

**การจำแนกเจตนาการสนทนา เพื่อการสร้างแชทบอท**

**Intent Classification for Building Chatbot**

**โดย**

**นายณัฐพล เดชประมวลลพล รหัสนักศึกษา 6110210129**

**นางสาววริศรา พิสุทธิ์เธียร รหัสนักศึกษา 6110210373**

**โครงงานนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิทยาศาสตรบัณฑิต**

**สาขาวิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยสงขลานครินทร์**

**ปีการศึกษา 2564**

# กิตติกรรมประกาศ

การทำโครงงานทางวิทยาการคอมพิวเตอร์ “การจำแนกเจตนาการสนทนา เพื่อการสร้างแชทบอท” ในครั้งนี้สำเร็จลุล่วงได้ด้วยดี เพราะความอนุเคราะห์จากบุคคลหลายๆ ฝ่าย ดังนี้

ขอขอบคุณอาจารย์ ดร.นิวรรณ วัฒนกิจรุ่งโรจน์ อาจารย์ประจำสาขาวิชาวิทยาศาสตร์การคำนวณ คณะวิทยาศาสตร์ มหาวิทยาลัยสงขลานครินทร์ วิทยาเขตหาดใหญ่ ซึ่งเป็นอาจารย์ที่ปรึกษาโครงงานที่ได้ให้คำปรึกษา แนะนำ และความช่วยเหลือในทุกๆ เรื่องสำหรับการทำงาน ให้ข้อคิด แรงกระตุ้น และกำลังใจในการพัฒนางาน นอกจากนี้ท่านยังสละเวลาในการติดตามผลการทำโครงงานครั้งนี้มาโดยตลอด

ขอขอบคุณอาจารย์ ผศ.ดร.จารุณี ดวงสุวรรณ และอาจารย์ผศ.ดร.วิภาดา เวทย์ประสิทธิ์ คณะกรรมการในการสอบโครงงานที่กรุณาทำการสอบและให้คำชี้แนะที่ดีในการทำโครงงาน

ขอขอบคุณภาควิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยายลัยสงขลานครินทร์ ที่เอื้อเผื่อสถานที่และอุปกรณ์ในการทำโครงงาน และการสอบโครงงาน

ทางคณะผู้จัดทำรู้สึกซาบซึ้งและขอขอบคุณทุกท่านเป็นอย่างสูงมา ณ โอกาสนี้

คณะผู้จัดทำ

# สารบัญ

หน้า

[กิตติกรรมประกาศ 1](#_Toc66572944)

[สารบัญ 2](#_Toc66572945)

[รายการตาราง 4](#_Toc66572946)

[รายการภาพประกอบ 5](#_Toc66572947)

[บทที่ 1 บทนำ 1](#_Toc66572948)

[1.1 ความเป็นมาของโครงงาน 1](#_Toc66572949)

[1.2 วัตถุประสงค์ 1](#_Toc66572950)

[1.3 ขอบเขตของโครงงาน 1](#_Toc66572951)

[1.4 ขั้นตอนและระยะเวลาในการดำเนินงาน 1](#_Toc66572952)

[1.5 แผนภาพระยะเวลาดำเนินงาน 2](#_Toc66572953)

[1.6 ประโยชน์ที่คาดว่าจะได้รับ 4](#_Toc66572954)

[1.7 สถานที่และเครื่องมือที่ใช้ทำโครงงาน 4](#_Toc66572955)

[1.8 อาจารย์ที่ปรึกษาโครงงาน 4](#_Toc66572956)

[1.9 ผู้จัดทำโครงงาน 4](#_Toc66572957)

[บทที่ 2 ทฤษฏีและหลักการที่เกี่ยวข้อง 5](#_Toc66572958)

[2.1 องค์ประกอบแชทบอท 5](#_Toc66572959)

[2.1.1 Intent 5](#_Toc66572960)

[2.1.2 Response 5](#_Toc66572961)

[2.1.2 Story (flow) 5](#_Toc66572962)

[2.2 ประเภทของแชทบอท 5](#_Toc66572963)

[2.2.1 Rule-base Chatbot 5](#_Toc66572964)

[2.2.2 Artificial Intelligence (AI) Chatbot 6](#_Toc66572965)

[2.3 การสร้างเวกเตอร์แทนข้อความ 6](#_Toc66572966)

[2.3.1 Binary vector 6](#_Toc66572967)

[2.3.2 Term Frequency (TF) 7](#_Toc66572968)

