Year 2018

หัวข้อโครงงาน การพัฒนาโมเดลการจำแนกความรู้สึกของข้อความภาษาไทยโดยใช้เทคนิคการเรียนรู้ด้วยเครื่อง

โดย นายนิติกร องค์ศิริมงคล

สาขาวิชา วิทยาการคอมพิวเตอร์

อาจารย์ที่ปรึกษาโครงงานหลัก ผู้ช่วยศาสตราจารย์ ดร.ภควรรณ ปักษี

ภาควิชาคณิตศาสตร์และวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้นับโครงงานฉบับนี้เป็นส่วนหนึ่ง ของการศึกษาตามหลักสูตรปริญญาบัณฑิต ในรายวิชา 2301499 โครงงานวิทยาศาสตร์ (Senior Project)

.....
(ศาสตราจารย์ ดร.กฤษณะ เนียมมณี) หัวหน้าภาควิชาคณิตศาสตร์และวิทยาการคอมพิวเตอร์

คณะกรรมการสอบโครงงาน

.....
(ผู้ช่วยศาสตราจารย์ ดร.ภควรรณ ปักษี) อาจารย์ที่ปรึกษาโครงงานหลัก

.....
(ผู้ช่วยศาสตราจารย์ ดร.กิติพร พลายมาศ) กรรมการ

.....
(รองศาสตราจารย์ ดร.พิเชฐ ชาวหา) กรรมการ

นายนิติกร องค์ศิริมงคล: การพัฒนาโมเดลการจำแนกความรู้สึกของข้อความภาษาไทยโดยใช้เทคนิคการเรียนรู้ด้วยเครื่อง (Development of a classification model for Thai statement sentiments using ML techniques) อ.ที่ปรึกษาโครงงานหลัก: ผู้ช่วยศาสตราจารย์ ดร.ภควรรณ ปักษี, 61 หน้า.

ในปัจจุบันข้อมูลมีอยู่ในหลากหลายรูปแบบ หนึ่งในนั้นคือข้อความในรูปแบบตัวอักษร (text) ที่มีอยู่เป็นจำนวนมากในอินเทอร์เน็ต ซึ่งข้อมูลเหล่านี้สามารถนำไปใช้ทำประโยชน์ได้ในหลาย ๆ ด้าน เช่น การสร้างระบบแนะนำสินค้า (product recommender system) การวิเคราะห์ความรู้สึกจากข้อความ (sentiment analysis) การทำเหมืองข้อมูล (data mining) และอื่น ๆ ดังนั้นผู้จัดทำได้เล็งเห็นถึงความสำคัญของการวิเคราะห์ความรู้สึกจากข้อความ เนื่องจากหากสามารถแบ่งแยกข้อความที่แสดงความรู้สึกทางด้านบวก ทางด้านลบ และไม่แสดงความรู้สึกหรือเป็นกลางออกจากกันได้จะเป็นประโยชน์ในการควบคุมคุณภาพสินค้าหรือรักษาคุณภาพการให้บริการ และการปรับปรุงคุณภาพสินค้าหรือการให้บริการที่ถูกกล่าวถึงในข้อความให้ดีขึ้นได้ โครงการนี้จะเก็บรวบรวมข้อมูลจากรีวิวในกลุ่มโรงแรม ร้านอาหาร สถานที่ท่องเที่ยว และสายการบิน จากเว็บไซต์ tripadvisor โดยจะแยกผลลัพธ์ออกเป็น 3 กลุ่ม คือ ข้อความแสดงความรู้สึกทางด้านบวก ทางด้านลบ และไม่แสดงความรู้สึกหรือเป็นกลาง และนำข้อมูลข้างต้นมาสร้างโมเดลที่ใช้ในการจำแนกความรู้สึกจากข้อความด้วยวิธีการเรียนรู้เชิงลึก ซึ่งมีการทดลองสร้างโมเดลหลาย ๆ รูปแบบ จึงเลือกโมเดลที่ให้ผลลัพธ์ดีที่สุด คือ โมเดลที่สร้างจากโครงข่ายประสาทเทียมแบบสังวัตนาการต่อกันจำนวนสามชั้น ผลการทดสอบประสิทธิภาพของโมเดลพบว่า ได้ผลการจำแนกที่ถูกต้องและแม่นยำมากกว่า 80% ในทุกกลุ่มข้อความ ดังนั้นโมเดลนี้สามารถช่วยให้ผู้ใช้สะดวกในการจำแนกข้อความเป็นประเภทต่าง ๆ ตามความรู้สึกของข้อความเพื่อนำไปใช้ในการควบคุม หรือปรับปรุงสินค้าหรือบริการให้มีความพึงพอใจต่อผู้บริโภคหรือผู้รับบริการมากยิ่งขึ้น

ภาควิชา คณิตศาสตร์และวิทยาการคอมพิวเตอร์ ลายมือชื่อนิสิต นิติกร องค์ศิริมงคล

สาขาวิชา วิทยาการคอมพิวเตอร์

ปีการศึกษา 2561 ลายมือชื่อ อ.ที่ปรึกษาโครงงานหลัก ภควรรณ ปักษี

5833637423: MAJOR COMPUTER SCIENCE

KEYWORDS: MACHINE LEARNING / SENTIMENT ANALYSIS / CLASSIFICATION

NITIKORN ONGSIRIMONGKOL: DEVELOPMENT OF A CLASSIFICATION MODEL FOR THAI STATEMENT SENTIMENTS USING ML TECHNIQUES. ADVISOR: ASST PROF. PAKAWAN PUGSEE, Ph.D., 61 pp.

At present, information is available in a variety of formats. One of them is the text that has a lot of text on the internet. This information can be used in many ways such as product recommender system, sentiment analysis, data mining and others. Therefore, the developer can see the importance of feeling analyzing from messages because if you are able to distinguish messages that show negative feelings, positive feelings and non-expressed feelings or neutrality, it will be useful in controlling the product quality or maintaining the quality of service and improving the quality of products or services that are mentioned in the message. This project will collect data from reviews in hotels, restaurants, tourist attractions and airlines from tripadvisor website, which will separate the results into 3 groups: a message showing a positive and negative feeling and not showing feelings or neutrality. Then, the previous data is used for creating a model by deep learning methods which get from the experiment of developing many various models. After that, the model that gives the best results is chosen. This model is constructed from convolutional neural networks with three connected layers and the results of the model's performance test showed an accuracy and precision of the results are more than 80% in all groups. Therefore, this model can help users to easily classify messages into different groups according to the feelings of the message to be used in the control and improvement of products or services to be more satisfied with consumers or customers.

Department :Mathematics and Computer Science Student's Signature .....

Field of Study : Computer Science

Academic Year : 2018 Advisor's Signature Pakawan Pugsee

กิตติกรรมประกาศ

การจัดทำโครงงานพัฒนาโมเดลจำแนกความรู้สึกจากข้อความภาษาไทยสามารถสำเร็จลุล่วงไปด้วยดี ทั้งนี้เนื่องจากได้รับความอนุเคราะห์และความช่วยเหลือจากคณาจารย์และบุคลากรต่าง ๆ หลายท่าน ดังนี้

ขอขอบพระคุณผู้ช่วยศาสตราจารย์ ดร.ภควรรณ ปักษี อาจารย์ที่ปรึกษาโครงงานที่คอยให้คำแนะนำ คำปรึกษาและข้อเสนอแนะ อีกทั้งยังช่วยแก้ไขและชี้แนะแนวทางในการทำงานตลอดการทำงานทั้งโครงงาน

ขอขอบพระคุณคณะกรรมการสอบ ได้แก่ ผู้ช่วยศาสตราจารย์ ดร.กิติพร พลายมาศและรองศาสตราจารย์ ดร.พิเชฐ ชาวหา ที่ช่วยให้คำแนะนำ และข้อเสนอแนะ ซึ่งช่วยในการนำเสนอและพัฒนาโครงงานและยังช่วยแก้ไขแนวทางในการทำงานให้มีประสิทธิภาพที่ดียิ่งขึ้น

ขอขอบพระคุณ คุณพ่อคุณแม่ที่คอยให้กำลังใจตลอดการพัฒนาโครงงาน แม้ในยามที่ต้องเจอกับปัญหาและอุปสรรคต่าง ๆ

ขอขอบคุณเพื่อน ๆ ภาควิชาคณิตศาสตร์และวิทยาการคอมพิวเตอร์ที่คอยให้กำลังใจและความเข้าอกเข้าใจ รวมถึงความช่วยเหลือต่าง ๆ ที่ช่วยให้การพัฒนาโครงงานดำเนินไปอย่างราบรื่นและสำเร็จลุล่วงไปด้วยดี

สารบัญ

หน้า

สารบัญตาราง

หน้า

สารบัญรูป

หน้า

บทที่ 1
บทนำ

1.1 ความเป็นมาและเหตุผลของโครงงาน

ในปัจจุบันข้อมูลมีอยู่ในหลากหลายรูปแบบ หนึ่งในนั้นคือข้อความในรูปแบบตัวอักษร (text) ที่มีอยู่เป็นจำนวนมากในอินเทอร์เน็ต ซึ่งข้อมูลเหล่านี้สามารถนำไปใช้ทำประโยชน์ได้ในหลาย ๆ ด้าน เช่น การสร้างระบบแนะนำสินค้า (product recommender system) การวิเคราะห์ความรู้สึกจากข้อความ (sentiment analysis) การทำเหมืองข้อมูล (data mining) และอื่น ๆ แต่ผู้จัดทำได้เล็งเห็นถึงความสำคัญของการวิเคราะห์ความรู้สึกจากข้อความ เนื่องจากหากสามารถแบ่งแยกข้อความที่แสดงความรู้สึกทางด้านบวก ทางด้านลบและไม่แสดงความรู้สึกหรือเป็นกลางได้ออกจากกันได้จะเป็นประโยชน์ในการควบคุมคุณภาพสินค้าหรือรักษาคุณภาพการให้บริการ และการปรับปรุงคุณภาพสินค้าหรือการให้บริการที่ถูกกล่าวถึงในข้อความให้ดีขึ้นได้

แต่ในการพัฒนาระบบวิเคราะห์ความรู้สึกจากข้อความด้วยเทคนิคการเรียนรู้ด้วยเครื่อง (machine learning) ในวิธีหนึ่งจำเป็นต้องใช้รายการคำ (word list) ที่ประกอบด้วยคำที่แสดงความรู้สึกด้านบวก และคำที่แสดงความรู้สึกด้านลบ ซึ่งการระบุคำเหล่านี้ให้ครอบคลุมทุกคำเป็นเรื่องยาก ทางผู้จัดทำจึงวิเคราะห์ข้อมูลประเภทข้อความและสร้างโมเดลในการกำหนดความรู้สึกของข้อความโดยใช้เทคนิคการเรียนรู้ด้วยเครื่อง ซึ่งแบ่งข้อความออกเป็นสามกลุ่ม คือ ข้อความที่แสดงความรู้สึกด้านบวกข้อความที่แสดงความรู้สึกด้านลบ และข้อความที่ไม่แสดงความรู้สึกหรือเป็นกลาง เพื่อสามารถนำไปใช้ในการวิเคราะห์ความรู้สึกจากข้อความได้

การเรียนรู้ด้วยเครื่อง คือ การให้ระบบหรือโปรแกรมมีการเรียนรู้และแก้ปัญหาหรือตัดสินใจด้วยตัวระบบเองโดยใช้ข้อมูลเป็นตัวเรียนรู้ ซึ่งส่วนใหญ่ต้องใช้ข้อมูลจำนวนมากในการสอนให้ระบบหรือโปรแกรมตัดสินใจได้อย่างถูกต้องและสร้างโมเดลในการแก้ปัญหา จึงต่างจากการเขียนโปรแกรมโดยตรงที่จะมีการใส่คำสั่งการทำงานและใส่ข้อมูลเพื่อให้ได้คำตอบ แต่การเรียนรู้ด้วยเครื่องจะใช้การใส่ข้อมูลสอนและระบุคำตอบเพื่อสร้างแบบจำลองหรือโมเดลในการหาคำตอบ ซึ่งมีหลากหลายเทคนิค หนึ่งในนั้นคือการเรียนรู้เชิงลึก (deep learning)

การเรียนรู้เชิงลึกเป็นกระบวนการเลียนแบบระบบเซลล์ประสาทในสมองของมนุษย์ (Neural Network) โดยเซลล์ประสาทแต่ละตัวจะเชื่อมต่อกันด้วยเส้นประสาท สามารถจำลองเป็นโครงข่ายประสาทเทียม (Artificial Neural Network) แบ่งเป็น 3 ส่วน 1. ส่วนเซลล์ประสาทที่รับข้อมูลเข้า (input layer) 2. ส่วนระบบประสาทประมวลผล (hidden layer) 3. ส่วนเซลล์ประสาทที่ส่งผลลัพธ์ของข้อมูลหลังประมวลผล (output layer) ซึ่งแต่ละส่วนมีการเชื่อมกันด้วยเส้นประสาทเทียมและมีการถ่วงน้ำหนัก

(weight) ในการจำลองโครงข่ายประสาทเทียมสามารถทำได้หลายรูปแบบ เช่น โครงข่ายประสาทเทียมแบบสังวัตนาการ (Convolutional Neural Networks) ซึ่งมีความสามารถในการจำแนกข้อความในรูปแบบตัวอักษร [1] และโครงข่ายประสาทแบบ LSTM (Long Short-Term Memory) มีความสามารถในการรองรับข้อมูลแบบมีลำดับเวลา (time series data) ซึ่งข้อมูลแบบข้อความประกอบด้วยลำดับของคำที่ต่อกันทำให้เกิดเป็นประโยค (sentence) ประโยคย่อย (clause) หรือวลี (phrase) จึงสามารถใช้ในการจำแนกข้อความในรูปแบบตัวอักษรได้ [2]

โครงงานนี้จึงจะจัดทำเพื่อสร้างโมเดลโดยด้วยเทคนิคการเรียนรู้เชิงลึกของการเรียนรู้ด้วยเครื่อง ซึ่งมีการสร้างโครงข่ายประสาทเทียมในรูปแบบสังวัตนาการหรือ LSTM เพื่อใช้ในการจำแนกข้อความภาษาไทย โดยแบ่งเป็นกลุ่มของข้อความที่แสดงความรู้สึกด้านบวกและกลุ่มของข้อความที่แสดงความรู้สึกด้านลบ และกลุ่มของข้อความที่ไม่แสดงความรู้สึกหรือเป็นกลาง

1.2 วัตถุประสงค์ของโครงงาน

1. เพื่อศึกษาวิธีการวิเคราะห์และจำแนกความรู้สึกของข้อความภาษาไทย
2. เพื่อสร้างโมเดลในการจำแนกความรู้สึกของข้อความภาษาไทย

1.3 ขอบเขตของโครงงาน

1. แหล่งข้อมูลจากรีวิวในกลุ่มโรงแรม ร้านอาหาร สถานที่ท่องเที่ยว และสายการบิน จากเว็บไซต์ tripadvisor [3] อย่างน้อย 1,000 รีวิว
2. ครอบคลุมเฉพาะข้อความหรือคำภาษาไทยที่สะกดถูกต้องตามไวยกรณ์ในภาษาไทยเท่านั้น
3. ผลลัพธ์การแบ่งกลุ่มข้อความที่ได้จากโมเดลมีสามรูปแบบคือ ข้อความที่แสดงความรู้สึกด้านบวก ข้อความที่แสดงความรู้สึกด้านลบ และไม่มีความรู้สึกหรือเป็นกลาง
4. การพัฒนาโมเดลจะใช้ภาษาไพทอน 3 (Python 3)

1.4 ขั้นตอนการดำเนินงาน

1. ค้นหาและศึกษาบทความรวมถึงองค์ความรู้ที่เกี่ยวข้องกับโครงงาน
2. ศึกษาเครื่องมือ โปรแกรมและเทคนิคที่ใช้ในโครงงาน
3. กำหนดขอบเขตของโครงงานและขั้นตอนดำเนินงาน
4. รวบรวมข้อความภาษาไทย
5. วิเคราะห์และออกแบบวิธีการที่ใช้ในการวิเคราะห์และจำแนกข้อความ
6. พัฒนาโมเดลจำแนกข้อความภาษาไทย
7. ตรวจสอบความถูกต้องของโมเดลที่พัฒนาขึ้น
8. สรุปผลการดำเนินการ ข้อเสนอแนะและจัดทำเอกสาร

ตารางเวลาการดำเนินงาน

ขั้นตอนการดำเนินงาน	ปี 2561				ปี 2562		
	ก.ย.	ต.ค.	พ.ย.	ธ.ค.	ม.ค.	ก.พ.	มี.ค.
1.ค้นหาและศึกษาบทความรวมถึงเนื้อหาที่ เกี่ยวข้องกับโครงงาน							
2.ศึกษาเครื่องมือ โปรแกรมและเทคนิคที่ ใช้ในโครงงาน							
3. กำหนดขอบเขตของโครงงานและ ขั้นตอนดำเนินงาน							
4.รวบรวมข้อความภาษาไทย							
5.วิเคราะห์และออกแบบวิธีการที่ใช้ในการวิเคราะห์และจำแนกข้อความ							
6.พัฒนาโมเดลจำแนกข้อความภาษาไทย							
7.ตรวจสอบความถูกต้องของโมเดลที่ พัฒนาขึ้น							
8.สรุปผลการดำเนินการ ข้อเสนอแนะและ จัดทำเอกสาร							

1.5 ประโยชน์ที่ได้รับ

1. ประโยชน์ต่อผู้จัดทำโครงงาน
   - ได้ความรู้และความเข้าใจเกี่ยวกับการทำงานของการเรียนรู้ด้วยเครื่องในการวิเคราะห์ความรู้สึกจากข้อความและการจัดการข้อมูลที่อยู่ในรูปแบบตัวอักษร
   - มีความเข้าใจและมีทักษะในการใช้ภาษาไพทอน (python) และเครื่องมือต่าง ๆ ที่ใช้ในการพัฒนา
2. ประโยชน์ต่อผู้นำไปใช้งาน
   - มีโมเดลที่มีความแม่นยำในการวิเคราะห์ความรู้สึกจากข้อความภาษาไทย
   - ช่วยอำนวยความสะดวกในการวิเคราะห์และสรุปผลข้อมูลเพื่อใช้ในการตัดสินใจได้

