

Product classification by product details

- | | |
|----------------------------|-------------|
| 1. นายธันยบุรณ หิรัญดำรงค์ | 59070501039 |
| 2. นายนพชัย จินตามณีโชติ | 59070501042 |
| 3. นายอภิสิทธิ์ สมพันธ์ | 59070501081 |

Abstract

เป็นการทำ Text classification ที่สามารถทำนายหมวดหมู่ของสินค้า 12 หมวดหมู่จากรายละเอียดของสินค้าที่ดึงมาจากเว็บ Lazada โดยจะนำรายละเอียดของสินค้าทั้งหมด 29,110 รายการมาทำการ clean โดยการตัดตัวอักษรพิเศษหรือคำที่ไม่เกี่ยวข้องออกแล้วนำมาแบ่งคำ หลังจากที่ได้คำมาแล้วก็จะแปลงให้เป็นตัวเลข เพื่อนำไปใช้ในการเทรนโมเดลต่อไป โดยจะทดลองเทรนโดยใช้ Neuron Network 3 ตัวเพื่อหาตัวที่ให้ผลลัพธ์ที่ดีที่สุดซึ่งใช้ Neuron Network 3 ตัวดังต่อไปนี้ 1.LSTM (Long Shot Term Memory) 2.Bidirectional LSTM และ 3.CNN (Convolution Neuron Network) หลังจากที่ได้เทรนเรียบร้อยแล้ว จะทำการประเมินผลโมเดลด้วยค่าของ precision, recall และ F-1 score ซึ่งได้ผลลัพธ์ออกมาว่า CNN ได้คะแนนสูงที่สุดรองลงมาคือ LSTM แล้วน้อยที่สุดคือ Bidirectional LSTM

1. Introduction

Product classification by product details เป็นการทำ Text Classification โดยการนำรายละเอียดสินค้าที่ได้จากเว็บไซต์ Lazada มาทำนายว่ารายละเอียดสินค้าแบบนี้จะจัดให้อยู่ในหมวดหมู่ใดภายใน 12 หมวดหมู่ โดยใช้ LSTM, Bidirectional LSTM, .CNN ในการลองเทรนโมเดล โดยจะทำการดึงข้อมูลสินค้ามาจากเว็บ Lazada แล้วนำมาแบ่งแยกข้อมูลออกเป็นตัว features กับ labels เพื่อที่จะนำไปสร้างโมเดลสำหรับการจำแนกสินค้า จากนั้นทำการ clean รายละเอียดสินค้าในแต่ละ set หลังจากที่ได้ทำการ clean ข้อมูลเสร็จจะนำข้อมูลเหล่านั้นมาแบ่งและแปลงให้เป็น sequence ของตัวเลขเพื่อให้โมเดลสามารถเรียนรู้ข้อมูลที่เป็นข้อความได้

2. Theory

LSTM

เป็น Neuron Network ที่มี memory อยู่ภายใน ซึ่งภายใน memory จะมี gate คอยบอกว่า ควรจะ write, forget หรือ อนุญาตให้ read ได้ตอนไหนโดยจะมี cell state เป็นตัวเก็บ state ของ memory cell เอาไว้ โดยจะมีสถานะต่างๆดังนี้

1. Forget gate เป็น gate ที่คอยตัดสินใจว่าจะลบข้อมูลนี้หรือไม่โดยใช้ sigmoid function ในการตัดสินใจผลลัพธ์ที่ได้จะได้เป็น $[0,1]$ 0 หมายถึงไม่มีข้อมูลที่ไหลไปได้ 1 หมายถึงปล่อยให้ข้อมูลไหลไปได้ทั้งหมด

$$f_t = \sigma(w_f \cdot [h_{t-1}, x_t] + b_f)$$

ภาพที่ 1 Sigmoid function สำหรับ forget gate

2. Input gate เป็นตัวที่เอาไว้สำหรับการบันทึกข้อมูลลงไปในแต่ละ node (write)

$$i_t = \sigma(w_i \cdot [h_{t-1}, x_t] + b_i)$$

ภาพที่ 2 Sigmoid function สำหรับการ write

3. Update cell state ระหว่างที่ input gate จะรับข้อมูลมาจดจำใน node ตัวเองจะมีอีก node ไปเข้า tanh function เพื่อให้ได้ข้อมูลที่ใกล้เคียงกับข้อมูลขาออกจาก node นี้

$$C_t = \tanh(w_c \cdot [h_{t-1}, x_t] + b_c)$$

ภาพที่ 3 tanh function สำหรับการ อัปเดต cell state

4. Output gate ใช้สำหรับบอกว่าข้อมูลนี้พร้อมที่จะเป็นข้อมูลขาออกแล้วจะมี 2 ค่าคือ 1.ค่าที่ผ่านการ update จาก update cell state แล้วจะถูกส่งต่อทันทีไม่ผ่านฟังก์ชันใดๆ 2.ค่าของข้อมูลขาเข้าที่ถูกตัดแปลงผ่านการคำนวณใน output gate คำนี้นี้จะถูกส่งต่อไปเป็นข้อมูลขาเข้าของ node ถัดไป โดยใช้ sigmoid function

$$o_t = \sigma(w_o \cdot [h_{t-1}, x_t] + b_o)$$

ภาพที่ 4 sigmoid function สำหรับ output gate

หลังจากที่ผ่านการคำนวณค่า sigmoid แล้วจะถูกนำมา pointwise กับสถานะปัจจุบันของ node ซึ่งค่าสถานะจะถูกนำไปเข้า tanh function ก่อนแล้วค่อยนำผลลัพธ์นั้นมา pointwise กับค่าที่ได้จาก sigmoid function โดยผลลัพธ์จะถูกแบ่งเป็นสองข้อมูลคือผลลัพธ์ของ node นั้นๆ และข้อมูลที่จะส่งต่อไปเป็น input ของ node ต่อไป

$$h_t = o_t \cdot \tanh(C_t)$$

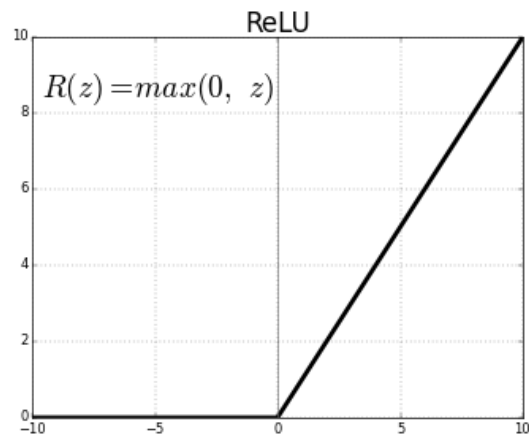
ภาพที่ 5 tanh function สำหรับนำไป pointwise กับค่าที่ได้จาก sigmoid function