[2.3.3 Inverse Document Frequency (IDF) 8](#_Toc66572969)

**สารบัญ (ต่อ)**

หน้า

[2.3.4 ค่าน้ำหนัก tf-idf 9](#_Toc66572970)

[2.4 ความคล้ายและความต่าง 9](#_Toc66572971)

[2.4.1 ความคล้ายกัน (similarity) 10](#_Toc66572972)

[2.4.2 ความต่างกัน (Distance) 11](#_Toc66572973)

[2.5 การใช้การเรียนรู้ของเครื่อง 12](#_Toc66572974)

[2.5.1 การเรียนรู้แบบไม่มีผู้สอน 12](#_Toc66572975)

[2.5.2 การเรียนรู้แบบมีผู้สอน 15](#_Toc66572976)

[2.6 เครื่องมือที่ใช้ในการสร้างแชทบอท 19](#_Toc66572977)

[2.6.1 Dialogflow 19](#_Toc66572978)

[2.6.2 chatterbot 22](#_Toc66572979)

[2.6.3 flow.ai 23](#_Toc66572980)

[บทที่ 3 การวิเคราะห์และขั้นตอนวิธี 25](#_Toc66572981)

[3.1 การจัดกลุ่มข้อความเพื่อกำหนด intent 25](#_Toc66572982)

[3.1.1 การทำความสะอาดข้อความ (Cleaning messages) 27](#_Toc66572983)

[3.1.2 การสร้างชุดคำศัพท์ (Preparing term set) 27](#_Toc66572984)

[3.1.3 การสกัดคุณลักษณะของข้อมูล (Feature extraction) 28](#_Toc66572985)

[3.1.4 การแบ่งกลุ่ม (Clustering) 28](#_Toc66572986)

[3.2 การสร้างโมเดลและทดสอบโมเดลในการจำแนก intent 29](#_Toc66572987)

[3.3 ชุดข้อมูลที่ใช้ทำการทดลองและตัวชี้วัดประสิทธิภาพ 32](#_Toc66572988)

[3.3.1 ชุดข้อมูลที่ใช้ 32](#_Toc66572989)

[3.3.2 ข้อมูลจำลองเพื่อสร้างแชทบอทโดยอาศัยวิธีการจัดกลุ่มและจำแนกที่นำเสนอในโครงงานนี้ 32](#_Toc66572990)

[3.3.3 การประเมินผล (Evaluation) 32](#_Toc66572991)

[3.4 การประยุกต์ใช้เพื่อพัฒนาต้นแบบแชทบอทและการอัพเดตแชทบอท 34](#_Toc66572992)

[3.4.1 การสร้างแชทบอท 34](#_Toc66572993)

[3.4.2 การอัพเดตแชทบอท 36](#_Toc66572994)

[เอกสารอ้างอิง 38](#_Toc66572995)

# รายการตาราง

หน้า

[ตารางที่ 2.1 ตารางค่า tf ของคำว่า the และ doctor ใน document 1 และ document 2 9](#_Toc61295376)

[ตารางที่ 2.2 ตารางค่า tf-idf ของคำว่า the และ doctor ใน document 1 และ document 2 9](#_Toc61295377)

# รายการภาพประกอบ

หน้า

[ภาพที่ 2.1 แสดงตัวอย่างลำดับของคำที่กำหนดให้ 6](#_Toc66446051)

[ภาพที่ 2.2 แสดงตัวอย่างของการแทนข้อความด้วยเวกเตอร์จากการค้นหาคำในข้อความ 6](#_Toc66446052)

[ภาพที่ 2.3 ตัวอย่างข้อความซึ่งอยู่ใน document 1 7](#_Toc66446054)

[ภาพที่ 2.4 การนับคำที่ปรากฎในข้อความ 7](#_Toc66446055)

[ภาพที่ 2.5 การทำค่า tft,d จาก document 1 8](#_Toc66446056)

[ภาพที่ 2.6 ตัวอย่างข้อความที่มีคำเหมือนและต่างกัน 8](#_Toc66446058)

[ภาพที่ 2.7 ตัวอย่างการหาค่า idf ของคำที่สนใจ 9](#_Toc66446059)

[ภาพที่ 2.8 ภาพวิธีการหาความคล้ายกันของ Jaccard similarity 10](#_Toc66446062)

[ภาพที่ 2.9 การวัดระยะทางแบบ Manhattan distance 11](#_Toc66446064)