1.6 โครงสร้างของรายงาน

บทที่ 2 จะกล่าวถึงความรู้พื้นฐานที่เกี่ยวข้อง

บทที่ 3 จะกล่าวถึงการรวบรวมและวิเคราะห์ข้อมูล

บทที่ 4 จะกล่าวถึงการพัฒนาโมเดล

บทที่ 5 จะกล่าวถึงผลการทดสอบโมเดล

บทที่ 6 จะกล่าวถึงข้อสรุปและข้อเสนอแนะ

บทที่ 2
ความรู้พื้นฐานที่เกี่ยวข้อง

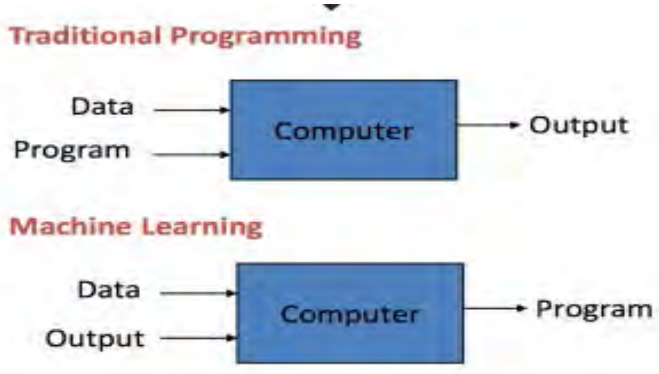
ในบทนี้จะกล่าวถึงความรู้เบื้องต้นและทฤษฎีที่นำมาประยุกต์ใช้กับการพัฒนาโมเดลจำแนกความรู้สึกของข้อความภาษาไทย ดังรายละเอียดต่อไปนี้

2.1 คำที่แสดงความรู้สึกในภาษาไทย

ในทุกภาษาก็มีคำที่แสดงอารมณ์ ความรู้สึกด้านบวก และด้านลบ อย่างในภาษาไทยก็มีคำเหล่านั้น เช่น กลัว ทรมาน โกรธ รำคาญ ยุ่งยากลำบาก เศร้า ไร้เหตุผล ไม่ชอบ ความไม่พอใจ รบกวน สดชื่น เอาใจใส่ เชื่อมั่น มีความสุข สนุกสนาน และร่าเริง ซึ่งคำเหล่านี้สามารถแสดงความรู้สึกได้ว่า ผู้ใช้มีความรู้สึกด้านบวกต่อสิ่งนั้น หรือมีความรู้สึกด้านลบอยู่ โดยความรู้สึกด้านบวกนั้นก็คือ อารมณ์หรือความรู้สึกที่ผู้ใช้รู้สึกพึงพอใจ หรือเห็นด้วยต่อสิ่งนั้น ซึ่งจะต่างจากความรู้สึกด้านลบที่ ผู้ใช้มีอารมณ์หรือความรู้สึกไม่พอใจหรือไม่เห็นด้วยต่อสิ่งนั้น ๆ นั่นเอง

2.2 การเรียนรู้ด้วยเครื่อง

การเรียนรู้ด้วยเครื่อง คือ การทำให้ระบบหรือโปรแกรมมีการเรียนรู้และแก้ปัญหาหรือตัดสินใจด้วยตัวระบบเองโดยใช้ข้อมูลเป็นตัวเรียนรู้ ซึ่งส่วนใหญ่ต้องใช้ข้อมูลจำนวนมากในการสอนให้ระบบหรือโปรแกรมตัดสินใจได้อย่างถูกต้องและสร้างโมเดลในการแก้ปัญหา จึงต่างจากการเขียนโปรแกรมโดยตรงที่จะมีการใส่คำสั่งการทำงานและใส่ข้อมูลเพื่อให้ได้คำตอบ แต่การเรียนรู้ด้วยเครื่องจะใช้การใส่ข้อมูลสอนและระบุคำตอบเพื่อสร้างโมเดลหรือโมเดลในการหาคำตอบ [4] ซึ่งมีหลากหลายเทคนิค หนึ่งในนั้นคือ การเรียนรู้เชิงลึก (deep learning) โดยรูปที่ 2.1 แสดงการเปรียบเทียบการเขียนโปรแกรมโดยตรงกับการเขียนโปรแกรมแบบการเรียนรู้ด้วยเครื่อง

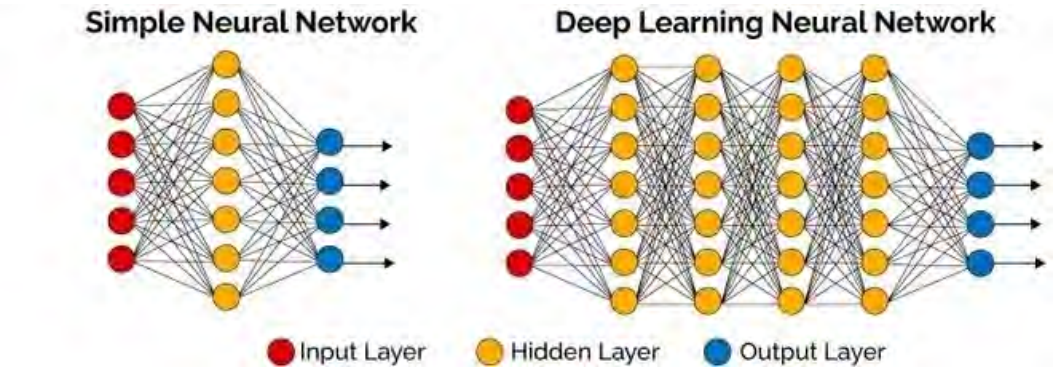


ที่มา: https://blog.finnomena.com/machine-learning-คืออะไร-fa8bf6663c07

รูปที่ 2.1 การเปรียบเทียบการเขียนโปรแกรมโดยตรงกับการเขียนโปรแกรมแบบการเรียนรู้ด้วยเครื่อง

2.3 การเรียนรู้เชิงลึก

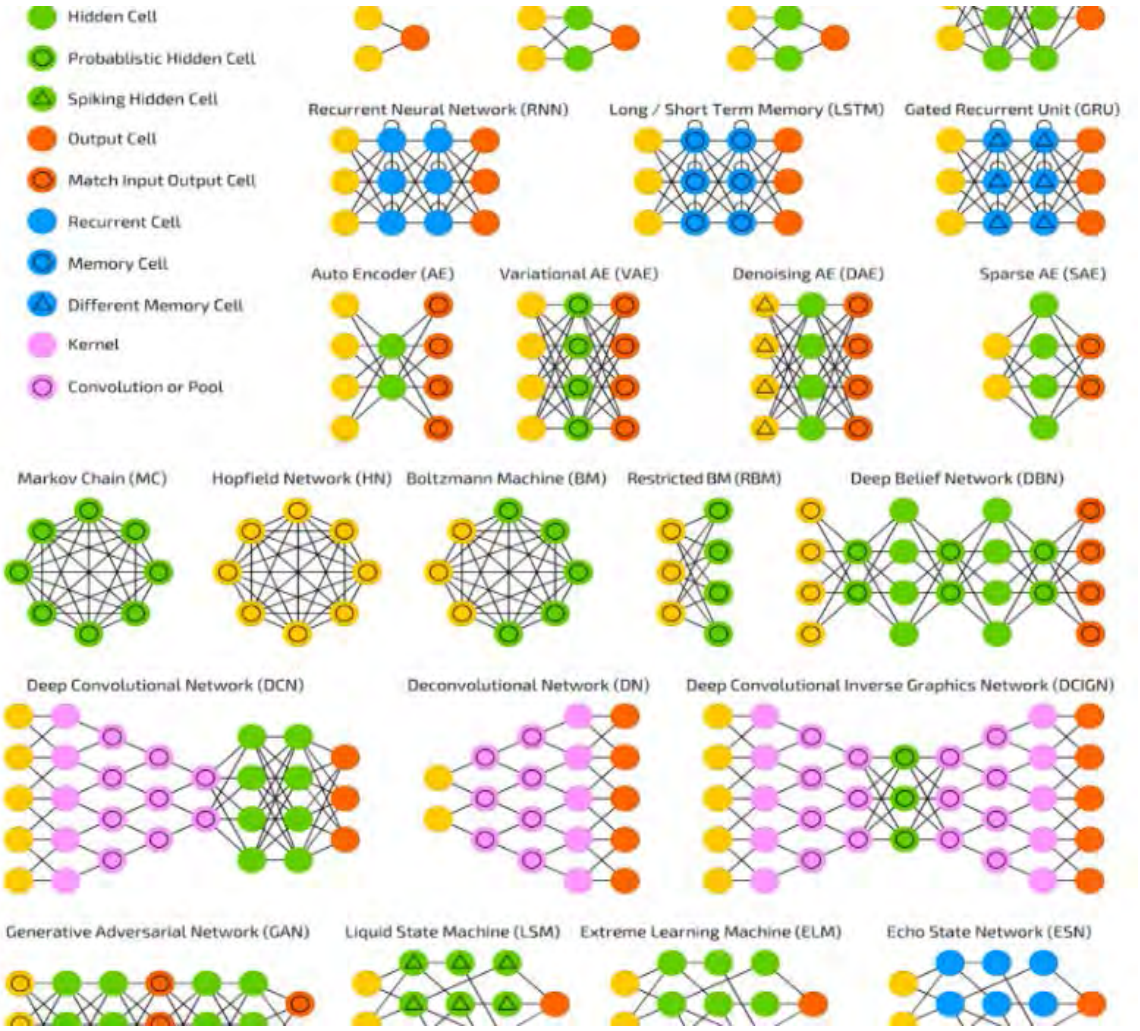
การเรียนรู้เชิงลึกเป็นกระบวนการเลียนแบบระบบเซลล์ประสาทในสมองของมนุษย์ หรือที่เรียกว่าโครงข่ายประสาท (Neural Network) โดยเซลล์ประสาทแต่ละตัวจะเชื่อมต่อกันด้วยเส้นประสาท ดังนั้นการเรียนรู้เชิงลึกจึงจำลองเป็นโครงข่ายประสาทเทียม (Artificial Neural Network) แบ่งเป็น 3 ส่วน คือ 1. ส่วนเซลล์ประสาทที่รับข้อมูลเข้า (input layer) 2. ส่วนระบบประสาทประมวลผล (hidden layer) 3. ส่วนเซลล์ประสาทที่ส่งผลลัพธ์ของข้อมูลหลังประมวลผล (output layer) ซึ่งแต่ละส่วนมีการเชื่อมกันด้วยเส้นประสาทเทียมและมีการถ่วงน้ำหนัก (weight) โดยจะมีการต่อกันของชั้นส่วนระบบประสาทประมวลผลหลาย ๆ ชั้น [5] ดังรูปที่ 2.2 ซึ่งมีอยู่หลายประเภทดังรูปที่ 2.3



ที่มา: https://blog.appliedai.com/how-neural-networks-work/

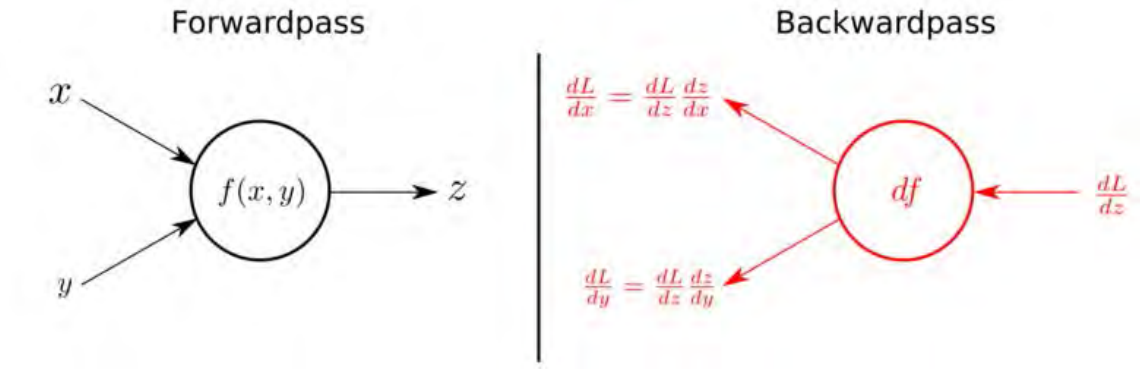
รูปที่ 2.2 ตัวอย่างโครงข่ายการเรียนรู้เชิงลึก

ซึ่งในแต่ละชั้นจะเห็นว่ามีโหนดอยู่ด้านใน โดยแต่ละโหนดจะเป็นตัวคำนวณค่าต่าง ๆ ตามฟังก์ชันการกระตุ้น (activation function) ที่ถูกกำหนดไว้และตัวโหนดจะถูกเชื่อมด้วยเส้นน้ำหนักซึ่งเส้นน้ำหนักจะเป็นตัวควบคุมการไหลของข้อมูลระหว่างโหนดที่เชื่อมกัน โดยที่การเรียนรู้ของโครงข่ายประสาทเทียมเกิดจากการปรับเส้นน้ำหนักให้เหมาะสมกับข้อมูลที่ถูกส่งผ่านเข้ามาและผลลัพธ์ที่ส่งออกไป [6] โดยการปรับเส้นน้ำหนักจะมีการปรับตามการสูญเสีย (loss) ของแต่ละโหนดที่เกิดจากการเทียบอนุพันธ์ของค่าการสูญเสีย (dL) กับอนุพันธ์ของผลลัพธ์ของโหนดตัวสุดท้าย (dz) เมื่อ L คือ ค่าการสูญเสียที่เกิดขึ้นและ z คือผลลัพธ์ที่เกิดขึ้นจากโหนดตัวสุดท้าย และสามารถดูได้ตามตัวอย่างการคำนวณในรูปที่ 2.4 ดังนั้นจึงใช้ค่า $\frac{dL}{dz}$ ในการปรับค่าเส้นน้ำหนักที่ได้ผลลัพธ์การคำนวณออกมาจากโหนดนั้น



ที่มา: https://towardsdatascience.com/the-mostly-complete-chart-of-neural-networks-explained-3fb6f2367464

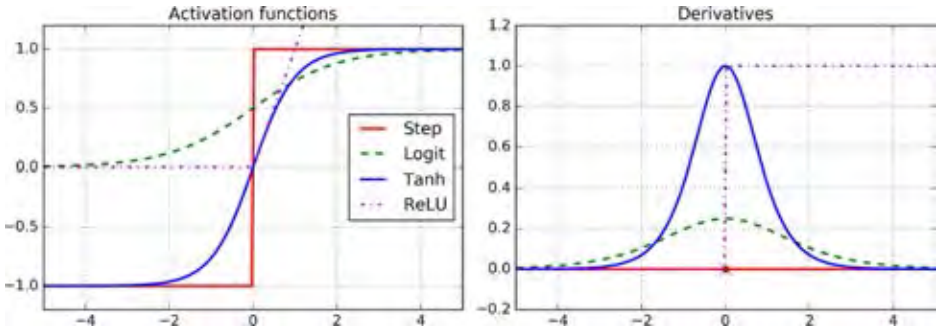
รูปที่ 2.3 ประเภทการเรียนรู้เชิงลึก



ที่มา : https://medium.com/mmp-li/deep-learning-แบบฉบับคนสามัญชน-ep-1-neural-network-history-f7789236a9a3

รูปที่ 2.4 ตัวอย่างการคิดค่าการสูญเสียของแต่ละโหนด

จะเห็นได้ว่าการคิดคำนวณค่าการสูญเสียของแต่ละเส้นจะใช้การหาอนุพันธ์ และจะมีการใช้ฟังก์ชันการกระตุ้นที่ไม่เป็นสมการเส้นตรงเพื่อให้หาผลลัพธ์ที่มีความซับซ้อนได้ดีขึ้น โดยมีรูปแสดงการหาอนุพันธ์ของฟังก์ชันการกระตุ้นดังรูปที่ 2.5 และเมื่อโครงข่ายประสาทมีความซับซ้อนการปรับแต่ละเส้นหรือคิดคำนวณจึงเป็นเรื่องยากจึงเกิดอัลกอริทึมการเพิ่มประสิทธิภาพ (Optimization Algorithms) ช่วยในการคิดคำนวณและปรับเส้นน้ำหนักแต่ละเส้นในโครงข่ายประสาทเทียม โดยจะเรียกตัวคิดคำนวณการสูญเสียและปรับเส้นน้ำหนักว่า ออพติไมเซอร์ (optimizer)

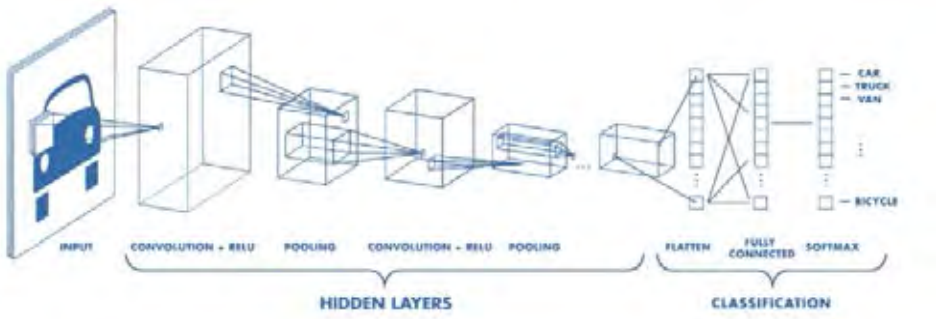


ที่มา : https://www.oreilly.com/library/view/neural-networks-and/9781492037354/ch01.html

รูปที่ 2.5 ตัวอย่างฟังก์ชันการกระตุ้นและอนุพันธ์ของฟังก์ชันการกระตุ้น

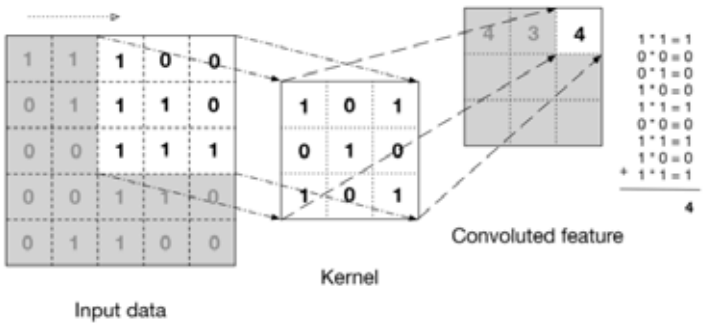
2.4 โครงข่ายประสาทเทียมแบบสังวัตนาการ (Convolutional Neural Networks: CNN)