ซึ่งจะใช้ LSTM มาทำ Text classification เนื่องจากว่าต้องการให้ข้อมูลมีการเรียนรู้แบบเป็นลำดับเช่น ใส่ข้อมูลว่า “หน้าจอ เฟอร์นิเจอร์และอุปกรณ์จัดเก็บอุปกรณ์สำหรับจัดเก็บกล่องและตะกร้าเก็บของตู้ลิ้นชัก 5 ชั้น ลิ้นชักพลาสติก ลิ้นชักเก็บของ ลิ้นชักใส่ของ (ถอดประกอบได้)” ก็จะได้ผลลัพธ์ออกมาว่า “บ้านและไลฟ์สไตล์” นอกจากนั้น LSTM ยังสามารถจดจำข้อมูลได้มากกว่า RNN ซึ่งนั่นเป็นข้อดีสำหรับการทำ Text classification ที่ต้องใช้ข้อมูลเยอะ

ReLU, Softmax (Activation Function)

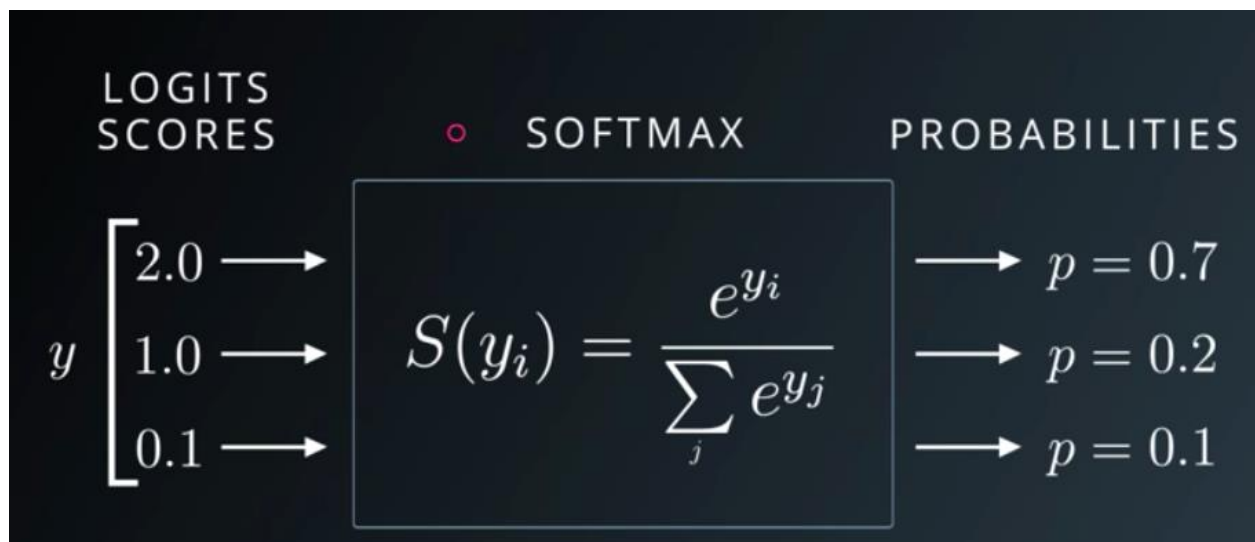
เป็นฟังก์ชันที่เอาไว้รับผลรวมการประมวลทั้งหมดจากทุก Input ภายใน 1 นิวรอนแล้วพิจารณาว่าจะส่งต่อเป็น Output เท่าไร โดยมีตัวอย่างของ activation function ดังนี้

1.ReLU (Rectified Linear Unit) เป็นฟังก์ชันที่ Slope จะมีค่าเท่ากับ 1 เสมอเมื่อ Input เป็นบวกทำให้ไม่เกิด Vanishing Gradient ส่งผลให้สามารถเทรนโมเดลได้เร็ว



ภาพที่ 6 กราฟของ ReLU function

2.Softmax เป็น activation function ที่นำข้อมูลประเภทตัวเลขมาเปลี่ยนเป็นค่าความน่าจะเป็น เหมาะสำหรับการทำ classification สำหรับ output layer



ภาพที่ 7 หลักการของ softmax function

Sparse categorical crossentropy (Loss function)

เป็น loss function ที่เอาไว้หา weight สำหรับข้อมูลที่เป็นแบบ single integer

Adam (Optimizer)

มีชื่อเต็มคือ Adaptive Moment Estimation เป็น optimizer ที่สามารถปรับค่า learning rate สำหรับพารามิเตอร์ในแต่ละครั้งได้ และยังสาสามารถลบปัญหาการ decaying learning rate ได้อีกด้วย

3. Experimental Design and Results

3.1 ชุดข้อมูล

ข้อมูลที่ใช้ในครั้งนี้ คือ ข้อมูลของสินค้าชนิดต่างๆ จากเว็บไซต์ Lazada โดยใช้ selenium ของ python ในการดึงข้อมูลตามหมวดหมู่ของสินค้าที่กำหนดไว้บนหน้าเว็บไซต์ทั้งหมด 12 หมวดหมู่ดังรูป ตัวอย่าง เช่น สินค้าในหมวดหมู่ อุปกรณ์ อิเล็กทรอนิกส์ นั้นจะทำการดึงสินค้าจาก หมวดโทรศัพท์ แท็บเล็ต แล็ปท็อป เป็นต้นซึ่งในแต่ละหมวดนั้นจะดึงมาอย่างละ 5 สินค้า ในกรณีที่สินค้านั้นมีหลายแบรนด์ ก็ จะทำการดึงสินค้าจากแต่ละแบรนด์มา 5 ชนิด

อุปกรณ์ อิเล็กทรอนิกส์ >	โทรศัพท์มือถือ
อุปกรณ์เสริม อิเล็กทรอนิกส์	แท็บเล็ต
ทีวีและเครื่องใช้ไฟฟ้าในบ้าน	แล็ปท็อป
สุขภาพและความงาม	คอมพิวเตอร์ตั้งโต๊ะ
เด็กอ่อน และของเล่น	กล้อง DSLR
ซูเปอร์มาร์เก็ต และสัตว์เลี้ยง	กล้องมิลเลอร์เลส
บ้านและไลฟ์สไตล์	กล้องคอมแพค
แฟชั่นผู้หญิง	กล้องอินสแตนท์
แฟชั่นผู้ชาย	กล้องแอ็คชั่นแคม
เครื่องประดับ	โดรน
กีฬาและ การเดินทาง	กล้องรักษาความปลอดภัย
ยานยนต์ และอุปกรณ์	เครื่องเล่นเกมคอนโซล

ภาพที่ 8 จำนวนหมวดหมู่สินค้าบนหน้าเว็บไซต์

โดยข้อมูลที่เก็บนั้นจะประกอบด้วยชื่อสินค้า หมวดหมู่ของสินค้า รายละเอียดของสินค้า และ ข้อมูลเฉพาะของสินค้า

คอมพิวเตอร์ & แล็ปท็อป > แล็ปท็อป > แล็ปท็อป > ACER SWIFT-SF314-56G-589T NOTEBOOK /I5-8265U/RAM 8 GB DDR4/HDD 1TB+16 GB OPTANE/MX:

ภาพที่ 9 ตัวอย่างหมวดหมู่ของสินค้า

รายละเอียดสินค้า ACER SWIFT-SF314-56G-589T NOTEBOOK /I5-8265U/RAM 8 GB DDR4/HDD 1TB+16 GB OPTANE/MX250 2 GB/N...

