



ระบบประเมินภาวะซึมเศร้าด้วยการเรียนรู้ของเครื่อง

Depression analysis with machine learning

โดย

นายพีรพัฒน์ ไพบูลย์ รหัสนิสิต 6230200503

นายอภิสิทธิ์ สุขขวัญ รหัสนิสิต 6230200881

โครงการวิทยาการคอมพิวเตอร์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
วิทยาศาสตรบัณฑิต (สาขาวิทยาการคอมพิวเตอร์)

คณะวิทยาศาสตร์ ศรีราชา

มหาวิทยาลัยเกษตรศาสตร์ วิทยาเขตศรีราชา

ปีการศึกษา 2565

ใบรับรองโครงการวิทยาการคอมพิวเตอร์
คณะวิทยาศาสตร์ ศรีราชา
ปริญญาวิทยาศาสตรบัณฑิต สาขาวิทยาการคอมพิวเตอร์

ระบบประเมินภาวะซึมเศร้าด้วยการเรียนรู้ของเครื่อง
Depression analysis with machine learning

นายพีรพัฒน์ ไพบูลย์ รหัสนิต 6230200503
นายอภิสิทธิ์ สุขขวัญ รหัสนิต 6230200881

ได้รับการตรวจสอบและอนุมัติ ให้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร

ประธานหลักสูตร	
(สาขาวิชาวิทยาการคอมพิวเตอร์)	(ดร.จิรพรรณ เจริญสุข)

กรรมการสอบ	
(สาขาวิชาวิทยาการคอมพิวเตอร์)	(ดร.ฉัตรชัย เกษมทวีโชค)

กรรมการสอบ	
(สาขาวิชาวิทยาการคอมพิวเตอร์)	(ผศ.ดร.อรรพรรณ วัชรภาพร)

อาจารย์ที่ปรึกษา	
	(ดร.ชโลธร ชูทอง)

ชื่อเรื่อง (ไทย)	ระบบประเมินภาวะซึมเศร้าด้วยปัญญาประดิษฐ์
ชื่อเรื่อง (English)	Depression analysis with machine learning
ชื่อผู้เขียน	นายพิรพัฒน์ ไพบูลย์ , นายอภิสิทธิ์ สุขขวัญ
ชื่อปริญญา	วิทยาศาสตร์บัณฑิต (วิทยาการคอมพิวเตอร์)
อาจารย์ที่ปรึกษา	ดร.ชโลธร ชูทอง
คำสำคัญ	ภาวะซึมเศร้า, โมเดล, เซโรโทนิน, เว็บไซต์สำหรับจองห้องรับคำปรึกษา, แบบประเมิน PHQ-9, แชนบอท,
Keyword	Colab, Model, cleansing data, Line OA, API, Support Vector Machine, Naive Bayes, K-nearest neighbor, Random forest, Python, Tokenization

บทคัดย่อ

เนื่องด้วยปัจจุบันปัญหาภาวะด้านจิตใจในสังคมนั้นเพิ่มขึ้นอย่างต่อเนื่อง โดยไม่ได้มีเพียงวัยทำงานเท่านั้นที่มีความเสี่ยง แม้แต่นิสิตในมหาวิทยาลัยเองนั้นก็มีความเสี่ยงในเรื่องของปัญหาด้านจิตใจทั้งสิ้น ซึ่งในบางกรณีผู้ที่มีความเสี่ยงอาจไม่รู้ตัวและละเลยจนเกิดเป็นภาวะซึมเศร้า และส่งผลกระทบต่อการใช้ชีวิตประจำวัน มหาวิทยาลัยเกษตรศาสตร์ วิทยาเขตศรีราชาให้บริการให้คำแนะนำด้านจิตวิทยากับนิสิตจากนักจิตวิทยา เพื่อคอยให้คำแนะนำและช่วยเหลือนิสิตให้สามารถปรับตัวและบรรเทาปัญหาทางด้านจิตใจ แต่เนื่องจากจำนวนผู้ใช้บริการที่มากขึ้นทำให้ภาระด้านการตรวจสอบและคัดกรองตกอยู่กับเจ้าหน้าที่เพียงผู้เดียว จึงได้มีการพัฒนาระบบประเมินภาวะซึมเศร้าด้วยการเรียนรู้ของเครื่องเพื่อนำมาช่วยในการประเมินอาการของผู้ที่มีความเสี่ยงเบื้องต้น ซึ่งระบบดังกล่าวเป็นการนำข้อความที่สื่อถึงอาการของภาวะซึมเศร้ามาวิเคราะห์และนำมาใช้ในการสร้างโมเดล เพื่อเรียนรู้และประเมินข้อความต่างๆจากผู้ใช้งานว่ามีอาการของภาวะซึมเศร้าหรือไม่ โดยมีโมเดลที่ทำการศึกษาทั้งหมด 4 โมเดลได้แก่ 1.) Support Vector Machine (SVM) 2.) Naive Bayes 3.) K-nearest neighbor 4.) Random forest และนำ Model ดังกล่าวมาประยุกต์ใช้กับ Line OA โดยหวังให้นำไปใช้ประโยชน์กับห้องรับคำปรึกษาของมหาวิทยาลัยเกษตรศาสตร์ วิทยาเขตศรีราชา เพื่อให้เกิดประโยชน์แก่นิสิตและเป็นการแบ่งเบาภาระกับเจ้าหน้าที่

กิตติกรรมประกาศ

รายงานผลการดำเนินการวิชาสหกิจศึกษาฉบับนี้สำเร็จสมบูรณ์ได้ด้วยความกรุณาจาก
นักจิตวิทยาประจำห้องรับคำปรึกษา มหาวิทยาลัยเกษตรศาสตร์ วิทยาเขตศรีราชาที่ให้ความรู้คำแนะนำ
กระทั่งสำเร็จลุล่วงด้วยดีผู้จัดทำรู้สึกซาบซึ้งเป็นอย่างยิ่งจึงขอขอบพระคุณเป็นอย่างสูงไว้ ณ โอกาสนี้

ขอขอบพระคุณ ดร.ชโลธร ชูทอง (อาจารย์ที่ปรึกษา) ที่คอยดูแลให้คำแนะนำเกี่ยวกับการทำงาน
ตลอดจนให้แนวคิดแนวทางต่าง ๆ และนำมาอ้างอิงในการจัดทำโครงงานครั้งนี้จนกระทั่งสำเร็จลุล่วง
ด้วยดี

ท้ายสุดนี้ ขอขอบคุณ เพื่อนๆนิสิตชั้นปีที่ 4 ทุกท่านที่ให้การช่วยเหลือและกำลังใจด้วยดีตลอดมา
ข้าพเจ้าขอขอบพระทัยอันพึงมีจากการศึกษาครั้งนี้ให้กับทุกท่านที่กล่าวมาด้วยความยินดียิ่ง

นายพีรพัฒน์ ไพบูลย์

นายอภิสิทธิ์ สุขขวัญ

สารบัญ

บทที่ 1 บทนำ.....	1
1.1 หลักการและเหตุผล.....	1
1.2 ปัญหา	1
1.3 วัตถุประสงค์ของโครงการ	2
1.4 ประโยชน์ที่คาดว่าจะได้รับ	2
1.5 ขอบเขตการทำงาน	2
1.6 ระยะเวลาและแผนการดำเนินงาน	3
1.7 เครื่องมือที่ใช้ในการพัฒนาระบบ.....	4
บทที่ 2 ทฤษฎีที่เกี่ยวข้อง	5
2.1 ภาวะซึมเศร้า (Depression).....	5
2.2 แชทบอท (Chatbot).....	7
2.3 Google colab	8
2.4 Python.....	9
2.5 Support Vector Machine (SVM).....	11
2.6 Naïve Bayes.....	12
2.7 K-nearest neighbor	13
2.8 Random forest.....	13
บทที่ 3 การจัดการข้อมูลและการออกแบบ	14
3.1 การวิเคราะห์และออกแบบระบบ.....	14
3.2 ข้อมูลที่ใช้ในการพัฒนาระบบ	14
3.3 การทำงานของระบบ	16
3.4 Usecase diagram	17
3.6 การพัฒนาโดยใช้โมเดล Naïve Bayes.....	21

3.7	การพัฒนาโดยใช้โมเดล K-nearest	22
3.8	การพัฒนาโดยใช้โมเดล Random-forest	22
บทที่ 4 การทำงานและทดสอบระบบ		24
4.1	การทำงานของระบบใน Line OA	24
4.2	การทำงานของระบบใน Dialogflow.....	28
4.3	การสร้างและการทำงานของ API	34
4.4	การใช้งาน Google cloud.....	39
4.5	ผลลัพธ์ของระบบ	42
บทที่ 5 การสรุปผลและข้อเสนอแนะ		43
5.1	สรุปผลโครงการ	43
5.2	ข้อเสนอแนะ.....	43

สารบัญภาพ

รูปที่ 1 Google colab icon	8
รูปที่ 2 Python Icon.....	9
รูปที่ 3 ตัวอย่างแนวคิด SVM	11
รูปที่ 4 ตัวอย่างแนวคิด K-NN	13
รูปที่ 5 ตัวอย่างแนวคิด random forest	13
รูปที่ 6 กราฟแสดงจำนวน data	14
รูปที่ 7 ตัวอย่างการทำTokenization.....	15
รูปที่ 8 การทำงานของระบบประเมินภาวะซึมเศร้าด้วยปัญญาประดิษฐ์.....	16
รูปที่ 9 การทำงานของระบบกับผู้ใช้	17
รูปที่ 10 ผลลัพธ์ของโมเดล Support vector machine.....	21
รูปที่ 11 ผลลัพธ์ของโมเดล Naïve Bayes	21
รูปที่ 12 ผลลัพธ์ของโมเดล Naïve Bayes	22
รูปที่ 13 ผลลัพธ์ของโมเดล Random-forest	22
รูปที่ 14 กราฟผลลัพธ์ค่าความถูกต้องของทั้ง 4 โมเดล.....	23
รูปที่ 15 ตัวอย่างข้อความแนะนำการใช้งานระบบ	24
รูปที่ 16 ตัวอย่างริชเมนูของระบบ	25
รูปที่ 17 เภณฑ์การให้คะแนน	25
รูปที่ 18 ตัวอย่างการตอบกลับหลังตอบคำถามครบ	27
รูปที่ 19 ระบบส่งผลลัพธ์กลับมายังผู้ใช้.....	27
รูปที่ 20 การสร้างโปรเจกต์Dialogflow.....	28
รูปที่ 21 การกำหนด Training phrases	28
รูปที่ 22 การกำหนดรูปแบบการตอบกลับ	29
รูปที่ 23 สร้าง follow up intent เพื่อรับข้อมูลจากผู้ใช้	30
รูปที่ 24 กำหนด parameter เพื่อรับค่าจากผู้ใช้.....	30
รูปที่ 25 เปิดใช้งาน Inline editor ใน fulfillment	31

รูปที่ 26 สร้างฟังก์ชันสำหรับนำข้อมูลมาประมวลผล	31
รูปที่ 27 รวมคะแนนจากแบบประเมิน.....	31
รูปที่ 28 เรียกใช้ API เพื่อนำข้อมูลไปประมวลผล	32
รูปที่ 29 รับค่าผลลัพธ์จากโมเดล	33
รูปที่ 30 กำหนดผลลัพธ์การตอบกลับ.....	33
รูปที่ 31 library ที่มีการเรียกใช้งาน	34
รูปที่ 32 การโหลดโมเดลและคลังคำศัพท์	34
รูปที่ 33 ทำการตัดคำ	35
รูปที่ 34 นำข้อมูลมาประมวลผลด้วยโมเดล.....	35
รูปที่ 35 กำหนด Path ของ Google sheet สำหรับบันทึกข้อมูล.....	36
รูปที่ 36 โค้ดการทำงานของ API	37
รูปที่ 37 การทดสอบ API โดยใช้ Postman	38
รูปที่ 38 รายละเอียดไฟล์ที่จำเป็น	39
รูปที่ 39 เมนูการเปิดใช้งาน Services	39
รูปที่ 40 ติดตั้ง Gcloud CLI	40
รูปที่ 41 ตัวอย่างการแสดงบัญชี.....	40
รูปที่ 42 เลือกรายการโปรเจกต์ที่ต้องการ	40
รูปที่ 43 ตัวอย่างการเริ่มการ Deploy	41
รูปที่ 44 แสดงผลลัพธ์การทาง และ URL สำหรับเรียกใช้ API.....	41
รูปที่ 47 เกณฑ์การประเมินระดับ"เสียงปานกลาง"	42
รูปที่ 45 เกณฑ์การประเมินระดับ"ไม่มีอาการ"	42
รูปที่ 48 เกณฑ์การประเมินระดับ"เสียงมาก".....	42
รูปที่ 49 เกณฑ์การประเมินระดับ"เสียงรุนแรง"	42
รูปที่ 46 เกณฑ์การประเมินระดับ"เสียงน้อย"	42

สารบัญตาราง

ตารางที่ 1	ระยะเวลาในการดำเนินงาน	3
ตารางที่ 2	เครื่องมือที่ใช้ในการพัฒนาระบบ.....	4
ตารางที่ 3	ตารางแสดงคุณลักษณะของข้อมูล	15
ตารางที่ 4	ตัวอย่างข้อมูลที่นำมาใช้ในการฝึกสอนเครื่อง	15
ตารางที่ 5	คำอธิบาย Use case ประเมินสุขภาพ.....	17
ตารางที่ 6	คำอธิบาย Use case กรอกรอกการเบื้องต้น	18
ตารางที่ 7	คำอธิบาย Use case เข้าลิ้งค์จองห้องรับคำปรึกษา.....	18
ตารางที่ 7	คำอธิบาย Use case ประมวลผลด้วยการเรียนรู้ของเครื่อง	19
ตารางที่ 8	คำอธิบาย Use case เก็บข้อมูลใน Google sheet.....	19
ตารางที่ 9	คำอธิบาย Use case แจ้งผลการประเมิน	20
ตารางที่ 10	คำอธิบาย Use case แจ้งลิ้งค์จองห้องรับคำปรึกษา	20
ตารางที่ 11	ตัวอย่างคำถามจากแบบประเมิน PHQ-9	26