CNN คือ โครงข่ายประสาทเทียมประเภทหนึ่งซึ่งนิยมใช้กับข้อมูลที่เป็นรูปภาพ มีความสามารถแยกคุณลักษณะ (feature) ของข้อมูลออกมาเป็นลักษณะย่อย ๆ ได้ดี โดยใช้การคำนวณเปรียบเทียบตัวกรอง (filter) และเคอร์เนล (kernel) ที่ช่วยดึงคุณลักษณะที่ใช้ในการเก็บรายละเอียดคุณลักษณะของวัตถุ โดยปกติตัวกรองและเคอร์เนลอันหนึ่งจะดึงคุณลักษณะที่สนใจออกมาได้หนึ่งอย่าง จึงจำเป็นต้องใช้ตัวกรองหลายตัวกรองทำงานร่วมกัน เพื่อหาคุณลักษณะทางพื้นที่หลายอย่างประกอบกัน [7] รูปที่ 2.6 และ 2.7 แสดงการทำงานและความสัมพันธ์ของ CNN ตามลำดับ และแสดงตัวอย่างของ CNN ที่ทำงานกับข้อมูลคำในรูปที่ 2.8



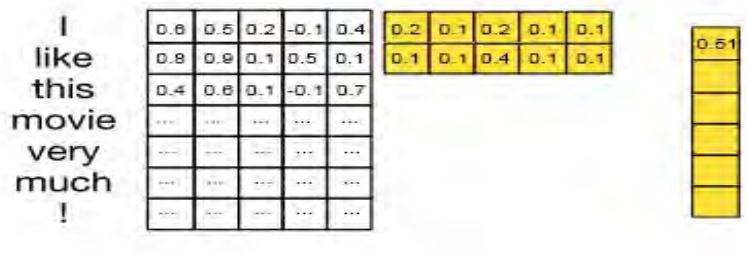
ที่มา: https://medium.com/@natthawatphongchit/มาลองดูวิธีการคิดของ-cnn-กัน-e3f5d73eebaa

รูปที่ 2.6 การทำงานของโครงข่ายประสาทเทียมแบบ CNN



ที่มา: http://www.davidsbatista.net/blog/2018/03/31/ SentenceClassificationConvNets

รูปที่ 2.7 ความสัมพันธ์ของ CNN

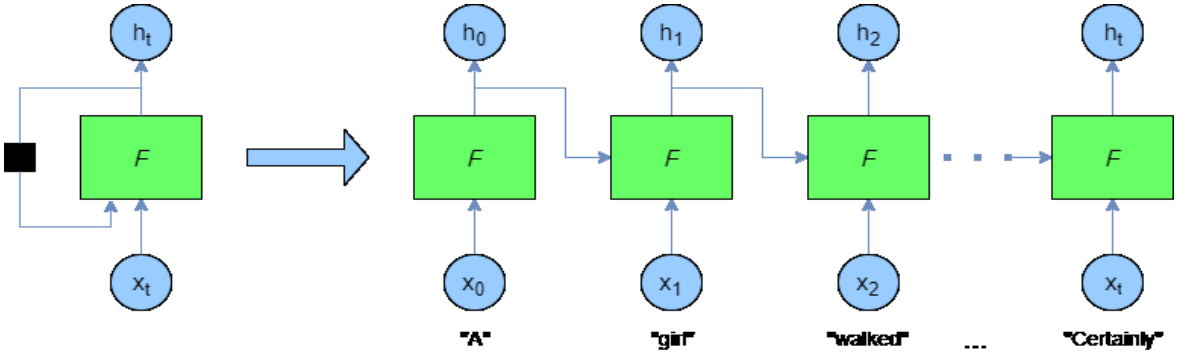


ที่มา: http://www.joshuakim.io/understanding-how-convolutional-neural-network-cnn-perform-text-classification-with-word-embeddings/

รูปที่ 2.8 การทำงานของ CNN กับคำ

2.5 โครงข่ายประสาทแบบ LSTM (Long Short-Term Memory: LSTM)

LSTM คือ โครงข่ายประสาทเทียมแบบหนึ่งของโครงข่ายประสาทเทียมแบบวนซ้ำ (Recurrent Neural Network) โดยที่สร้างขึ้นมาเพื่อจำลองรูปแบบความจำของคน (memory) ที่มีความจุของความทรงจำอยู่จำกัด เมื่อมีเหตุการณ์ใหม่ ๆ เข้ามาในความทรงจำ สมองจะเลือกที่จะรับหรือไม่รับเหตุการณ์ใหม่เข้ามาในความทรงจำ ดังนั้น LSTM จึงมีความสามารถในการรองรับข้อมูลแบบมีลำดับเวลา [8] โดยแสดงตัวอย่างการทำงานของ LSTM ตามรูปที่ 2.9

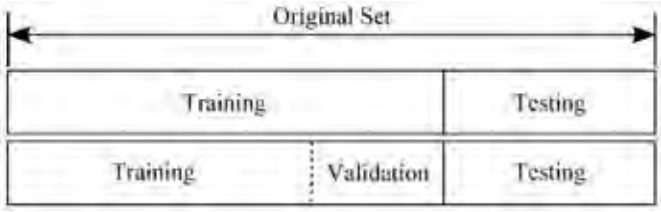


ที่มา: https://adventuresinmachinelearning.com/recurrent-neural-networks-lstm-tutorial-tensorflow/

รูปที่ 2.9 ตัวอย่างการทำงานของ LSTM

2.6 การประเมินประสิทธิภาพของโมเดลการเรียนรู้เชิงลึก

การเรียนรู้เชิงลึกมักใช้กับปัญหาที่มีชุดข้อมูลขนาดใหญ่ ดังนั้นต้องมีชุดทดสอบที่มีประสิทธิภาพซึ่งช่วยให้สามารถประเมินประสิทธิภาพการทำงานของโมเดลในข้อมูลที่มองไม่เห็นและเปรียบเทียบประสิทธิภาพกับการกำหนดค่าอื่น ๆ ได้อย่างน่าเชื่อถือ โดยทั่วไปแล้วจะใช้การแยกข้อมูลอย่างง่ายโดยจะแบ่งข้อมูลทั้งหมดเป็นชุดข้อมูลสอน (training datasets) และชุดข้อมูลทดสอบ (test datasets) หรือชุดข้อมูลสอนและชุดข้อมูลตรวจสอบความถูกต้อง (validation datasets) [9] เพื่อนำไปใช้คำนวณหาความถูกต้อง (accuracy) ของโมเดลต่อไป รูปที่ 2.10 แสดงตัวอย่างการแบ่งข้อมูลเพื่อประเมินประสิทธิภาพโมเดล



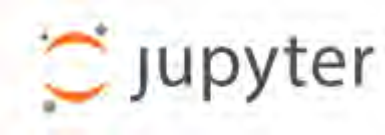
ที่มา: https://www.intechopen.com/books/advances-in-data-mining-knowledge-discovery-and-applications/selecting-representative-data-sets

รูปที่ 2.10 ตัวอย่างการแบ่งข้อมูลเพื่อประเมินประสิทธิภาพโมเดล

2.7 เครื่องมือที่เกี่ยวข้องในการพัฒนาโมเดล

2.7.1 จูปิเตอร์ โน้ตบุ๊ก (Jupyter Notebook)

จูปิเตอร์โน้ตบุ๊กเป็นเครื่องมือช่วยในการเขียนภาษาไพทอน ที่สามารถหาข้อผิดพลาดของโปรแกรมและช่วยให้การพัฒนาโปรแกรมทำงานได้ราบรื่นขึ้น โดยมีสัญลักษณ์ดังรูปที่ 2.11



ที่มา: https://jupyter.org/

รูปที่ 2.11 สัญลักษณ์ของ Jupyter notebook

2.7.2 Keras library

Keras library คือ ไลบรารีที่ใช้ในการสร้างโครงข่ายประสาทเทียมของตัวโมเดลและมีการเลือกใช้คำสั่งต่าง ๆ เช่น model = Sequential() ใช้ในการสร้างโมเดลและ model.add(Dense(units=64, activation='relu', input_dim=100)) จะใช้ในการเพิ่มจำนวนชั้นของโครงข่ายประสาทเทียมในโมเดล และยังสามารถเรียกใช้งานคำสั่งต่าง ๆ เกี่ยวกับการเรียนรู้เชิงลึกได้อีกมาก โดยมีสัญลักษณ์ดังรูปที่ 2.12

ที่มา: https://keras.io/

รูปที่ 2.12 สัญลักษณ์ของ Keras library

2.7.3 Deep cut library

Deep cut library คือ ไลบรารีที่ใช้ในการตัดข้อความเป็นคำในภาษาไทยโดยสร้างจากโครงข่ายประสาทแบบ CNN ที่ใช้ข้อมูลจากกลุ่มคำจากคลังข้อความภาษาไทยของศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติ (NECTEC) ซึ่งประกอบด้วย 4 กลุ่ม คือ บทความ ข่าว นิยาย และสารานุกรม โดยใช้กลุ่มคำทั้งหมดในการตัดสินว่าตัวอักษรแต่ละตัวเป็นตัวอักษรขึ้นต้นของคำใหม่หรือไม่ โดยสามารถรับข้อความที่เป็นตัวอักษรเพื่อตัดคำออกมาเป็นอาเรย์ของคำได้ โดยอย่างการใช้งาน คือ import deepcut เป็นคำสั่งที่ใช้ในการเรียกใช้งานไลบรารี deepcut และคำสั่ง deepcut.tokenize('ตัดคำได้ดีมาก') ใช้ในการสั่งตัดคำ ซึ่งผลลัพธ์ที่จะได้ออกมา คือ ['ตัดคำ','ได้','ดี','มาก']

บทที่ 3
การรวบรวมและวิเคราะห์ข้อมูล

ในบทนี้จะกล่าวถึง ส่วนการรวบรวมข้อมูลที่ใช้ในการพัฒนาโมเดลและส่วนการวิเคราะห์ข้อมูล ซึ่งประกอบด้วยการเตรียมข้อมูลและการทดลองเพื่อเลือกการพัฒนาโมเดลและประสิทธิภาพของโมเดลแบบต่าง ๆ ที่ได้ทดลอง ดังรายละเอียดต่อไปนี้

3.1 การรวบรวมข้อมูลและรูปแบบการเก็บข้อมูล

ผู้พัฒนาได้เก็บรวบรวมข้อความเพื่อจะนำไปใช้พัฒนาโมเดลจำแนกความรู้สึกจากเว็บไซต์ www.tripadvisor.com ซึ่งเป็นเว็บไซต์ที่ช่วยอำนวยความสะดวกในการหาที่พัก ร้านอาหาร หรือสถานที่ท่องเที่ยว และเที่ยวบินต่าง ๆ โดยจะเปิดให้ผู้ใช้บริการสามารถรีวิวถึงสถานที่หรือการบริการที่ได้รับได้อย่างอิสระ โดยจะรวบรวมข้อมูลมาในรูปแบบของข้อความจากรีวิวในหมวดต่าง ๆ 4 หมวด คือ โรงแรม ร้านอาหาร สายการบิน และสถานที่ท่องเที่ยว รวมทั้งหมด 12,596 ข้อความ โดยข้อความที่เก็บรวบรวมได้จะถูกแบ่งประเภทออกเป็น 3 กลุ่ม คือ

1.ข้อความที่แสดงความรู้สึกด้านบวก แทนด้วยเลข 0

2.ข้อความที่แสดงความรู้สึกด้านลบ แทนด้วยเลข 1

3.ข้อความที่ไม่แสดงความรู้สึกหรือเป็นกลาง แทนด้วยเลข 2

รายละเอียดตามตารางที่ 3.1

ตารางที่ 3.1 ข้อมูลที่รวบรวมได้เป็นหมวดต่าง ๆ

หมวดหมู่	ความรู้สึกด้านบวก	ความรู้สึกด้านลบ	ไม่แสดงความรู้สึกหรือเป็นกลาง	รวม
โรงแรม	1,544	1,183	713	3,440
ร้านอาหาร	1,401	713	920	3,034
สถานที่ท่องเที่ยว	1,286	723	1,027	3,036
สายการบิน	1,470	636	980	3,086
รวม	5,701	3,255	3,640	12,596

โครงการนี้พิจารณาข้อความที่ไม่แสดงความรู้สึกกับข้อความที่มีความรู้สึกเป็นกลางไว้ในกลุ่มเดียวกันเนื่องจาก โมเดลที่พัฒนาขึ้นสนใจเฉพาะแพตเทิร์น (pattern) ข้อความด้านบวกและแพตเทิร์นด้านลบดังนั้นข้อความสองประเภทข้างต้นจึงจัดเป็นกลุ่มเดียวกัน เพื่อง่ายต่อการพัฒนาโมเดล อีกทั้งผู้พัฒนาเห็นว่าข้อความทั้งสองประเภทข้างต้นเป็นข้อความที่ไม่มีผลในการพัฒนาคุณภาพสินค้าและบริการ โดยมีตัวอย่างของข้อมูลที่รวบรวมได้ในหมวดร้านอาหารดังรูปที่ 3.1

```
อาหารซีฟู้ดสดมากเหมือนขึ้นมาจากทะเลใหม่ ไม่ต้องไปกินไกลถึงทะเล,0
ราคาไม่แรงมากพอรับได้ เหมาะสมกับราคาค่ะ เดินทางสะดวก,0
ร้านตกแต่งน่ารัก บรรยากาศสงบ,0
บรรยากาศร้านดีมาก อาหารก็ตกแต่งสวยและอร่อย ,0
 บรรยากาศตกแต่งดูดีมากเน้นเป็นสีขาว,0
แต่รสชาดก็ถือว่าใช้ได้ อาหารเสริฟค่อนข้างเร็วทีเดียวไม่ต้องเสียเวลานาน,0
```

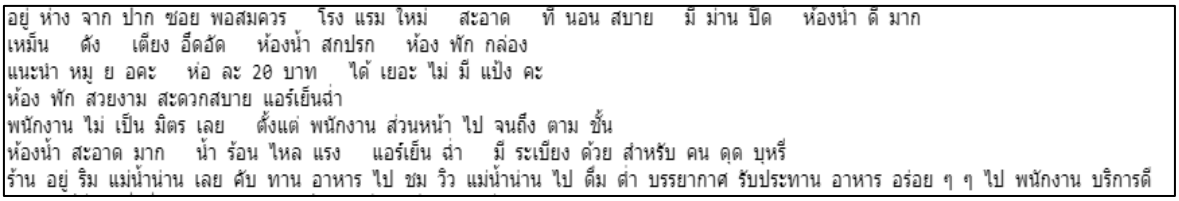
รูปที่ 3.1 ตัวอย่างข้อมูลหมวดร้านอาหารที่ให้ความรู้สึกด้านบวก

3.2 การวิเคราะห์ข้อมูล

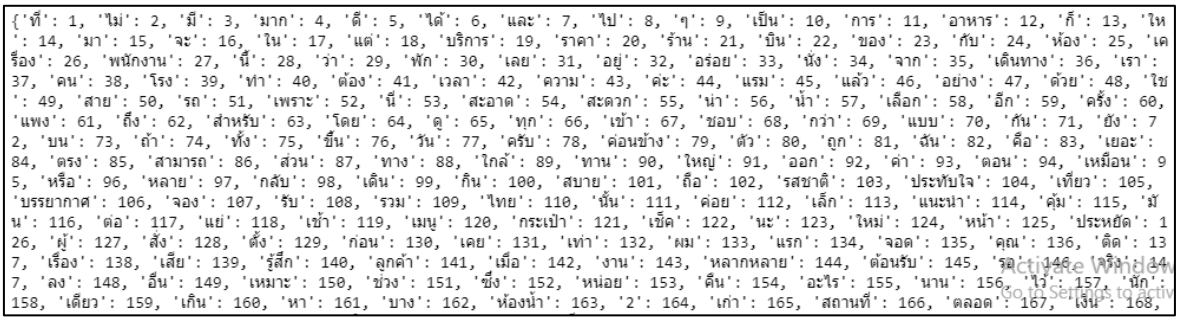
จากข้อมูลที่ได้จากการรวบรวมข้อมูลจะนำมาวิเคราะห์โดยเริ่มจากการเตรียมข้อมูล (data preparation) เพื่อให้พร้อมที่จะนำไปประมวลผลรายละเอียดดังนี้

3.2.1 การเตรียมข้อมูล

ในส่วนของการเตรียมข้อมูลนี้ ผู้พัฒนาได้มีการดึงข้อมูลมาอย่างละ 1,800 ข้อความจากแต่ละหมวด (โรงแรม ร้านอาหาร สายการบิน และสถานที่ท่องเที่ยว) โดยแบ่งเป็น ข้อความแสดงความรู้สึกด้านบวก ข้อความแสดงความรู้สึกด้านลบ และข้อความไม่แสดงความรู้สึกหรือเป็นกลาง ประเภทละ 600 ข้อความ รวมเป็นข้อมูลทั้งหมดเป็น 7,200 ข้อความ เพื่อขจัดปัญหาการไม่สมดุลของข้อมูลที่ใช้ในการพัฒนาโมเดล และได้แบ่งเป็นข้อมูลสำหรับสอน 5,400 ข้อความ และข้อมูลสำหรับตรวจสอบความถูกต้อง 1,800 ข้อความ ซึ่งในการแบ่งเกิดจากการสุ่มในแต่ละหมวดที่มีจำนวนเท่า ๆ กัน หลังจากนั้นได้นำข้อมูลที่ได้ไปตัดคำด้วยไลบรารี deep cut โดยข้อความที่ผ่านการตัดคำจะอยู่ในลักษณะของอาร์เรย์ (array) ของคำโดยมีตัวอย่างแสดงตามรูปที่ 3.2 และนำไปจับคู่กับโทเคน (token) เพื่อแปลงเป็นตัวเลขที่มีความเฉพาะต่อคำนั้น ๆ โดยจะมีจำนวนโทเคนทั้งหมด 7,240 โทเคน หรือก็คือ 7,240 คำที่ไม่ซ้ำกัน โดยมีตัวอย่างโทเคนตามรูปที่ 3.3 และเนื่องจากข้อความแต่ละข้อความมีจำนวนคำไม่เท่ากัน ดังนั้นข้อความที่มีคำน้อยกว่าจำนวนโหนดข้อมูลนำเข้า (input node) ของโมเดลจะถูกเติมเลข 0 ด้านหน้าให้มีจำนวนเท่ากับจำนวนข้อมูลที่นำเข้าสู่โมเดลที่สร้างโดยวิธีการเรียนรู้เชิงลึก