- ACER SWIFT มาพร้อมกับการออกแบบที่เพรียวบางและเบา เพราะตัวเครื่องทำจากอะลูมิเนียมสีเงิน ที่ให้ความรู้สึกเย็นเมื่อสัมผัส และดูทันสมัย
- มีความเพรียวบาง และน้ำหนักเบาเพียง 1.5 กก. ทำให้ง่ายต่อการพกพา
- กราฟิก NVIDIA GeForce MX250 และจอภาพขนาด 14 inch Full HD ที่ให้ความคมชัดในการรับชมวิดีโอ
- หน่วยประมวลผล Intel Core i5-8265U ความจุ 8 GB DDR4 และฮาร์ดดิสก์ 1 TB + 16 GB OPTANE
- ให้สัญญาณด้วยระบบไร้สาย 802.11ac Wi-Fi + Bluetooth 5.0 ที่แรงสม่ำเสมอ

ภาพที่ 10 ตัวอย่างรายละเอียดของสินค้า

ข้อมูลเฉพาะของ ACER SWIFT-SF314-56G-589T NOTEBOOK /I5-8265U/RAM 8 GB DDR4/HDD 1TB+16 GB OPTANE/MX250 2 GB/NO ...

แบรนด์	SKU
No Brand	462982418_TH-850898986
ประเภทของการรับประกัน	ขนาดหน้าจอ
การรับประกันจากโรงงานในประเทศ	14
รุ่น	ประเภทหน่วยประมวลผล
ACER SWIFT-SF314-56G-589T	Intel Core i5-8265U
ระยะเวลาการรับประกัน	
2 ปี	

ภาพที่ 11 ตัวอย่างข้อมูลเฉพาะของสินค้า

จากนั้นทำการเก็บข้อมูลที่ได้ลงบน excel ซึ่งสินค้าทั้งหมดที่เก็บรวบรวมมามีทั้งสิ้น 29,110 รายการ

id	category	amount
0	อุปกรณ์อิเล็กทรอนิกส์	826
1	อุปกรณ์เสริม อิเล็กทรอนิกส์	2345
2	ทีวีและเครื่องใช้ไฟฟ้าในบ้าน	2370
3	สุขภาพและความงาม	2665
4	เด็กอ่อน และของเล่น	3853
5	ซูเปอร์มาร์เก็ต และสัตว์เลี้ยง	2335
6	บ้านและไลฟ์สไตล์	3897
7	แฟชั่นผู้หญิง	2133
8	แฟชั่นผู้ชาย	1860
9	เครื่องประดับ	819
10	กีฬาและ การเดินทาง	2875
11	ยานยนต์ และอุปกรณ์	3132
	ทั้งหมด	29110

ภาพที่ 12 จำนวนของข้อมูลสินค้าในแต่ละหมวดหมู่

0	คอมพิวเตอร์ & แล็ปท็อปแล็ปท็อปแล็ปท็อป	ACER SWIFT-SF314-56G-589T SILVERCPUINTEL CORE i5-8265URAM8 GB DDR4STORAGE1 TB + 16 GBDISP
0	คอมพิวเตอร์ & แล็ปท็อปแล็ปท็อปแล็ปท็อป	Acer Notebook Nitro 5 AN515-52-53TU/T012(NH.Q3MST.012) i5-8300H/8GB/SSD512GB/GTX 1050 4GB/
0	คอมพิวเตอร์ & แล็ปท็อปแล็ปท็อปแล็ปท็อป	Acer Core i5 GEN 5 Ram 8 GB HDD 1000 GB การ์ดจอแยก NVIDIA สเปคคนทำงาน หรือเล่นเกม สบายๆAcer
0	คอมพิวเตอร์ & แล็ปท็อปแล็ปท็อปแล็ปท็อป	ACER A315-55G-540P :Ci5-8265U/8GB/1TB/GF MX230 2GB/15.6/Win10 :2YProcessor : Intel Core i5-8265
0	คอมพิวเตอร์ & แล็ปท็อปแล็ปท็อปแล็ปท็อป	Notebook Acer A315-53G-59ZGการรับประกันสินค้า2 YearsบริการหลังการขายAcer Service Center : 0-268
0	คอมพิวเตอร์ & แล็ปท็อปแล็ปท็อปแล็ปท็อป	โน้ตบุ๊ก HP Probook 450G1 - Ram 4GB HDD 320GB สภาพดี!! มีประกัน!! [พร้อมเคอร์สินค้าพร้อมจัดส่ง 29 ๗

ภาพที่ 13 ตัวอย่างข้อมูลที่ได้ใน excel

3.2 การเตรียมและประมวลผลข้อมูล

- Import data

นำข้อมูลที่ import เข้ามาจากไฟล์ Excel ที่อยู่คนละชีทกัน นำมารวมกันแล้วเก็บไว้ใน dataframe อันเดียว โดยมีข้อมูลอยู่ 2 columns คือ รายละเอียดสินค้า และ ประเภทของรายละเอียด ทั้งหมดจำนวน 29,110 rows

Category		Description
0	0	SmartphonesSAMSUNG Galaxy A20s *ประกันศูนย์ 1ป...
1	0	SmartphonesSamsung A30 Ram4/64GB (เครื่องศูนย์...
2	0	SmartphonesSamsung Galaxy A10 Ram2/32GB จอ 6.2...
3	0	SmartphonesSamsung Galaxy M30s (4/64GB) (โทรศ...
4	0	SmartphonesSamsung Galaxy A10 Ram2/32GB จอ 6.2...
...
29105	11	ยานยนต์และอุปกรณ์โมเตอร์ไซด์เครื่องแต่งกายและห...
29106	11	ยานยนต์และอุปกรณ์โมเตอร์ไซด์เครื่องแต่งกายและห...
29107	11	ยานยนต์และอุปกรณ์โมเตอร์ไซด์เครื่องแต่งกายและห...
29108	11	ยานยนต์และอุปกรณ์โมเตอร์ไซด์เครื่องแต่งกายและห...
29109	11	ยานยนต์และอุปกรณ์โมเตอร์ไซด์เครื่องแต่งกายและห...