บทที่ 1

บทนำ

1.1 หลักการและเหตุผล

ในปัจจุบันปัญหาด้านต่างๆ ที่เกิดขึ้นในชีวิตประจำวันนั้นมีมากมายไม่ว่าจะเป็นปัญหาเล็กๆ หรือปัญหาที่อาจเปลี่ยนแปลงการใช้ชีวิตในมหาลัย แต่ไม่ว่าปัญหาดังกล่าวจะเล็กหรือใหญ่ ก็มีผลกระทบต่อสภาพจิตใจและสะสมมากขึ้น จึงอาจก่อให้เกิดเป็นภาวะซึมเศร้า (Depression) หรือภาวะซึมเศร้าได้ ภาวะซึมเศร้า เป็นภาวะทางจิตเวชประเภทหนึ่ง เกิดจากความผิดปกติของสารเคมีในสมอง ‘เซโรโทนิน (Serotonin)’ มีปริมาณลดลง ทำให้ผู้ป่วยมีอาการป่วยทั้งร่างกาย จิตใจ และความคิด รู้สึกว่าตัวเองไม่มีความสุข มีแต่ความวิตกกังวล และหากปล่อยไว้ ผู้ป่วยอาจคิดสั้นฆ่าตัวตายได้ สาเหตุหรือปัจจัยที่เป็นตัวกระตุ้นได้แก่ ความเครียดสะสม การสูญเสียครั้งใหญ่ สภาพจิตใจที่เกิดจากการเลี้ยงดู ความสัมพันธ์กับคนรอบข้างไม่ราบรื่น ความเสี่ยงทางพันธุกรรม เป็นต้น ผู้ที่มีภาวะซึมเศร้าจะแสดงอาการออกมาผ่านทางพฤติกรรมหรือคำพูดและผู้ที่มิภาวะดังกล่าว อาจไม่ทราบถึงอาการหรือพฤติกรรมที่เปลี่ยนแปลงไปของตน

มหาวิทยาลัยเกษตรศาสตร์ วิทยาเขตศรีราชาให้บริการให้คำแนะนำด้านจิตวิทยากับนิสิตจากนักจิตวิทยา เพื่อคอยให้คำแนะนำและช่วยเหลือนิสิตให้สามารถปรับตัวและบรรเทาปัญหาทางด้านจิตใจ แต่เนื่องจากจำนวนผู้เข้าใช้บริการที่มากขึ้นทำให้ภาระด้านการตรวจสอบและคัดกรองตกอยู่กับเจ้าหน้าที่ ทั้งนี้จึงเป็นเหตุผลการพัฒนาระบบประเมินภาวะซึมเศร้าด้วยปัญญาประดิษฐ์ เพื่อให้ผู้ใช้สามารถประเมินอาการของตนเบื้องต้น และลดภาระการทำงานของเจ้าหน้าที่

จากที่กล่าวมาข้างต้น จึงได้มีการสร้างระบบที่ใช้ในการประเมินผู้ที่อาจมีภาวะซึมเศร้าโดยใช้ Line Chatbot สอบถามอาการและความรู้สึกของผู้ใช้ในวันนั้นๆ และนำข้อความคำตอบของผู้ใช้มาวิเคราะห์โดยใช้ Machine Learning เพื่อจัดกลุ่มข้อความของผู้ใช้ว่าอยู่ในกลุ่มของผู้ที่มีภาวะซึมเศร้าหรือไม่ และแนะนำให้ผู้ใช้ที่มีอาการเสี่ยงที่จะเป็นภาวะซึมเศร้าให้ปรึกษาแพทย์ต่อไป

1.2 ปัญหา

- 1.2.1 ผู้มีภาวะซึมเศร้าอาจไม่สามารถบอกเล่าให้ผู้อื่นฟังได้
- 1.2.2 ผู้ที่มีภาวะซึมเศร้าอาจไม่ทราบถึงอาการของตน
- 1.2.3 การจองบริการของห้องรับคำปรึกษาเข้าถึงได้ยาก

1.3 วัตถุประสงค์ของโครงการ

เพื่อพัฒนาระบบประเมินภาวะซึมเศร้าด้วยการเรียนรู้ของเครื่องให้สามารถระบบสามารถประเมินอาการของผู้ใช้ได้อย่างแม่นยำ และนำมาใช้ได้จริง เพื่อให้ผู้ใช้สามารถประเมินความเสี่ยงต่อภาวะซึมเศร้าได้ด้วยตนเองเบื้องต้น และลดภาระการทำงานของเจ้าหน้าที่ลง

1.4 ประโยชน์ที่คาดว่าจะได้รับ

ระบบสามารถประเมินอาการของผู้ใช้ได้อย่างแม่นยำเพื่อให้ผู้ใช้รับรู้อาการของตนและสามารถดูแลตนเองได้อย่างถูกต้อง รวมถึงเป็นช่องทางหนึ่งเพื่อให้เข้าใช้บริการของห้องรับคำปรึกษาได้สะดวกยิ่งขึ้น

1.5 ขอบเขตการทำงาน

ระบบ

- สามารถประเมินสุขภาพจิตของผู้ใช้ได้อย่างถูกต้อง
- แจ้งผลการประเมินให้ผู้ใช้รับรู้ถึงระดับความเสี่ยงของตน
- บันทึกผลการประเมินและอาการของผู้ใช้
- นำผู้ใช้ไปยังหน้าเว็บไซต์สำหรับจองห้องรับคำปรึกษา

ผู้ใช้

- สามารถทำแบบประเมิน PHQ-9 เพื่อประเมินสุขภาพตนเอง
- เข้าดูข้อมูลวิธีการดูแลสุขภาพจิต
- เข้าเว็บไซต์สำหรับจองห้องรับคำปรึกษา

1.6 ระยะเวลาและแผนการดำเนินงาน

ตารางที่ 1 ระยะเวลาในการดำเนินงาน

ระยะเวลาการดำเนินงาน โครงการวิทยากรคอมพิวเตอร์ ปีการศึกษา 2/2565					
ขั้นตอนการดำเนินงาน	ธันวาคม 2565	มกราคม 2566	กุมภาพันธ์ 2566	มีนาคม 2566	เมษายน 2566
1. วางแผนหัวข้อโครงการ	← - - - - - → ← - - - - - →				
2. ศึกษา software ระบบ	← - - - - - → ← - - - - - →				
3. รวบรวมและวิเคราะห์	← - - - - - → ← - - - - - →				
4. ออกแบบระบบการทำงาน		← - - - - - → ← - - - - - →			
5. ออกแบบ User Interface, Workflow, ระบบฐานข้อมูล		← - - - - - → ← - - - - - →			
6. พัฒนาระบบ		← - - - - - → ← - - - - - →			
7. ทดสอบระบบ			← - - - - - → ← - - - - - →		
8. จัดทำเอกสารโครงการ			← - - - - - → ← - - - - - →		

*หมายเหตุ

← - - - - - → คาดการณ์แทนระยะเวลา

← - - - - - → แทนระยะเวลาทำงานจริง

1.7 เครื่องมือที่ใช้ในการพัฒนาระบบ

ตารางที่ 2 เครื่องมือที่ใช้ในการพัฒนาระบบ

	Hardware	Software
เครื่องมือพัฒนาระบบ (User Interface)	Dell latitude 3480 Processor : Intel(R) Core(TM) i7-7500U CPU @ 2.70GHz RAM : 8 GB SSD : 256 GB	OS : Windows 10 Pro - Line office account - Dialogflow
เครื่องมือพัฒนาระบบ (API)	Dell latitude 3480 Processor : Intel(R) Core(TM) i7-7500U CPU @ 2.70GHz RAM : 8 GB SSD : 256 GB	OS : Windows 10 Pro - Line
เครื่องมือในการพัฒนาระบบ (Model)	Dell latitude 3480 Processor : Intel(R) Core(TM) i7-7500U CPU @ 2.70GHz RAM : 8 GB SSD : 256 GB	OS : Windows 10 Pro - Google Colab

บทที่ 2

ทฤษฎีที่เกี่ยวข้อง

ในการพัฒนาระบบประเมินภาวะซึมเศร้าด้วยปัญญาประดิษฐ์นี้ ผู้พัฒนาได้ศึกษาหลักการ ทฤษฎีและเทคโนโลยีต่าง ๆ ที่เกี่ยวข้องกับการประยุกต์ใช้ในการพัฒนาระบบ ซึ่งประกอบด้วยหัวข้อต่าง ๆ ดังต่อไปนี้

2.1 ภาวะซึมเศร้า (Depression)

2.1.1 ภาวะซึมเศร้า คืออะไร

คนส่วนใหญ่มักคิดว่าภาวะซึมเศร้าฟังดูไม่คุ้นหูถ้าพูดถึงเรื่องซึมเศร้าเรามักจะนึกกันว่าเป็นเรื่องของอารมณ์ความรู้สึกที่เกิดจากความผิดหวังหรือการสูญเสียมากกว่าที่จะเป็นภาวะซึ่งตามจริงแล้วที่เราพบกันในชีวิตประจำวันส่วนใหญ่ก็จะเป็นเรื่องของอารมณ์ความรู้สึกธรรมดาๆ ที่มีกันในชีวิตประจำวัน อย่างไรก็ตามในบางครั้ง ถ้าอารมณ์เศร้าที่เกิดขึ้นนั้นเป็นอยู่นานโดยไม่มีที่ท่าว่าจะดีขึ้น หรือเป็นรุนแรง มีอาการต่างๆ ติดตามมา เช่น นอนหลับๆ ตื่นๆ เบื่ออาหาร น้ำหนักลดลงมาก หมดความสนใจต่อโลกภายนอก ไม่คิดอยากมีชีวิตอยู่อีกต่อไป ก็อาจจะเข้าข่ายของภาวะซึมเศร้าแล้ว คำว่า “ภาวะ” บ่งบอกว่าเป็นความผิดปกติทางการแพทย์ ซึ่งจำเป็นต้องได้รับการดูแลรักษาเพื่อให้อาการทุเลา ต่างจากภาวะอารมณ์เศร้าตามปกติธรรมดาที่ถ้าเหตุการณ์ต่างๆ รอบตัวคลี่คลายลงหรือมีคนเข้าใจเห็นใจอารมณ์เศร้านี้ก็อาจหายได้

ผู้ที่ป่วยเป็นภาวะซึมเศร้านอกจากมีอาการซึมเศร้าร่วมกับอาการต่างๆ แล้วการทำงานหรือการประกอบกิจวัตรประจำวันก็แย่ลงด้วย หากจะเปรียบกับภาวะทางร่างกายก็คล้ายๆ กัน เช่น ในภาวะหัวใจ ผู้ที่เป็นก็จะมีอาการต่างๆ ร่วมกับการทำอะไรต่างๆ ได้น้อยหรือไม่ดีเท่าเดิม ดังนั้น การเป็นภาวะซึมเศร้าไม่ได้หมายความว่า ผู้ที่เป็นเป็นคนอ่อนแอ คิดมาก หรือเป็นคนไม่สู้ปัญหา แต่ที่เขาเป็นนั้นเป็นเพราะตัวภาวะ กล่าวได้ว่าถ้าได้รับการรักษาที่ถูกต้องเหมาะสม ภาวะก็จะทุเลาลง เขาก็จะกลับมาเป็นผู้ที่จิตใจแจ่มใส พร้อมจะทำกิจวัตรต่างๆ ดังเดิม ผู้ที่ป่วยเป็นภาวะซึมเศร้าจะมีการเปลี่ยนแปลงไปจากเดิมค่อนข้างมาก การเปลี่ยนแปลงหลักๆ จะเป็นในด้านอารมณ์ ความรู้สึกนึกคิด พฤติกรรม ร่วมกับอาการทางร่างกายต่างๆ

2.1.2 เกณฑ์การวินิจฉัย

มีอาการดังต่อไปนี้ 5 อาการหรือมากกว่า

1. มีอารมณ์ซึมเศร้าแทบทั้งวัน (ในเด็กและวัยรุ่นอาจเป็นอารมณ์หงุดหงิดก็ได้)
2. ความสนใจหรือความเพลินใจในกิจกรรมต่างๆ แทบทั้งหมดลดลงอย่างมากแทบทั้งวัน
3. น้ำหนักลดลงหรือเพิ่มขึ้นมาก (น้ำหนักเปลี่ยนแปลงมากกว่าร้อยละ 5 ต่อเดือน) หรือมีการเบื่ออาหารหรือเจริญอาหารมาก
4. นอนไม่หลับ หรือหลับมากไป
5. กระวนกระวาย อยู่ไม่สุข หรือเชื่องช้าลง
6. อ่อนเพลีย ไร้เรี่ยวแรง
7. รู้สึกตนเองไร้ค่า
8. สมาธิลดลง ใจลอย หรือลังเลใจไปหมด คิดเรื่องการตาย คิดอยากตาย

* ต้องมีอาการในข้อ 1 หรือ 2 อย่างน้อย 1 ข้อ

* ต้องมีอาการเป็นอยู่นาน 2 สัปดาห์ขึ้นไป และต้องมีอาการเหล่านี้อยู่เกือบตลอดเวลา แทบทุกวัน ไม่ใช่เป็นๆหายๆ เป็นเพียงแคว้นสองวันหายไปแล้วกลับมาเป็นใหม่

2.2 แชทบอท (Chatbot)

2.2.1 แชทบอทคืออะไร

แอปพลิเคชันซอฟต์แวร์ที่ถูกออกแบบให้สามารถพูดคุย หรือสนทนาได้เหมือนกับมนุษย์ผ่านทางข้อความ หรือการพูดคุยโดยมีเทคโนโลยี AI เป็นผู้อยู่เบื้องหลัง รวมถึงเจ้าหน้าที่เสมือน (Virtual agent) เจ้าหน้าที่เสมือนในการโต้ตอบกับลูกค้า (Interactive agents) และผู้ช่วยดิจิทัล (Digital assistants) หรือ AI เชิงสนทนา (Conversational AI) โดยปกติแล้วแชทบอทจะถูกใส่รวมเข้าไปในแอปพลิเคชัน หรือแพลตฟอร์มที่ใช้ในการส่งข้อความต่างๆเพื่อใช้ในการให้บริการลูกค้าโดยไม่ต้องใช้พนักงานตัวเป็นๆในการตอบคำถาม

2.2.2 ประเภทของแชทบอท

แชทบอทที่ธุรกิจสามารถนำไปใช้งานได้แบ่งได้เป็น 2 ประเภท ได้แก่ Transactional chatbot และ Conversational chatbots ซึ่งทั้ง 2 ประเภทแตกต่างกันที่ระดับความซับซ้อนในการทำงาน