รูปที่ 3.2 ตัวอย่างข้อความที่ผ่านการตัดคำด้วยไลบรารี deep cut



รูปที่ 3.3 โทเคนที่ใช้ในการแทนคำ

ในส่วนของผลลัพธ์ของข้อความจะมีการปรับเปลี่ยนค่าจากเลข 0 1 และ 2 ให้อยู่ในรูปแบบของอาร์เรย์ 3 ช่องที่มีค่าเป็น 0 หรือ 1 โดยมีรูปแบบดังนี้

จากผลลัพธ์ที่ให้ความรู้สึกด้านบวกมีค่าที่แทนด้วย 0 เป็น [1,0,0]

จากผลลัพธ์ที่ให้ความรู้สึกด้านลบมีค่าที่แทนด้วย 1 เป็น [0,1,0]

จากผลลัพธ์ที่เป็นกลางหรือไม่ให้ความรู้สึกมีค่าที่แทนด้วย 2 เป็น [0,0,1]

3.2.2 ผลการทดลองของโมเดลชนิดต่าง ๆ

ในการทดลองจะวัดประสิทธิภาพของการจำแนกข้อมูล ด้วยความถูกต้องของการทำนายด้วยข้อมูลชุดตรวจสอบ (validation accuracy) ซึ่งข้อมูลชุดตรวจสอบนี้เปรียบเสมือนกับข้อมูลที่โมเดลยังไม่เคยเรียนรู้มาก่อน ทำให้สามารถนำมาใช้ในการประเมินประสิทธิภาพโมเดลได้ โดยยิ่งมีเปอร์เซ็นต์ของความถูกต้องมากยิ่งมีประสิทธิภาพมาก การแสดงผลการทดสอบจะอยู่ในรูปแบบของ “เปอร์เซ็นต์ความถูกต้องของข้อมูลชุดตรวจสอบ/เวลาที่ใช้ในการเรียนรู้(วินาที)” ซึ่งถ้ามีเปอร์เซ็นต์ความถูกต้องใกล้เคียงกัน อันที่ใช้เวลาในการเรียนรู้น้อยกว่าจะถือว่ามีประสิทธิภาพมากกว่า

สำหรับจำนวนโหนดข้อมูลนำเข้า มีข้อแนะนำว่าควรมีมากกว่าหรือเท่ากับจำนวนคำของข้อความที่มากที่สุดที่ใช้ในการสอนโมเดล ซึ่งทางผู้พัฒนาได้กำหนดเป็นจำนวน 100 ตัว เพราะคาดการณ์ว่าจำนวนคำในกลุ่มของข้อความที่พิจารณา เมื่อตัดคำออกมาแล้วจะมีจำนวนคำไม่เกิน 100 คำเรียงต่อกัน และทางผู้พัฒนาได้ทดลองกับโมเดลอย่างง่ายแล้วพบว่า จำนวนโหนดข้อมูลนำเข้าที่มากกว่าจำนวนคำในข้อความ มีผลลัพธ์ไม่แน่นอนว่าจะทำให้ความถูกต้องของโมเดลเพิ่มมากขึ้น แต่ส่งผลให้การเรียนรู้ใช้เวลาเพิ่มมากขึ้น

อย่างแน่นอน ตามตารางที่ 3.2 ดังนั้นหากข้อความมีจำนวนคำที่มากกว่า 100 คำ โมเดลจะตัดออกและใช้เป็นโหนดข้อมูลนำเข้าแค่ 100 คำแรก โดยในการทดลองจะมุ่งเน้นถึงประสิทธิภาพในการสร้างหรือการต่อโมเดลด้วยโครงข่ายประสาทแบบ CNN และโครงข่ายประสาทแบบ LSTM เท่านั้น โดยได้กำหนดการเรียนรู้ 20 รอบเพราะเพียงพอต่อการแสดงให้เห็นถึงความคงที่ของประสิทธิภาพในการเรียนรู้ว่าจะไม่มีประสิทธิภาพในการเรียนรู้ที่ดีหรือลดลงกว่าเดิม

ตารางที่ 3.2 ผลลัพธ์การตรวจสอบจำนวนโหนดข้อมูลนำเข้า

ลักษณะโมเดล	ความถูกต้อง (เปอร์เซ็นต์) /เวลา (วินาที)
CNN(32,1) 100 input	80.68/237
CNN(32,1) 150 input	80.29/278
CNN(32,2) 100 input	81.17/277
CNN(32,2) 150 input	81.31/329

ในขั้นแรกผู้พัฒนาจะทดสอบด้วยการใช้โครงข่ายประสาทแบบเดียวนั่นคือ CNN โดยโครงข่ายประสาทเทียม CNN จะมีค่าที่ส่งผลต่อความถูกต้องที่สามารถปรับเปลี่ยนได้อยู่ 2 ค่า คือ

1.จำนวนตัวกรอง คือ จำนวนคุณลักษณะที่จะต้องการจากข้อมูล

2.ขนาดของเคอร์เนล คือ จำนวนความกว้างของช่องที่จะใช้ในการประมวลผลเพื่อหาคุณลักษณะ

โดยใช้สัญลักษณ์ระบุจำนวนตัวกรองและขนาดของเคอร์เนล คือ CNN(จำนวนตัวกรอง,ขนาดของเคอร์เนล) เช่น CNN(16,3) หมายถึง โครงข่ายประสาทแบบ CNN ที่มีจำนวนตัวกรอง 16 ตัว และมีขนาดของเคอร์เนล 3 ช่อง

สำหรับโครงข่ายประสาทแบบ LSTM จะมีค่าที่ส่งผลต่อความถูกต้อง 1 ค่า คือ จำนวนเซลล์ความจำ โดยจะใช้สัญลักษณ์ระบุจำนวนเซลล์ความจำเป็น LSTM(จำนวนเซลล์ความจำ) เช่น LSTM(1) หมายถึง โครงข่ายประสาทแบบ LSTM ที่มีจำนวนเซลล์ความจำ 1 หน่วย

ทางผู้พัฒนาจึงเริ่มการทดสอบหาจำนวนตัวกรองที่เหมาะสมโดยกำหนดให้ขนาดของเคอร์เนลมีค่าคงที่เท่ากับ 3 เพราะส่วนมากในการใช้งาน CNN ที่ใช้กับรูปภาพจะมีกำหนดให้มีขนาดเคอร์เนลเท่ากับ 3 [10] และกำหนดให้จำนวนตัวกรองเริ่มต้นมีค่าเท่ากับ 16 และเพิ่มเป็นจำนวนสองเท่า เพราะเป็นวิธีการปรับค่าตัวกรองที่นิยมในการทดลองเพื่อหาโมเดลที่เหมาะสมกับข้อมูล เมื่อได้จำนวนตัวกรองที่เหมาะสมแล้ว จึงนำไปทดลองเพื่อหาขนาดของเคอร์เนลที่เหมาะสมอีกครั้งหนึ่ง โดยจะแสดงผลลัพธ์การทดสอบตามจำนวนตัวกรอง ในตารางที่ 3.3

ตารางที่ 3.3 ผลลัพธ์การทดสอบปริมาณตัวกรอง

ลักษณะโมเดล	ความถูกต้อง (เปอร์เซ็นต์) /เวลา (วินาที)
CNN(16,3)	78.95/334
CNN(32,3)	80.57/358
CNN(64,3)	80.45/396
CNN(128,3)	80.50/442

จากผลการทดสอบจำนวน ตัวกรอง พบว่าจำนวน 32 มีค่าความถูกต้องที่ 80.57% ซึ่งมีค่าความถูกต้องใกล้เคียงกับการทดสอบที่จำนวน 64 และ 128 แต่เวลาที่ใช้ในการเรียนรู้นั้นน้อยกว่าทางผู้พัฒนาจึงเลือกปริมาณตัวกรองเท่ากับ 32 ไปใช้ในการทดสอบหาขนาดเคอร์เนลที่เหมาะสมต่อไป โดยแสดงผลลัพธ์การทดสอบขนาดเคอร์เนลในตารางที่ 3.4

ตารางที่ 3.4 ผลลัพธ์การทดสอบขนาดเคอร์เนล

ลักษณะโมเดล	ความถูกต้อง (เปอร์เซ็นต์) /เวลา (วินาที)
CNN(32,1)	80.68/237
CNN(32,2)	81.17/277
CNN(32,3)	80.57/358

จะเห็นได้ว่าขนาดเคอร์เนลเท่ากับ 2 ให้ความถูกต้องที่ดีที่สุด ในขั้นต่อไปผู้พัฒนาจึงเลือกโมเดลชั้นเดียวที่ดีที่สุด คือ CNN(32,2) ไปทำการทดลองเพิ่มจำนวนชั้นของ CNN เป็น 2 ชั้นเพื่อดูผลการทดสอบโดยแสดงตามตารางที่ 3.5

ตารางที่ 3.5 ผลลัพธ์การทดสอบเมื่อเพิ่มจำนวนชั้นของ CNN

ลักษณะโมเดล	ความถูกต้อง (เปอร์เซ็นต์) /เวลา (วินาที)
CNN(32,2)+ CNN(16,2)	81.47/316
CNN(32,2)+ CNN(32,2)	81.69/326
CNN(32,2)+ CNN(64,2)	81.69/302
CNN(32,2)+ CNN(128,2)	81.43/357
CNN(32,2)+ CNN(64,1)	81.11/305
CNN(32,2)+ CNN(64,3)	81.69/336

เมื่อดูจากผลการทดลองจะเห็นว่ามีการทดสอบทั้งการปรับเปลี่ยนปริมาณตัวกรองเพื่อหาตัวที่ดีที่สุด คือ CNN(32,2)+ CNN(64,2) จากนั้นนำไปปรับเปลี่ยนขนาดเคอร์เนลให้มากขึ้นหรือลดลง ผลปรากฏขนาดเคอร์เนลเท่ากับชั้นแรก คือ 2 ให้ผลดีที่สุดจึงได้ข้อสังเกตว่าจำนวนตัวกรองที่ใช้มากขึ้นในชั้นถัดไปอาจทำให้ประสิทธิภาพดีขึ้นแต่การลดขนาดเคอร์เนลในชั้นถัดไปจะทำให้ประสิทธิภาพลดลงอย่างแน่นอน และการเพิ่มขนาดเคอร์เนลในชั้นถัดไปนอกจากจะไม่ช่วยให้ประสิทธิภาพดีขึ้นยังทำให้เวลาที่ใช้ในการเรียนรู้เพิ่มมากขึ้นอีกด้วย จึงนำข้อสังเกตเหล่านี้ไปใช้ในการเพิ่มจำนวนชั้นที่ 3 และ 4 ของ CNN ต่อไปโดยมีผลลัพธ์แสดงดังตารางที่ 3.6 แสดงผลลัพธ์ของ CNN 3 ชั้นและ 4 ชั้น

ตารางที่ 3.6 ผลลัพธ์ของ CNN 3 ชั้นและ 4 ชั้น

ลักษณะโมเดล	ความถูกต้อง (เปอร์เซ็นต์) /เวลา (วินาที)
CNN(32,2)+CNN(64,2)+ CNN(128,2)	82.02/403
CNN(32,2)+CNN(64,2)+CNN(128,2)+ CNN(256,2)	81.85/642

จากผลการทดสอบพบว่าการเพิ่มชั้นของ CNN ให้มีจำนวน 3 ชั้น คือ CNN(32,2+CNN(64,2)+ CNN(128,2) โดยใช้ข้อสังเกตการเพิ่มชั้นของ CNN ในชั้นที่สองส่งผลให้ประสิทธิภาพดีที่สุดในการต่อกันของ CNN และจากผลการทดลองยังแสดงให้เห็นว่าความลึกหรือจำนวนมีผลทำให้ความถูกต้องมีค่าลดลงได้เช่นกัน ทางผู้พัฒนาจึงหยุดการทดสอบเกี่ยวกับ CNN และเลือกให้โมเดล CNN(32,2+CNN(64,2)+ CNN(128,2) เป็นโมเดลที่จะนำไปทดสอบการต่อกับ LSTM ต่อไป

ในลำดับต่อไปทางผู้พัฒนาได้ทำการทดสอบโครงข่ายประสาท LSTM โดยโครงข่ายประสาท LSTM จะมีค่าที่มีผลต่อความถูกต้อง คือ จำนวนความจำ โดยจะเริ่มทำการทดลองหาจำนวนเซลล์ความจำที่เหมาะสมต่อข้อมูลเป็นอันดับแรก โดยผลลัพธ์การทดสอบแสดงดังตารางที่ 3.7

ตารางที่ 3.7 ผลลัพธ์ของการทดสอบจำนวนเซลล์ความจำของโครงข่ายประสาท LSTM

ลักษณะโมเดล	ความถูกต้อง (เปอร์เซ็นต์) /เวลา (วินาที)
LSTM(1)	59.50/619
LSTM(2)	73.92/617
LSTM(3)	69.80/627

เมื่อดูผลการทดสอบแล้วพบว่าโครงข่ายประสาท LSTM ให้มีประสิทธิภาพดีเมื่อมีจำนวนเซลล์ความจำเท่ากับ 2 จึงเลือกไปทำการเพิ่มจำนวนชั้นของ LSTM ในลำดับต่อไป ซึ่งได้ผลลัพธ์การทดสอบตามตารางที่ 3.8

ตารางที่ 3.8 ผลลัพธ์ของการเพิ่มจำนวนชั้นของโครงข่ายประสาท LSTM

ลักษณะโมเดล	ความถูกต้อง (เปอร์เซ็นต์) /เวลา (วินาที)
LSTM(2)+ LSTM(1)	61.99/872
LSTM(2)+ LSTM(2)	69.25/876
LSTM(2)+ LSTM(3)	72.75/917
LSTM(2)+ LSTM(4)	68.37/961

จากผลการทดสอบพบว่าการเพิ่มจำนวนชั้นของโครงข่ายประสาทแบบ LSTM ส่งผลให้ประสิทธิภาพลดลง ทางผู้พัฒนาจึงเลือกใช้ LSTM เพียงชั้นเดียว คือ LSTM(2) ในการทดสอบการต่อกันของ LSTM และ CNN

หลังจากได้ทดสอบการสร้างโมเดลที่ใช้ CNN หรือโครงข่ายประสาท LSTM เพียงอย่างเดียวแล้ว ทางผู้พัฒนาจึงทำการทดสอบโดยนำโครงข่ายประสาทเทียมทั้งสองมาเชื่อมต่อกันโดยเริ่มจากโมเดลที่ให้ผลดีที่สุดในแต่ละชั้น คือ นั่นคือ CNN(32,2) เป็นตัวแทนของ CNN จำนวนชั้นเดียว CNN(32,2)+ CNN(64,2) เป็นตัวแทนของ CNN 2 ชั้น และสุดท้าย คือ CNN(32,2+CNN(64,2)+CNN(128,2) เป็นตัวแทนสำหรับ CNN 3 ชั้น โดยจะนำทั้งสามแบบข้างต้นมาต่อกับ LSTM เพื่อหา LSTM ที่เหมาะสมของแต่ละแบบต่อไป และทดลองสร้างโมเดลใหม่โดยเริ่มชั้นแรก คือ LSTM ต่อด้วย CNN ซึ่งจะเริ่มด้วย LSTM(2) เพื่อต่อกับ CNN โดยผลลัพธ์การทดสอบ CNN ต่อกับ LSTM ได้ผลลัพธ์ตามตารางที่ 3.9 และผลลัพธ์การทดสอบ LSTM ต่อกับ CNN ได้ผลลัพธ์ตามตารางที่ 3.10

ตารางที่ 3.9 ผลลัพธ์ของการต่อกันของ CNN กับโครงข่ายประสาท LSTM

ลักษณะโมเดล	ความถูกต้อง (เปอร์เซ็นต์) /เวลา (วินาที)
CNN(32,2)+LSTM(1)	62.03/538
CNN(32,2)+LSTM(2)	76.05/574
CNN(32,2)+LSTM(3)	77.07/546
CNN(32,2)+LSTM(10)	78.99/553
CNN(32,2)+LSTM(11)	80.01/542
CNN(32,2)+LSTM(12)	78.61/559
CNN(32,2)+LSTM(15)	77.68/567
CNN(32,2)+CNN(64,2)+LSTM(11)	78.70/608
CNN(32,2)+CNN(64,2)+LSTM(13)	78.90/616
CNN(32,2)+CNN(64,2)+LSTM(15)	79.55/689
CNN(32,2)+CNN(64,2)+LSTM(20)	79.28/668
CNN(32,2)+CNN(64,2)+CNN(128,2)+LSTM(1)	32.27/711
CNN(32,2)+CNN(64,2)+CNN(128,2)+LSTM(2)	68.55/722
CNN(32,2)+CNN(64,2)+CNN(128,2)+LSTM(3)	68.69/711
CNN(32,2)+CNN(64,2)+CNN(128,2)+LSTM(5)	78.05/701
CNN(32,2)+CNN(64,2)+CNN(128,2)+LSTM(10)	81.02/752
CNN(32,2)+CNN(64,2)+CNN(128,2)+LSTM(12)	82.20/805
CNN(32,2)+CNN(64,2)+CNN(128,2)+LSTM(13)	80.88/771
CNN(32,2)+CNN(64,2)+CNN(128,2)+LSTM(15)	32.27/718

ตารางที่ 3.10 ผลลัพธ์ของการต่อกันของโครงข่ายประสาท LSTM กับ CNN

ลักษณะโมเดล	ความถูกต้อง (เปอร์เซ็นต์) /เวลา (วินาที)
LSTM(2)+CNN(32,2)+CNN(64,2)+CNN(128,2)	80.01/784
LSTM(2)+CNN(32,2)	79.47/713
LSTM(2)+CNN(64,2)	79.74/641
LSTM(2)+CNN(128,2)	74.36/671
LSTM(2)+CNN(32,1)	75.11/602
LSTM(2)+CNN(32,3)	78.01/672
LSTM(2)+CNN(32,2)+CNN(64,2)	76.39/698