[ภาพที่ 2.10 การวัดระยะทางแบบ Euclidean distance 12](#_Toc66446065)

[ภาพที่ 2.11 ภาพอธิบายขั้นตอนของ K-Means Clustering 13](#_Toc66446067)

[ภาพที่ 2.12 ภาพตัวอย่างชุดข้อมูลที่ใช้กับ DBSCAN 14](#_Toc66446068)

[ภาพที่ 2.13 ตัวอย่าง DBSCAN 15](#_Toc66446069)

[ภาพที่ 2.14 ตัวอย่าง Decision tree 16](#_Toc66446071)

[ภาพที่ 2.15 ตัวอย่างของฟังก์ชันการแปลง 17](#_Toc66446072)

[ภาพที่ 2.16 Neural Network แบบชั้นเดียว 18](#_Toc66446073)

[ภาพที่ 2.17 Neural Network แบบหลายชั้น 18](#_Toc66446074)

[ภาพที่ 2.18 หน้าจอ intent ส่วนของ contexts 20](#_Toc66446076)

[ภาพที่ 2.19 หน้าจอแสดงผลการทดสอบแชทบอท Dialogflow (1) 20](#_Toc66446077)

[ภาพที่ 2.20 หน้าจอแสดงผลการทดสอบแชทบอท Dialogflow (2) 21](#_Toc66446078)

[ภาพที่ 2.21 หน้าจอโปรแกรมเขียน train ให้กับ chatbot 22](#_Toc66446080)

[ภาพที่ 2.22 หน้าจอผลลัพธ์จากการรันโปรแกรมที่เขียนด้วย chatterbot 22](#_Toc66446081)

[ภาพที่ 2.23 หน้าจอหลักของ flow.ai 23](#_Toc66446083)

[ภาพที่ 2.24 หน้าจอแสดงผลการรันของ flow.ai 24](#_Toc66446084)

**รายการประกอบภาพ (ต่อ)**

หน้า

[ภาพที่ 3.1 ขั้นตอนการทำงาน A: การจัดกลุ่มข้อความเพื่อกำหนด intent 26](#_Toc66446085)

[ภาพที่ 3.2 ขั้นตอนการทำงาน B1: การสร้างโsมเดล 30](#_Toc66446089)

[ภาพที่ 3.3 ขั้นตอนการทำงาน B2: ทดสอบโมเดลในการจำแนก intent 31](#_Toc66446090)

[ภาพที่ 3.4 ขั้นตอนการทำงาน C: พัฒนาต้นแบบแชทบอท 35](#_Toc66446095)

[ภาพที่ 3.5 ขั้นตอนการทำงาน D: การอัพเดตแชทบอท 37](#_Toc66446096)

# บทที่ 1 บทนำ

## 1.1 ความเป็นมาของโครงงาน

การตอบข้อความของลูกค้าหรือผู้ที่เข้ามาสอบถามเป็นส่วนหนึ่งของงานที่มีเจ้าหน้าที่คอยให้บริการอยู่เสมอ แต่ในการตอบกลับเหล่านั้นยังมีข้อจำกัดในหลายๆ ด้าน เช่น ต้องรอเวลาทำการเจ้าหน้าที่จึงสามารถตอบได้ หรือต้องรอให้เจ้าหน้าที่ตรวจสอบว่ามีข้อความเข้าจึงจะได้รับคำตอบ ซึ่งทำให้การบริการหรือคำถามไม่ได้คำตอบในทันทีทันใด และอาจทำให้ลูกค้าหรือผู้ใช้เกิดความไม่พึงพอใจขึ้นได้อีกด้วย ในปัจจุบันจึงมีตัวช่วยที่เรียกกันว่าแชทบอทเข้ามาเป็นส่วนหนึ่งในการให้บริการทางด้านนี้โดยเฉพาะ

ปัจจุบันแชทบอทหรือโปรแกรมคอมพิวเตอร์ชนิดหนึ่งที่ใช้ในการตอบกลับการสนทนากำลังเป็นที่แพร่หลายในองค์กรหรือบริษัทห้างร้าน เนื่องจากความสามารถในการช่วยอำนวยความสะดวกให้แก่ลูกค้าและเจ้าหน้าที่ เช่น ช่วยให้การสอบถามข้อมูลและบริการสะดวกรวดเร็วทั้งในและนอกเวลาทำการ ลดปัญหาเจ้าหน้าที่ไม่เพียงพอต่อความต้องการของลูกค้า เป็นต้น ความท้าทายอย่างหนึ่งในการสร้างแชทบอท คือ การระบุเจตนาจากการสนทนาหรือเรียกว่า intent ซึ่ง intent หนึ่งๆ มีข้อความได้หลากหลาย เราอาจใช้วิธีให้ผู้พัฒนาแชทบอทระบุข้อความที่เป็นไปได้ สำหรับแต่ละ intent แต่อาจไม่ครบถ้วน และต้องใช้เวลานาน