- **Transactional chatbots** : หรือเรียกว่า Task-oriented or Declarative chatbot เป็นแชทบอทที่ถูกสร้างขึ้นมาเพื่อใช้งานในวัตถุประสงค์เดียว โดยโฟกัสให้แชทบอทสามารถทำงาน หรือหน้าที่นั้นๆได้แบบอัตโนมัติ โดยแชทบอทประเภทนี้ถูกออกแบบมาเพื่อให้ทางเลือกที่ได้มีการตั้งค่าไว้ล่วงหน้ากับผู้ใช้งานได้เลือก ขึ้นอยู่กับผู้ใช้งานว่าผู้ใช้งานเลือกทางเลือกไหน หรือปัญหาด้านใดที่ผู้ใช้งานต้องการได้รับการแก้ไข
- **Conversational Chatbots** : Conversational chatbots จะมีความซับซ้อนกว่า และเป็นรูปแบบปฏิสัมพันธ์ของแชทบอทที่ใช้ Natural language processing (NLP) ในการตอบคำถามให้เหมาะกับแต่ละบุคคลได้(Personalization) Conversational บอทได้ใช้ AI เชิงสนทนา ร่วมกับ NLP และการเข้าถึงฐานข้อมูล และข้อมูลอื่นๆเพื่อสามารถตรวจจับคำถามและคำตอบที่อาจจะมีความแตกต่างกันเพียงเล็กน้อยได้เป็นอย่างดี และสามารถปรับเปลี่ยนและให้คำตอบได้อย่างแม่นยำเหมือนกับพูดคุยกับพนักงานจริงๆ

2.3 Google colab

Google Colab (Google Colaboratory) เป็นบริการ Software as a Service (SaaS) โฮสต์โปรแกรม Jupiter Notebook บน Cloud จาก Google สามารถใช้ Google Colab เขียนโปรแกรมภาษา Python ได้ฟรี และยังมี GPU, TPU ให้ใช้ฟรีอีกครั้งละ 12 ชั่วโมง



รูปที่ 1 Google colab icon

2.4 Python

2.4.1 Python คืออะไร

Python เป็นภาษาการเขียนโปรแกรมที่ใช้อย่างแพร่หลายในเว็บแอปพลิเคชัน การพัฒนาซอฟต์แวร์ วิทยาศาสตร์ข้อมูล และการเรียนรู้ของเครื่อง (Machine learning) นักพัฒนาใช้ Python เนื่องจากมีประสิทธิภาพ เรียนรู้ง่าย และสามารถทำงานบนแพลตฟอร์มต่างๆ ได้มากมาย ทั้งนี้ซอฟต์แวร์ Python สามารถดาวน์โหลดได้ฟรี ผสานการทำงานร่วมกับระบบทุกประเภท และเพิ่มความเร็วในการพัฒนา



รูปที่ 2 Python Icon

2.4.2 ข้อดีของ Python

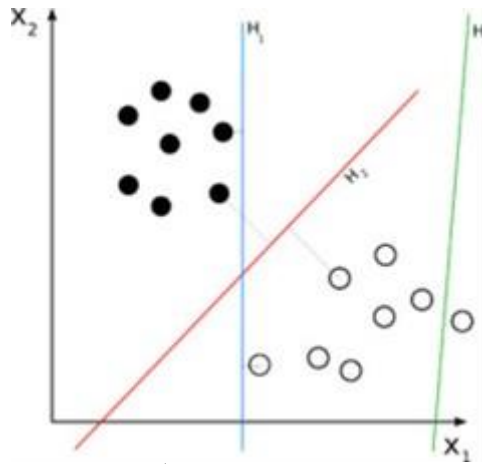
- นักพัฒนาสามารถอ่านและทำความเข้าใจโปรแกรม Python ได้อย่างง่ายดาย เนื่องจากมีไวยากรณ์พื้นฐานเหมือนภาษาอังกฤษ
- Python ทำให้นักพัฒนาทำงานได้อย่างมีประสิทธิภาพมากขึ้นเนื่องจากพวกเขาสามารถเขียนโปรแกรม Python ได้โดยใช้โค้ดน้อยลงเมื่อเปรียบเทียบกับภาษาอื่นๆ อีกมากมาย
- Python มีไลบรารีมาตรฐานขนาดใหญ่ที่มีโค้ดที่ใช้งานได้สำหรับเกือบทุกงาน ด้วยเหตุนี้ นักพัฒนาจึงไม่ต้องเขียนโค้ดขึ้นใหม่ทั้งหมด
- นักพัฒนาสามารถใช้ Python ร่วมกับภาษาการเขียนโปรแกรมยอดนิยมอื่นๆ เช่น Java, C และ C++ ได้
- Python สามารถใช้งานได้บนระบบปฏิบัติการคอมพิวเตอร์ต่างๆ เช่น Windows, macOS, Linux และ Unix

2.4.3 คุณสมบัติของ Python

- Python เป็นภาษาที่แปลผลแล้ว ซึ่งหมายความว่าสามารถเรียกใช้โค้ดที่ละบรรทัดได้โดยตรงหากมีข้อผิดพลาดในโค้ดโปรแกรมก็จะหยุดทำงานทันทีที่ตรงนั้น โปรแกรมเมอร์จึงสามารถค้นหาข้อผิดพลาดในโค้ดได้อย่างรวดเร็ว
- Python ใช้คำที่เหมือนในภาษาอังกฤษ ซึ่งแตกต่างจากภาษาการเขียนโปรแกรมอื่นๆ เนื่องจาก Python ไม่ใช่วงเล็บปีกกา แต่จะการใช้การเยื้องแทน
- ไม่ต้องระบุประเภทตัวแปรเมื่อเขียนโค้ดเนื่องจาก Python จะกำหนดไว้ที่รันไทม์ ด้วยเหตุนี้ จึงสามารถเขียนโปรแกรม Python ได้รวดเร็วขึ้น
- Python มีความใกล้เคียงกับภาษามนุษย์มากกว่าภาษาการเขียนโปรแกรมอื่นๆ ดังนั้นจึงไม่ต้องกังวลกับฟังก์ชันการทำงานพื้นฐานต่างๆ เช่น สถาปัตยกรรมและการจัดการหน่วยความจำ

2.5 Support Vector Machine (SVM)

อัลกอริทึมที่สามารถนำมาช่วยแก้ปัญหาการจำแนกข้อมูล ซึ่งเป็นการเรียนรู้แบบมีผู้สอน ใช้ในการวิเคราะห์ข้อมูลและจำแนกข้อมูลโดยอาศัยหลักการของการหาสมประสิทธิ์ของสมการเพื่อสร้างเส้นแบ่งแยกกลุ่มข้อมูลที่ถูกต้องเข้าสู่กระบวนการสอนให้ระบบเรียนรู้โดยเน้นไปยังเส้นแบ่งแยกแยะกลุ่มข้อมูลได้ดีที่สุด โดยเฉพาะกับปัญหาที่มีขนาดไม่ใหญ่มาก หลักการทำงานของ SVM จะอาศัยการใช้เส้นตรงในการแบ่งข้อมูล (Hyperplane)



รูปที่ 3 ตัวอย่างแนวคิด SVM

โดย SVM นั้น มีสูตรการทำงานดังสมการ (1)

$$\begin{aligned} h_{(0)}(x) &= w_1x_1 + w_2x_2 + \dots + w_nx_n + \\ &= w^tx + b \end{aligned} \quad (1)$$

2.6 Naïve Bayes

เป็นการจำแนกข้อมูลโดยวิธีการที่ไม่ซับซ้อน โดยการเรียนรู้ของอัลกอริทึมจะใช้หลักการของความน่าจะเป็น ซึ่งมีพื้นฐานมาจากทฤษฎีของเบย์ หรือทฤษฎีว่าด้วยโอกาสที่จะเกิดของเหตุการณ์ต่างๆ ซึ่งจะใช้การคำนวณความน่าจะเป็นแบบมีเงื่อนไขที่เรียกว่า Conditional Probability แสดงได้ดังสมการ (2)

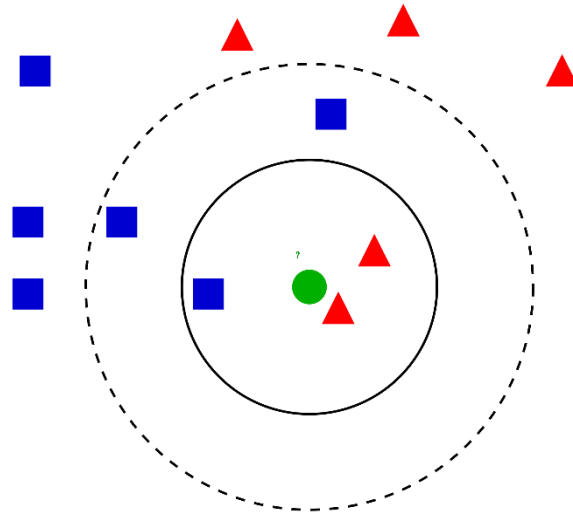
$$P(h|D) = \frac{P(D|h) \times P(h)}{P(D)} \quad (2)$$

ในปัญหาที่มีตัวแปรต้นหรือข้อมูลที่ต้องพิจารณามากกว่า 1 ค่าความน่าจะเป็นของเหตุการณ์สามารถคำนวณได้จากผลการคูณของความน่าจะเป็นของแต่ละข้อมูลที่มีคลาสเป็น h ดังสมการ (3)

$$P(h|D) = P(D_1|h) \times P(D_2|h) \times \dots \times P(D_n|h) \times h \quad (3)$$

2.7 K-nearest neighbor

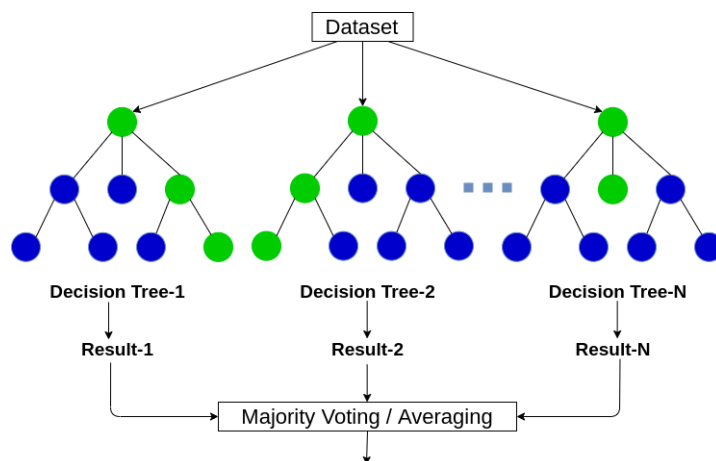
นับเป็นเทคนิคที่มีวิธีการไม่ซับซ้อนและเข้าใจได้ง่ายที่สุดที่ใช้ในการจำแนกประเภทข้อมูล โดยหลักการทำงานคือ จะใช้หลักการเปรียบเทียบความคล้ายคลึงกันของข้อมูลที่สนใจกับข้อมูลอื่นว่า มีความคล้ายคลึงหรืออยู่ใกล้กับข้อมูลใดมากที่สุด k ตัว จากนั้นจะทำการตัดสินใจว่าคำตอบของข้อมูลที่สนใจนั้นควรเป็นคำตอบเดียวกับข้อมูลที่อยู่ใกล้ที่สุด k ตัวนั้น ทั้งนี้ k คือความถี่ของข้อมูลที่อยู่ใกล้กับข้อมูลที่สนใจ โดยกำหนดค่าพารามิเตอร์ $n = 5$, $\text{algorithm} = \text{'auto'}$



รูปที่ 4 ตัวอย่างแนวคิด K-NN

2.8 Random forest

มีหลักการทำงาน คือ จะแบ่งข้อมูลออกเป็นต้นไม้ตัดสินใจหลายๆ ต้น โดยแต่ละต้นจะได้รับคุณลักษณะ และข้อมูลที่ไม่เหมือนกันทั้งหมด เพื่อให้ได้ต้นไม้ที่มีความหลากหลาย และมีความอิสระต่อกันมากขึ้น



รูปที่ 5 ตัวอย่างแนวคิด random forest

บทที่ 3

การจัดการข้อมูลและการออกแบบ

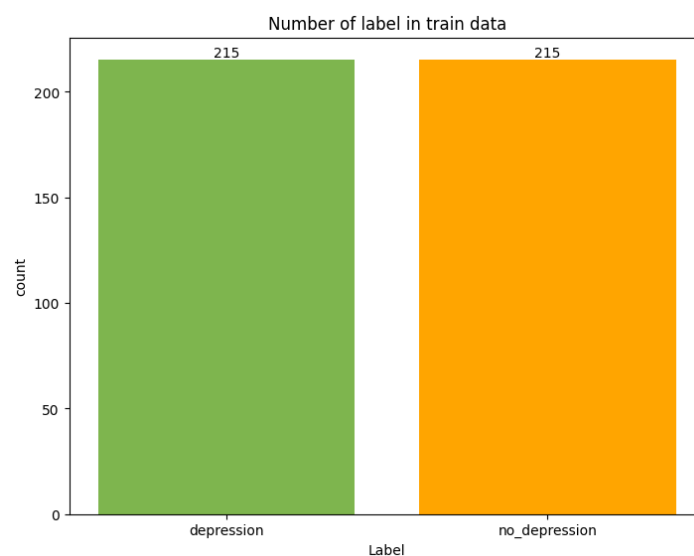
ในบทนี้จะกล่าวถึงข้อมูลที่นำมาใช้และการจัดการข้อมูลเพื่อให้อยู่ในรูปแบบที่ใช้งานได้ และการทำงานของระบบประเมินภาวะซึมเศร้าด้วยการเรียนรู้ของเครื่อง พร้อมทั้งอธิบายถึงการออกแบบ ในรูปแบบของแผนภาพต่างๆ

3.1 การวิเคราะห์และออกแบบระบบ

การที่จะประเมินอาการของผู้ที่เสี่ยงมีภาวะซึมเศร้าได้นั้น คือการจัดทำชุดข้อมูลประโยคความรู้สึกเพื่อใช้ในการจำแนกกลุ่มคำที่บ่งบอกถึงความเสี่ยงที่จะมีภาวะซึมเศร้า เช่น “เครียด”, “เบื่อ”, “ไม่อยากมีชีวิต” เป็นต้น โดยนำชุดข้อมูลมาฝึกสอนให้กับโมเดลการเรียนรู้ของเครื่อง แต่อย่างไรก็ตามอารมณ์ความรู้สึกของมนุษย์นั้นซับซ้อนเกินกว่าจะอาศัยเพียงแค่ข้อมูลประโยคความรู้สึก เนื่องจากความสามารถในการจัดการกับอารมณ์ ความรู้สึกจะแตกต่างกันไปในแต่ละบุคคล ดังนั้นเราจึงได้มีการใช้แบบประเมินสุขภาพจิต เข้ามาเก็บข้อมูลคะแนนของคำถามแต่ละข้อเพื่อให้โมเดลการเรียนรู้ของเครื่อง ประเมินอาการของผู้ที่เสี่ยงมีภาวะซึมเศร้าได้มีประสิทธิภาพมากขึ้น