29110 rows × 2 columns

ภาพที่ 11 dataframe ของข้อมูลที่ทำกร import จาก Excel แล้วนำมารวมกัน ประกอบด้วย 2 columns คือ Category และ Description มีขนาด 29110 rows x 2 columns

จากนั้นทำการแบ่งแยกข้อมูลออกเป็นตัว features กับ labels เพื่อที่จะนำไปสร้างโมเดลสำหรับการจำแนกสินค้า โดย Description หรือ รายละเอียดสินค้า ให้เป็น features ที่มี Category เป็น labels บอกว่ารายละเอียดสินค้านี้อยู่ในหมวดหมู่สินค้าไหน จำนวนหมวดหมู่ทั้งหมด 12 หมวดหมู่

id	category
0	อุปกรณ์อิเล็กทรอนิกส์
1	อุปกรณ์เสริม อิเล็กทรอนิกส์
2	ทีวีและเครื่องใช้ไฟฟ้าในบ้าน
3	สุขภาพและความงาม
4	เด็กอ่อน และของเล่น
5	ซูเปอร์มาร์เก็ต และสัตว์เลี้ยง
6	บ้านและไลฟ์สไตล์
7	แฟชั่นผู้หญิง
8	แฟชั่นผู้ชาย
9	เครื่องประดับ
10	กีฬาและ การเดินทาง
11	ยานยนต์ และอุปกรณ์

ภาพที่ 12 จำนวนหมวดหมู่สินค้าทั้งหมด 12 หมวดหมู่ที่ใช้จำแนกสินค้า

แบ่งออกเป็น train set และ test set เพื่อให้โมเดลที่สร้างขึ้นทำการเรียนรู้จาก train set เท่านั้น แล้วใช้ test set เพื่อทำการประเมินผลที่ได้ โดยแบ่ง test set ให้มีขนาด 0.2 เท่าของข้อมูลที่มีทั้งหมด และทำการแบ่งข้อมูลอีกครั้งจาก train set ที่ได้มา ให้แบ่งเป็น train set และ validation set อีกต่อหนึ่ง โดย validation set นี้ใช้ตรวจสอบโมเดลในแต่ละ epoch ขณะทำการ train โมเดล

```
train size = 20959
test size = 2911
validation size = 5240
```

ภาพที่ 13 ขนาดของข้อมูลใน train, test และ validation set

- Cleaning data

เมื่อทำการแบ่งข้อมูลเรียบร้อยแล้ว ขั้นตอนต่อไปทำการ clean รายละเอียดสินค้าในแต่ละ set โดยจะทำการลบพวกอักขระพิเศษและ stop words (import จาก nltk) ทั้งในภาษาไทยและภาษาอังกฤษ ออก แล้วเลือกเก็บไว้แต่ตัวอักษรภาษาอังกฤษทั้งตัวพิมพ์เล็กและใหญ่ ตัวเลข และตัวอักษรภาษาไทย

- Split text into word

เมื่อผ่านการ clean รายละเอียดสินค้าแล้ว ก็จะนำข้อมูลรายละเอียดสินค้าที่ยังเป็นข้อความยาวอยู่มาแบ่งออกเป็นหน่วยย่อยในที่นี้แบ่งออกเป็นคำโดยใช้ library PythaiNLP มาทำการแบ่งคำภาษาไทยด้วย newmm โดยจะยังไม่แบ่งในขั้นตอนนี้ แต่จะทำการ mark จุดแบ่งด้วยสัญลักษณ์ “|” ไว้ก่อนด้วย และทำการแปลงรูปของคำในภาษาไทยให้เป็นคำพื้นฐานด้วยโดย import wordnet synsets มาใช้เทียบ

```
'เสื้อ|แจ๊คเก็ต|มอเตอร์ไซค์',
'ความสูง|ด้านบน|สันเท้า|เหนือ|สันเท้า',
'จอ|ole|สัมผัส|สี|ฟ้า|คมชัด',
'ฟัง|คั่น|คุณสมบัติ|น. |วัน',
'ชนิด|การรับประกัน|ไม่มี|ประกัน',
'อุปกรณ์|รถ',
```

ภาพที่ 14 ตัวอย่างการแบ่งคำด้วย “|”

- Convert text to sequence

ขั้นตอนต่อมาคือการแปลงข้อความให้เป็น sequence ของตัวเลขเพื่อให้โมเดลสามารถการเรียนรู้ข้อมูลที่เป็นข้อความได้ โดยจะใช้ Tokenizer ใน keras ในการแปลง เริ่มจากกำหนดค่า parameter สำหรับ Tokenizer และสำหรับ input ใน Embedding layer

```
max_word = 10000
max_len = 150
embedding_size = 128
```

ภาพที่ 15 กำหนดค่า parameter สำหรับ Tokenizer และการ embedding word

โดยค่าของ max_word คือ จำนวนคำที่มากที่สุดที่จะให้โมเดลเรียนรู้ มาจากความถี่ของคำ และ max_len คือ ความยาวของ input ใน embedding layer ว่าจะให้ input แต่ละตัวมีความยาวเท่าไร ส่วน embedding_size คือ ขนาดของ output ที่ทำการ embedding แล้ว เมื่อกำหนด Tokenizer เรียบร้อยแล้วก็ทำการ fit_on_text กับข้อมูล train set ด้วย Tokenizer จะได้ออกมาเป็นคลังคำศัพท์ที่ map ระหว่าง คำศัพท์ กับ ตัวเลขที่เป็น index แล้วจากนั้นทำการแปลงข้อความทั้งหมดให้เป็น sequence ของตัวเลข ด้วยคลังคำศัพท์ที่สร้างขึ้นมา แต่ว่า sequence ของตัวเลขที่ได้มามีขนาดที่ไม่เท่ากันอยู่ จะต้องทำการ padding ให้มีขนาดเท่ากับ max_len ที่กำหนดไว้ โดยตัวที่สั้นกว่า max_len จะทำการเติม 0 จนกว่าขนาดจะเท่าที่กำหนด ส่วนตัวที่ยาวกว่าจะทำการตัดให้เท่ากับ max_len ก็จะได้ sequence ของตัวเลขที่มีขนาดเท่ากันทั้งหมดพร้อมที่จะนำไปเข้าสู่โมเดลแล้ว