การเรียนรู้ของเครื่อง (Machine Learning) เป็นเทคนิคที่สามารถนำมาใช้ในการจำแนกข้อมูลได้ ผู้จัดทำโครงงานจึงสนใจศึกษาการนำการเรียนรู้ของเครื่องมาช่วยในการจำแนก intent เพื่อการสร้าง  
แชทบอท โดยจะใช้กรณีศึกษาจากชุดข้อมูลมาตรฐาน และบทสนทนาถามตอบเกี่ยวกับการศึกษาของนักศึกษาคณะวิทยาศาสตร์ มหาวิทยาลัยสงขลานครินทร์

## 1.2 วัตถุประสงค์

1. เพื่อศึกษาการจำแนก intent โดยใช้การเรียนรู้ของเครื่องสำหรับสร้างแชทบอท
2. เพื่อสร้างแชทบอทตอบคำถามข้อมูลเบื้องต้นของสถาบันการศึกษา ซึ่งจะช่วยตอบคำถามให้แก่ผู้ใช้นอกช่วงเวลาทำการ

## ขอบเขตของโครงงาน

ศึกษาการจำแนก intent โดยใช้การเรียนรู้ของเครื่อง อย่างน้อย 2 เทคนิค บนชุดข้อมูลมาตรฐาน และกรณีศึกษาแชทบอทตอบคำถามข้อมูลเบื้องต้นเกี่ยวกับการศึกษา สำหรับนักศึกษาคณะวิทยาศาสตร์ มหาวิทยาลัยสงขลานครินทร์ โดยใช้ภาษา python เป็นหลัก

## 1.4 ขั้นตอนและระยะเวลาในการดำเนินงาน

1. คิดหัวข้อโครงงาน
2. ศึกษาความเป็นไปได้ของโครงงาน
3. กำหนดขอบเขต
4. ศึกษาเครื่องมือที่เกี่ยวข้อง
5. ศึกษาเทคนิคที่เกี่ยวข้อง
6. วางแผนการทดสอบ
7. ทำการทดลอง
8. สรุปและวิเคราะห์ผลการทดลอง
9. นำ intent ไปสร้างแชทบอท
10. ทดสอบการใช้แชทบอท
11. ประเมินผลการใช้แชทบอท

## 1.5 แผนภาพระยะเวลาดำเนินงาน

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ขั้นตอนการดำเนินงาน | ปีการศึกษา 2563 | | | | | | ปีการศึกษา 2564 | | | | | |
| พ.ย. | ธ.ค. | ม.ค. | ก.พ. | มี.ค. | ม.ย. | พ.ค. | ม.ย. | ก.ค. | ส.ค. | ก.ย. | ต.ค. |
| 1. คิดหัวข้อโครงงาน |  |  |  |  |  |  |  |  |  |  |  |  |
| 1. ศึกษาความเป็นไปได้ของโครงงาน |  |  |  |  |  |  |  |  |  |  |  |  |
| 1. กำหนดขอบเขต |  |  |  |  |  |  |  |  |  |  |  |  |
| 1. ศึกษาเครื่องมือที่เกี่ยวข้อง |  |  |  |  |  |  |  |  |  |  |  |  |
| 1. ศึกษาเทคนิคที่เกี่ยวข้อง |  |  |  |  |  |  |  |  |  |  |  |  |
| 1. วางแผนการทดสอบ |  |  |  |  |  |  |  |  |  |  |  |  |
| 1. ทำการทดลอง |  |  |  |  |  |  |  |  |  |  |  |  |
| 1. สรุปและวิเคราะห์ผลการทดลอง |  |  |  |  |  |  |  |  |  |  |  |  |
| 1. นำ intent ไปสร้าง แชทบอท |  |  |  |  |  |  |  |  |  |  |  |  |
| 1. ทดสอบการใช้แชทบอท |  |  |  |  |  |  |  |  |  |  |  |  |
| 1. ประเมินผลการใช้ แชทบอท |  |  |  |  |  |  |  |  |  |  |  |  |