3.2 ข้อมูลที่ใช้ในการพัฒนาระบบ

เพื่อการพัฒนาโมเดลให้สามารถนำมาใช้งานได้ คณะผู้จัดทำได้นำข้อมูลจากห้องรับคำปรึกษาที่เก็บรวบรวมไว้จากผู้ที่ใช้บริการของห้อง โดยข้อมูลที่นำมาใช้คือ ข้อมูลอาการเบื้องต้นหรือความรู้สึกของผู้ที่ใช้บริการ ซึ่งข้อมูลดังกล่าวอยู่ในรูปแบบของประโยคภาษาไทย โดยแยกเป็น 2 label คือ depression, no depression ทั้งหมด 430 แถว ซึ่งจะแบ่งเป็น depression จำนวน 215 ประโยค และ no depression จำนวน 215 ประโยค



รูปที่ 6 กราฟแสดงจำนวน data

โดยที่คุณลักษณะ และตัวอย่างของชุดข้อมูลมีดังนี้

ตารางที่ 3 ตารางแสดงคุณลักษณะของข้อมูล

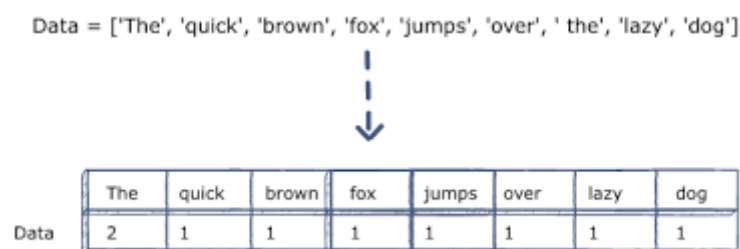
depression	อาการหรือความรู้สึกของผู้ที่มีความเสี่ยงภาวะซึมเศร้า
no depression	อาการหรือความรู้สึกของผู้ที่ไม่มีความเสี่ยงภาวะซึมเศร้า

ตารางที่ 4 ตัวอย่างข้อมูลที่นำมาใช้ในการฝึกสอนเครื่อง

ไม่อยากเกิดมาเลย	depression
ไม่อยากมีชีวิตตน	depression
หมดความตั้งใจทุกอย่าง	depression
บางครั้งก็จิกขาตัวเอง คิดว่าตัวเองไร้ค่า	depression
อารมณ์ดิ่งมาก	depression
ผู้คนน่ารักรายล้อม	no_depression
ฉันท่องมากแล้ว	no_depression
มีน้ำใจ	no_depression
ฉันจะคอยอยู่เคียงข้างเธอเอง	no_depression
ตอนนี้ก็มีโปรโมชั่นลดราคาก็เลยส่งให้แม่ดู	no_depression

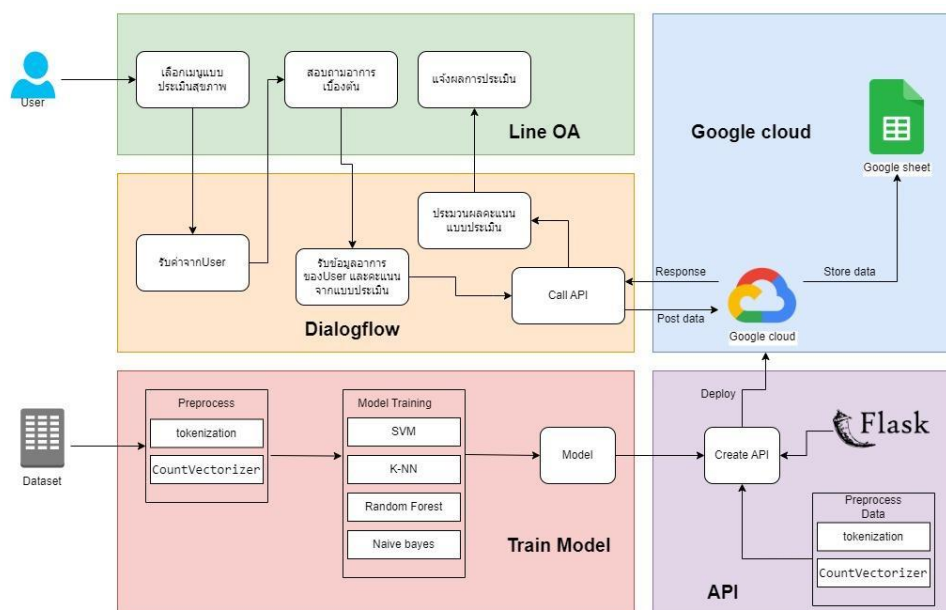
3.2.1 การจัดการข้อมูล

เราได้ทำการนำข้อมูลทั้งหมด ทำการ tokenization หรือทำการตัดคำ โดยการตัดคำเราได้เลือกใช้ attacut library จะได้ คำที่แยกออกเป็น token และนำ token ที่ได้มาสร้างเป็น matrix โดยใช้กลุ่มของคำที่มีเป็นตัวอ้างอิง เพื่อให้ข้อมูลอยู่ในรูปแบบที่พร้อมใช้งานก่อนที่จะนำไปฝึกสอนให้กับโมเดลการเรียนรู้ของเครื่อง



รูปที่ 7 ตัวอย่างการทำTokenization

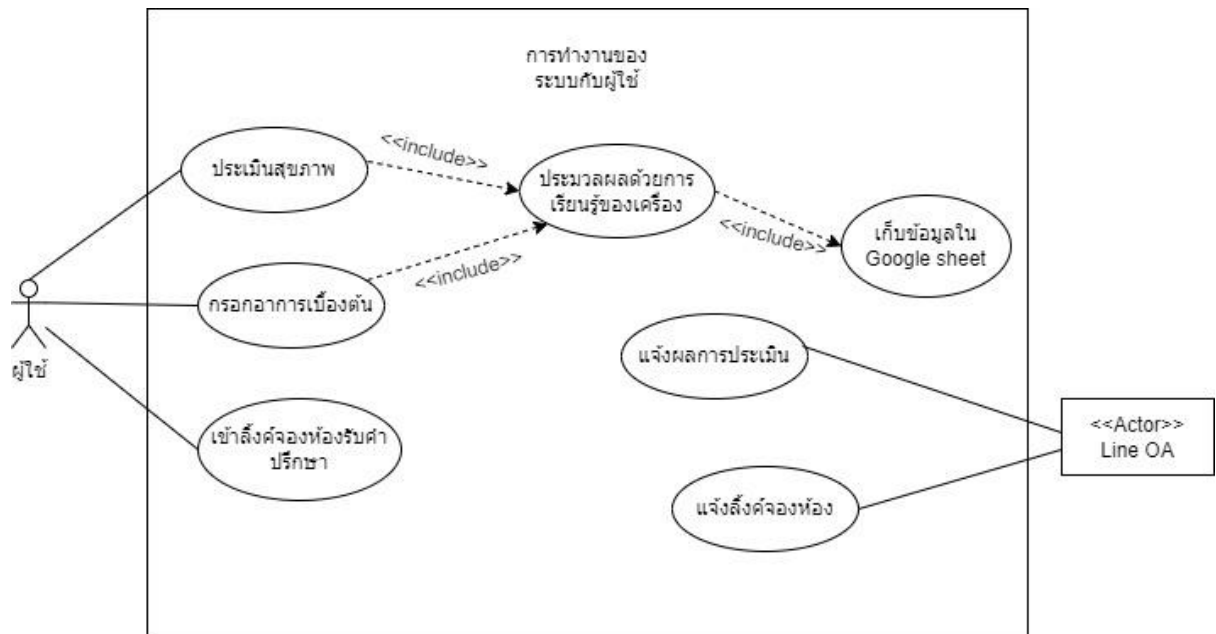
3.3 การทำงานของระบบ



รูปที่ 8 การทำงานของระบบประเมินภาวะซึมเศร้าด้วยปัญญาประดิษฐ์

เริ่มต้นจากการที่ผู้ใช้เลือกใช้เมนูการประเมินสุขภาพจิตผ่าน Line OA และตอบแบบประเมินสุขภาพจนครบ จากนั้นระบบจะรับข้อมูลคำตอบของผู้ใช้เข้ามาคำนวณผ่าน Dialogflow และทำการสอบถามอาการของผู้ใช้เบื้องต้น และส่งข้อมูลอาการของผู้ใช้และคะแนนจากแบบประเมินมาคำนวณด้วย Machine Learning ในส่วนของ Model นั้นได้นำข้อมูลมาทำการ Preprocessing เพื่อให้ข้อมูลอยู่ในรูปแบบที่ใช้งานได้ และนำมา Train ด้วย Model ต่างๆ หลังจากที่เสร็จสิ้นการคำนวณด้วย Machine Learning แล้วจะบันทึกข้อมูลอาการของผู้ใช้, คะแนนจากแบบประเมินสุขภาพจิต และผลการคำนวณไว้ใน Google cloud เพื่อนำข้อมูลมาใช้พัฒนาระบบต่อไป

3.4 Usecase diagram



รูปที่ 9 การทำงานของระบบกับผู้ใช้

Use Case Description

คำอธิบายรายละเอียดของ Use Case ทั้ง 7 Use Case มีดังนี้

1) Use case ประเมินสุขภาพ

ตารางที่ 5 คำอธิบาย Use case ประเมินสุขภาพ

Use case name	ประเมินสุขภาพ
Participating	ผู้ใช้
Entry Condition	เพื่อประเมินสุขภาพด้วยแบบประเมิน PHQ-9
Flow of Event	ผู้ใช้ตอบคำถามทั้งหมด 9 ข้อ โดยตอบเป็นตัวเลข 0-3
Exit Condition	ตอบคำถามครบทั้งหมด

2) Use case กรอกอาการเบื้องต้น

ตารางที่ 6 คำอธิบาย Use case กรอกอาการเบื้องต้น

Use case name	กรอกอาการเบื้องต้น
Participating	ผู้ใช้
Entry Condition	กรอกอาการหรือความรู้สึกที่ไม่ปกติ
Flow of Event	ผู้ใช้กรอกอาการหรือความรู้สึกที่ผิดปกติ ในระยะ 2-3 สัปดาห์ที่ผ่านมา โดยกรอกเป็นข้อความโดยการพิมพ์เข้ามาเพียงครั้งเดียว
Exit Condition	ผู้ใช้ส่งข้อมูลอาการ

3) Use case เข้าลิ้งค์จองห้องรับคำปรึกษา

ตารางที่ 7 คำอธิบาย Use case เข้าลิ้งค์จองห้องรับคำปรึกษา

Use case name	เข้าลิ้งค์จองห้องรับคำปรึกษา
Participating	ผู้ใช้
Entry Condition	เข้าเว็บไซต์สำหรับจองใช้ห้องรับคำปรึกษา
Flow of Event	1.ผู้ใช้เลือกเมนูจองห้องรับคำปรึกษา 2.เข้าหน้าเว็บไซต์ของห้องรับคำปรึกษา
Exit Condition	ผู้ใช้ออกจากเว็บไซต์

4) Use case ประมวลผลด้วยการเรียนรู้ของเครื่อง

ตารางที่ 7 คำอธิบาย Use case ประมวลผลด้วยการเรียนรู้ของเครื่อง

Use case name	ประมวลผลด้วยการเรียนรู้ของเครื่อง
Participating	ผู้ใช้
Entry Condition	นำข้อมูลอาการและคะแนนจากแบบประเมินมาประมวลผลด้วยการเรียนรู้ของเครื่อง
Flow of Event	1.ส่งข้อมูลด้วย api 2.เรียกใช้ Model เพื่อประมวลผล
Exit Condition	ประมวลผลเสร็จสิ้น

5) Use case เก็บข้อมูลใน Google sheet

ตารางที่ 8 คำอธิบาย Use case เก็บข้อมูลใน Google sheet

Use case name	เก็บข้อมูลใน Google sheet
Participating	ผู้ใช้
Entry Condition	นำข้อมูลอาการและคะแนนจากแบบประเมินมาบันทึกใน Google sheet
Flow of Event	บันทึกข้อมูลคำตอบของแบบประเมิน PHQ-9, ผลคะแนนรวมของแบบประเมิน, ข้อมูลอาการของผู้ใช้และผลการประเมินที่ประมวลผลไว้ใน Google sheet
Exit Condition	บันทึกข้อมูลเสร็จสิ้น

6) Use case แจ้างผลการประเมิน

ตารางที่ 9 คำอธิบาย Use case แจ้างผลการประเมิน

Use case name	แจ้างผลการประเมิน
Participating	Line OA
Entry Condition	แจ้างผลการประเมินของผู้ใช้
Flow of Event	แจ้างผลการประเมินของผู้ใช้ตามเกณฑ์ ความเสี่ยงโดยแบ่งเป็น ไม่มีอาการ,น้อย, ปานกลาง,มากและรุนแรง
Exit Condition	แจ้างผลการประเมินกลับไปหาผู้ใช้

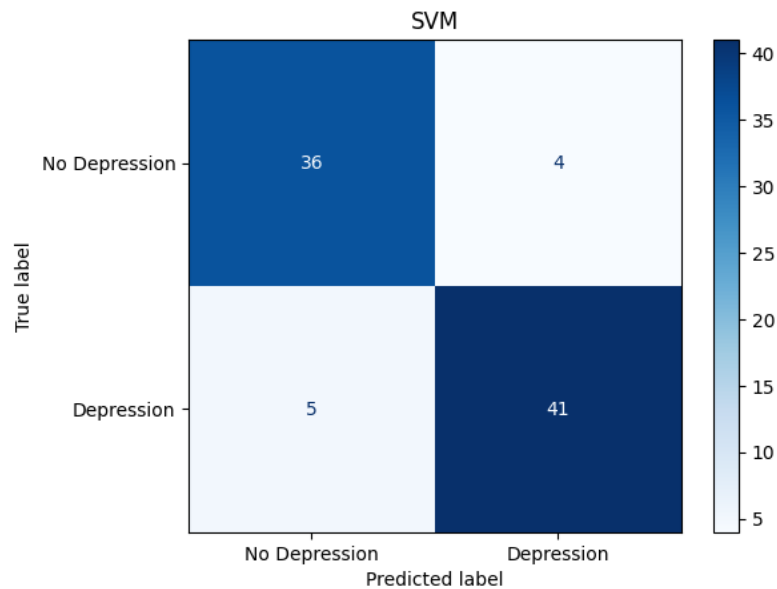
7) Use case แจ้างลิ้งค์จองห้องรับคำปรึกษา

ตารางที่ 10 คำอธิบาย Use case แจ้างลิ้งค์จองห้องรับคำปรึกษา

Use case name	แจ้างลิ้งค์จองห้องรับคำปรึกษา
Participating	Line OA
Entry Condition	แจ้างลิ้งค์สำหรับจองห้องรับคำปรึกษาให้กับ ผู้ใช้
Flow of Event	แจ้างลิ้งค์จองห้องรับคำปรึกษาไปยังผู้ใช้
Exit Condition	แจ้างลิ้งค์จองห้องรับคำปรึกษาไปยังผู้ใช้ เสร็จสิ้น