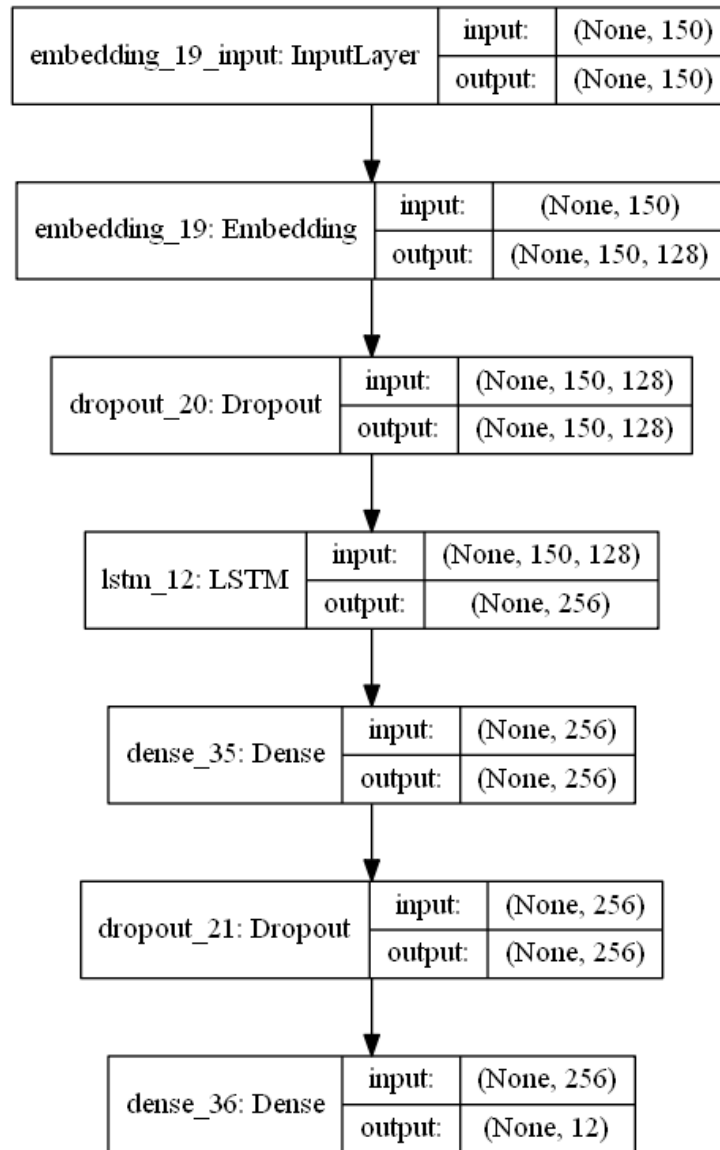
```
array([ 0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,
        0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,
        0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,
        0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,
        0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,
        12, 21, 31, 12, 148, 511, 5176, 836, 2103, 115, 22,
        30, 88, 39, 113, 2266, 76, 122, 53, 469, 809, 322,
        1280, 193, 3, 900, 2, 1, 8, 10, 22, 23, 12,
        5611])
```

ภาพที่ 16 ตัวอย่างของ sequence ของตัวเลขที่สร้างจากคลังคำศัพท์และผ่านการ padding เรียบร้อยแล้วที่ขนาด max_len = 100

3.3 การสร้างโมเดล

เป้าหมายของเราคือการสร้างโมเดลที่สามารถใช้ในการจำแนกหมวดหมู่ของสินค้าได้จาก รายละเอียดสินค้า ซึ่งรายละเอียดสินค้าจะมีลักษณะเป็นข้อความยาวหลายข้อความ ตัวโมเดลต้องทำการ เรียนรู้ข้อความนั้นแล้วบอกได้ว่าอยู่หมวดหมู่ไหน ซึ่งเราได้ทำการสร้างโมเดลต่างๆ ดังนี้

- LSTM (Long Short-Term Memory)

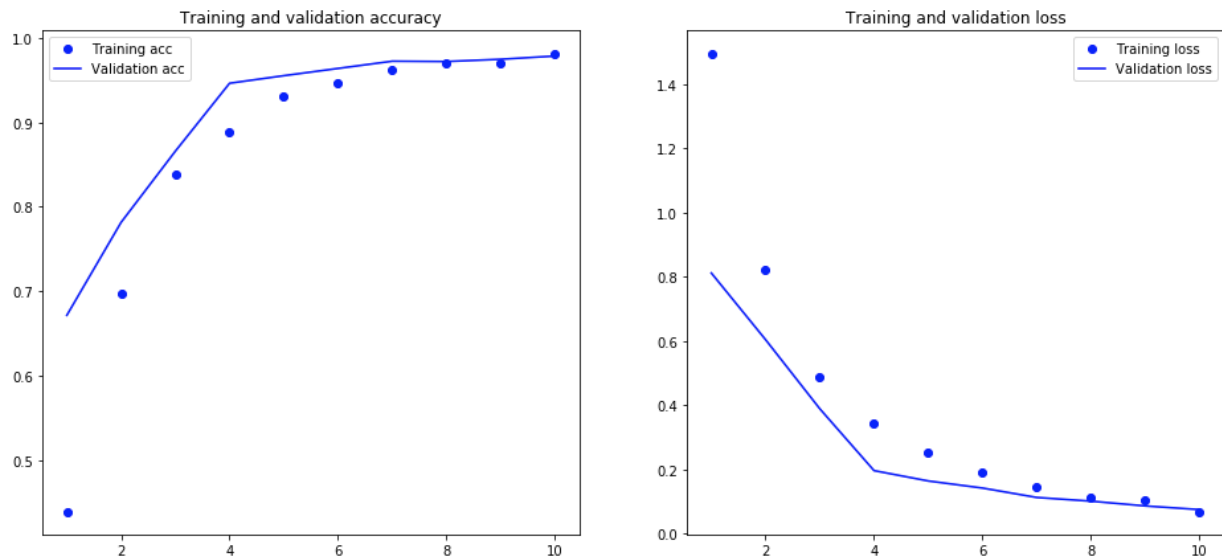


ภาพที่ 17 โครงสร้างโมเดลการจำแนกสินค้าด้วย LSTM

โดยเริ่มจาก Input layer ที่เป็น sequence ของตัวเลขที่มีความยาว sequence ละ 150 เข้าไปสู่ Embedding layer เพื่อสร้าง vector ที่มีขนาดเท่ากันขึ้นที่ขนาด (64,150,128) ผ่านการ Dropout เพื่อป้องกันการ overfit ที่อัตรา 0.2 แล้ว vector นั้นก็จะผ่านเข้าสู่ชั้นของ LSTM ที่มี 256 units ทำการเรียนรู้ vector ของคำที่แปลงมาจากรายละเอียดสินค้า แล้วส่งผ่าน output ไปที่ hidden layer อีกชั้นหนึ่งที่มี activation function = relu แล้วทำการ Dropout อีก 0.5 ก่อนที่จะไปยัง layer สุดท้าย หรือ output layer ที่มี activation function = softmax ที่มี 12 nodes สำหรับ 12 หมวดหมู่สินค้า