## 1.6 ประโยชน์ที่คาดว่าจะได้รับ

1) ได้วิธีการจำแนก intent เพื่อการสร้างแชทบอทที่มีประสิทธิภาพ

2) ได้ต้นแบบแชทบอท เพื่อการสนทนาเกี่ยวกับการเรียน สำหรับนักศึกษาคณะวิทยาศาสตร์

## 1.7 สถานที่และเครื่องมือที่ใช้ทำโครงงาน

สถานที่ คือ อาคารภาควิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยสงขลานครินทร์ วิทยาเขตหาดใหญ่

เครื่องมือที่ใช้

* ฮาร์ดแวร์
  + Processor: 2.4 GHz Intel Core i5
  + Ram: 8 GB
* ซอฟต์แวร์
  + Python 3.6.8
  + ระบบปฎิบัติการ windows 10

## 1.8 อาจารย์ที่ปรึกษาโครงงาน

อาจารย์ ดร.นิวรรณ วัฒนกิจรุ่งโรจน์

## 1.9 ผู้จัดทำโครงงาน

นายณัฐพล เดชประมวลพล รหัสนักศึกษา 6110210129

นางสาววริศรา พิสุทธิ์เธียร รหัสนักศึกษา 6110210373

# บทที่ 2 ทฤษฏีและหลักการที่เกี่ยวข้อง

## 2.1 องค์ประกอบแชทบอท

แชทบอท (Chatbot) คือ แอปพลิเคชันซอฟแวร์ที่ใช้ในการสนทนาออนไลน์ผ่านข้อความหรือเสียง แทนการติดต่อโดยตรงกับมนุษย์ ซึ่งถูกออกแบบมาเพื่อตอบคำถามอัตโนมัติให้กับผู้ใช้อย่างเป็นธรรมชาติ โดยแชทบอทถูกออกแบบให้ใช้เป็นระบบ โต้ตอบ เพื่อวัตถุประสงค์ต่างๆ เช่น การรับออเดอร์จากลูกค้า, การสอบถามข้อมูล และการขอรับบริการ เป็นต้น ซึ่งแชทบอทจะประกอบด้วยส่วนหลักทั้งหมด 3 ส่วนที่มีความสำคัญ และจะทำงานร่วมกันเพื่อให้แชทบอทสามารถใช้งานได้อย่างมีประสิทธิภาพ โดยส่วนประกอบทั้งหมดมีดังนี้

2.1.1 Intent คือ ส่วนที่แชทบอทจะดึงเจตนารมณ์ที่ซ่อนอยู่ภายในข้อความที่ได้รับจากผู้ใช้ หรือความต้องการที่ผู้ใช้ต้องการกล่าวถึง ซึ่งอาจตรวจจับจาก keyword และช่วงคำหลัก โดยจะคำนึงถึงความจำเพาะที่บอทถูกสร้างขึ้นมาใช้งาน

2.1.2 Response คือ การตอบกลับข้อความ ซึ่งจะเกิดขึ้นเมื่อมีการระบุ intent จากข้อความที่ผู้ใช้ส่งเข้ามา โดยแชทบอทจะตอบด้วยคำตอบที่เหมาะสมที่สุด

2.1.2 Story (flow) คือ การวางแผนการตัดสินใจ หรือแผนผังตรรกะของการสนทนาเส้นตรง โดยคำนึงถึงการสนทนาจริง เพื่อเป็นการต่อประโยคการสนทนาอย่างเป็นธรรมชาติ และหากผู้ใช้ถามคำถามอื่นขึ้นมา โดยคำถามนั้นไม่มีความเกี่ยวข้องกับหัวข้อที่คุยกันอยู่ในช่วงก่อนหน้า แชทบอทสามารถเปลี่ยนการสนทนาให้เป็นไปตาม intent ที่ผู้ใช้ถามเข้ามาได้

## 2.2 ประเภทของแชทบอท

โดยทั่วไปแชทบอทถูกนำไปใช้เพื่อให้บริการในการตอบโต้บทสนทนาผ่านแอปพลิเคชัน หรือบนหน้าเว็บของแชทบอทเอง และวิธีการในการโต้ตอบกับแชทบอทแตกต่างกันไปตามประเภท ซึ่งสามารถแบ่งประเภทของแชทบอทได้ตามวิธีการทำงานของแชทบอท ดังนี้