3.5 การพัฒนาโดยใช้โมเดล Support Vector Machine

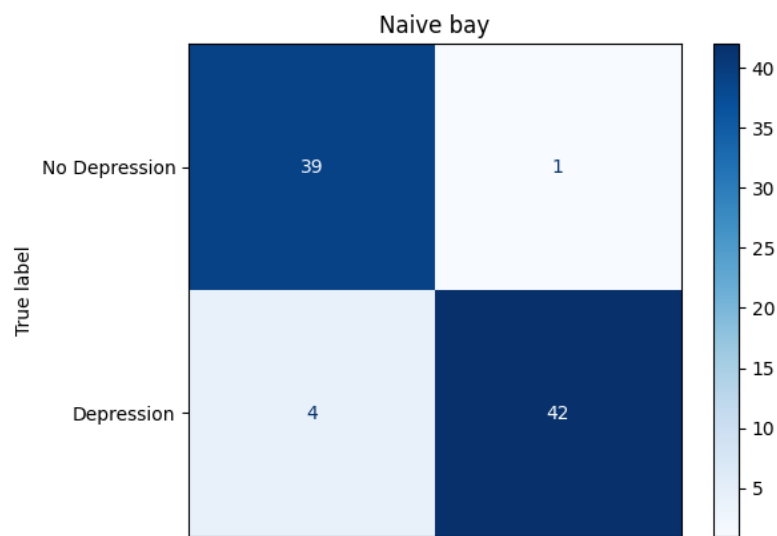
การใช้โมเดล Support Vector Machine ในการประเมินความเสี่ยงภาวะซึมเศร้าโดยแบ่งข้อมูลออกเป็น 2 ชุด โดยใช้ Train test split ในการแบ่งข้อมูล ซึ่งจะแบ่งเป็นชุดข้อมูลสำหรับฝึกสอน 80% และชุดข้อมูลสำหรับทดสอบ 20% เมื่อทำการฝึกสอนเสร็จสิ้น ผลลัพธ์ที่ได้จากโมเดล ดังรูปที่ 10



รูปที่ 10 ผลลัพธ์ของโมเดล Support vector machine

3.6 การพัฒนาโดยใช้โมเดล Naïve Bayes

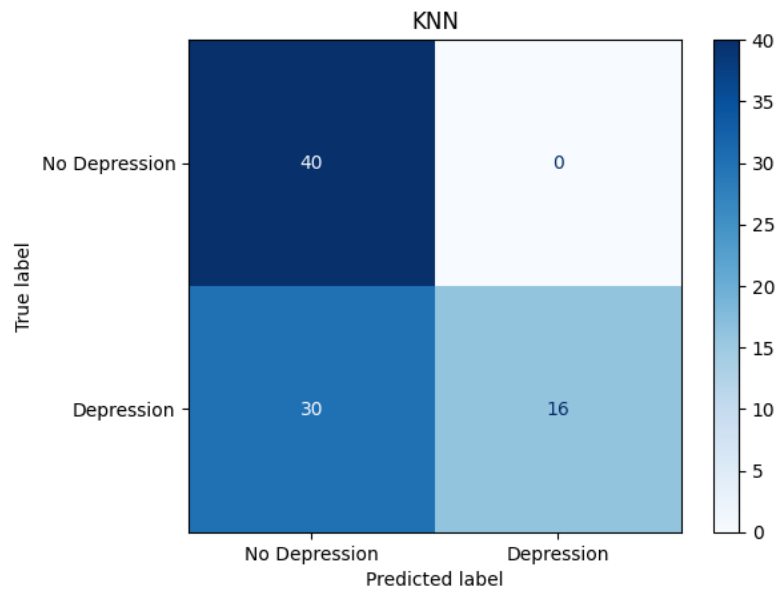
การใช้โมเดล Naïve Bayes ในการประเมินความเสี่ยงภาวะซึมเศร้าโดยแบ่งข้อมูล ออกเป็น 2 ชุด โดยใช้ Train test split ในการแบ่งข้อมูล ซึ่งจะแบ่งเป็นชุดข้อมูลสำหรับฝึกสอน 80% และชุดข้อมูลสำหรับทดสอบ 20% เมื่อทำการฝึกสอนเสร็จสิ้น ผลลัพธ์ที่ได้จากโมเดล ดังรูปที่ 11



รูปที่ 11 ผลลัพธ์ของโมเดล Naïve Bayes

3.7 การพัฒนาโดยใช้โมเดล K-nearest

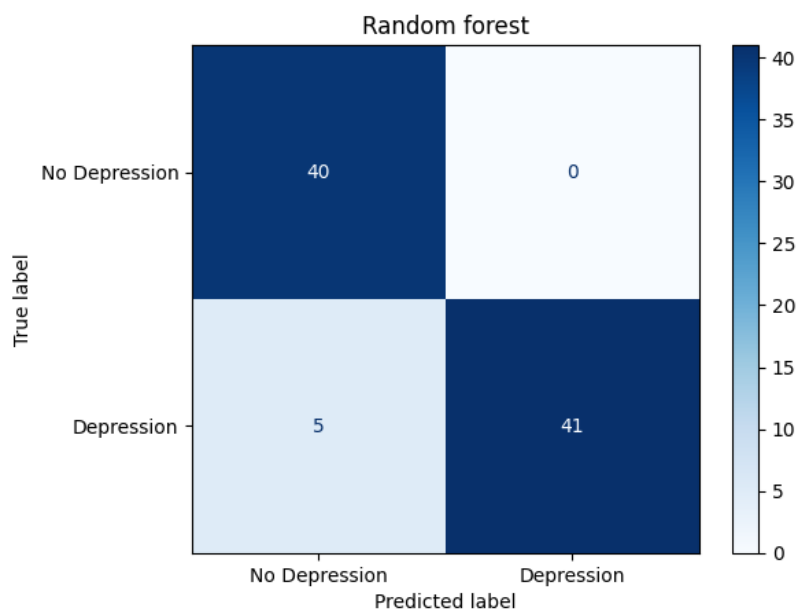
การใช้โมเดล K-nearest neighbor ในการประเมินความเสี่ยงภาวะซึมเศร้าโดยแบ่งข้อมูลออกเป็น 2 ชุด โดยใช้ Train test split ในการแบ่งข้อมูล ซึ่งจะแบ่งเป็นชุดข้อมูลสำหรับฝึกสอน 80% และชุดข้อมูลสำหรับทดสอบ 20% เมื่อทำการฝึกสอนเสร็จสิ้น ผลลัพธ์ที่ได้จากโมเดล ดังรูปที่ 12



รูปที่ 12 ผลลัพธ์ของโมเดล Naïve Bayes

3.8 การพัฒนาโดยใช้โมเดล Random-forest

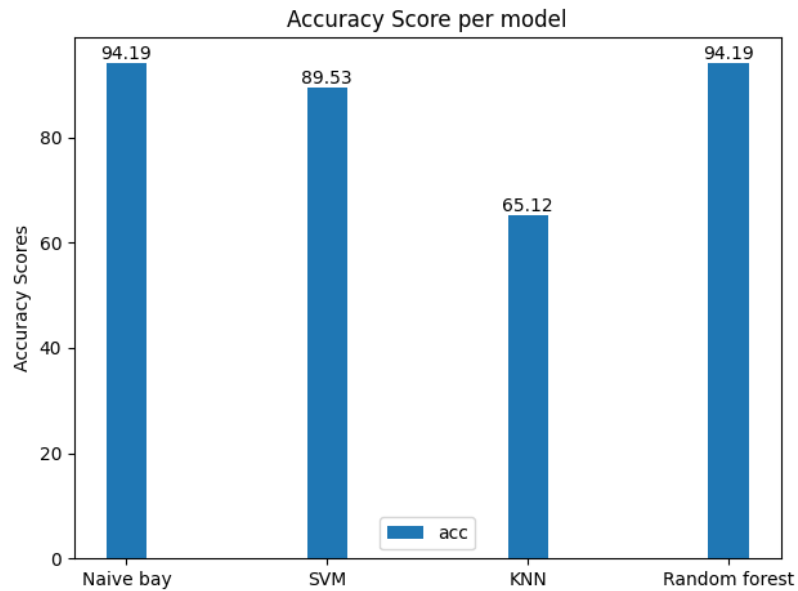
การใช้โมเดล Random-forest ในการประเมินความเสี่ยงภาวะซึมเศร้าโดยแบ่งข้อมูลออกเป็น 2 ชุด โดยใช้ Train test split ในการแบ่งข้อมูล ซึ่งจะแบ่งเป็นชุดข้อมูลสำหรับฝึกสอน 80% และชุดข้อมูลสำหรับทดสอบ 20% เมื่อทำการฝึกสอนเสร็จสิ้น ผลลัพธ์ที่ได้จากโมเดล ดังรูปที่ 13



รูปที่ 13 ผลลัพธ์ของโมเดล Random-forest

3.9 ค่าความถูกต้องของโมเดลการเรียนรู้ของเครื่อง

จากการฝึกสอนโมเดลการเรียนรู้ของเครื่อง ในการประเมินความเสี่ยงภาวะซึมเศร้าโดยแบ่งข้อมูลออกเป็น 2 ชุด ในแต่ละโมเดลจะมีค่าความถูกต้องดังรูปที่ 14



รูปที่ 14 กราฟผลลัพธ์ค่าความถูกต้องของทั้ง 4 โมเดล

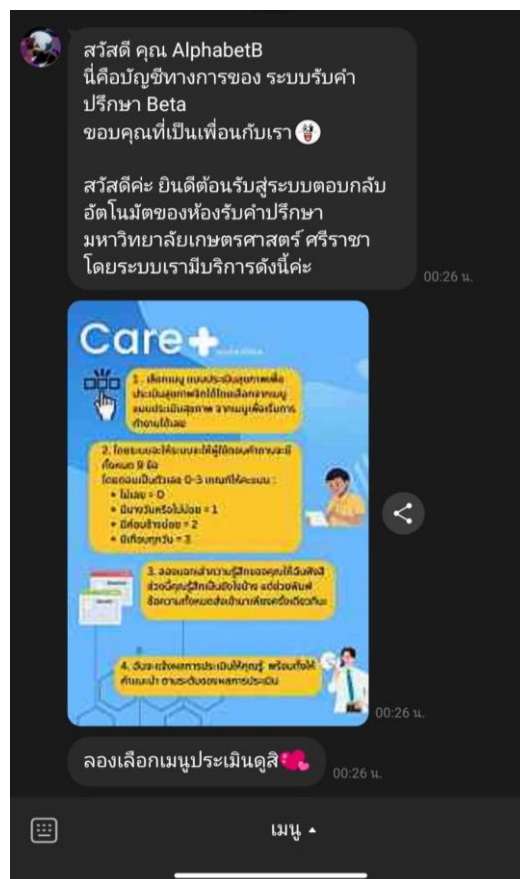
บทที่ 4

การทำงานและทดสอบระบบ

โดยในบทนี้ จะอธิบายถึงขั้นตอนการใช้งานของ”ระบบประเมินภาวะซึมเศร้าด้วยการเรียนรู้ของเครื่อง” รวมถึงผลลัพธ์ที่ได้จากระบบประเมิน โดยมีขั้นตอนการทำงานของระบบดังต่อไปนี้

4.1 การทำงานของระบบใน Line OA

ในส่วนแรกสุด คือการทำงานของ Line ซึ่งเป็นส่วนที่ใช้ในการติดต่อกับผู้ใช้งาน และรับข้อมูลจากผู้ใช้งานเพื่อส่งไปประมวลผลในโมเดลการเรียนรู้ของเครื่อง โดยเริ่มแรกหลังจากทำการเพิ่มเพื่อนกับ Line ของระบบประเมินภาวะซึมเศร้า ระบบจะส่งข้อความทักทาย และแนะนำวิธีการใช้งานระบบ โดยข้อความที่ระบบส่งเข้ามาเป็นดัง รูปที่15



รูปที่ 15 ตัวอย่างข้อความแนะนำการใช้งานระบบ

เมื่อเข้าสู่ระบบ Line แล้ว ผู้ใช้สามารถเลือก “เมนู” เพื่อเปิดริชเมนูของระบบและเลือกใช้เมนูที่ต้องการได้ โดยหน้าริชเมนูของระบบเป็นดัง รูปที่16

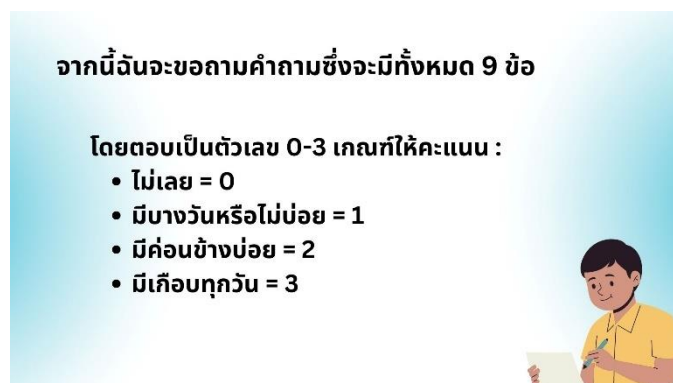


รูปที่ 16 ตัวอย่างริชเมนูของระบบ

โดยริชเมนูของระบบจะมีเมนูให้เลือกใช้ 4 รายการ ดังนี้

- แบบประเมินสุขภาพจิต
เริ่มการประเมินภาวะซึมเศร้าของผู้ใช้งาน
- ลิงค์สำหรับจองห้อง
ช่องทางการจองห้องรับคำปรึกษา โดยจะส่งผู้ไปยังหน้าเว็บไซต์ของห้องรับคำปรึกษา
- ข้อมูลการดูแลสุขภาพจิต
ศึกษาข้อมูลการดูแลสุขภาพจิต จากเว็บไซต์ของโรงพยาบาลมหารมย์
- เกี่ยวกับระบบ
แนะนำการใช้งานระบบ

การทำงานของระบบจะเริ่มเมื่อผู้ใช้เลือกเมนู “แบบประเมินสุขภาพจิต” หรือพิมพ์ “แบบประเมิน” และส่งผ่านแชทเพื่อเริ่มการทำงาน โดยจะให้ผู้ใช้ตอบคำถามทั้งหมด 9 ข้อ โดยตอบเป็นตัวเลข 0-3 ซึ่งมีเกณฑ์การให้คะแนนดัง รูปที่ 17



รูปที่ 17 เกณฑ์การให้คะแนน

คำถามทั้ง 9 ข้อ อ้างอิงมาจากแบบประเมินสุขภาพจิต PHQ-9 ของโรงพยาบาลรามาริบดี คณะแพทยศาสตร์โรงพยาบาลรามาริบดี มหาวิทยาลัยมหิดล โดยมีรายละเอียดของคำถาม ดัง ตารางที่ 10

