

การจำแนกเจตนาการสนทนาเพื่อการสร้างแชทบอท

Intent Classification for Building Chatbot

อาจารย์ที่ปรึกษา
อาจารย์ ดร.นิวอรณ วัฒนกิจรุ่งโรจน์

จัดทำโดย
นายณัฐพล เดชประมวลพลด รหัสนักศึกษา 6110210129
นางสาววริศรา พิสุทธิ์เรียม รหัสนักศึกษา 6110210373



หัวข้อที่นำเสนอ

- ที่มาและความสำคัญ 3
- วัตถุประสงค์ 4
- ขอบเขตของโครงการ 5
- การทดลองและผลการทดลอง 6

ที่มาและความสำคัญ

ในการตอบข้อความของลูกค้าผ่านทางออนไลน์ จำเป็นจะต้องมีเจ้าหน้าที่ในการให้บริการ ซึ่งในปัจจุบันได้มีเทคโนโลยีทางคอมพิวเตอร์ที่เรียกว่า แชทบอท (chatbot) เข้ามาช่วย แต่ความท้าทายของการสร้างแชทบอท คือ “การจำแนกเจตนาจากการสนทนา” หรือ “intent” ซึ่งมีความหลากหลายอย่างมาก

ทางคณะผู้จัดทำจึงมีความสนใจศึกษาการนำการเรียนรู้ของเครื่อง (Machine Learning) มาช่วยในการจำแนก intent เพื่อสร้างแชทบอท

วัตถุประสงค์

1. เพื่อคึกคักจำแนก intent โดยใช้การเรียนรู้ของเครื่องในการสร้างแซกบอท
2. เพื่อสร้างแซกบอทตอบคำถามข้อมูลเบื้องต้นของสถานศึกษา ซึ่งจะช่วยตอบคำถามให้แก่ผู้ใช้ นอกช่วงเวลาทำการ



ขอบเขตของโครงงาน

ศึกษาการจำแนก intent โดยใช้การเรียนรู้ของเครื่องอย่างน้อย 2 เทคนิค บนชุดข้อมูลมาตรฐาน และกรณีศึกษา Zachbok ตอบคำถามข้อมูลเบื้องต้นเกี่ยวกับสถานศึกษา โดยใช้ภาษา Python เป็นหลัก