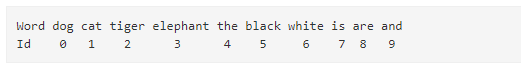
2.2.1 Rule-base Chatbot คือ แชทบอทแบบมีข้อกำหนด นิยมแบบมีปุ่มตัวเลือกให้ผู้ใช้เลือกถามคำถาม เพื่อเป็นการกำหนดให้ผู้ใช้ทำตามกฎที่เขียนไว้ล่วงหน้า ซึ่งจะทำให้ได้รับคำตอบที่ผู้สร้างได้กำหนดไว้ และผู้สร้างต้องมีการสร้างกฎขึ้นหลายข้อ เพื่อครอบคลุมหลายกรณีที่ต้องการให้แชทบอทสามารถโต้ต้อบได้ และหากผู้ใช้ต้องการถามคำถามนอกเหนือจากที่มี แชทบอทจะไม่สามารถตอบคำถามเหล่านั้นได้ โดยการทำงานเช่นนี้ของแชทบอทจะไม่มีการใช้งานปัญญาประดิษฐ์ จึงมีข้อจำกัดในการใช้งานอยู่มาก [1]

2.2.2 Artificial Intelligence (AI) Chatbot คือ แชทบอทที่มีการใช้งานปัญญาประดิษฐ์ เพื่อช่วยในการตอบคำถาม แต่ในกรณีนี้ยังมีความไม่ชัดเจนเรื่องแนวคิดเบื้องหลังของระบบคอมพิวเตอร์ที่ทำให้เกิดการ ‘คิดเหมือนมนุษย์’ จึงทำให้มีการใช้ตรรกะทางความคิด การวางแผน และการเข้าใจภาษาเข้ามาเกี่ยวข้องด้วย ซึ่งการเข้าใจภาษามนุษย์จะทำให้แชทบอทมีการตอบโต้ที่ดูเป็นธรรมชาติ อันเกิดจากการใช้อัลกอริทึม หรือ Neural Networks เข้ามาประมวลผลภาษาธรรมชาติ จึงเป็นที่มาของ Natural Language Processing หรือ NLP ซึ่งทำให้มีความยากมากกว่าแชทบอทประเภทแรก [1]

## 2.3 การสร้างเวกเตอร์แทนข้อความ

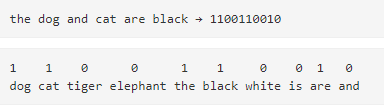
การสร้างเวกเตอร์แทนข้อความ เป็นการสร้างเวกเตอร์เพื่อใช้แทนข้อความต่างๆ ซึ่งจะพิจารณาว่าคำดังกล่าวปรากฎอยู่ในข้อความหรือไม่ และไม่คำนึงถึงลำดับของคำ สามารถสร้างได้หลายวิธี ในที่นี้จะกล่าวถึงวิธีการสร้างเวกเตอร์แทนข้อความ ด้วยวิธี 4 วิธีดังนี้

2.3.1 Binary vector ซึ่งจะประกอบด้วย 0 หรือ 1 เพื่อแสดงว่าไม่มีหรือมีคำนั้นอยู่ในข้อความตามลำดับ [2] ตัวอย่างมีคำดังนี้



ภาพที่ 2.1 แสดงตัวอย่างลำดับของคำที่กำหนดให้

(ที่มา สืบค้นจาก: <https://ichi.pro/th/kar-thaen-kha-laea-khxkhwam-ni-kar-pramwl-phl-phas-a-thrrmchati-56300952731182> )

 เมื่อมีประโยคมาให้ และเขียนเปรียบเทียบเป็นเลข 0 หรือ 1 จะได้ว่า

1

1100110011

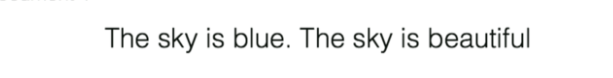
(ที่มา สืบค้าจาก: <https://ichi.pro/th/kar-thaen-kha-laea-khxkhwam-ni-kar-pramwl-phl-phas-a-thrrmchati-56300952731182>)

ภาพที่ 2.2 แสดงตัวอย่างของการแทนข้อความด้วยเวกเตอร์จากการค้นหาคำในข้อความ

และเขียนให้อยู่ในรูปของเวกเตอร์จะได้ว่า [1 1 0 0 1 1 0 0 1 1]T

2.3.2 Term Frequency (TF) เป็นการหาเวกเตอร์ค่าน้ำหนักของความถี่คำ ซึ่งไม่จำเป็นต้องเป็นเลข 0 หรือ 1 หากมีจำนวนคำมากกว่า 1 ขึ้นไป [2] พิจารณาได้ 2 ลักษณะ