ตารางที่ 11 ตัวอย่างคำถามจากแบบประเมิน PHQ-9

ข้อที่	คำถาม
1	เบื่อ ทำอะไร ๆ ก็ไม่เพลิดเพลิน
2	ไม่สบายใจ ซึมเศร้า หรือท้อแท้
3	หลับยาก หรือหลับ ๆ ตื่น ๆ หรือหลับมากเกินไป
4	เหนื่อยง่าย หรือไม่ค่อยมีแรง
5	เบื่ออาหาร หรือกินมากเกินไป
6	รู้สึกไม่ดีกับตัวเอง คิดว่าตัวเองล้มเหลว หรือเป็นคน ทำให้ตัวเอง หรือครอบครัวผิดหวัง
7	สมาธิไม่ดีเวลาทำอะไร เช่น ดูโทรทัศน์ ฟังวิทยุ หรือ ทำงานที่ต้องใช้ความตั้งใจ
8	พูดหรือทำอะไรซ้ำจนคนอื่นมองเห็น หรือ กระสับกระส่ายจนท่านอยู่ไม่นิ่งเหมือนเคย
9	คิดทำร้ายตนเอง หรือคิดว่าถ้าตาย ๆ ไปเสียคงจะดี

หลังจากที่ผู้ใช้ ตอบคำถามครบทั้ง 9 ข้อ ระบบจะสอบถามความรู้สึกที่ผ่านมาในช่วง 2-3 สัปดาห์
 ของผู้ใช้ โดยพิมพ์บอกเล่าความรู้สึกของผู้ใช้ผ่านทางข้อความ หลังจากนั้นระบบจะนำข้อมูล
 ไปประมวลผล และแจ้งผลลัพธ์ของการประเมินด้วยโมเดลและผลลัพธ์จากแบบประเมิน PHQ-9
 กลับมายังผู้ใช้ ดังรูปที่ 18 และรูปที่ 19



รูปที่ 19 ตัวอย่างการตอบกลับหลังตอบคำถามครบ



รูปที่ 18 ระบบส่งผลลัพธ์กลับมายังผู้ใช้

4.2 การทำงานของระบบใน Dialogflow

ทำการสร้างโปรเจกใน Dialogflow กำหนดชื่อโปรเจก และภาษา, โซนเวลาเป็นไทยดัง รูปที่ 21

Agent name CREATE

DEFAULT LANGUAGE ? Thai — th
Primary language for your agent. Other languages can be added later.

DEFAULT TIME ZONE (GMT+7:00) Asia/Bangkok
Date and time requests are resolved using this timezone if not provided in the API requests.

GOOGLE PROJECT Create a new Google project
Enables Cloud functions, Actions on Google and permissions management.

AGENT TYPE Set as Mega Agent
Combine multiple Dialogflow agents (i.e. sub agents) into a single agent (i.e. mega agent).

รูปที่ 20 การสร้างโปรเจกDialogflow

ทำการสร้าง intent เพื่อตอบกลับสิ่งที่ผู้ใช้ต้องการ โดยในโครงงานนี้ intent หลักที่ใช้ในการทำงานของระบบ ใช้ชื่อ intent ว่า แบบประเมินPHQ-9 ภายใน intent ให้กำหนดรูปแบบของคำ หรือประโยคที่คาดว่าผู้ใช้จะสื่อว่าต้องการใช้งานในส่วนนี้ใน Training phrases ดังตัวอย่างในรูปที่ 21

Dialogflow Essentials Global SAVE

Training phrases Search training phrases

Template phrases are deprecated and will be ignored in training time. More details [here](#).

When a user says something similar to a training phrase, Dialogflow matches it to the intent. You don't have to create an exhaustive list. Dialogflow will fill out the list with similar expressions. To extract parameter values, use [annotations](#) with available [system](#) or [custom](#) entity types.

รูปที่ 21 การกำหนด Training phrases

กำหนดการตอบกลับไปยังผู้ใช้งานในส่วนของ Responses โดยการตอบกลับนั้นสามารถกำหนดรูปแบบได้หลากหลายไม่ว่าจะเป็นการตอบกลับด้วยข้อความ, รูปภาพ, การ์ด, หรือกำหนดรูปแบบโดยเขียนรูปแบบการตอบกลับด้วย Json ดังรูปที่ 22

Responses ?

DEFAULT LINE +

Responses from this tab will be sent to the Line integration.
Use responses from the DEFAULT tab as the first responses.

Image

จากนี้มันจะออกค่าตามนี้จะมียกยอด 9 ข้อ

โดยตอบเป็นตัวเลข 0-3 เกมที่ให้คะแนน :

- ไม่เลย = 0
- เริ่มรู้สึกเริ่มไม่พอใจ = 1
- เริ่มไม่ชอบ = 2
- เริ่มเกลียด = 3

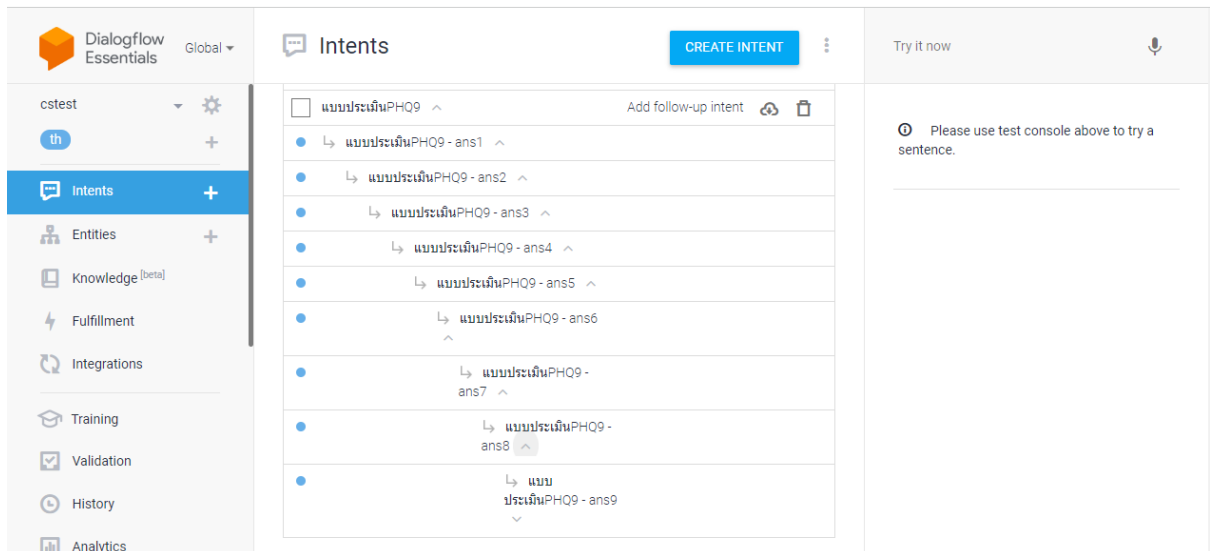
<https://scontent.fbkk12-3.fna.fbcdn.net/v/>

Text Response

1	เขานั่นจะเริ่มเล่นนะ
2	Enter a text response variant

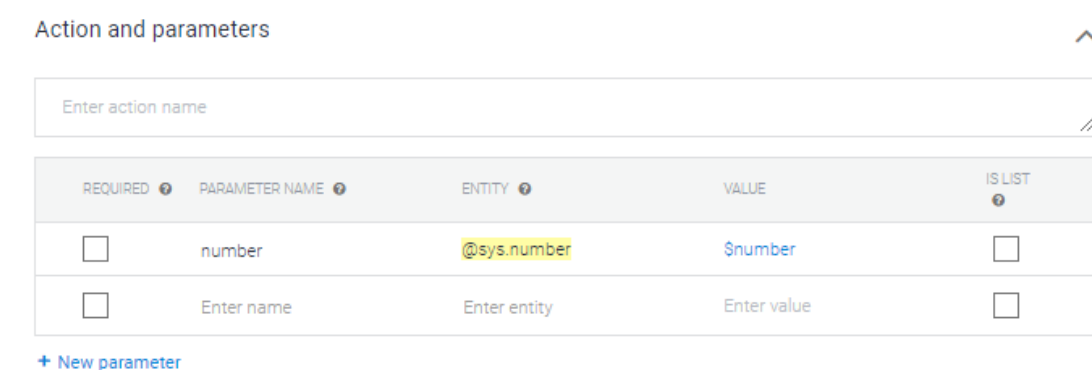
รูปที่ 22 การกำหนดรูปแบบการตอบกลับ

หลังจากที่สร้าง intent สำหรับเริ่มการทำงานแล้ว ให้ทำการสร้าง intent เพื่อรับค่าคำตอบจากผู้ใช้ โดยการเลือกเมนู add follow-up intent ซึ่งจะเป็น intent ที่ทำงานถัดจาก intent หลักต่อกันในรูปแบบของลำดับขั้น โดยในโครงงานนี้จะรับค่าคำตอบจากผู้ใช้ทั้งหมด 10 ครั้ง คือคำตอบจากคำถามทั้ง 9 ข้อ และข้อความบอกความรู้สึกของผู้ใช้ ดังรูปที่ 23



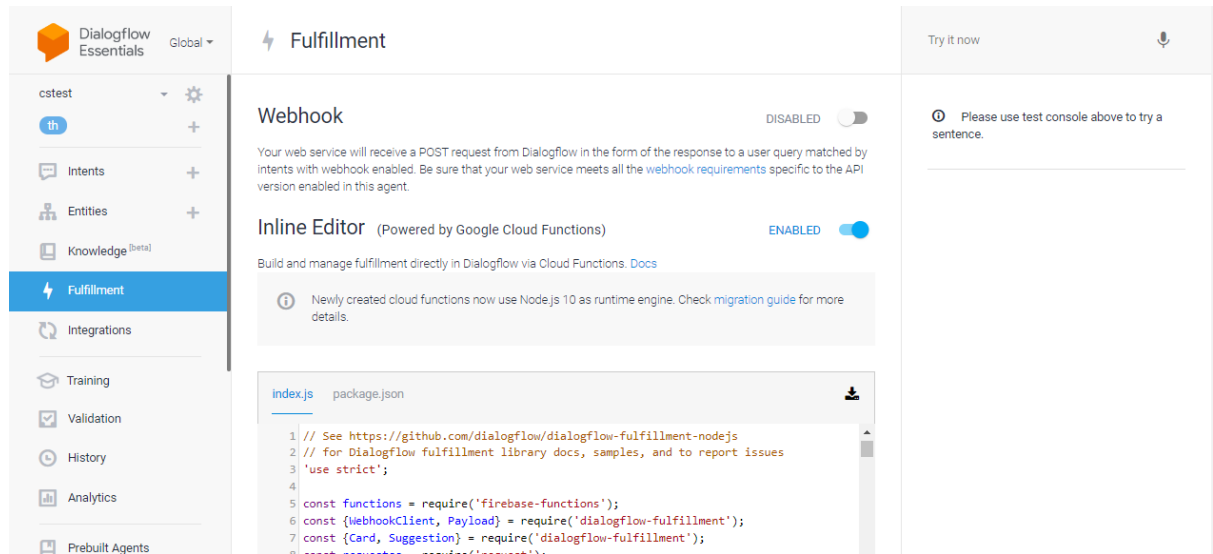
รูปที่ 23 สร้าง follow up intent เพื่อรับข้อมูลจากผู้ใช้

หลังจากที่สร้าง Follow-up เรียบร้อยแล้ว ในการรับค่าจากคำตอบของผู้ใช้ จะต้องทำการกำหนด Parameter ขึ้นมาเพื่อรับค่าคำตอบของผู้ใช้มาเก็บไว้ โดยกำหนด Parameter และชนิดของค่าที่จะรับมาใน Action and Parameter ดังตัวอย่างใน รูปที่ 24



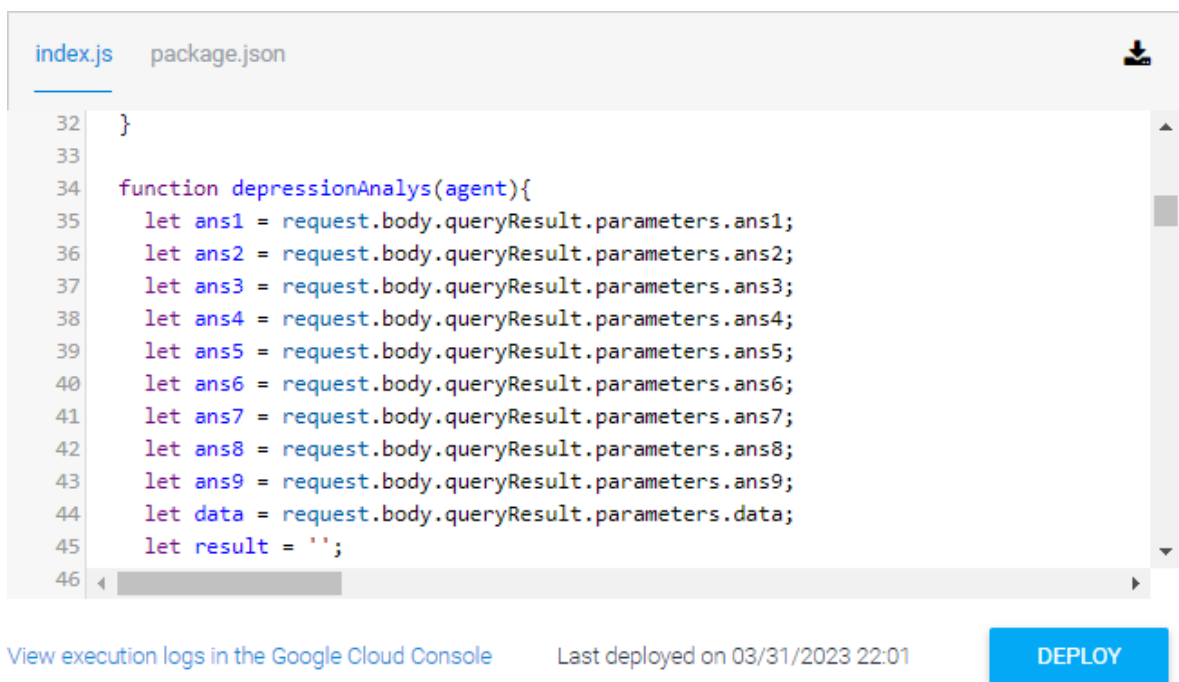
รูปที่ 24 กำหนด parameter เพื่อรับค่าจากผู้ใช้

เมื่อรับค่าคำตอบจากผู้ใช้ทั้งหมดแล้ว เลือกเมนู Fulfillment และเปิดใช้งาน Inline editor เพื่อนำข้อมูลที่ได้รับมาคำนวณผลการประเมิน ดังรูปที่ 25