การทดลองและการทดลอง

ชุดข้อมูลที่ใช้

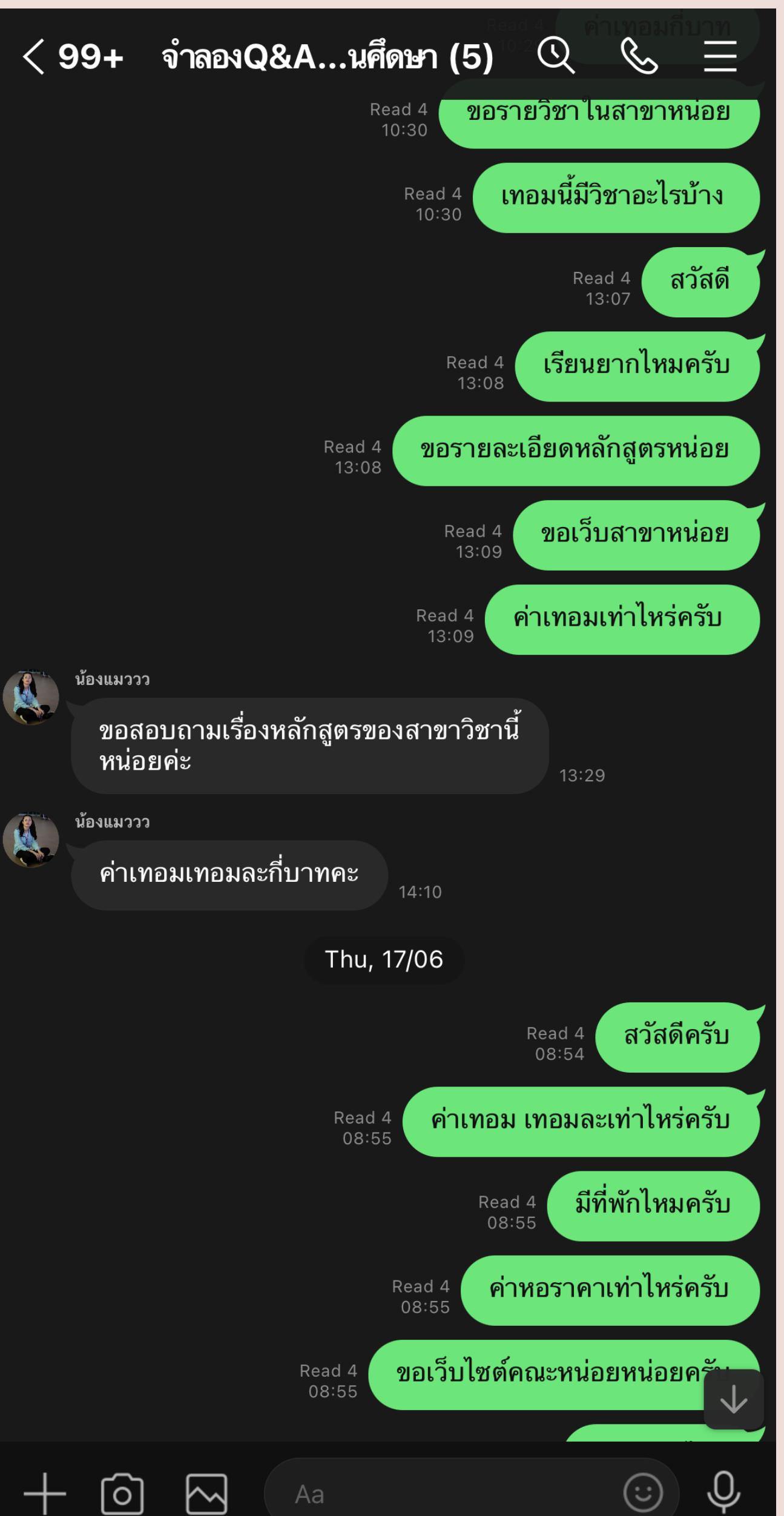
ชุดข้อมูลมาตรฐาน

1. ชุดข้อมูล ATIS หรือ Airline Travel Information System เป็นชุดข้อมูลมาตรฐานสำหรับสร้าง chatbot ที่ใช้กันอย่างแพร่หลาย มีจำนวนข้อมูล **4978** ตัว
2. ชุดข้อมูล Corona Dataset เป็นชุดข้อมูลที่ใช้สำหรับสร้าง chatbot ตอบคำถามเกี่ยวกับ corona virus มีจำนวนข้อมูล **1053** ตัว
3. ชุดข้อมูล Case Routing Intent เป็นชุดข้อมูลการถามตอบเกี่ยวกับการซื้อสินค้า มีจำนวนข้อมูล **150** ตัว

ชุดข้อมูลที่ใช้

ชุดข้อมูลที่สร้างขึ้น

เป็นชุดข้อมูลจำลองที่ถูกสร้างขึ้นภายใต้หัวข้อ “แยกบทสำหรับสถานศึกษา” โดยมีจำนวนข้อมูลไม่ต่ำกว่า 500 ตัว



ตัวชี้วัดประสิทธิภาพ

การประเมินผลการแบ่งกลุ่ม

Purity เป็นตัวชี้วัดผลการจัดกลุ่ม โดยข้อมูลที่อยู่ในกลุ่มเดียวกัน ควรมาจากการเดียวกัน

$$P_{ij} = \frac{m_{ij}}{m_i} \dots\dots\dots (1)$$

$$P_i = \max_{j=1}^l P_{ij} \dots\dots\dots (2)$$

$$purity = \sum_{i=1}^k \frac{m_i}{m} P_i \dots\dots\dots (3)$$

ตัวชี้วัดประสิทธิภาพ

การเปิดเมินผลการแบ่งกลุ่ม

Sum Square of Error หรือ **SSE** เป็นตัวชี้วัดการจัดกลุ่มโดยการหาผลรวมค่าระยะห่างของข้อมูลแต่ละตัวกับจุดกึ่งกลางของกลุ่มข้อมูลนั้นๆ

$$SSE = \sum_{i=1}^K \sum_{x \in C_i} dist(c_i, x)^2$$

ตัวชี้วัดประสิทธิภาพ

การประเมินผลการจำแนก

Precision, Recall, F1-Score เป็นการวัดประสิทธิภาพการจำแนกข้อมูล โดยผ่านตัวแปร 4 ตัว ได้แก่