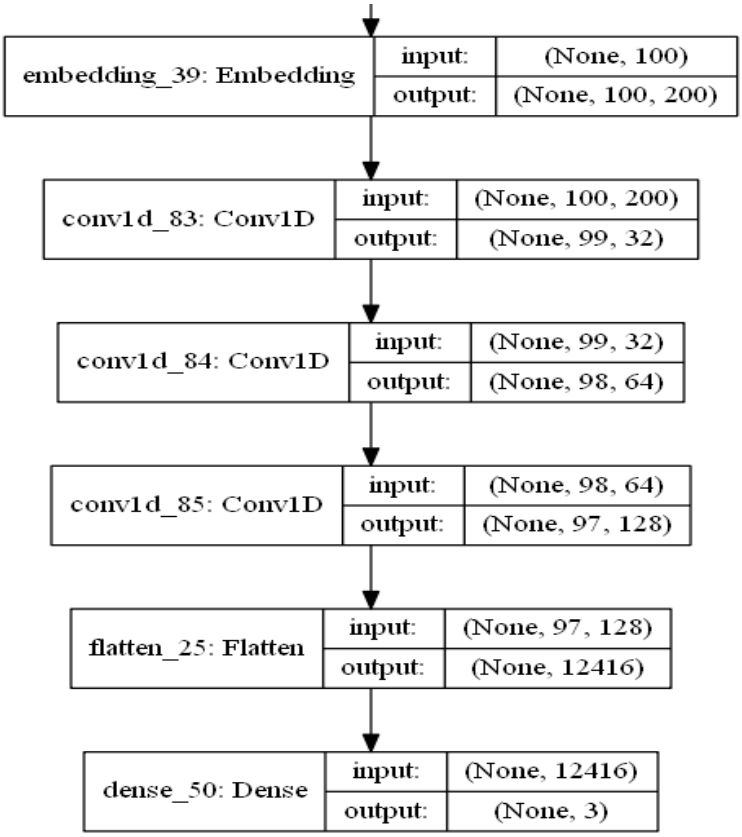
จากผลการทดลองทั้งหมดทำให้ผู้พัฒนาเลือกโมเดล CNN(32,2)+CNN(64,2)+ CNN(128,2) มาใช้ในการพัฒนาโมเดลจำแนกความรู้สึกของข้อความภาษาไทยเพราะ มีความถูกต้องที่ 82.02% ซึ่งมีความใกล้เคียงกับโมเดลแบบ CNN(32,2)+CNN(64,2)+CNN(128,2)+LSTM(12) ที่มีความถูกต้อง 82.20% แต่เวลาที่ใช้ในการเรียนรู้ของโมเดลที่เลือกมาใช้เวลาเรียนรู้ 403 วินาที ซึ่งเป็นครึ่งหนึ่งของโมเดล CNN(32,2)+CNN(64,2)+CNN(128,2)+LSTM(12) ที่ใช้เวลาเรียนรู้ถึง 805 วินาที

บทที่ 4
การพัฒนาโมเดล

ในบทนี้จะกล่าวถึง การพัฒนาโมเดล การออกแบบ การออกแบบวิธีการเตรียมข้อมูล การออกแบบโมเดลที่ใช้ในการจำแนกความรู้สึกของข้อความ

4.1 การพัฒนาโมเดล

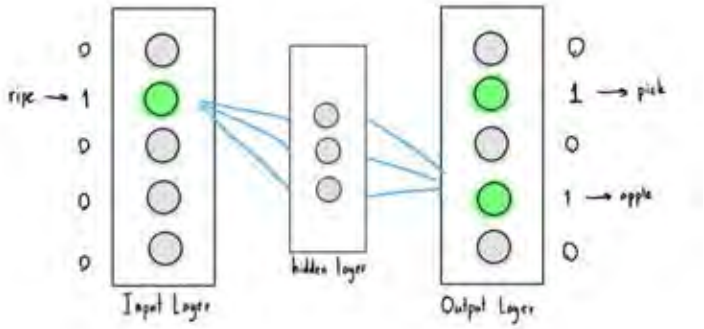
การพัฒนาโมเดลเพื่อนำไปใช้พัฒนาเป็นโมเดลจำแนกความรู้สึกของข้อความภาษาไทย จากโมเดลที่เลือกมา คือ CNN(32,2)+CNN(64,2)+CNN(128,2) คือ โครงข่ายประสาทที่มีส่วนระบบประสาทประมวลผล 3 ชั้นคือ โครงข่ายประสาทแบบสังวัตนาการต่อกันสามชั้น โดยมีจำนวนตัวกรองเท่ากับ 32 64 และ 128 ตามลำดับและมีขนาดเคอร์เนลเท่ากับ 2 เท่ากันทั้งสามชั้นซึ่งมีลักษณะดังรูปที่ 4.1



รูปที่ 4.1 ภาพรวมของโมเดล

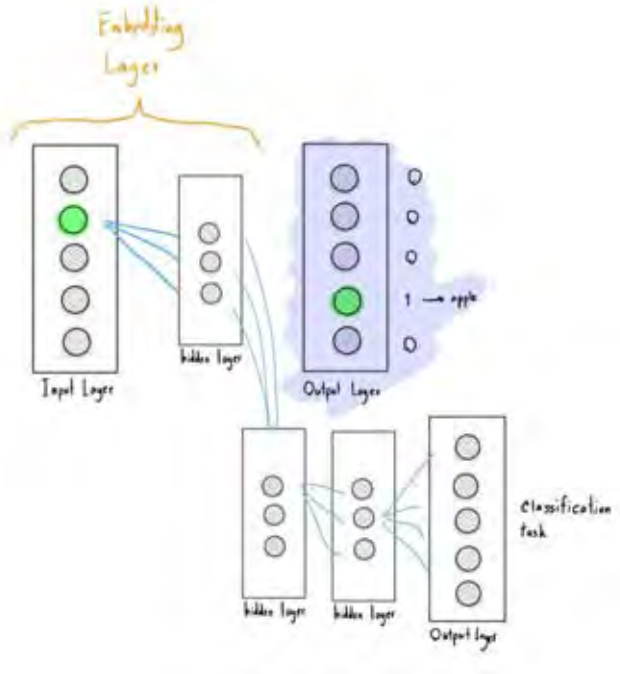
จากรูปที่ 4.1 จะเห็นได้ว่ามีชั้น Embedding ที่ใช้จัดการข้อมูลที่รับเข้ามาให้เหมาะสมกับโมเดลการเรียนรู้เชิงลึก [11] โดยจะเปลี่ยนคำให้อยู่ในรูปแบบของเวกเตอร์เป็นจำนวนจริงเรียงต่อกัน

(เช่น “กิน” เปลี่ยนเป็น (...., 0.24, ..., 0.56, ..., 0.21, ...)) [6] โดยมีจำนวนข้อมูลนำเข้า 100 ตัว และมีการปรับให้เป็นเวกเตอร์ของแต่ละคำให้มีขนาดเท่ากับ 200 คอลัมน์ และในชั้น Embedding มีการทำงานโดยจะนำคำทุกคำที่มีอยู่มาเรียนรู้ในโครงข่ายประสาทเทียมที่มีการกำหนดไว้ด้วยไลบรารี Keras เพื่อประมวลผลหาเวกเตอร์ที่เหมาะสมของแต่ละคำ ซึ่งตัวโครงข่ายประสาทเทียมจะสร้างเวกเตอร์โดยดูจากความคล้ายกันของคำที่จะมาต่อคำนั้น ถ้าคำที่มาต่อเหมือนกันเวกเตอร์ที่สร้างออกมาก็จะมีค่าใกล้เคียงกัน ตามตำแหน่งและการปรากฏของคำ สามารถดูรูปโครงข่ายการทำงานดังรูปที่ 4.2 และมีการนำชั้น Embedding ไปใช้ต่อกับชั้นประมวลผลอื่น ๆ ในโครงข่ายดังรูปที่ 4.3



ที่มา : https://lukkiddd.com/word2vec-ทำอย่างไร-b3de9d9a38b3

รูปที่ 4.2 โครงข่ายประสาทการคำนวณชั้น Embedding



ที่มา : https://lukkiddd.com/word2vec-ทำอย่างไร-b3de9d9a38b3

รูปที่ 4.3 การนำชั้น Embedding ไปใช้

โดยในการกำหนดจำนวนคอลัมน์นั้นผู้พัฒนาคาดว่าจำนวน 200 เพียงพอต่อข้อมูลของคำนำเข้าทั้งหมดที่ใช้ในการพัฒนาโมเดล และได้มีการทดลองในภายหลังพบว่าเพียง 150 ก็เพียงพอต่อข้อมูลที่ทำการทดลองแล้ว โดยผลลัพธ์ได้แสดงตามตารางที่ 4.1

ตารางที่ 4.1 ผลลัพธ์ของจำนวนคอลัมน์ในชั้น embedding ของต่อประสิทธิภาพของโมเดล

ลักษณะโมเดล	ความถูกต้อง (เปอร์เซ็นต์) /เวลา (วินาที)
CNN(32,1)(150 คอลัมน์)	78.09/184
CNN(32,1)(200 คอลัมน์)	78.09/277
CNN(32,1)(250 คอลัมน์)	78.04/304
CNN(64,2)(100 คอลัมน์)	78.75/182
CNN(64,2)(150 คอลัมน์)	79.30/262
CNN(64,2)(200 คอลัมน์)	79.34/292

หลังจากชั้น Embedding จะต่อด้วยตัวโมเดลหลักที่ศึกษาและพัฒนาขึ้น นั่นคือ CNN 3 ชั้น ซึ่งจากจำนวนโหนดนำเข้า 100 ตัว จะเหลือผลลัพธ์ออกเป็น 99 ตัว เนื่องจากขนาดเคอร์เนลมีค่าเท่ากับ 2 ทำให้ 2 ตัวแรกของโหนดนำเข้ารวมกัน เมื่อส่งต่อให้ชั้นถัดไปจึงเหลือเพียง 98 และ 97 ตัว ตามลำดับ ต่อด้วยชั้น Flatten ซึ่งเป็นชั้นเปลี่ยนอาร์เรย์ของข้อมูลทั้งหมดให้เป็นเวกเตอร์ (รูปแสดงตัวอย่างของชั้น Flatten รูปที่ 4.4) และนำไปใช้ในการหาผลลัพธ์ในชั้น Dense ซึ่งเป็นชั้นสุดท้าย เพื่อส่งผลลัพธ์ออกมาเป็นอาร์เรย์ขนาด 3 ช่อง

ที่มา : https://www.superdatascience.com/blogs/convolutional-neural-networks-cnn-step-3-flattening

รูปที่ 4.4 ตัวอย่างการทำงานชั้น flatten

4.2 การออกแบบวิธีการเตรียมข้อมูล

ในส่วนการออกแบบวิธีการเตรียมข้อมูลจะกล่าวถึงการแปลงข้อมูลให้อยู่ในรูปของตัวเลข และการปรับขนาดข้อมูลให้มีขนาดเท่ากับจำนวนข้อมูลนำเข้าของโมเดลโดยมีขั้นตอนดังนี้

ขั้นตอนที่ 1 นำข้อความที่ได้จากการรวบรวมข้อมูล โดยแสดงตัวอย่างของข้อมูลดังรูปที่ 4.5 ซึ่งอยู่ในรูปแบบของข้อความตามด้วยหมายเลขกลุ่มของข้อความ มาทำการตัดแบ่งด้วยไลบรารี deep cut โดยจะได้ลักษณะดังรูปที่ 4.6 และจะมีตัวเลขแสดงกลุ่มของข้อความที่ถูกจำแนกตามหลังข้อความที่ถูกตัดเป็นคำ โดยข้อความที่แสดงความรู้สึกด้านบวกแทนด้วยเลข 0 ข้อความที่แสดงความรู้สึกด้านลบแทนด้วยเลข 1 และข้อความที่ไม่แสดงความรู้สึกหรือเป็นกลางแทนด้วยเลข 2

text	label
แต่ช่วยนี้ใครจะไปก็ต้องเช็คน้ำหนักให้ดี เนื่องจากขึ้นเครื่อง กระเป๋าได้สูงสุด2ใบ รวมกันห้ามเกิน7กิโล ซึ่งมีการเช็คจริง ชั่งจริง ห้ามเกินเด็ดขาด	2
ห้องน้ำสะอาดมาก น้ำร้อนไหลแรง แอร์เย็นฉ่ำ มีระเบียงด้วยสำหรับคนดูดบุหรี่	0
จุดขายอยู่ตรงที่มีรถรับส่งจากสนามบินทุกชั่วโมง และระยะทางที่ใกล้มากไม่ต้องระแวงว่าจะตกเครื่อง	0
บิน UA นานหลายช.ม. ที่นั่งแคบ อาหารสู้สายการบินเอเชียไม่ได้ เที่ยวเดียวก็พอแล้ว	1
airasiax เดินทางไปเกาหลีประหยัดและคุ้มมาก	0

รูปที่ 4.5 ข้อความที่เก็บรวบรวมมา

แต่ ช่วย นี้ ใคร จะ ไป ก็ ต้อง เช็ค น้ำหนัก ให้ ดี เนื่อง จาก ขึ้น เครื่อง กระเป๋า ได้ สูง สุด 2 ใบ รวม กัน ห้าม เกิน 7 กิโล ซึ่ง มี การ เช็ค จริง ชั่ง จริง ห้าม เกิน เด็ดขาด	2
ห้องน้ำ สะอาด มาก น้ำ ร้อน ไหล แรง แอร์เย็น ฉ่ำ มี ระเบียง ด้วย สำหรับ คน ดูด บุหรี่	0
จุด ขาย อยู่ ตรง ที่ มี รถ รับ ส่ง จาก สนามบิน ทุก ชั่วโมง และ ระยะ ทาง ที่ ใกล้ มาก ไม่ ต้อง ระแวง ว่า จะ ตก เครื่อง	0
บิน UA นาน หลาย ช.ม. ที่นั่ง แคบ อาหาร สู้ สาย การ บินเอเชีย ไม่ ได้ เที่ยว เดียว ก็ พอ แล้ว	1
airasiax เดินทาง ไป เกาหลีประหยัด และ คุ้ม มาก	0

รูปที่ 4.6 ข้อความที่ผ่านการตัดคำด้วยไลบรารี deep cut

ขั้นตอนที่ 2 เปลี่ยนคำที่อยู่ในรูปแบบของภาษาไทยเป็นตัวเลขด้วยโทเคนและเติมค่าให้มีจำนวนเท่ากับจำนวนข้อมูลเข้าของโมเดล คือ 100 ตัว สำหรับข้อความที่มีคำไม่ครบจำนวน 100 ตัว จะเติมเลข 0 ด้านหน้าให้ครบ 100 ตัว โดยมีการเติมเลข 0 ด้านหน้าเพราะเป็นผลลัพธ์ที่ได้จากการใช้งานคำสั่งในไลบรารี Keras (คำสั่ง pad_sequences(sequences, maxlen= 100)) หลังจากนั้นแบ่งจำนวนข้อมูลเป็นข้อมูลที่ใช้ในการสอนและข้อมูลที่ใช้ในการตรวจสอบ ลักษณะข้อมูลนำเข้าของ 1 ข้อความ แสดงดังรูปที่ 4.5

```
[   0,    0,    0,    0,    0,    0,    0,    0,    0,    0,    0,
    0,    0,    0,    0,    0,    0,    0,    0,    0,    0,    0,
    0,    0,    0,    0,    0,    0,    0,    0,    0,    0,    0,
    0,    0,    0,    0,    0,    0,    0,    0,    0,    0,    0,
    0,    0,    0,    0,    0,    0,    0,    0,    0,    0,    0,
    0,    0,    0,    0,    0,    0,    0,    0,    0,    0,    0,
    0,    0,    0,    0,    0,    0,    0,    0,    0,    0,    0,
    0,    0,    0,    0,    0,    0,    0,  164,   54,    4,   57,
  253,  760,  422,  896, 1446,    3, 1162,   48,   63,   38, 4471,
  575],
```

รูปที่ 4.5 ข้อมูลนำเข้าสู่ชั้น Embedding ของ 1 ข้อความ

ขั้นตอนที่ 3 แปลงผลลัพธ์ของแต่ละข้อความให้อยู่ในรูปแบบ one-hot คือ อาร์เรย์ของตัวเลขที่มีค่า 0 หรือ 1 เท่านั้น แสดงตามรูปที่ 4.7

```
array([[0., 0., 1.],
       [1., 0., 0.],
       [1., 0., 0.],
       [0., 1., 0.],
       [1., 0., 0.]],
```

รูปที่ 4.7 ผลลัพธ์ของข้อความที่พร้อมนำเข้าเรียนรู้

4.3 การออกแบบโมเดลที่ใช้ในการจำแนกความรู้สึกของข้อความ

ในส่วนการออกแบบโมเดลที่ใช้นี้แบ่งออกเป็น 3 ส่วน ได้แก่ การสร้างและกำหนดค่าโมเดลที่ใช้ อธิบายในหัวข้อ 4.3.1 การนำข้อมูลเข้าไปเรียนรู้ อธิบายในหัวข้อ 4.3.2 และการจำแนกข้อความด้วยโมเดล อธิบายในหัวข้อ 4.3.3

4.3.1 สร้างและกำหนดค่าโมเดลที่ใช้

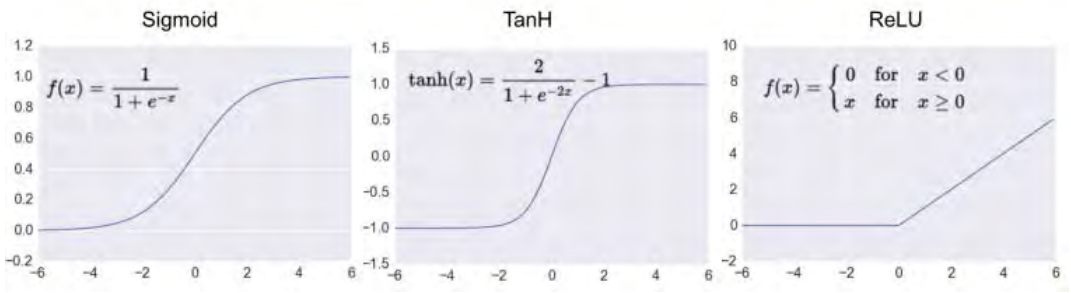
ในการสร้างและกำหนดค่าโมเดล CNN(32,2)+CNN(64,2)+CNN(128,2) จะใช้คำสั่งดังรูปที่ 4.8

```
model2 = models.Sequential()
model2.add(Embedding(Vocab_size, Embed_size, input_length=100))
model2.add(Conv1D(32, 2, activation='relu'))
model2.add(Conv1D(64, 2, activation='relu'))
model2.add(Conv1D(128, 2, activation='relu'))
model2.add(Flatten())
model2.add(Dense(3, activation='softmax'))
model2.compile(loss='categorical_crossentropy', optimizer='adam',    metrics=['accuracy'])
```