รูปที่ 25 เปิดใช้งาน Inline editor ใน fulfillment

สร้างฟังก์ชันสำหรับเรียกใช้ Parameter ที่รับจากผู้ใช้ใน index.js และนำคะแนนจากแบบประเมินทั้งหมดมารวมกัน เพื่อใช้ในการประเมินผล ดังรูปที่ 26 และรูปที่ 27



รูปที่ 26 สร้างฟังก์ชันสำหรับนำข้อมูลมาประมวลผล

```
let depress = (ans1+ans2+ans3+ans4+ans5+ans6+ans7+ans8+ans9);
```

รูปที่ 27 รวมคะแนนจากแบบประเมิน

จากนั้นทำการเรียกใช้โมเดลการเรียนรู้ของเครื่องผ่าน API โดยส่งข้อมูลทั้งหมดไปยัง Google cloud เพื่อทำการประมวลผล และบันทึกข้อมูลไว้ใช้พัฒนาระบบต่อไป โดยข้อมูลที่ส่งไปประมวลผล และเก็บบันทึกไว้มีดังนี้

- Sentence : ประโยคที่ผู้ใช้งานกรอกเข้ามา
- Q1 : คำตอบจากคำถามข้อที่ 1
- Q2 : คำตอบจากคำถามข้อที่ 2
- Q3 : คำตอบจากคำถามข้อที่ 3
- Q4 : คำตอบจากคำถามข้อที่ 4
- Q5 : คำตอบจากคำถามข้อที่ 5
- Q6 : คำตอบจากคำถามข้อที่ 6
- Q7 : คำตอบจากคำถามข้อที่ 7
- Q8 : คำตอบจากคำถามข้อที่ 8
- Q9 : คำตอบจากคำถามข้อที่ 9
- Qscore : ผลรวมจากคำถามทั้ง 9 ข้อ

จากนั้นทำการเรียก API เพื่อนำข้อมูลไปประมวลผลดังรูปที่ 28

```
requestss.post(  
    'https://predictapi-fdy3hvaf6q-as.a.run.app/',  
    { json: { sentence: data,  
              Q1:ans1,  
              Q2:ans2,  
              Q3:ans3,  
              Q4:ans4,  
              Q5:ans5,  
              Q6:ans6,  
              Q7:ans7,  
              Q8:ans8,  
              Q9:ans9,  
              Qscore:depress} }},
```

รูปที่ 28 เรียกใช้ API เพื่อนำข้อมูลไปประมวลผล

ทำการรับค่าผลลัพธ์ที่ได้จากโมเดลกลับมาที่ Dialogflow และทำนำผลลัพธ์ที่ได้มารวมกับผลคะแนนจากแบบประเมิน เพื่อกำหนดผลการตอบกลับไปยังผู้ใช้อย่างรูปที่ 29 และรูปที่ 30

```
function (error, response, body) {  
  if (!error && response.statusCode == 200) {  
    predict = body.prediction;  
  }  
  else if(response.statusCode != 200){  
    //return 'API Error';  
  }  
}
```

รูปที่ 29 รับค่าผลลัพธ์จากโมเดล

```
switch (true) {  
  case depress <= 4:  
    result = 1;  
    break;  
  case depress >= 5 && depress <= 8:  
    result = 2;  
    break;  
  case depress >= 9 && depress <= 14:  
    result = 3;  
    break;  
  case depress >= 15 && depress <= 19:  
    result = 4;  
    break;  
}
```

รูปที่ 30 กำหนดผลลัพธ์การตอบกลับ

4.3 การสร้างและการทำงานของ API

เริ่มต้นด้วยการเรียกใช้ Library ต่างๆที่จำเป็นในการสร้าง API เพื่อให้เรียกใช้โมเดลการเรียนรู้

ของเครื่องได้ โดย Library ที่มีการเรียกใช้งานมีดัง รูปที่ 31

```
import numpy as np
from attacut import tokenize
from sklearn.feature_extraction.text import CountVectorizer
import joblib
from flask import Flask, request, jsonify
import pandas as pd
import gspread
from gspread_dataframe import set_with_dataframe
from google.oauth2.service_account import Credentials
from pydrive.auth import GoogleAuth
from pydrive.drive import GoogleDrive
```

รูปที่ 31 library ที่มีการเรียกใช้งาน

ทำการโหลดโมเดลที่ใช้ โดยในโครงงานนี้ได้เลือกใช้ Naïve Bayes ซึ่งมีค่าความถูกต้องมากที่สุดมาใช้งาน และโหลดคลังคำศัพท์ขึ้นมาเพื่อรอการเรียกใช้จาก API ดังรูปที่ 32

```
f = open('Naive.pickle', 'rb')
classifier = joblib.load(f)
f.close()

f = open('CountVector.pickel', 'rb')
token = joblib.load(f)
f.close()
```

รูปที่ 32 การโหลดโมเดลและคลังคำศัพท์

จากข้อมูลที่ได้รับมาจากผู้ใช้สร้างฟังก์ชันสำหรับการตัดคำด้วย library attacut เพื่อให้ข้อมูลอยู่ในรูปแบบที่สามารถนำไปใช้งานได้ ดังรูปที่ 33

```
def sent2idx(x):  
    xidx = []  
    for w in tokenize(x):  
        xidx.append(w)  
    return np.array(xidx)
```

รูปที่ 33 ทำการตัดคำ

สร้างฟังก์ชันสำหรับการทำนายผลเริ่มด้วยการรับค่าจาก API จากนั้นเรียกใช้งานฟังก์ชันสำหรับการตัดคำ หลังจากนั้นทำการนับเทียบกับคลังคำศัพท์ที่ใช้สอนโมเดล แล้วนำเข้าโมเดลเพื่อให้ทำนายผลออกมา ดังรูปที่ 34

```
def prediction(x):  
    word = []  
    word.append(sent2idx(x))  
  
    predict_list = [ ', '.join(tkn) for tkn in word]  
    cvec_train = CountVectorizer(analyzer=lambda x:x.split(','))  
    c_feat = cvec_train.fit(token)  
    c_feat_train = cvec_train.transform(predict_list)  
    predict = classifier.predict(c_feat_train)  
    return predict
```

รูปที่ 34 นำข้อมูลมาประมวลผลด้วยโมเดล

ทำการเชื่อมต่อกับ API ของ Google เพื่อส่งข้อมูลที่ได้รับจาก API มาบันทึกใน Google sheet เพื่อเป็นการเก็บข้อมูลไว้ใช้พัฒนาระบบต่อไป สิ่งที่สำคัญคือ credentials ที่ได้จาก Google API และ key ที่ใช้เชื่อมกับ Google sheet หลังจากนั้นเลือก Sheet ที่ต้องการบันทึกข้อมูล ดังรูปที่ 35

```
scopes = ['https://www.googleapis.com/auth/spreadsheets',  
| | | | 'https://www.googleapis.com/auth/drive']  
  
credentials = Credentials.from_service_account_file('cspro-381810-5602e6154eea.json', scopes=scopes)  
  
gc = gspread.authorize(credentials)  
  
gauth = GoogleAuth()  
drive = GoogleDrive(gauth)  
  
# open a google sheet  
gs = gc.open_by_key('1BNbitueWA4MTv2ufNpyjAxMdPRcMhnmR5u7Mh-pg5xc')  
# select a work sheet from its name  
worksheet1 = gs.worksheet('Sheet1')
```

รูปที่ 35 กำหนด Path ของ Google sheet สำหรับบันทึกข้อมูล

ในส่วนของ API จะเริ่มด้วยการตรวจสอบ Method ที่เรียกเข้ามาว่าเป็น Post หรือไม่ แล้วทำการดึงข้อมูลที่ส่งมาเป็น Json เข้ามาและตรวจสอบว่าค่าที่รับเข้ามานั้นเป็นค่าว่างหรือไม่ หลังตรวจสอบเสร็จแล้ว นำข้อมูลที่รับเข้ามาไปผ่านการตัดคำและทำนายความเสี่ยงการเป็นภาวะซึมเศร้า

สร้างรูปแบบการแสดงผลข้อมูลตอบกลับจาก API

สร้างรูปแบบการเก็บข้อมูลเข้า Google sheet

จากนั้นทำการบันทึกข้อมูลใน Google sheet จากนั้นทำการส่ง Response กลับไปที่ Dialogflow โดยมรการทำงาน ดังรูปที่ 36

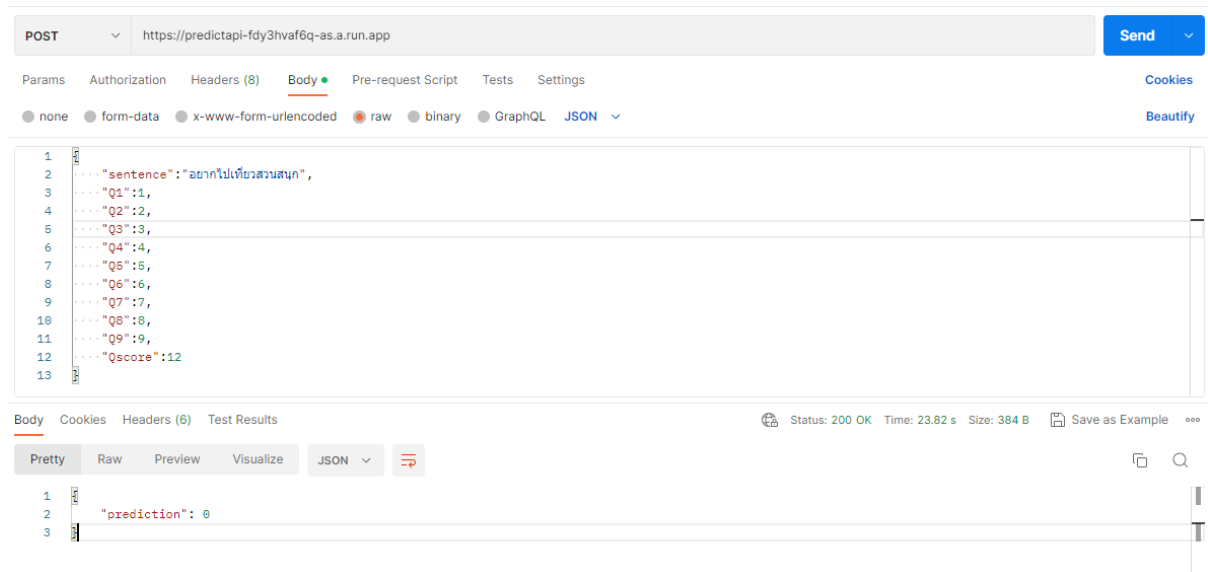
```
@app.route("/", methods=["GET", "POST"])
def index():
    if request.method == "POST":
        request_data = request.get_json()
        if request_data is None:
            return jsonify({"error": "no data"})

        try:
            sent = request_data['sentence']
            print(request_data)
            app.logger.info('', request_data)
            predic = prediction(sent)
            data = {"prediction": int(predic)}
            sheet = {
                "sentence": request_data['sentence'],
                "Q1": request_data['Q1'],
                "Q2": request_data['Q2'],
                "Q3": request_data['Q3'],
                "Q4": request_data['Q4'],
                "Q5": request_data['Q5'],
                "Q6": request_data['Q6'],
                "Q7": request_data['Q7'],
                "Q8": request_data['Q8'],
                "Q9": request_data['Q9'],
                "Qscore": request_data['Qscore'],
                "predic": predic
            }
            df = pd.DataFrame(sheet, index=[0])
            df_values = df.values.tolist()
            gs.values_append('Sheet1', {'valueInputOption': 'RAW'}, {'values': df_values})

            return jsonify(data)
        except Exception as e:
            return jsonify({"error": str(e)})
    return "OK"
```

รูปที่ 36 โค้ดการทำงานของ API

ทดสอบการเรียก API โดยในโครงงานนี้ ได้มีการใช้งาน Postman ในการเข้ามาช่วยทดสอบระบบ โดยค่า Parameter ที่ใช้คือค่าเดียวกันที่กำหนดไว้ใน Dialogflow

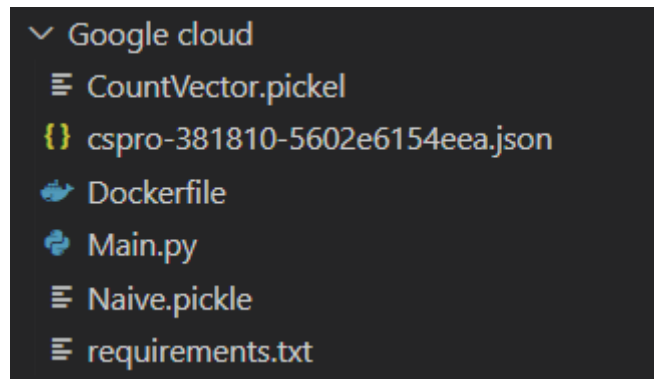


รูปที่ 37 การทดสอบ API โดยใช้ Postman

จาก รูปที่ 37 การทดสอบ API โดยใช้ Postman ในการทำงานได้ผลว่า API สามารถทำการส่งข้อมูลไปยังโมเดลการเรียนรู้ของเครื่อง เพื่อนำข้อมูลไปประมวลผล จากนั้นรับค่าผลลัพธ์จากโมเดล และส่งกลับมาแสดงผลได้อย่างถูกต้อง และนำข้อมูลทั้งหมดไปบันทึกใน Google sheet ได้ครบถ้วน

4.4 การใช้งาน Google cloud

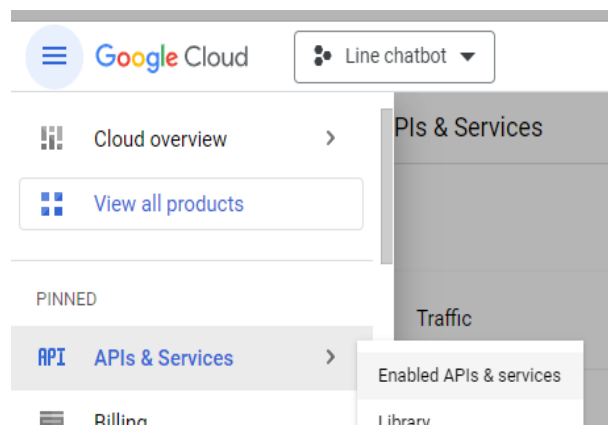
การที่จะให้ระบบเรียกใช้งานโมเดลได้ตลอดเวลา ในโครงการนี้ได้มีการใช้บริการของ Google cloud เพื่อนำโมเดลและ API บันทึกไว้โดยสร้างโฟลเดอร์ สำหรับไฟล์ที่จำเป็นต้องบันทึกไว้ ซึ่งประกอบไปด้วยรายละเอียดดัง รูปที่ 38