Confusion Matrix			
TP	FP		
		FN	TN

$$\text{Precision} = \frac{TP}{TP + FP}$$

$$\text{Recall} = \frac{TP}{TP + FN}$$

$$F1 - Score = \frac{2 * Precision * Recall}{Precision + Recall}$$

การจัดกลุ่มข้อความ

- จำจัดคำหยุด = คำ ครับ หน่อย คำ วี๊ะ
- ใช้ TF-IDF ในการสกัดคุณลักษณะแทนข้อความ โดยใช้วิธีการ PCA มาช่วยในการลดมิติของข้อมูล

ตารางเปรียบเทียบจำนวนมิติข้อมูลก่อนและหลังทำ PCA

ชุดข้อมูล	เวลาที่ใช้ (วินาที)	จำนวนมิติข้อมูล	จำนวนมิติหลังทำ PCA
Case routing intent	0.022895098	197	64
Corona	0.264032841	609	172
ATIS	0.433625937	498	111

การกำหนดพารามิเตอร์สำหรับการจัดกลุ่ม

- **K-means Optimal** กำหนดจำนวน centroids โดยใช้วิธีการ Silhouette Method
- **DBSCAN Optimal** กำหนด MinPts = 3 และ 5 และประยุกต์ใช้ฟังก์ชัน NearestNeighbors ในการกำหนดระยะ eps

ผลการจัดกลุ่มก่อนและหลังทำ PCA

ข้อมูล	วิธีการ	ใช้ PCA	S	D	purity	เวลาจัดกลุ่ม
case_routing_intent	K-means	no	NaN	k=14	0.6667	0.1180
case_routing_intent	K-means	yes	NaN	k=14	0.6333	0.1090
case_routing_intent	DBSCAN	no	minPts=3	eps=0.6	0.3133	0.0618
case_routing_intent	DBSCAN	yes	minPts=3	eps=0.1	0.2933	0.0608
case_routing_intent	DBSCAN	no	minPts=5	eps=0.6	0.2933	0.0609
case_routing_intent	DBSCAN	yes	minPts=5	eps=0.1	0.2933	0.0590
corona	K-means	no	NaN	k=99	0.5745	1.8632
corona	K-means	yes	NaN	k=102	0.6135	0.8775
corona	DBSCAN	no	minPts=3	eps=0.3	0.0997	0.3211
corona	DBSCAN	yes	minPts=3	eps=0.1	0.0969	0.3012
corona	DBSCAN	no	minPts=5	eps=0.3	0.0551	0.2940
corona	DBSCAN	yes	minPts=5	eps=0.1	0.0551	0.2579
atis_intents	K-means	no	NaN	k=489	0.6135	28.8498
atis_intents	K-means	yes	NaN	k=485	0.8763	10.0008
atis_intents	DBSCAN	no	minPts=3	eps=0.1	0.7704	1.7692
atis_intents	DBSCAN	yes	minPts=3	eps=0.1	0.7808	1.6541
atis_intents	DBSCAN	no	minPts=5	eps=0.1	0.7521	1.5810
atis_intents	DBSCAN	yes	minPts=5	eps=0.1	0.7618	1.4044