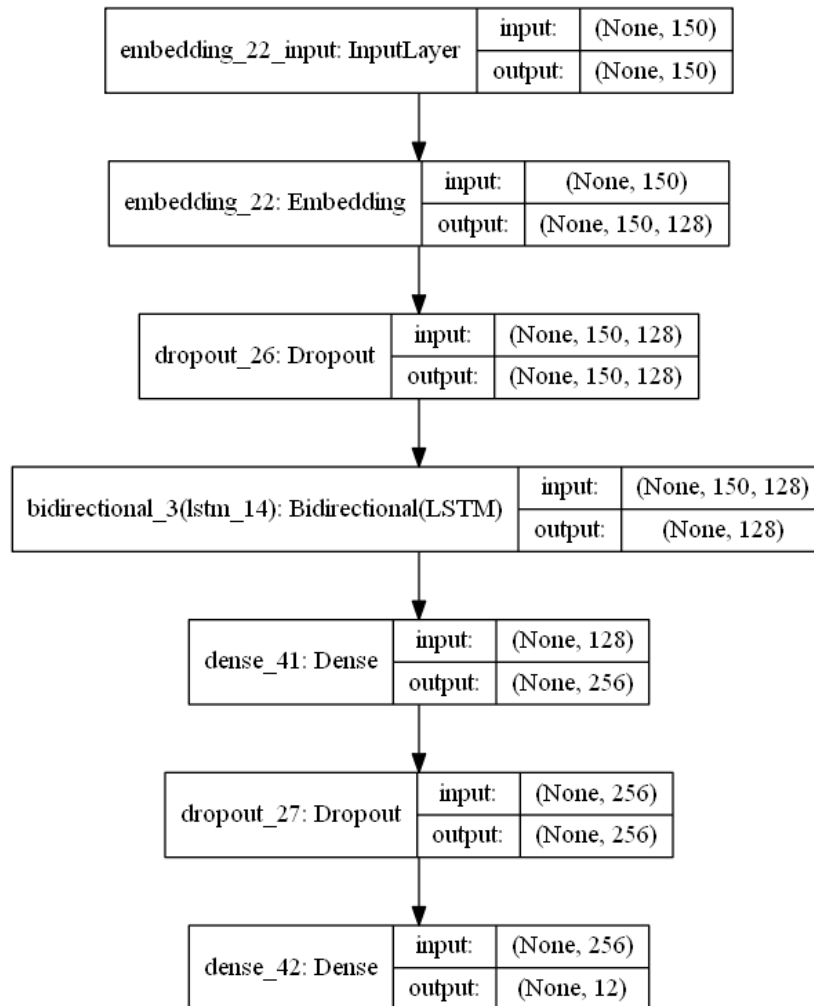
ซึ่งโมเดล LSTM นี้ในขั้นตอนการเทรนกำหนดค่า parameter ต่างๆ คือ batch size = 64, epoch = 10, optimizer = adam และ loss = sparse_categorical_crossentropy ซึ่งเป็น loss function ที่ไม่ต้องแปลง labels ให้กลายเป็น one-hot encoding สามารถใช้เป็นตัวเลขได้เลย

ผลลัพธ์จากการเทรนได้ค่า accuracy = 0.9818, loss = 0.0693 และ validation accuracy = 0.9786, validation loss = 0.075 ซึ่งเป็นค่าที่สูงมาก โดยในแต่ละ epoch ใช้เวลาประมาณ 80 วินาที



ภาพที่ 18 กราฟแสดงค่า accuracy และ loss ในแต่ละ epoch ของการเทรนโมเดล LSTM

- Bidirectional LSTM

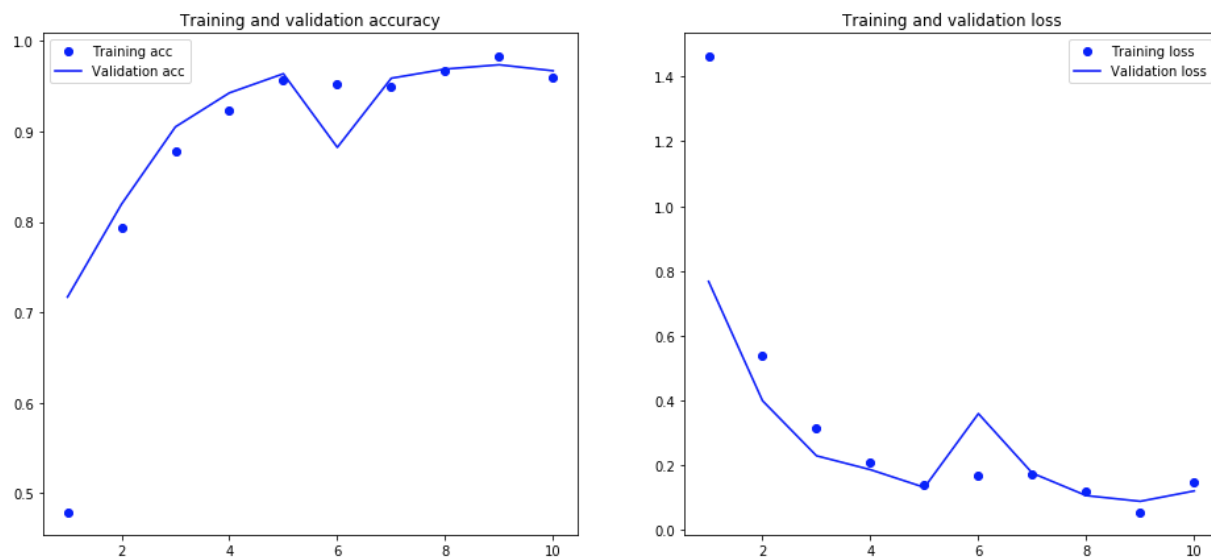


ภาพที่ 19 โครงสร้างของโมเดลที่ใช้ Bidirectional LSTM

เริ่มจาก Input layer มี input เข้ามาเป็น sequence ของตัวเลขที่มีความยาว 150 แล้วไปที่ Embedding layer เพื่อแปลง sequence ของตัวเลขนั้นให้กลายเป็น vector แล้วทำการ Dropout ที่ 0.2 ก่อนแล้วจึงส่งไปยัง layer ที่เป็น Bidirectional LSTM ที่มี 64 units หลังจากได้ output จาก Bidirectional LSTM แล้วส่งต่อไปที่ hidden layer ที่มี activation = relu แล้วผ่านการ Dropout อีก 0.5 ก่อนที่จะไปยัง output layer สุดท้ายที่มี activation function = softmax ที่มี 12 nodes สำหรับ 12 หมวดหมู่สินค้า