* ค่าน้ำหนักความถี่โดยตรง แทนด้วย tf(org)
* ค่าน้ำหนักความถี่ที่ได้รับการทำให้เป็นมาตราฐาน แทนด้วย tf

โดยพิจารณาตัวอย่างข้อความที่มีบางคำซ้ำกันบ้าง ไม่ซำกันบบ้าง

ภาพที่ 2.3 ตัวอย่างข้อความซึ่งอยู่ใน document 1

(ที่มา TF-IDF คำไหนสำคัญนะ

สืบค้นจาก: [https://lukkiddd.com/tf-idf-คำไหนสำคัญนะ-dd1e1568312e](https://lukkiddd.com/tf-idf-%E0%B8%84%E0%B8%B3%E0%B9%84%E0%B8%AB%E0%B8%99%E0%B8%AA%E0%B8%B3%E0%B8%84%E0%B8%B1%E0%B8%8D%E0%B8%99%E0%B8%B0-dd1e1568312e))

ให้ tf(org)t,d แทนจำนวนครั้งของการปรากฎของคำ t(term) ในข้อความ d(document)

ภาพที่ 2.4 การนับคำที่ปรากฎในข้อความ

(ที่มา TF-IDF คำไหนสำคัญนะ

สืบค้นจาก: [https://lukkiddd.com/tf-idf-คำไหนสำคัญนะ-dd1e1568312e](https://lukkiddd.com/tf-idf-%E0%B8%84%E0%B8%B3%E0%B9%84%E0%B8%AB%E0%B8%99%E0%B8%AA%E0%B8%B3%E0%B8%84%E0%B8%B1%E0%B8%8D%E0%B8%99%E0%B8%B0-dd1e1568312e))

 ทำค่าให้อยู่เป็นช่วงค่ามาตราฐาน (Normalization) โดยใช้สูตร 1+log(tforg) แต่หากไม่มีคำนั้นปรากฎอยู่ให้ค่าเป็น 0 ทันที

ภาพที่ 2.5 การทำค่า tft,d จาก document 1

(ที่มา TF-IDF คำไหนสำคัญนะ

สืบค้นจาก: [https://lukkiddd.com/tf-idf-คำไหนสำคัญนะ-dd1e1568312e](https://lukkiddd.com/tf-idf-%E0%B8%84%E0%B8%B3%E0%B9%84%E0%B8%AB%E0%B8%99%E0%B8%AA%E0%B8%B3%E0%B8%84%E0%B8%B1%E0%B8%8D%E0%B8%99%E0%B8%B0-dd1e1568312e))

จากการทำในภาพที่ 2.5 จะสามารถเขียนออกมาในรูปเวกเตอร์ได้ว่า [1.3 1.3 1.3 1 1]T

2.3.3 Inverse Document Frequency (IDF) เนื่องจากคำบางคำมีปรากฎอยู่ในข้อความเกือบทุกข้อความที่สนใจ ซึ่งหมายความได้ว่าคำเหล่านั้นแทบไม่มีนัยสำคัญต่อการค้นหา ในทางกลับกันคำที่ปรากฎอยู่เพียงไม่กี่ข้อความจะถือว่ามีความสำคัญมาก เมื่อมีการสอบถามและปรากฎคำที่สำคัญ จึงต้องสนใจคำที่สำคัญนั้นมากกว่าคำที่พบในเกือบทุกข้อความ

ให้ dft แทน จำนวนข้อความที่มีคำ t ปรากฎอยู่ ค่าผกผันของ dft นิยามด้วย idft และ N คือจำนวนข้อความที่มีอยู่ จะใช้สูตร

ในการหาค่า idft จากข้อความตัวอย่าง

ภาพที่ 2.6 ตัวอย่างข้อความที่มีคำเหมือนและต่างกัน

(ที่มา TF-IDF คำไหนสำคัญนะ

สืบค้นจาก: [https://lukkiddd.com/tf-idf-คำไหนสำคัญนะ-dd1e1568312e](https://lukkiddd.com/tf-idf-%E0%B8%84%E0%B8%B3%E0%B9%84%E0%B8%AB%E0%B8%99%E0%B8%AA%E0%B8%B3%E0%B8%84%E0%B8%B1%E0%B8%8D%E0%B8%99%E0%B8%B0-dd1e1568312e))