ผลการจัดกลุ่มระหว่าง K-means และ DBSCAN

ข้อมูล	วิธีการ	sse	purity
case_routing_intent	K-mean	85.5811	0.6333
case_routing_intent	DBSCAN(minPts=3)	116.5259	0.2933
case_routing_intent	DBSCAN(minPts=5)	116.5259	0.2933
corona	K-mean	356.5779	0.6135
corona	DBSCAN(minPts=3)	715.9563	0.0969
corona	DBSCAN(minPts=5)	763.7182	0.0551
atis_intents	K-mean	1146.6829	0.8763
atis_intents	DBSCAN(minPts=3)	3207.7457	0.7808
atis_intents	DBSCAN(minPts=5)	3509.3300	0.7618
ข้อมูลที่สร้างขึ้นเอง	K-mean	118.8681	NaN
ข้อมูลที่สร้างขึ้นเอง	dbSCAN(minPts = 3)	171.8013	NaN
ข้อมูลที่สร้างขึ้นเอง	dbSCAN(minPts = 5)	243.9920	NaN

Deep K-means

คือการใช้ K-means แบ่งกลุ่มลึกลงไป 3 ชั้น (level) มีขั้นตอนดังนี้

1. Level1 : จัดกลุ่มโดยใช้ K-means แบบ Optimal
2. สำหรับแต่ละกลุ่มที่ j คำนวณทางเฉลี่ยจากข้อมูลในกลุ่มไปยังจุดศูนย์กลางของกลุ่ม แทนด้วย $d(j)$ (สมการที่ 1)
3. หากาเฉลี่ยนจากค่าที่ในขั้นตอนที่ 2. ของทุกกลุ่ม แทนด้วย $D(\text{avg})$ (สมการที่ 2)

$$d_j = \frac{\sum_{i=1}^{n_j} |\text{centroid}_j - x_i|}{n_j} \dots\dots\dots (1)$$

$$D_{\text{avg}} = \frac{\sum_{j=1}^N d_j}{N} \dots\dots\dots (2)$$

ผลการจัดกลุ่มระหว่าง Deep K-means, K-means และ DBSCAN

ข้อมูล	วิธีการ	sse	purity
case_routing_intent	K-mean	85.5811	0.6333
case_routing_intent	deep K-mean	32.3764	0.86
case_routing_intent	DBSCAN(minPts=3)	116.5259	0.2933
case_routing_intent	DBSCAN(minPts=5)	116.5259	0.2933
corona	K-mean	356.5779	0.6135
corona	deep K-mean	140.9437	0.8072
corona	DBSCAN(minPts=3)	715.9563	0.0969
corona	DBSCAN(minPts=5)	763.7182	0.0551
atis_intents	K-mean	1146.6829	0.8763
atis_intents	deep K-mean	493.2257	0.9249
atis_intents	DBSCAN(minPts=3)	3207.7457	0.7802
atis_intents	DBSCAN(minPts=5)	3509.3300	0.7616
ข้อมูลที่สร้างขึ้นเอง	K-mean	118.8681	NaN
ข้อมูลที่สร้างขึ้นเอง	deep K-mean	34.4778	NaN
ข้อมูลที่สร้างขึ้นเอง	dbSCAN(minPts = 3)	171.8013	NaN
ข้อมูลที่สร้างขึ้นเอง	dbSCAN(minPts = 5)	243.9920	NaN

การระบุ intent ให้กับกลุ่มของข้อมูลหลังจาก ผ่านการจัดกลุ่มด้วย Deep K-means

เขียนโปรแกรมสำหรับระบุ intent โดยมีการทำงาน
ดังนี้

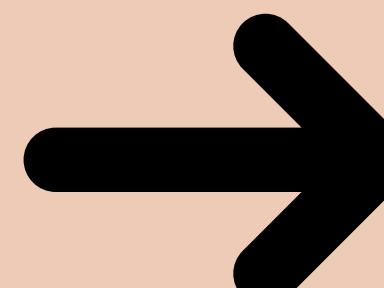
1. แสดงตัวอย่างข้อความภายในกลุ่ม โดยเลือกจาก
ข้อความที่มีระยะทางใกล้จุดศูนย์กลางของกลุ่มมาก
ที่สุด
2. ระบุ intent จากตัวอย่างที่แสดง
3. บันทึกเป็นไฟล์ .csv ตามกลุ่ม intent ระบุ