รูปที่ 38 รายละเอียดไฟล์ที่จำเป็น

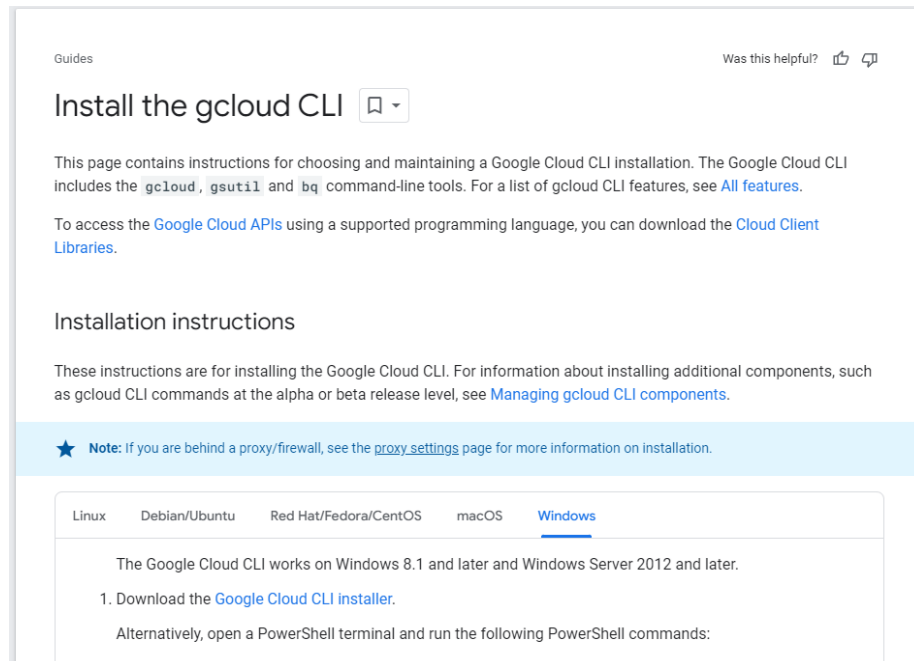
- ไฟล์คลังคำศัพท์จากข้อมูลที่ใช้ในการสอนโมเดล
- ไฟล์ Credentials ที่ใช้ในการเชื่อมต่อกับ Google sheet
- ไฟล์ Docker ไว้ใช้สำหรับ Deploy API และ Model
- ไฟล์ Python ที่ใช้สำหรับเรียก Model และ API
- ไฟล์ Model ที่ผ่านการ Train แล้ว
- ไฟล์ที่เก็บ Library ที่จำเป็นสำหรับ API

ในบริการของ Google cloud ทำการสร้างโปรเจกต์สำหรับการบันทึกโมเดล และ API จากนั้นทำการเปิดใช้งาน Cloud Run API และ Cloud Build API โดยเลือกที่ Enable API & Services ดังตัวอย่างในรูปที่ 39



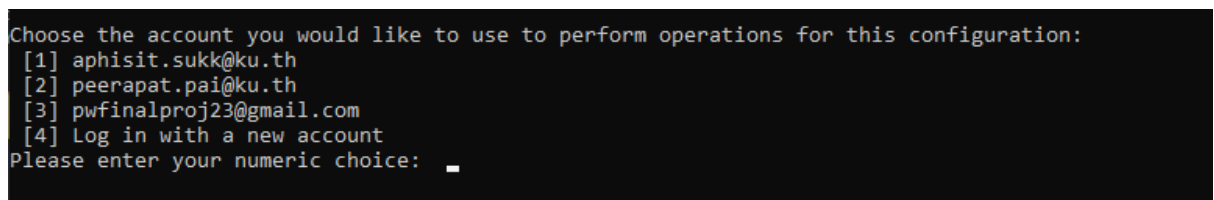
รูปที่ 39 เมนูการเปิดใช้งาน Services

ทำการติดตั้ง Gcloud CLI เพื่อใช้สำหรับการส่งโมเดลและ API ไปทำงานบน Google cloud โดยสามารถติดตั้งได้จาก [Install the gcloud CLI | Google Cloud](#) ดังตัวอย่างในรูปที่ 40



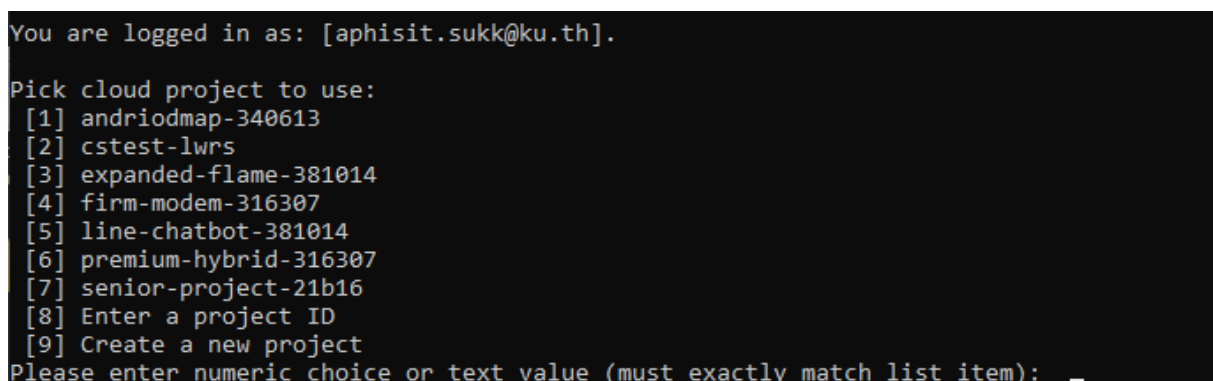
รูปที่ 40 ติดตั้ง Gcloud CLI

หลังจากที่ติดตั้งสำเร็จให้เปิด Google Cloud SDK Shell เพื่อเริ่มต้นใช้งานโดยใช้คำสั่ง `gcloud init` ระบบจะแสดงบัญชีที่เข้าใช้งาน Google cloud ให้ทำการเลือกบัญชีที่สร้างโปรเจกต์สำหรับการทำงานดังรูปที่ 41



รูปที่ 41 ตัวอย่างการแสดงผลบัญชี

เลือกโปรเจกต์ที่สร้างไว้สำหรับ Line chatbot ดังรูปที่ 42



รูปที่ 42 เลือกรายการโปรเจกต์ที่ต้องการ

เริ่มขั้นตอนการ Deploy โดยใช้คำสั่ง

```
gcloud builds submit --tag gcr.io/<Your Project ID>/index
```

ดังตัวอย่างในรูปที่ 43

```
D:\Final\Google cloud>gcloud builds submit --tag gcr.io/line-chatbot-381014/index
Creating temporary tarball archive of 6 file(s) totalling 107.0 KiB before compression.
Uploading tarball of [.] to [gs://line-chatbot-381014_cloudbuild/source/1680686569.538416-ffb4ce7fbaf94221aad1737295e15115.tgz]
Created [https://cloudbuild.googleapis.com/v1/projects/line-chatbot-381014/locations/global/builds/0f4414dc-e7e3-4318-bf43-7f871e8d19a8].
Logs are available at [ https://console.cloud.google.com/cloud-build/builds/0f4414dc-e7e3-4318-bf43-7f871e8d19a8?project=280241517567 ].
----- REMOTE BUILD OUTPUT -----
starting build "0f4414dc-e7e3-4318-bf43-7f871e8d19a8"
```

รูปที่ 43 ตัวอย่างการเริ่มการ Deploy

หลังจากที่การทำงานเสร็จสิ้นให้ใช้คำสั่งถัดไปเพื่อ Deploy model

```
gcloud run deploy --image gcr.io/<Your Project ID>/index --platform managed
```

จากนั้นกำหนดชื่อบริการที่ใช้ และทำการเลือกเซิร์ฟเวอร์ที่ต้องการใช้งาน หลังจากการทำงานเสร็จ

ระบบจะแสดงผลดังรูปที่ 44 และจะได้รับ URL สำหรับการเรียกใช้ API

```
Deploying container to Cloud Run service [bot1] in project [line-chatbot-381014] region [asia-southeast2]
OK Deploying... Done.
  OK Creating Revision...
  OK Routing traffic...
Done.
Service [bot1] revision [bot1-00002-tek] has been deployed and is serving 100 percent of traffic.
Service URL: https://bot1-an5zzi55ta-et.a.run.app
```

รูปที่ 44 แสดงผลลัพธ์การทาง และ URL สำหรับเรียกใช้ API

4.5 ผลลัพธ์ของระบบ

หลังจากที่โมเดลทำการประมวลผลลัพธ์ และส่งค่ากลับมายัง Dialogflow แล้วนั้น จะนำค่าผลรวมคะแนนของแบบประเมินสุขภาพจิต และผลของโมเดล ตอบกลับไปยังผู้ใช้ โดยส่งผลลัพธ์จากแบบประเมิน PHQ-9 เป็นข้อความตอบกลับในรูปแบบของ Flex message จากนั้นแจ้งผลการประเมินกลับไปยังผู้ใช้ ซึ่งเกณฑ์การประเมินนั้นแบ่งออกเป็น 5 ระดับ จากไม่มีอาการ, เสี่ยงน้อย, เสี่ยงปานกลาง, เสี่ยงมาก, เสี่ยงรุนแรง ดังตัวอย่างในรูปที่ 45 – 49



รูปที่ 47 เกณฑ์การประเมินระดับ"ไม่มีอาการ"



รูปที่ 48 เกณฑ์การประเมินระดับ"เสี่ยงน้อย"



รูปที่ 46 เกณฑ์การประเมินระดับ"เสี่ยงปานกลาง"



รูปที่ 45 เกณฑ์การประเมินระดับ"เสี่ยงมาก"



รูปที่ 49 เกณฑ์การประเมินระดับ"เสี่ยงรุนแรง"

บทที่ 5

การสรุปผลและข้อเสนอแนะ

5.1 สรุปผลโครงการ

จากการทดสอบการทำงานของระบบประเมินภาวะซึมเศร้าด้วยการเรียนรู้ของเครื่อง โดยโมเดลทั้งหมด 4 โมเดล ได้ผลสรุปว่า Naïve Bayes มีค่าความถูกต้องมากที่สุด จึงได้เลือกโมเดลดังกล่าวมาใช้ในระบบ โดยการนำโมเดลและระบบ API บันทึกไว้ใน Google cloud โดยใช้ Cloud run service เพื่อให้สามารถเรียกใช้งานได้ตลอดเวลา เมื่อผู้ใช้กรอกข้อมูลเพื่อทำการประเมินเสร็จสิ้นแล้ว ระบบจะนำข้อมูลไปประเมินผลและส่งผลลัพธ์กลับมาแจ้งผู้ใช้งานตามเกณฑ์การประเมินของระบบ และผลข้อมูลทั้งหมดไว้ใน Google sheet

5.2 ข้อเสนอแนะ

ระบบการประเมินภาวะซึมเศร้าด้วยการเรียนรู้ของเครื่อง ออกแบบมาเพื่อให้ผู้ใช้สามารถประเมินอาการของตนเองได้เบื้องต้น และอำนวยความสะดวกในการเข้าถึงบริการของห้องรับคำปรึกษา มหาวิทยาลัยเกษตรศาสตร์ วิทยาเขตศรีราชา นอกจากนั้นยังลดภาระการทำงานของเจ้าหน้าที่ในการคัดกรองอาการเบื้องต้นอีกด้วย แต่ด้วยจำนวนข้อมูลที่มีน้อย ทำให้โมเดลการเรียนรู้ของเครื่องนั้นยังไม่สามารถประมวลผลลัพธ์ออกมาได้มีประสิทธิภาพมากเท่าที่คาดหวัง

จึงมีข้อเสนอแนะว่า ด้วยข้อมูลที่มีการเก็บรวบรวมในระบบปัจจุบัน ควรนำข้อมูลดังกล่าวมาทำการฝึกสอน และพัฒนาระบบให้สามารถประมวลผลลัพธ์ได้มีประสิทธิภาพมากขึ้น นอกจากนั้น ข้อมูลคะแนนคำตอบจากแบบประเมินสุขภาพจิตนั้น สามารถนำมากำหนดน้ำหนักของคะแนนในแต่ละข้อคำถาม เพื่อนำข้อมูลเหล่านั้นมาพัฒนาโมเดลการเรียนรู้ของเครื่อง ให้สามารถประเมิน

อาการของผู้ใช้ได้ดียิ่งขึ้นต่อไป

- [1] Jirawatee (2561). **เรียนรู้การ Integrate LINE Bot เข้ากับ Dialogflow และ Firebase ผ่าน BMI Bot.** สืบค้นเมื่อ 27 กุมภาพันธ์ 2566. จาก <https://medium.com/lindevth/เรียนรู้การ-Integrate-line-Bot-เข้ากับ-dialogflow-และ-firebase-ผ่าน-bmi-bot-5a30a672f6ae>
- [2] Patrick Loeber(2565). **How To Deploy ML Models With Google Cloud Run.** สืบค้นเมื่อ 23 มีนาคม 2566. จาก <https://www.youtube.com/watch?v=vieoHqt7pxo>
- [3] Tracyrenee(2564). **How sklearn's CountVectorizer and TfidfTransformer compares with TfidfVectorizer.** สืบค้นเมื่อ 26 มีนาคม 2566. จาก <https://medium.com/geekculture/how-sklearns-countvectorizer-and-tfidftransformer-compares-with-tfidfvectorizer-a42a2d6d15a2>.
- [4] Faculty of Medicine Ramathibodi Hospital(2561). **แบบประเมินภาวะซึมเศร้า PHQ-9.** สืบค้นเมื่อ 13 กุมภาพันธ์ 2566. จาก https://www.rama.mahidol.ac.th/th/depression_risk
- [5] รศ.ดร.ไกรศักดิ์ เกษร(2564). **k nearest neighbor.** สืบค้นเมื่อ 19 มีนาคม 2566. จาก <https://csit.nu.ac.th/kraisak/ds/ds/chapter05/Chapter05.pdf>
- [6] รศ.ดร.ไกรศักดิ์ เกษร(2564). **Random Forest.** สืบค้นเมื่อ 19 มีนาคม 2566. จาก <https://csit.nu.ac.th/kraisak/ds/ds/chapter05/Chapter05.pdf>
- [7] รศ.ดร.ไกรศักดิ์ เกษร(2564). **Support Vector Machine.** สืบค้นเมื่อ 19 มีนาคม 2566. จาก <https://csit.nu.ac.th/kraisak/ds/ds/chapter06/Chapter06.pdf>
- [8] รศ.ดร.ไกรศักดิ์ เกษร(2564). **Naïve Bayes.** สืบค้นเมื่อ 19 มีนาคม 2566. จาก <https://csit.nu.ac.th/kraisak/ds/ds/chapter06/Chapter06.pdf>