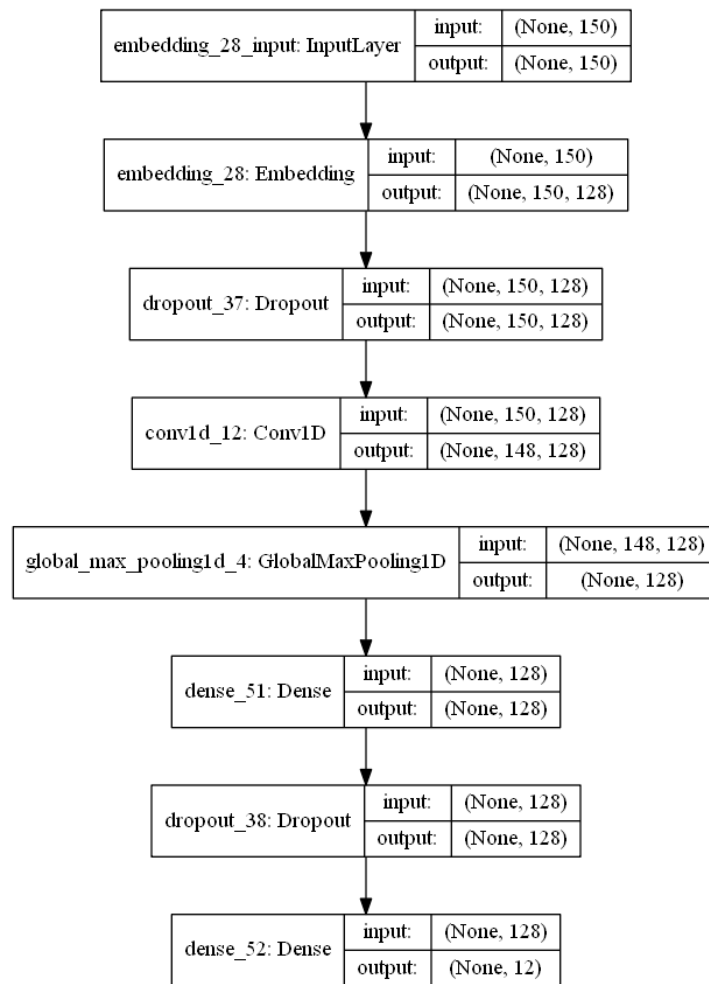
โดยในการเทรนโมเดลนี้ได้กำหนด batch size = 64, epoch = 10, optimizer = adam และ loss = sparse_categorical_crossentropy

ผลลัพธ์ที่ได้จากการเทรน accuracy = 0.9708, loss = 0.1165 และ validation accuracy = 0.9519, validation loss = 0.1780 ซึ่งค่าที่ได้น้อยกว่า โมเดล LSTM โดยในแต่ละ epoch ใช้เวลาประมาณ 127 วินาที



ภาพที่ 20 กราฟแสดงค่า accuracy และ loss ในแต่ละ epoch การเทรนโมเดล Bidirectional LSTM

- CNN (Convolution Neural Network)

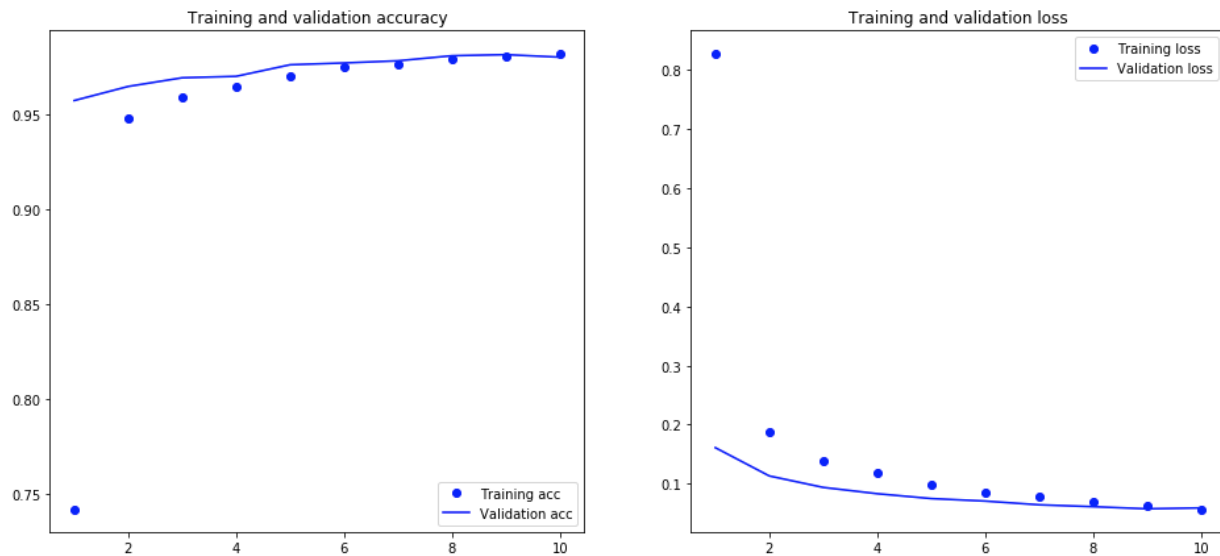


ภาพที่ 21 โครงสร้างของโมเดลที่ใช้ CNN ในการเรียนรู้ข้อมูลสินค้า

โมเดลนี้ใช้ CNN ในการเรียนรู้รูปแบบที่อยู่ภายในรายละเอียดสินค้า ต่างจาก LSTM ที่เรียนรู้แบบเป็นลำดับ โดยโครงสร้างโมเดลเริ่มจาก Input layer ที่มี input เป็น sequence ของตัวเลขความยาว 150 เข้ามา ทำการเปลี่ยนให้เป็น vector ที่ Embedding layer จากนั้นทำการ Dropout 0.2 ก่อนที่จะนำไปเข้าสู่ Conv1D layer ที่ใช้ activation = relu เพื่อทำ convolution โดยกำหนดขนาด kernel_size = 2 และ filters = 100 หลังจากนั้นทำ GlobalMaxPooling1D เหลือข้อมูลเป็น 1 มิติแล้วนำไปผ่าน hidden layer ที่มี activation function = relu จากนั้น Dropout ที่ 0.5 ก่อนที่จะไปยัง output layer ที่มี activation function = softmax ที่มี 12 nodes สำหรับ 12 หมวดหมู่สินค้า

โดยในการเทรนโมเดล CNN นี้ได้กำหนด batch size = 50, epoch = 10, optimizer = adadelta และ loss = sparse_categorical_crossentropy

ผลลัพธ์ที่ได้จากการเทรนนั้นมีค่า loss: 0.0571 - acc: 0.9824 - val_loss: 0.0591 - val_acc: 0.9803
ซึ่งเป็นค่าที่สูงมากและใช้เวลาในการเทรนที่น้อยมากเมื่อเทียบกับ LSTM และ Bidirectional LSTM โดย
ในแต่ละ epoch ใช้เวลาประมาณ 3 วินาที



ภาพที่ 22 กราฟแสดงค่าของ accuracy และ loss ของการเทรนโมเดล CNN ในแต่ละ epoch

3.4 การประเมินผลโมเดล

- การประเมินผลด้วยข้อมูล test set

หลังจากที่เทรนโมเดลทั้งหมดเรียบร้อยแล้วนั้น ก็จะทำการประเมินผลโมเดลด้วยค่าของ precision, recall และ F-1 score แล้วนำมาเปรียบเทียบกัน โดยข้อมูลที่ใช้ในการประเมินคือ ข้อมูลชุด test set ที่ถูกแบ่งเอาไว้ตั้งแต่ตอนเตรียมข้อมูล เป็นข้อมูลที่ไม่ได้ผ่านการเทรนเลยหรือก็คือโมเดลยังไม่เคยเห็น โดยจะได้ค่าของแต่ละโมเดลดังนี้