example message : ชื่อของคุณคืออะไร
assigned intent : ชื่อผู้ลงทะเบียน
example message : เรียนกี่ปีจบ
assigned intent : ระยะเวลาเรียน
example message : ค่าเทอมที่นาท
assigned intent : ค่าเทอม
example message : เรียนแบบห้อง
assigned intent : เรียนคอมพิวเตอร์
example message : มีที่พักไหม
assigned intent : ข้อมูลหอพัก
example message : เรียนจบไปทำงานอะไร
assigned intent : งานในอนาคต
example message : มีหอพักสำหรับนักศึกษามั้ยคะ
assigned intent : ข้อมูลหอพัก
example message : ค่าหอเท่าไหร่คะ
assigned intent : ข้อมูลหอพัก
example message : เรียนยกใหม่ครับ
assigned intent : การเรียน
example message : ขอ恕เบี้ยนการหอพัก
assigned intent : ข้อมูลหอพัก
example message : เปิดห้องวันไหน

นำข้อมูลขึ้น Dialogflow

- ระบุ response สำหรับตอบ User ในไฟล์ .csv แต่ละไฟล์ตามความเหมาะสม
- เขียนโปรแกรมสำหรับแปลงไฟล์ .csv เป็น json format สำหรับ upload ขึ้น Dialogflow
- หลังจากนั้น Import ไฟล์ที่ได้ขึ้น Dialogflow

	A	B	C	D	E	F
1	intent	query	response			
2	ขอบคุณ	ขอบคุณครับ	ด้วยความเต็มใจครับ			
3	ขอบคุณ	ขอบคุณ	ยินดีครับ			
4	ขอบคุณ	ขอบคุณครับ	ด้วยความยินดีครับ			
5	ขอบคุณ	ขอบคุณนะครับ				
6	ขอบคุณ	ขอบคุณครับ				
7	ขอบคุณ	ขอบคุณค่ะ				
8	ขอบคุณ	ขอบคุณค่ะ				
9	ขอบคุณ	ขอบคุณนะครับ				
10	ขอบคุณ	ขอบคุณครับ				
11	ขอบคุณ	ขอบคุณ				
12	ขอบคุณ	ขอบคุณครับ				