รูปที่ 4.8 คำสั่งที่ใช้ในการกำหนดค่าโมเดล

โดยโมเดลจะกำหนดชั้นข้อมูลนำเข้า ให้มีขนาด 100 ตามด้วย CNN ที่มี 32 64 และ 128 ตัวกรอง และขนาดเคอร์เนลเท่ากับ 2 ต่อกันทั้ง 3 ชั้น และมีตัวฟังก์ชันแต่ละตัวเป็นฟังก์ชันการกระตุ้นแบบ relu ที่เป็น สำหรับปรับค่าให้ผลลัพธ์ที่ออกมาเป็นค่าที่มากกว่า 0 เสมอ โดยค่าที่มาก

ที่สุดจะไม่เกิน 1 และตัวฟังก์ชัน relu ยังมีข้อดีมากกว่าฟังก์ชัน sigmoid และฟังก์ชัน tanh คือ การคำนวณไม่ต้องมีการคำนวณค่าเอกซ์โพเนนเชียล และตัวโหนดสามารถมีค่าเป็น 0 ได้ (แต่ sigmoid และ tanh ทำได้แค่เข้าใกล้ 0) ซึ่งส่งผลให้โครงข่ายมีการเรียนรู้ที่เร็วขึ้นและยังแนะนำให้ใช้เป็นฟังก์ชันการกระตุ้นของโครงข่ายประสาทแบบ CNN ในปัจจุบันอีกด้วย [12] โดยจะแสดงฟังก์ชันทั้งสามในรูปที่ 4.9 หลังจากนั้นรวมผลลัพธ์ทั้งหมดด้วยวิธีการ Flatten เพื่อเปลี่ยนให้ผลลัพธ์อยู่ในรูปแบบเวกเตอร์ และส่งไปยังชั้นผลลัพธ์สุดท้ายที่เป็นอาร์เรย์ขนาด 3 ช่อง โดยจะใช้ฟังก์ชันตัวสุดท้ายเป็น softmax ซึ่งเป็นฟังก์ชันการกระตุ้นสำหรับปรับค่าผลลัพธ์ให้อยู่ในรูปแบบของความน่าจะเป็นที่จะถูกจำแนกอยู่ในกลุ่มนั้น และทุกช่อง (3 ช่อง) รวมกันจะได้เท่ากับ 1 เท่านั้น และตัวคำนวณค่าการสูญเสียและปรับเส้นน้ำหนัก คือ adam เพราะมีงานวิจัยว่าเป็นตัวปรับน้ำหนักเส้นที่รวดเร็ว [13] เนื่องจากการปรับน้ำหนักเส้นมีจุดประสงค์เพื่อให้เข้าใกล้กับผลลัพธ์มากที่สุดและใช้เวลาน้อยที่สุดเพื่อลดเวลาในการสอน และใช้การคำนวณค่าการสูญเสียแบบ categorical cross-entropy ซึ่งเป็นการวัดค่าความต่างกันระหว่างความน่าจะเป็นของผลลัพธ์ที่ทำนายกับผลลัพธ์จริง โดยมีสูตร คือ $\mathrm{H(p\,,q)} = -\sum plogq$ โดย p คือความน่าจะเป็นของผลลัพธ์จริง และ q คือค่าความน่าจะเป็นของผลลัพธ์ที่ทำนาย ซึ่งมีความเหมาะสมต่อการคำนวณค่าการสูญเสียแบบหลายกลุ่มในการจำแนก



รูปที่ 4.9 ลักษณะฟังก์ชันของ Sigmoid TanH และ ReLU

4.3.2 การนำข้อมูลเข้าไปเรียนรู้

ในการนำข้อมูลเข้าสู่โมเดลสามารถทำได้โดยใช้ข้อมูลที่ทำการแปลงเป็นตัวเลข โดยแบ่งเป็นข้อมูลที่จะใช้สอนและข้อมูลที่ใช้ในการตรวจสอบประสิทธิภาพเพื่อนำเข้าสู่โมเดล โดยแยกพารามิเตอร์การนำเข้าข้อมูลตามรูปที่ 4.10

```
history2 = model1.fit(X_train, # Features
                      y_train, # Target vector
                      epochs=20, # Number of epochs
                      verbose=1, # Print description after each epoch
                      # Number of observations per batch
                      validation_data=(X_test, y_test),
                      callbacks=[cb]) # Data for evaluation
```

รูปที่ 4.10 คำสั่งการนำข้อมูลเข้าเพื่อเรียนรู้และตรวจสอบความถูกต้องของโมเดล

4.3.3 การจำแนกข้อความด้วยโมเดล

ผลของการจำแนกจะอยู่ในรูปแบบของ ความน่าจะเป็นของข้อความที่จะอยู่ในกลุ่มของความรู้สึกด้านบวก ความรู้สึกด้านลบ และเป็นกลางหรือไม่บ่งบอกความรู้สึก โดยช่องที่มีความน่าจะเป็นมากที่สุดจะเป็นตัวกำหนดว่าข้อความที่ทำนายควรอยู่ในกลุ่มใดดังรูปที่ 4.11

```
[1.0979222e-09 3.7854039e-05 9.9996209e-01] [0. 0. 1.]
[6.3581665e-13 1.4526248e-13 1.0000000e+00] [1. 0. 0.]
[1.1418816e-04 1.8891109e-05 9.9986696e-01] [1. 0. 0.]
[0.685911   0.30890468 0.00518424] [0. 1. 0.]
[9.9982554e-01 1.4904603e-04 2.5402338e-05] [1. 0. 0.]
```

รูปที่ 4.11 ตัวอย่างผลลัพธ์ที่ได้การจำแนกของโมเดลกับผลลัพธ์ที่แท้จริงที่ใช้ในการจำแนก

จากรูปที่ 4.11 จะเห็นได้ว่า โมเดลมีการทำนายออกมาในรูปแบบของความน่าจะเป็นที่ข้อความจะอยู่ในกลุ่มข้อความความรู้สึกด้านบวก ความรู้สึกด้านลบ และเป็นกลางหรือไม่บ่งบอกความรู้สึก โดยถ้ากลุ่มใดมีความน่าจะเป็นมากที่สุดจะถือว่าโมเดลจำแนกให้ข้อความอยู่กลุ่มนั้น ซึ่งจะเห็นว่าโมเดลทำนายได้ถูกต้อง 2 ข้อความ คือ ข้อความที่ 1 และ 5 ที่จำแนกให้อยู่กลุ่มเป็นกลางหรือไม่บ่งบอกความรู้สึก และ ความรู้สึกด้านบวก ตามลำดับ ส่วนข้อมูลที่ทำนายผิด คือ ข้อความที่ 2 และ 3 ที่เป็นข้อความความรู้สึกด้านบวก แต่โมเดลทำนายได้กลุ่มเป็นกลางหรือไม่บ่งบอกความรู้สึก รวมถึงข้อความที่ 4 ที่เป็นความรู้สึกด้านลบ แต่โมเดลทำนายว่าเป็นความรู้สึกด้านบวก

4.4 การใช้งานโมเดล

โมเดลที่ผู้พัฒนาได้พัฒนาขึ้นมีการใช้งานโดยให้ใส่ข้อความที่มีความยาวไม่เกิน 100 คำ ในฟังก์ชันที่มีการเขียนไว้เพื่อใช้ในการจำแนกประเภทของข้อความมีตัวอย่างตามรูปที่ 4.12

```
print(decision(test_model("ไม่แนะนำให้ใช้บริการ",model5,t)[0]))
```

รูปที่ 4.12 ฟังก์ชันการเรียกใช้โมเดล

โดยผลลัพธ์ที่ได้จะแสดงออกมาเป็นตัวเลข

ถ้าเลข 0 คือ อยู่ในกลุ่มที่แสดงความรู้สึกด้านบวก

ถ้าเลข 1 คือ อยู่ในกลุ่มที่แสดงความรู้สึกด้านลบ

ถ้าเลข 2 คือ อยู่ในกลุ่มที่แสดงไม่มีความรู้สึกหรือเป็นกลาง

```
ไม่แนะนำให้ใช้บริการ
['ไม่', 'แนะนำ', 'ให้', 'ใช้', 'บริการ']
[[2, 145, 19, 50, 34]]
1
```

รูปที่ 4.13 ผลลัพธ์ที่ได้จากการเรียกใช้งานโมเดล

4.5 ภาษาและโปรแกรมที่ใช้พัฒนาโมเดล

4.5.1 ภาษาที่ใช้พัฒนาโมเดล

ภาษาไพทอนใช้พัฒนาในส่วนของการวิเคราะห์ข้อมูล สร้างโมเดล และกำหนดค่าโมเดลและประเมินความถูกต้องของโมเดล

4.5.2 โปรแกรมที่ใช้พัฒนาโมเดล

- Jupyter Notebook เป็นเครื่องมือในการเขียนโปรแกรมภาษาไพทอน
- Notepad ใช้ในการเก็บข้อมูล
- Autohotkey ใช้เป็นเครื่องมือช่วยในการรวบรวมข้อมูล
- Keras library ไลบรารีภาษาไพทอนใช้ในการสร้างโมเดลและเป็นไลบรารีสำหรับการเรียนรู้เชิงลึกของภาษาไพทอน
- Deep cut library ใช้ในการตัดคำ

บทที่ 5
ผลการทดสอบโมเดล

ในบทนี้จะกล่าวถึง ผลการทดสอบ การวิเคราะห์ข้อมูลผลการจำแนกความรู้สึกจากข้อความ โดยจะใช้ข้อมูลที่ได้มาทั้งหมดซึ่งเป็นข้อมูลแบบไม่สมดุลกันในแต่ละหมวดมาใช้ในการเรียนรู้และตรวจสอบผลลัพธ์ และแสดงผลการตรวจสอบผลลัพธ์ด้วยคอนฟิวชันเมทริกซ์ (Confusion Matrix)

5.1 การทดสอบการจำแนกข้อความ

5.1.1 ข้อมูลที่ใช้ในการทดสอบโมเดล

จากข้อความ 12,596 ข้อความ ตามตาราง 3.1 ดังนี้

หมวดหมู่	ความรู้สึกด้านบวก	ความรู้สึกด้านลบ	ไม่แสดงความรู้สึกหรือเป็นกลาง
โรงแรม	1,544	1,183	713
ร้านอาหาร	1,401	713	920
สถานที่ท่องเที่ยว	1,286	723	1,027
สายการบิน	1,470	636	980

มีการทดสอบประสิทธิภาพของโมเดลโดยแบ่งข้อมูลทั้งหมดเป็น 3 ส่วนคือ ข้อมูลในการสอน ข้อมูลในการตรวจสอบ และข้อมูลในการทดสอบ วิธีการแบ่งข้อมูล คือ จาก 12,596 ข้อความ จะใช้เป็นข้อมูลทดสอบ 15% นั่นคือ 1,890 ข้อความ และข้อมูล 85% (10,706 ข้อความ) จะแบ่งเป็นข้อมูลสอน และข้อมูลตรวจสอบด้วยอัตราส่วน 80:20 ดังนั้นมีข้อมูลใช้สอน 8,564 ข้อความ และข้อมูลตรวจสอบประสิทธิภาพของโมเดล 2,142 ข้อความ กลุ่มของข้อมูลแสดงในตารางที่ 5.1

ตารางที่ 5.1 กลุ่มข้อความของข้อมูลใช้สอน ข้อมูลตรวจสอบ และข้อมูลทดสอบ

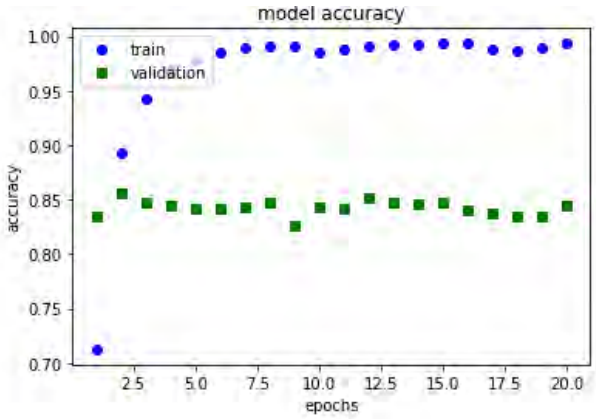
กลุ่มข้อมูล	ความรู้สึกด้านบวก	ความรู้สึกด้านลบ	ไม่แสดงความรู้สึกหรือเป็นกลาง
ข้อมูลใช้สอน (8,564)	3,898	2,220	2,446
ข้อมูลตรวจสอบ (2,142)	947	550	645
ข้อมูลทดสอบ (1,890)	856	485	549

5.1.2 การทดสอบโมเดลด้วยชุดข้อมูลตรวจสอบ

ผลการทดสอบโมเดลด้วยข้อมูลตรวจสอบ แสดงดังรูปที่ 5.1 โดยจะดูจากค่าความถูกต้องของการทำนายด้วยข้อมูลตรวจสอบ (val_acc) เป็นหลัก ซึ่งจะเห็นได้ว่าโมเดลมีความถูกต้องของการทำนายข้อมูลตรวจสอบอยู่ที่ประมาณ 82-85% จากการสอนทั้งหมด 20 รอบ โดยจะสามารถวาดกราฟระหว่างความถูกต้องและรอบการสอนได้ดังรูปที่ 5.2 และผลลัพธ์คอนฟิวชันเมทริกซ์ดังรูปที่ 5.3

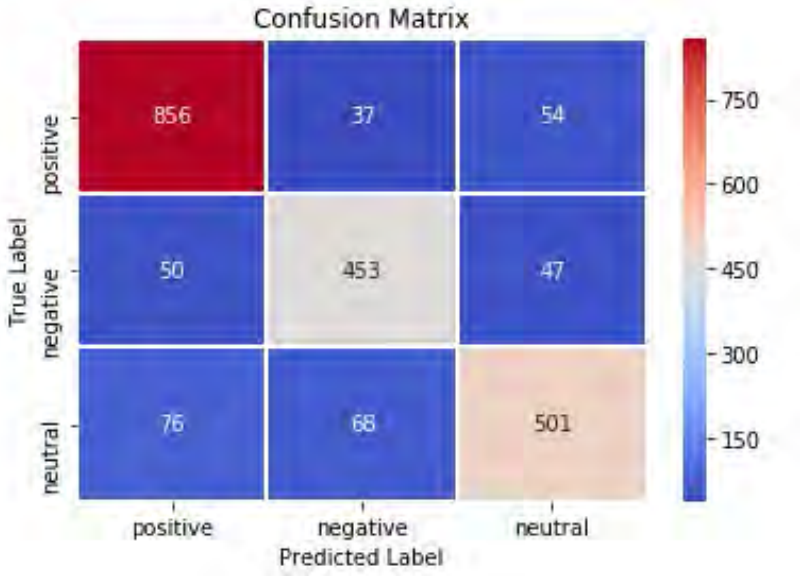
```
Train on 8564 samples, validate on 2142 samples
Epoch 1/20
8564/8564 [==============================] - 39s 5ms/step - loss: 0.6497 - acc: 0.7126 - val_loss: 0.4377 - val_acc: 0.8352
Epoch 2/20
8564/8564 [==============================] - 35s 4ms/step - loss: 0.3027 - acc: 0.8929 - val_loss: 0.4049 - val_acc: 0.8562
Epoch 3/20
8564/8564 [==============================] - 34s 4ms/step - loss: 0.1736 - acc: 0.9427 - val_loss: 0.4648 - val_acc: 0.8478
Epoch 4/20
8564/8564 [==============================] - 38s 4ms/step - loss: 0.1045 - acc: 0.9675 - val_loss: 0.5684 - val_acc: 0.8450
Epoch 5/20
8564/8564 [==============================] - 42s 5ms/step - loss: 0.0690 - acc: 0.9783 - val_loss: 0.6466 - val_acc: 0.8422
Epoch 6/20
8564/8564 [==============================] - 43s 5ms/step - loss: 0.0458 - acc: 0.9856 - val_loss: 0.7566 - val_acc: 0.8422
Epoch 7/20
8564/8564 [==============================] - 41s 5ms/step - loss: 0.0353 - acc: 0.9897 - val_loss: 0.8312 - val_acc: 0.8427
Epoch 8/20
8564/8564 [==============================] - 43s 5ms/step - loss: 0.0312 - acc: 0.9911 - val_loss: 0.8366 - val_acc: 0.8473
Epoch 9/20
8564/8564 [==============================] - 38s 4ms/step - loss: 0.0311 - acc: 0.9902 - val_loss: 0.9892 - val_acc: 0.8263
Epoch 10/20
8564/8564 [==============================] - 37s 4ms/step - loss: 0.0451 - acc: 0.9852 - val_loss: 0.8619 - val_acc: 0.8436
Epoch 11/20
8564/8564 [==============================] - 40s 5ms/step - loss: 0.0398 - acc: 0.9877 - val_loss: 0.8686 - val_acc: 0.8422
Epoch 12/20
8564/8564 [==============================] - 38s 4ms/step - loss: 0.0306 - acc: 0.9904 - val_loss: 0.8718 - val_acc: 0.8511
Epoch 13/20
8564/8564 [==============================] - 37s 4ms/step - loss: 0.0230 - acc: 0.9926 - val_loss: 0.9213 - val_acc: 0.8469
Epoch 14/20
8564/8564 [==============================] - 39s 5ms/step - loss: 0.0227 - acc: 0.9918 - val_loss: 0.9428 - val_acc: 0.8464
Epoch 15/20
8564/8564 [==============================] - 35s 4ms/step - loss: 0.0204 - acc: 0.9933 - val_loss: 1.0083 - val_acc: 0.8469
Epoch 16/20
8564/8564 [==============================] - 36s 4ms/step - loss: 0.0215 - acc: 0.9930 - val_loss: 1.0176 - val_acc: 0.8403
Epoch 17/20
8564/8564 [==============================] - 35s 4ms/step - loss: 0.0332 - acc: 0.9883 - val_loss: 1.1063 - val_acc: 0.8375
Epoch 18/20
8564/8564 [==============================] - 39s 5ms/step - loss: 0.0408 - acc: 0.9868 - val_loss: 0.9951 - val_acc: 0.8347
Epoch 19/20
8564/8564 [==============================] - 35s 4ms/step - loss: 0.0311 - acc: 0.9896 - val_loss: 0.9944 - val_acc: 0.8352
Epoch 20/20
8564/8564 [==============================] - 39s 5ms/step - loss: 0.0215 - acc: 0.9930 - val_loss: 1.0051 - val_acc: 0.8450
```