	Precision	Recall	F1-score
LSTM	0.98	0.98	0.98
Bidirectional LSTM	0.97	0.96	0.97
CNN	0.98	0.98	0.98

ตารางที่ 1 แสดงค่า Precision, Recall, F1-score ของ โมเดล LSTM, Bidirectional LSTM, CNN สำหรับการจำแนกหมวดหมู่สินค้าด้วยรายละเอียดสินค้าในชุดข้อมูล test set

พบว่าค่าของโมเดลทั้งสามมีค่าสูงทั้งหมดจากการทดสอบโดย test set ซึ่งโมเดล CNN มีค่ามากที่สุด เท่ากับ LSTM ส่วนที่น้อยสุดคือ Bidirectional LSTM

3.5 ทดลองใช้งานโมเดล

ลองนำโมเดลที่ได้มาทดสอบการจำแนกสินค้าจากรายละเอียดสินค้าที่กำหนด

รายละเอียดสินค้า: ช่วยเพิ่มความชื้นในอากาศทำให้ห้องสดชื่นเปลี่ยนสีได้ขนาดเล็กเคลื่อนย้ายง่ายใช้งานง่ายใช้เสียบ USB จุน้ำได้ 270 มล ทำงานได้ 69 ชั่วโมงในพื้นที่ 30 ตรม เหมาะกับผู้ที่นั่งในห้องแอร์เป็นเวลานานจะช่วยให้ผิวชุ่มชื้น

หมวดหมู่: โทรทัศน์และเครื่องใช้ไฟฟ้าในบ้าน

การจำแนกของแต่ละโมเดล:

LSTM: สุขภาพและความงาม

Bidirectional LSTM: โทรทัศน์และเครื่องใช้ไฟฟ้าในบ้าน

CNN: โทรทัศน์และเครื่องใช้ไฟฟ้าในบ้าน

จะพบว่าตัวโมเดลที่มีค่าจากการประเมินผลที่สูงก็อาจจะมีการทำนายที่ผิดพลาดได้ดังตัวอย่าง ซึ่งก็ขึ้นอยู่กับข้อมูลที่นำไปเทรนโมเดล และข้อมูลที่นำมาทดสอบด้วย

4. Conclusion

เราได้ทำการสร้างโมเดลสำหรับใช้ในการจำแนกหมวดหมู่ของสินค้าจากรายละเอียดสินค้าด้วย LSTM และ Bidirectional LSTM ที่เป็น recurrent neural network และ โมเดลที่เป็น CNN ในการเรียนรู้ข้อมูลรายละเอียดสินค้าเพื่อใช้จำแนกสินค้า หลังจากทำการเทรนโมเดลทั้งสามแล้ว ผลลัพธ์ที่ได้แสดงให้เห็นว่าโมเดลทั้งสามมีความแม่นยำในการจำแนกสินค้าจากการประเมินผลด้วยข้อมูลชุด test โดยที่แม่นยำที่สุดคือ CNN ซึ่งเท่ากับ LSTM และ Bidirectional LSTM มีค่าน้อยที่สุด

จากการเรียนรู้ที่แตกต่างกันของ RNN และ CNN ตรงที่ RNN จะทำการเรียนรู้ข้อมูลแบบเป็นลำดับ ส่วน CNN จะเรียนรู้หารูปแบบที่อยู่ในข้อมูล และจากผลลัพธ์ที่ได้น่าจะเป็นเพราะข้อมูลที่รวบรวมมาในแต่ละหมวดหมู่มีรูปแบบที่ปรากฏชัดเจนในแต่ละหมวดหมู่ทำให้โมเดล CNN นั้นมีค่า F1-score ที่สูงกว่าแบบ RNN เพราะเรียนรู้รูปแบบที่มีในข้อความของแต่ละหมวดหมู่ ไม่ได้เรียนรู้แบบเป็นลำดับ ซึ่งโมเดลประเภท CNN อาจมีความเหมาะสมในงานที่เป็นประเภท classification ใน text มากกว่า RNN

5. References

- [1] T.Sirinart (2017) “Long Short-Term Memory (LSTM)” สืบค้นเมื่อวันที่ 24/11/2019: <https://medium.com/@sinart.t/long-short-term-memory-lstm-e6cb23b494c6>
- [2] Doratong24 (2017) “[ML] LSTMs แบบ (เกือบ) ละเอียดยิบ ๆ” สืบค้นเมื่อวันที่ 24/11/2019: <https://medium.com/@tongkornkitt/ml-lstms-แบบ-เกือบ-ละเอียดยิบ-ๆ-a3a55cd37883>
- [3] Avinash Sharma (2017) “Understanding Activation Functions in Neural Networks” สืบค้นเมื่อวันที่ 25/11/2019: <https://medium.com/the-theory-of-everything/understanding-activation-functions-in-neural-networks-9491262884e0>
- [4] Keng Surapong (2019) “Activation Function คืออะไร ใน Artificial Neural Network, Sigmoid Function คืออะไร – Activation Function ep.1” สืบค้นเมื่อวันที่ 25/11/2019: <https://www.bualabs.com/archives/1261/what-is-activation-function-what-is-sigmoid-function-activation-function-ep-1/>
- [5] Keng Surapong (2019) “Activation Function คืออะไร ใน Artificial Neural Network, Sigmoid Function คืออะไร – Activation Function ep.1” สืบค้นเมื่อวันที่ 25/11/2019: <https://www.bualabs.com/archives/1355/what-is-relu-function-why-popular-deep-learning-training-deep-neural-network-activation-function-ep-3/>
- [6] Uniqtech (2018) “Understand the Softmax Function in Minutes” สืบค้นเมื่อวันที่ 25/11/2019: <https://medium.com/data-science-bootcamp/understand-the-softmax-function-in-minutes-f3a59641e86d>
- [7] Mr.P L (2019) “Deep Learning แบบฉบับสามัญชน EP 2” สืบค้นเมื่อวันที่ 25/11/2019: <https://medium.com/mmp-li/deep-learning-แบบฉบับสามัญชน-ep-2-optimization-activation-function-เรียนกันสบายๆสัปดาห์ที่9feb5a87e3b2>
- [8] Vitaly Bushaev (2018) “Adam — latest trends in deep learning optimization.” สืบค้นเมื่อวันที่ 25/11/2019: <https://towardsdatascience.com/adam-latest-trends-in-deep-learning-optimization-6be9a291375c>
- [9] Susan Li (2019) “Multi-Class Text Classification with LSTM.” สืบค้นเมื่อวันที่ 26/11/2019 : <https://towardsdatascience.com/multi-class-text-classification-with-lstm-1590bee1bd17>
- [10] “Report on Text Classification using CNN, RNN & HAN” สืบค้นเมื่อวันที่ 26/11/2019 : <https://medium.com/jatana/report-on-text-classification-using-cnn-rnn-han-f0e887214d5f>