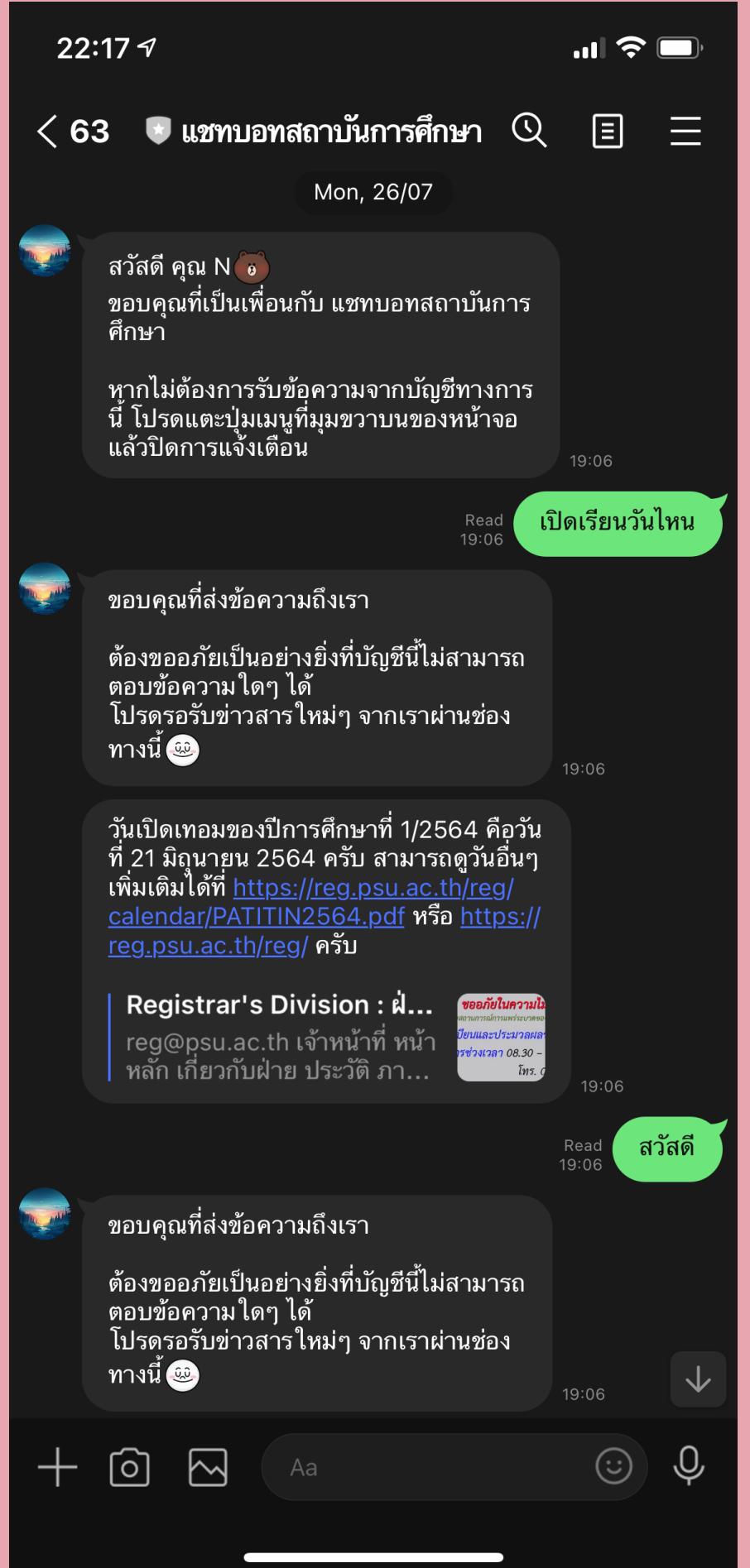
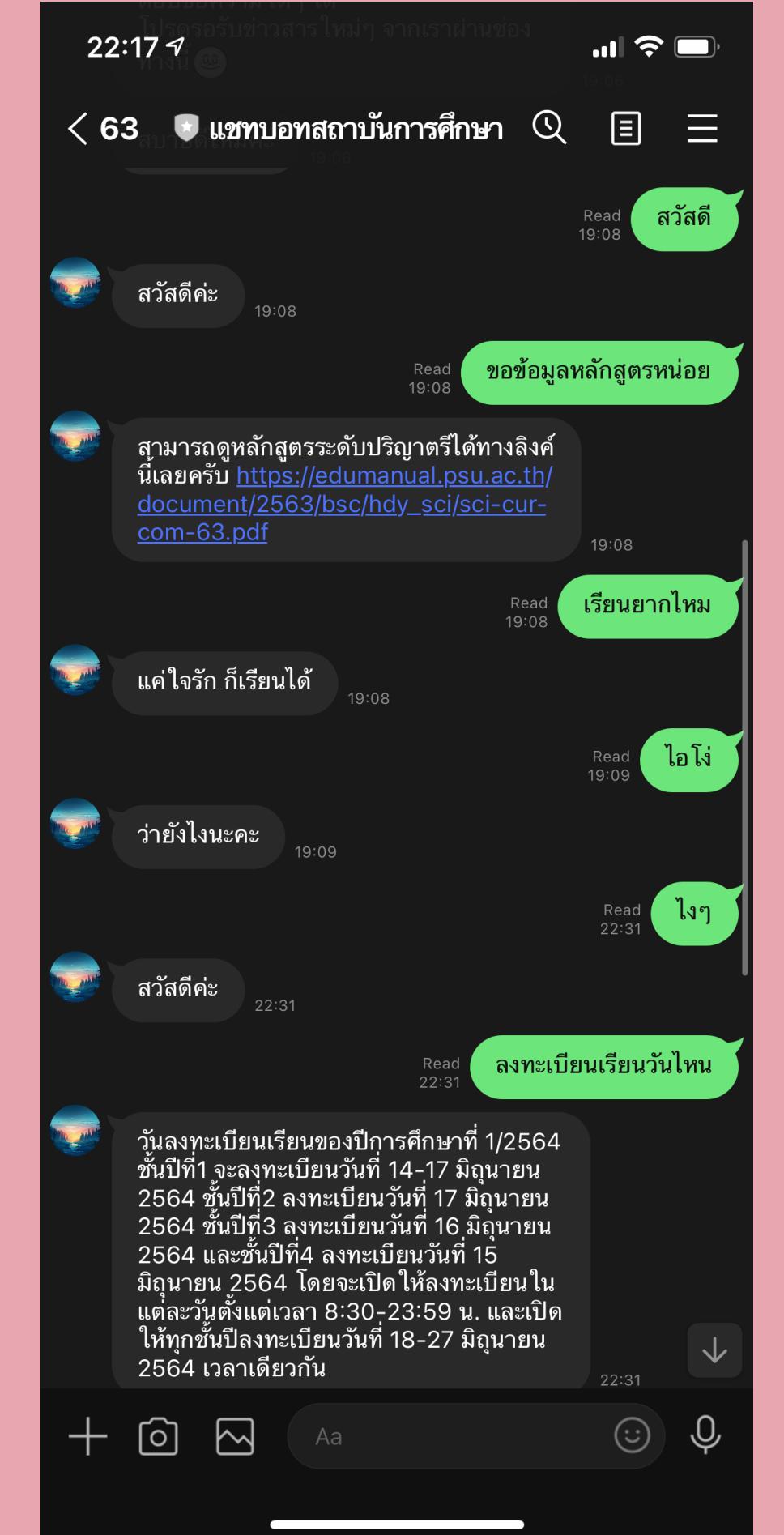
```
(/) ข้อมูล.json M X
dialogflow_file > {} ข้อมูล.json > ...
1 {
2   "name": "ขอบคุณ",
3   "auto": true,
4   "contexts": [],
5   "responses": [
6     {
7       "resetContexts": false,
8       "affectedContexts": [],
9       "parameters": [],
10      "messages": [
11        {
12          "type": 0,
13          "lang": "th",
14          "speech": ["ด้วยความเต็มใจครับ", "ยินดีครับ", "ด้วยความยินดีครับ"]
15        }
16      ],
17      "defaultResponsePlatforms": {},
18      "speech": []
19    },
20    "priority": 500000,
21    "webhookUsed": false,
22    "webhookForSlotFilling": false,
23    "fallbackIntent": false,
24    "events": []
25  }
26}
```

```
(/) ขอบคุณ_usersays_th.json M X
dialogflow_file > {} ขอบคุณ_usersays_th.json > {} 43 > [ ] data > {} 0 > userDef
1 [
2   {
3     "id": "0",
4     "data": [{"text": "ขอบคุณครับ", "userDefined": false}],
5     "isTemplate": false,
6     "count": 0,
7     "lang": "th"
8   },
9   {
10     "id": "1",
11     "data": [{"text": "ขอบคุณ", "userDefined": false}],
12     "isTemplate": false,
13     "count": 0,
14     "lang": "th"
15   },
16   {
17     "id": "2",
18     "data": [{"text": "ขอบคุณครับ", "userDefined": false}],
19     "isTemplate": false,
20     "count": 0,
21     "lang": "th"
22   },
23   {
24     "id": "3",
25     "data": [{"text": "ขอบคุณนะครับ", "userDefined": false}],
26     "isTemplate": false,
27     "count": 0,
28     "lang": "th"
29   }
30 ]
```

การทำ Line chatbot จาก Dialogflow

ขั้นตอนการทำ Line chatbot จาก Dialogflow

- สร้าง account Line developer
- สร้าง provider
- Enable Line ใน tab Integrations ของ Dialogflow หลังจากนั้นกรอกข้อมูลให้ครบ
- นำ web hook url จาก Dialogflow กรอกใน Line developer และเปิดการใช้งาน



THANK YOU