รูปที่ 5.1 ผลการเรียนรู้และผลการตรวจสอบของโมเดล



รูปที่ 5.2 กราฟแสดงความถูกต้องของชุดข้อมูลสอนและชุดข้อมูลตรวจสอบต่อรอบการสอนของโมเดล

จากรูปที่ 5.1 และ 5.2 จะเห็นได้ว่าโมเดลมีความถูกต้องของการทำนายข้อมูลตรวจสอบที่ดีตั้งแต่รอบที่ 2 จึงควรหยุดการสอนตั้งแต่รอบที่ 2 เพื่อไม่ให้โมเดลมีการจำข้อมูลที่ใช้ในการสอนมากเกินไป



รูปที่ 5.3 คอนฟิวชันเมทริกซ์ของการตรวจสอบโมเดลด้วยข้อมูลชุดตรวจสอบ

จากรูปที่ 5.3 อธิบายได้ว่า โมเดลทำนายข้อมูลตรวจสอบได้ถูกต้องและไม่ถูกต้องดังต่อไปนี้

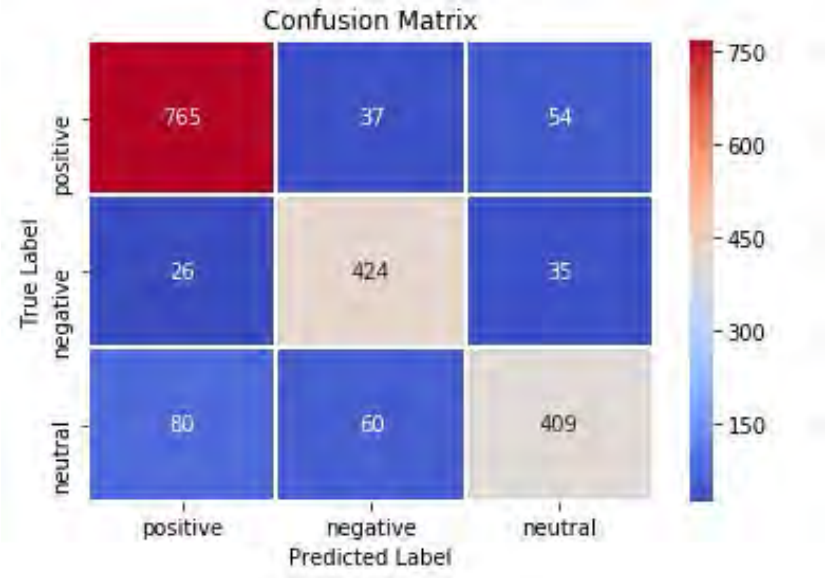
- กลุ่ม 0 ความรู้สึกด้านบวก ถูกต้องจำนวน 856 ข้อความ และไม่ถูกต้องจำนวน 50 + 76 เท่ากับจำนวน 126 ข้อความ

- กลุ่ม 1 ความรู้สึกด้านลบ ถูกต้องจำนวน 453 ข้อความ และไม่ถูกต้องจำนวน 37 + 68 เท่ากับจำนวน 105 ข้อความ

- กลุ่ม 2 ความรู้เป็นกลางหรือไม่แสดงความรู้สึก ถูกต้องจำนวน 501 ข้อความ และไม่ถูกต้องจำนวน 54 + 47 เท่ากับจำนวน 101 ข้อความ

5.1.3 การทดสอบโมเดลด้วยข้อมูลชุดทดสอบ

ผลการทดสอบโมเดลด้วยข้อมูลทดสอบสามารถแสดงเป็นคอนฟิวชันเมทริกซ์ได้ดังรูปที่ 5.4



รูปที่ 5.4 คอนฟิวชันเมทริกซ์ของการตรวจสอบโมเดลด้วยข้อมูลชุดทดสอบ

จากรูปที่ 5.4 อธิบายได้ว่า โมเดลทำนายข้อมูลตรวจสอบได้ถูกต้องและไม่ถูกต้องดังต่อไปนี้

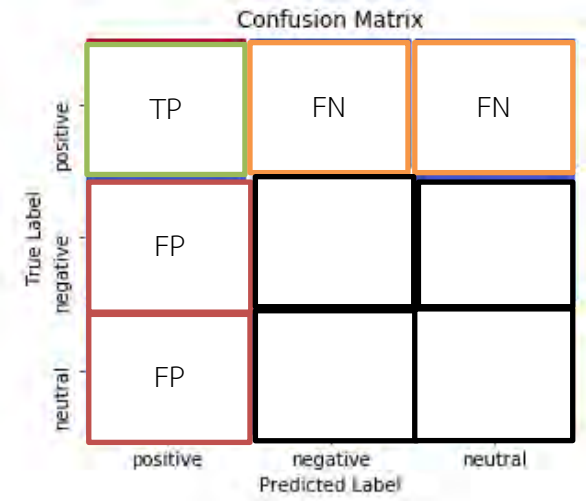
- กลุ่ม 0 ความรู้สึกด้านบวก ถูกต้องจำนวน 765 ข้อความ และไม่ถูกต้องจำนวน 26 + 80 เท่ากับจำนวน 106 ข้อความ

- กลุ่ม 1 ความรู้สึกด้านลบ ถูกต้องจำนวน 424 ข้อความ และไม่ถูกต้องจำนวน 37 + 60 เท่ากับจำนวน 97 ข้อความ

- กลุ่ม 2 ความรู้เป็นกลางหรือไม่แสดงความรู้สึก ถูกต้องจำนวน 409 ข้อความ และไม่ถูกต้องจำนวน 80 + 60 เท่ากับจำนวน 140 ข้อความ

5.1.4 สรุปผลการทดสอบ

สรุปผลการทดสอบจะวัดประสิทธิภาพของโมเดลด้วยค่าความถูกต้อง ความแม่นยำ (precision) รีคอล (recall) และ F1 score ซึ่งมีตัวอย่างการกำหนดค่าที่ใช้ในการคำนวณของกลุ่ม 0 ดังรูปที่ 5.5



รูปที่ 5.5 การกำหนดค่าในคอนฟิวชันเมทริกซ์โดยมีกลุ่มที่สนใจคือกลุ่ม 0

กำหนดให้

Positive (P) คือ เรื่องที่เราสนใจ Negative (N) คือ เรื่องที่ไม่สนใจ

True (T) คือ ถูก False (F) คือ ผิด

สูตรและตัวอย่างการคำนวณ ความแม่นยำ รีคอล และ F1 score ของกลุ่ม 0 ดังนี้

Precision= TP/ (TP + FP)

ตัวอย่างการคำนวณความแม่นยำกลุ่ม 0 ในข้อมูลชุดตรวจสอบ คือ 856 / (856 + (50 + 76)) เท่ากับ 0.8716

ตัวอย่างการคำนวณความแม่นยำกลุ่ม 0 ในข้อมูลชุดทดสอบ คือ 765 / (765 + (26 + 80)) เท่ากับ 0.8783

Recall= TP/ (TP + FN)

ตัวอย่างการคำนวณรีคอลกลุ่ม 0 ในข้อมูลชุดตรวจสอบ คือ 856 / (856 + (37 + 54)) เท่ากับ 0.9039

ตัวอย่างการคำนวณรีคอลกลุ่ม 0 ในข้อมูลชุดทดสอบ คือ 765 / (765 + (37 + 54)) เท่ากับ 0.8937

F1 = 2 * (Precision * Recall / (Precision + Recall))

ตัวอย่างการคำนวณ F1-score กลุ่ม 0 ในข้อมูลชุดตรวจสอบ คือ 2 *(0.8716 * 0.9039 / (0.8716 + 0.9093)) เท่ากับ 0.8874

ตัวอย่างการคำนวณ F1-score กลุ่ม 0 ในข้อมูลชุดทดสอบ คือ 2 *(0.8783 * 0.8937 / (0.8783 + 0.8937)) เท่ากับ 0.8859

จากคอนฟิวชันเมทริกซ์ในรูปที่ 5.3 และ 5.4 ของการทดสอบด้วยข้อมูลตรวจสอบและข้อมูลทดสอบ สรุปเป็นผลลัพธ์ค่าความถูกต้อง ความแม่นยำ (precision) รีคอล (recall) และ F1 score ได้ดังตารางที่ 5.2

ตารางที่ 5.2 ผลลัพธ์ของ ความถูกต้อง ความแม่นยำ รีคอล และ F1-score

ชนิดข้อมูล	ความถูกต้อง	ความแม่นยำ	รีคอล	F1-score	กลุ่ม
ข้อมูลชุดตรวจสอบ	84.50%	87.16%	90.39%	0.887	0 – บวก
		81.18%	82.36%	0.817	1 – ลบ
		83.22%	77.67%	0.803	2 - กลาง
ข้อมูลชุดทดสอบ	84.55%	87.83%	89.37%	0.886	0 – บวก
		81.38%	87.42%	0.843	1 – ลบ
		82.13%	74.50%	0.781	2 - กลาง

จากตารางที่ 5.2 พบว่าโมเดลมีประสิทธิภาพการจำแนกข้อความจากกลุ่มความรู้สึกด้านบวกได้ดีที่สุดทั้งในข้อมูลชุดตรวจสอบและข้อมูลชุดทดสอบ เนื่องจากค่าความแม่นยำ ค่ารีคอล และค่า F1-score มีค่าสูงที่สุด และตัวโมเดลมีค่า F1-score ที่มากกว่า 0.75 ในทุก ๆ กลุ่ม โดยยิ่งค่า F1 มีค่าใกล้กับ 1 หมายความว่าโมเดลมีค่า ความแม่นยำ และรีคอลที่ดี ซึ่งอาจกล่าวได้ว่าโมเดลนี้มีความสามารถจำแนกข้อมูลเป็นความรู้สึกด้านบวกได้ดีที่สุด แต่ประสิทธิภาพโดยรวมถือว่าอยู่ในมาตรฐานที่น่าพึงพอใจ

ดังนั้นสามารถสรุปผลได้ว่า โมเดลการจำแนกความรู้สึกจากข้อความที่พัฒนามาจากข้อมูลที่มีความสมดุลกันของแต่ละประเภท สามารถทำงานได้ดีแม้จะเป็นชุดข้อมูลที่ไม่มีความสมดุลกันของข้อมูลแต่ละประเภท โดยมีลักษณะของโมเดล คือ โครงข่ายประสาทที่มีส่วนระบบประสาทประมวลผล 3 ชั้น คือ โครงข่ายประสาทแบบ CNN ซึ่งมีตัวกรองขนาด 32 64 และ 128 ตามลำดับ และมีขนาดเคอร์เนลเท่ากับ 2 เท่ากันทั้งสามชั้น ซึ่งมีลักษณะดังรูปที่ 4.1 ที่มีการกล่าวมาข้างต้น

5.2 ข้อจำกัดของระบบ

1. สามารถจำแนกได้ดีกับข้อความที่อยู่ในรูปแบบของความรู้สึกที่เป็นไปในทางเดียวกันทั้งข้อความ
2. ยังไม่มีการพัฒนาในส่วนของการต่อประสานการใช้งานกับผู้ใช้งานทำให้ยุ่งยากซับซ้อนและไม่สบายตาต่อผู้ใช้งาน
3. หากข้อความที่ต้องการจำแนกมีหลายประโยคและมีความรู้สึกที่ต่างกัน มีโอกาสเกิดข้อผิดพลาดในการจำแนกกลุ่มของข้อความได้สูง
4. โมเดลยังไม่สามารถทำนายข้อความที่มีความหมายทั้งด้านบวกและด้านลบอยู่ด้วยกันได้อย่างมีประสิทธิภาพ
5. โมเดลยังไม่สามารถทำนายข้อความที่เป็นประโยคปฏิเสธซ้อนปฏิเสธได้ถูกต้อง

บทที่ 6
ข้อสรุปและข้อเสนอแนะ

6.1 สรุปผล

จากโมเดลที่ทดลองทั้งหมดได้โมเดลที่มีประสิทธิภาพดีที่ผู้พัฒนาเลือกมาใช้ซึ่งมีลักษณะเป็นการรับข้อมูลนำเข้าจำนวน 100 ตัวโครงข่ายประสาทที่มีส่วนระบบประสาทประมวลผล 3 ชั้นคือ โครงข่ายประสาทแบบสังวัตนาการต่อกันกับสังวัตนาการซึ่งมีฟิลเตอร์ขนาด 32 64 และ 128 ตามลำดับ โดยมีขนาดเคอร์เนล 2 เท่ากันทั้งสามชั้น และมีรูปแบบผลลัพธ์เป็นอาร์เรย์ของความน่าจะเป็นแต่ละกลุ่มจำนวน 3 ช่อง พัฒนาด้วยภาษาไพทอนบนเครื่องมือที่มีชื่อว่า จูปิเตอร์โน้ตบุ๊ก ซึ่งสามารถทำงานได้ดีทั้งข้อมูลที่มีความสมดุลกันและข้อมูลที่ไม่มีความสมดุลกัน และแสดงผลลัพธ์ของการทดสอบด้วยข้อมูลชุดตรวจสอบและข้อมูลชุดทดสอบออกมาเป็นคอนฟิวชันเมทริกซ์ที่มีความละเอียดของการแสดงผลลัพธ์การทำนายที่ชัดเจน โดยในการจำแนกระหว่างความรู้สึกด้านบวก ความรู้สึกด้านลบ และไม่แสดงความรู้สึกหรือเป็นกลางมีค่าความถูกต้องประมาณ 85% และความแม่นยำระหว่าง 81-87%

6.2 ผลที่ได้รับ

ผลที่ได้รับจากการพัฒนาโมเดลจำแนกความรู้สึกจากข้อความภาษาไทยจะมี 2 ส่วน คือ

ผลที่ได้รับต่อผู้พัฒนา

- ผู้พัฒนาได้เรียนรู้และเข้าใจเทคนิคการเรียนรู้ด้วยเครื่องและการเรียนรู้แบบเชิงลึก
- ได้ความรู้และความเข้าใจเกี่ยวกับการทำงานของการเรียนรู้ด้วยเครื่องในการวิเคราะห์ความรู้สึกจากข้อความและการจัดการข้อมูลที่อยู่ในรูปแบบตัวอักษร
- มีความเข้าใจและมีทักษะในการใช้ภาษาไพทอน (python) และเครื่องมือต่าง ๆ ที่ใช้ในการพัฒนา

ผลที่ได้รับต่อผู้ใช้งาน

- มีโมเดลที่มีความแม่นยำในการวิเคราะห์ความรู้สึกจากข้อความภาษาไทย
- ช่วยอำนวยความสะดวกในการวิเคราะห์และสรุปผลข้อมูลเพื่อใช้ในการตัดสินใจได้

6.3 ปัญหาและอุปสรรค

- ข้อมูลบางประเภทหาตัวอย่างข้อความได้น้อยกว่าประเภทอื่น ๆ เช่น ข้อความแสดงความรู้สึกด้านลบ
- ข้อความบางประโยคมีความกำกวมไม่สามารถตีความว่าเป็นด้านบวกหรือด้านลบได้อย่างชัดเจน ทำให้เกิดความสับสนในการรวบรวมข้อมูล

6.4 วิธีการแก้ปัญหา

- พยายามเก็บข้อมูลข้อความให้มากขึ้นเพื่อให้เพียงพอต่อการสร้างโมเดลที่มีประสิทธิภาพ
- กำหนดเป็นข้อจำกัดของโมเดลว่ามีการจำแนกประเภทประโยคด้านบวกและด้านลบตามการตีความหมายของผู้พัฒนา

6.5 ข้อเสนอแนะ

- ควรจะเก็บรวบรวมข้อมูลสอนให้มากขึ้น เพื่อให้โมเดลเรียนรู้ประโยคที่มีความหลากหลายได้สมบูรณ์ขึ้น
- เพิ่มมาตรฐานหรือเกณฑ์ในการจำแนกกลุ่มของข้อความที่แสดงความรู้สึกด้านบวก ด้านลบ หรือเป็นกลางอย่างชัดเจน

เอกสารอ้างอิง

[1] Chunting Zhou, Chonglin Sun, Zhiyuan Liu and Francis C.M. Lau. (2015). A C-LSTM Neural Network for Text Classification. arXiv. https://arxiv.org/pdf/1511.08630.pdf

[2] Xiang Zhang, Junbo Zhao and Yann LeCun. (2015). Character-level Convolutional Networks for Text Classification. In Proceedings of the 28th International Conference on Neural Information Processing Systems Vol. 1 (649-657). Montreal, Canada: MIT Press Cambridge. https://papers.nips.cc/paper/5782-character-level-convolutional-networks-for-text-classification.pdf.

[3] Tripadvisor. Retrived November 6, 2018, from https://th.tripadvisor.com

[4] Vithan Minaphinant. (2018). Machine Learning คืออะไร?. Retrived April 9, 2019, from https://blog.finnomena.com/machine-learning-fa8bf6663c07

[5] วิกิพีเดีย สารานุกรมเสรี. (2017). การเรียนรู้เชิงลึก. Retrived April 9, 2019, from https://th.wikipedia.org/wiki/การเรียนรู้เชิงลึก

[6] Lei Zhang, Shuai Wang and Bing Liu. (2018). Deep Learning for Sentiment Analysis: A Survey. arXiv. https://arxiv.org/ftp/arxiv/papers/1801/1801.07883.pdf

[7] Mc.ai. (2018). มาลองดูวิธีการคิดของ CNN กัน !!!. Retrived April 9, 2019, from https://mc.ai/มาลองดูวิธีการคิดของ-cnn-ก-2/

[8] Sirinart Tangruamsub. (2017). Long Short-Term Memory (LSTM). Retrived April 9, 2019, from https://medium.com/@sinart.t/long-short-term-memory-lstm-e6cb23b494c6

[9] Jason Brownlee. (2016). Evaluate the Performance Of Deep Learning Models in Keras. Retrived April 9, 2019, from https://machinelearningmastery.com/evaluate-performance-deep-learning-models-keras/

[10] Tobias Würfl. (2016). How can I decide the kernel size, output maps and layers of CNN? Retrived April 9, 2019, from https://www.quora.com/How-can-I-decide-the-kernel-size-output-maps-and-layers-of-CNN

[11] Collobert R, Weston J, Bottou L, Karlen M, Kavukcuoglu K, and Kuksa P. (2011). Natural language processing (almost) from scratch. Journal of Machine Learning Research, Vol 12, 2493-2537. http://www.jmlr.org/papers/volume12/collobert11a/collobert11a.pdf

[12] Jason Brownlee. (2019). A Gentle Introduction to the Rectified Linear Activation Function for Deep Learning Neural Networks. Retrived April 9, 2019, from https://machinelearningmastery.com/rectified-linear-activation-function-for-deep-learning-neural-networks/

[13] Diederik P. Kingma and Jimmy Lei Ba. (2015). adam: a method for stochastic optimization. International Conference on Learning Representations (ICLR). May 7-9, 2015, San Diego. https://arxiv.org/pdf/1412.6980v8.pdf

ภาคผนวก

ภาคผนวก ก

แบบเสนอหัวข้อโครงงาน รายวิชา 2301399 Project Proposal

ปีการศึกษา 2561

ชื่อโครงงาน (ภาษาไทย)	การพัฒนาโมเดลการจำแนกความรู้สึกของข้อความภาษาไทยโดยใช้เทคนิคการเรียนรู้ด้วยเครื่อง
ชื่อโครงงาน (ภาษาอังกฤษ)	Development of a classification model for Thai statement sentiments using ML techniques
อาจารย์ที่ปรึกษา	ผู้ช่วยศาสตราจารย์ ดร.ภควรรณ ปักษี
ผู้ดำเนินการ	นายนิติกร องค์ศิริมงคล เลขประจำตัวนิสิต 5833637423 สาขาวิชาวิทยาการคอมพิวเตอร์ ภาควิชาคณิตศาสตร์และวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย

หลักการและเหตุผล

ในปัจจุบันข้อมูลมีอยู่ในหลากหลายรูปแบบ หนึ่งในนั้นคือข้อความในรูปแบบตัวอักษร (text) ที่มีอยู่เป็นจำนวนมากในอินเทอร์เน็ต ซึ่งข้อมูลเหล่านี้สามารถนำไปใช้ทำประโยชน์ได้ในหลาย ๆ ด้าน เช่น การสร้างระบบแนะนำสินค้า (product recommender system) การวิเคราะห์ความรู้สึกจากข้อความ (sentiment analysis) การทำเหมืองข้อมูล (data mining) และอื่น ๆ แต่ผู้จัดทำได้เล็งเห็นถึงความสำคัญของการวิเคราะห์ความรู้สึกจากข้อความ เนื่องจากหากสามารถแบ่งแยกข้อความที่แสดงความรู้สึกทางด้านบวกและทางด้านลบออกจากกันได้จะเป็นประโยชน์ในการควบคุมคุณภาพสินค้าหรือรักษาคุณภาพการให้บริการ และการปรับปรุงคุณภาพสินค้าหรือการให้บริการที่ถูกกล่าวถึงในข้อความให้ดีขึ้นได้

แต่ในการพัฒนาระบบวิเคราะห์ความรู้สึกจากข้อความด้วยเทคนิคการเรียนรู้ด้วยเครื่อง (machine learning) ในวิธีหนึ่งจำเป็นต้องใช้รายการคำ (word list) ที่ประกอบด้วยคำที่แสดงความรู้สึกด้านบวก และคำที่แสดงความรู้สึกด้านลบ ซึ่งการระบุคำเหล่านี้ให้ครอบคลุมทุกคำเป็นเรื่องยาก ทางผู้จัดทำจึงจะวิเคราะห์ข้อมูลประเภทข้อความและสร้างโมเดลในการกำหนดความรู้สึกของข้อความโดยใช้เทคนิคการเรียนรู้ด้วยเครื่อง ซึ่งจะแบ่ง

ข้อความออกเป็นสองกลุ่ม คือ ข้อความที่แสดงความรู้สึกด้านบวกและข้อความที่แสดงความรู้สึกด้านลบ เพื่อสามารถนำไปใช้ในการวิเคราะห์ความรู้สึกจากข้อความได้

การเรียนรู้ด้วยเครื่อง คือ การให้ระบบหรือโปรแกรมมีการเรียนรู้และแก้ปัญหาหรือตัดสินใจด้วยตัวระบบเองโดยใช้ข้อมูลเป็นตัวเรียนรู้ ซึ่งส่วนใหญ่ต้องใช้ข้อมูลจำนวนมากในการสอนให้ระบบหรือโปรแกรมตัดสินใจได้อย่างถูกต้องและสร้างโมเดลในการแก้ปัญหา จึงต่างจากการเขียนโปรแกรมโดยตรงที่จะมีการใส่คำสั่งการทำงานและใส่ข้อมูลเพื่อให้ได้คำตอบ แต่การเรียนรู้ด้วยเครื่องจะใช้การใส่ข้อมูลสอนและระบุคำตอบเพื่อสร้างโมเดลหรือโมเดลในการหาคำตอบ ซึ่งมีหลากหลายเทคนิค หนึ่งในนั้นคือ การเรียนรู้เชิงลึก (deep learning)

การเรียนรู้เชิงลึกเป็นกระบวนการเลียนแบบระบบเซลล์ประสาทในสมองของมนุษย์ (Neural Network) โดยเซลล์ประสาทแต่ละตัวจะเชื่อมต่อกันด้วยเส้นประสาท สามารถจำลองเป็นโครงข่ายประสาทเทียม (Artificial Neural Network) แบ่งเป็น 3 ส่วน 1. ส่วนเซลล์ประสาทที่รับข้อมูลเข้า (input layer) 2. ส่วนระบบประสาทประมวลผล (hidden layer) 3. ส่วนเซลล์ประสาทที่ส่งผลลัพธ์ของข้อมูลหลังประมวลผล (output layer) ซึ่งแต่ละส่วนมีการเชื่อมกันด้วยเส้นประสาทเทียมและมีการถ่วงน้ำหนัก (weight) ในการจำลองโครงข่ายประสาทเทียมสามารถทำได้หลายรูปแบบ เช่น โครงข่ายประสาทเทียมแบบสังวัตนาการ (Convolutional Neural Networks) ซึ่งมีความสามารถในการจำแนกข้อความในรูปแบบตัวอักษร [1] และโครงข่ายประสาทแบบ LSTM (Long Short-Term Memory) มีความสามารถในการรองรับข้อมูลแบบมีลำดับเวลา (time series data) ซึ่งข้อมูลแบบข้อความประกอบด้วยลำดับของคำที่ต่อกันทำให้เกิดเป็นประโยค (sentence) ประโยคย่อย (clause) หรือวลี (phrase) จึงสามารถใช้ในการจำแนกข้อความในรูปแบบตัวอักษรได้ [2]

โครงงานนี้จึงจะจัดทำเพื่อสร้างโมเดลโดยด้วยเทคนิคการเรียนรู้เชิงลึกของการเรียนรู้ด้วยเครื่อง ซึ่งมีการสร้างโครงข่ายประสาทเทียมในรูปแบบสังวัตนาการหรือ LSTM เพื่อใช้ในการจำแนกข้อความภาษาไทย โดยแบ่งเป็นกลุ่มของข้อความที่แสดงความรู้สึกด้านบวกและกลุ่มของข้อความที่แสดงความรู้สึกด้านลบ

วัตถุประสงค์

1. เพื่อศึกษาวิธีการวิเคราะห์และจำแนกความรู้สึกของข้อความภาษาไทย
2. เพื่อสร้างโมเดลในการจำแนกความรู้สึกของข้อความภาษาไทย

ขอบเขตของโครงงาน

1. แหล่งข้อมูลจากรีวิวในกลุ่มโรงแรม ร้านอาหาร สถานที่ท่องเที่ยว และสายการบิน จากเว็บไซต์ tripadvisor [3] อย่างน้อย 1,000 รีวิว
2. ครอบคลุมเฉพาะข้อความหรือคำภาษาไทยที่สะกดถูกต้องตามไวยกรณ์ในภาษาไทยเท่านั้น โดย
3. ผลลัพธ์การแบ่งกลุ่มข้อความที่ได้จากโมเดลมีสามรูปแบบ คือ ข้อความที่แสดงความรู้สึกด้านบวก ข้อความที่แสดงความรู้สึกด้านลบ และข้อความที่เป็นกลางหรือไม่แสดงความรู้สึก
4. การพัฒนาโมเดลจะใช้ภาษาไพทอน 3 (Python 3)

วิธีการดำเนินงาน

1. ค้นหาและศึกษาบทความรวมถึงองค์ความรู้ที่เกี่ยวข้องกับโครงงาน
2. ศึกษาเครื่องมือ โปรแกรมและเทคนิคที่ใช้ในโครงงาน
3. กำหนดขอบเขตของโครงงานและขั้นตอนดำเนินงาน
4. รวบรวมข้อความภาษาไทย
5. วิเคราะห์และออกแบบวิธีการที่ใช้ในการวิเคราะห์และจำแนกข้อความ
6. พัฒนาโมเดลจำแนกข้อความภาษาไทย
7. ตรวจสอบความถูกต้องของโมเดลที่พัฒนาขึ้น
8. สรุปผลการดำเนินการ ข้อเสนอแนะและจัดทำเอกสาร

ขั้นตอนการดำเนินงาน	ปี 2561				ปี 2562		
	ก.ย.	ต.ค.	พ.ย.	ธ.ค.	ม.ค.	ก.พ.	มี.ค.
1.ค้นหาและศึกษาบทความรวมถึงเนื้อหาที่ เกี่ยวข้องกับโครงงาน							
2.ศึกษาเครื่องมือ โปรแกรมและเทคนิคที่ ใช้ในโครงงาน							
3. กำหนดขอบเขตของโครงงานและขั้นตอนดำเนินงาน							
4.รวบรวมข้อความภาษาไทย							

5.วิเคราะห์และออกแบบวิธีการที่ใช้ในการวิเคราะห์และจำแนกข้อความ							
6.พัฒนาโมเดลจำแนกข้อความภาษาไทย							
7.ตรวจสอบความถูกต้องของโมเดลที่พัฒนาขึ้น							
8.สรุปผลการดำเนินการ ข้อเสนอแนะ และ จัดทำเอกสาร							

ประโยชน์ที่คาดว่าจะได้รับ

ประโยชน์ต่อผู้จัดทำโครงงาน

- ได้ความรู้และความเข้าใจเกี่ยวกับการทำงานของการเรียนรู้ด้วยเครื่องในการวิเคราะห์ความรู้สึกจากข้อความและการจัดการข้อมูลที่อยู่ในรูปแบบตัวอักษร
- มีความเข้าใจและมีทักษะในการใช้ภาษาไพทอน (python) และเครื่องมือต่าง ๆ ที่ใช้ในการพัฒนา

ประโยชน์ต่อผู้นำไปใช้งาน

- มีโมเดลที่มีความแม่นยำในการวิเคราะห์ความรู้สึกจากข้อความภาษาไทย
- ช่วยอำนวยความสะดวกในการวิเคราะห์และสรุปผลข้อมูลเพื่อใช้ในการตัดสินใจได้

อุปกรณ์และเครื่องมือที่ใช้

1. ฮาร์ดแวร์

- เครื่องคอมพิวเตอร์แล็ปท็อป Operating System: Windows 10 Pro 64 bit

  System Manufacturer: Acer

  System Model: Aspire E5-571G

  Processor: intel Core i3-4005U (1.7 GHz,3MB L3 cache)

  Memory: 4096MB RAM

  HDD: 500 GB

2. ซอฟต์แวร์

- จูปิเตอร์โน้ตบุ๊ก (jupyter notebook) ใช้ในการเขียนโปรแกรม
- library deepcut ใช้ในการตัดคำหลังจากจัดเก็บในรูปแบบข้อความ
- Microsoft office 2016 ใช้ในการจัดทำเอกสาร
- AutoHotKey ใช้เป็นตัวช่วยในการจัดเก็บข้อมูล
- Notepad ใช้จัดเก็บข้อมูล

งบประมาณ

1. คีย์บอร์ดไร้สาย K520 + เมาส์ไร้สาย M310	ราคา	2,200	บาท
2. หูฟังไร้สาย Audio-Technica ATH-S200BT	ราคา	2,400	บาท
3. ค่าถ่ายเอกสารและทำรูปเล่ม	ราคา	400	บาท
รวม		5,000	บาท

หมายเหตุ ทั้งนี้งบประมาณที่ตั้งไว้ขอถัวเฉลี่ยทุกรายการ

เอกสารอ้างอิง

[1] Chunting Zhou, Chonglin Sun, Zhiyuan Liu and Francis C.M. Lau. (2015). A C-LSTM Neural Network for Text Classification. https://arxiv.org/pdf/1511.08630.pdf

[2] Xiang Zhang, Junbo Zhao and Yann LeCun. (2015). Character-level Convolutional Networks for Text Classification. Courant Institute of Mathematical Sciences, New York University. https://papers.nips.cc/paper/5782-character-level-convolutional-networks-for-text-classification.pdf.

[3] Tripadvisor. Retrived November 6, 2018, from https://th.tripadvisor.com

ภาคผนวก ข
ตัวอย่างโค้ดที่ใช้ในการพัฒนาโมเดล

1. ตัวอย่างโค้ดการโหลดไฟล์ข้อความเข้ามาในโปรแกรม

```
#โหลดไฟล์
import pandas as pd
category1="Hotel"
category2="Resturant"
category3="Travel"
category4="Airline"
#load data category 1
pp1=pd.read_csv(category1+'/textn.txt',error_bad_lines=True,names = ["text", "label"])
pn1=pd.read_csv(category1+'/textp.txt',error_bad_lines=True,names=["text","label"] )
pm1=pd.read_csv(category1+'/textme.txt',error_bad_lines=True,names=["text","labe l"])
# load data category 2
pp2=pd.read_csv(category2+'/textn.txt',error_bad_lines=True,names = ["text", "label"])
pn2=pd.read_csv(category2+'/textp.txt',error_bad_lines=True,names=["text","label"] )
pm2=pd.read_csv(category2+'/textme.txt',error_bad_lines=True,names=["text","labe l"])
# load data category 3
```

2. ตัวอย่างโค้ดที่ใช้ในการรวมข้อมูลและแบ่งชุดข้อมูลสอนและชุดข้อมูลตรวจสอบ

```
#mix  all data
df=pp1[:600]
df=df.append(pn1[:600],ignore_index=True)
df=df.append(pm1[:600],ignore_index=True)

df=df.append(pp2[:600],ignore_index=True)
df=df.append(pn2[:600],ignore_index=True)
df=df.append(pm2[:600],ignore_index=True)

df=df.append(pp3[:600],ignore_index=True)
df=df.append(pn3[:600],ignore_index=True)
df=df.append(pm3[:600],ignore_index=True)

df=df.append(pp4[:600],ignore_index=True)
df=df.append(pn4[:600],ignore_index=True)
df=df.append(pm4[:600],ignore_index=True)
#split data
from sklearn.model_selection import train_test_split
rawx_train, rawx_test, rawy_train, rawy_test = train_test_split(df['text'], df['label'],
```

3. ตัวอย่างโค้ดที่ใช้ในการตัดคำ

```
import deepcut
datatoken=[]



for index, rows in df.iterrows():
    row=deepcut.tokenize(str(rows['text']))
    row=' '.join(str(x) for x in row)
    df.loc[index, "text"]=row
```

4. ตัวอย่างโค้ดที่ใช้ในการจับคู่โทเคนและเติมเลข 0

```
from keras.preprocessing.text import Tokenizer
from keras.preprocessing.sequence import pad_sequences
t = Tokenizer()
# fit the tokenizer on the documents
t.fit_on_texts(rawx_train)
t.fit_on_texts(rawx_test)
# fill padding to 100
sequences = t.texts_to_sequences(rawx_train)
X_train = pad_sequences(sequences, maxlen= 100)
sequences = t.texts_to_sequences(rawx_test)
X_test = pad_sequences(sequences, maxlen= 100)
```

5. ตัวอย่างโค้ดที่ใช้ในการแปลงผลลัพธ์ของข้อมูลเป็นอาร์เรย์

```
from keras.utils import to_categorical
y_train = to_categorical(rawy_train)
y_test = to_categorical(rawy_test)
```

6. ตัวอย่างโค้ดที่ใช้ในการสร้างโมเดล

```
#Defind model
model5 = models.Sequential()
model5.add(Embedding(Vocab_size, Embed_size, input_length=100))
model5.add(Conv1D(32, 2, activation='relu'))
model5.add(Conv1D(64, 2, activation='relu'))
model5.add(Conv1D(128, 2, activation='relu'))
model5.add(Dense(3, activation='softmax'))
model5.compile(loss='categorical_crossentropy',optimizer='adam',metrics=['accurac
y'])
```

7. ตัวอย่างโค้ดในการสอนโมเดล

```
history10 = model5.fit(X_train, # Features
                       y_train, # Target vector
                       epochs=20, # Number of epochs
                       verbose=1, # Print description after each epoch
                       validation_data=(X_test, y_test),
                       callbacks=[cb]) # Data for evaluation
```

ประวัติผู้เขียน

Mr. Nitikorn Ongsirimongkol
นายนิติกร องค์ศิริมงคล
วัน เดือน ปีเกิด: 2 ตุลาคม 2539
สถานที่เกิด: จังหวัดนครปฐม
ชั้นปีที่ 4 คณะวิทยาศาสตร์
ภาควิชาคณิตศาสตร์และวิทยาการคอมพิวเตอร์
สาขาวิทยาการคอมพิวเตอร์ จุฬาลงกรณ์มหาวิทยาลัย
มือถือ: 064-8270258
อีเมล: niti2539@gmail.com