 หาค่า idft ได้จากสูตร โดยตัวอย่างจะสนใจคำว่า ‘the’ และคำว่า ‘doctor’ ที่มีปรากฎทั้งสองข้อความ และมีปรากฎใน document 1 เพียงอย่างเดียวตามลำดับ [2] ได้ว่า

ภาพที่ 2.7 ตัวอย่างการหาค่า idf ของคำที่สนใจ

(ที่มา TF-IDF คำไหนสำคัญนะ

สืบค้นจาก: [https://lukkiddd.com/tf-idf-คำไหนสำคัญนะ-dd1e1568312e](https://lukkiddd.com/tf-idf-%E0%B8%84%E0%B8%B3%E0%B9%84%E0%B8%AB%E0%B8%99%E0%B8%AA%E0%B8%B3%E0%B8%84%E0%B8%B1%E0%B8%8D%E0%B8%99%E0%B8%B0-dd1e1568312e))

2.3.4 ค่าน้ำหนัก tf-idf เป็นปัจจัยที่ส่งผลต่อการคาดคะเนความเกี่ยวข้อง โดยพิจารณาจากค่า tf ซึ่งบ่งบอกว่าคำเหล่านั้น และข้อความมีความเกี่ยวข้องกันเพียงใด และ idf ซึ่งบ่งบอกว่าหากมีค่ามาก คำนั้นสำคัญต่อการค้นหาเป็นอย่างมากเช่นกัน โดยในการคำนวณ tf-idf [2] จะหาจากสูตร

โดยนำตัวอย่างจากภาพที่ 2.6 หาค่า tf ของ the และdoctor ได้ว่า

|  |  |  |
| --- | --- | --- |
| คำ | tf ใน document1 | tf ใน document 2 |
| the | 1 | 1 |
| doctor | 1 | 0 |

**ตารางที่ 2.1 ตารางค่า tf ของคำว่า the และ doctor ใน document 1 และ document 2**

ได้ค่า tf-idf ของคำว่า the และ document จากสูตรดังนี้

|  |  |  |
| --- | --- | --- |
| คำ | tf-idf ของ document 1 | tf-idf ของ document 2 |
| the | 0 | 0 |
| doctor | 0.3 | 0 |

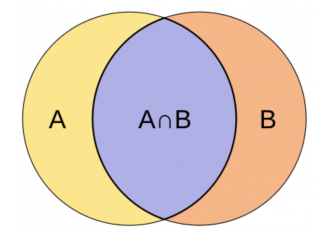
ตารางที่ 2.2 ค่า tf-idf ของคำว่า the และ doctor ใน document 1 และ document 2

## 2.4 ความคล้ายและความต่าง

เมื่อได้เวกเตอร์แทนข้อความแล้ว การตรวจสอบว่าข้อความที่สอบถาม กับข้อความที่มีอยู่เกี่ยวข้องกันมากน้อยเพียงใด พิจารณาได้ 2 แนวทาง ดังนี้

2.4.1 ความคล้ายกัน (similarity) คือการหาความคล้ายกันของเวกเตอร์ ซึ่งพิจารณาได้จากทิศทางของเวกเตอร์ ซึ่งพุ่งออกจากจุดกำเนิด ไปยังพิกัดของเวกเตอร์นั้นๆ หากเวกเตอร์ใดมีทิศใกล้เคียงกัน หมายความได้ว่ามีคุณลักษณะในแต่ละมิติการกระจายตัวคล้ายกัน โดยขอกล่าวถึง 2 วิธี

1. Jaccard Similarity หรือ Intersect over union เป็นวิธีการหา similarity ที่ได้จากค่า A intersect B ซึ่งเป็นค่าที่มีความคล้ายคลึงกันมากที่สุด ซึ่งค่าสูงสุดของ Jaccard คือ 1 จะเกิดขึ้นเมื่อ A intersect B มีค่าเท่ากับ A union B [3] โดยคำนวณด้วยสูตร

เขียนเป็นรูปให้เข้าใจได้ง่ายว่า

ภาพที่ 2.8 ภาพวิธีการหาความคล้ายกันของ Jaccard similarity

(ที่มา Jaccard index

สืบค้นจาก : <https://en.wikipedia.org/wiki/Jaccard_index>)

1. Cosine Similarity คือการดูความคล้ายคลึงด้วยองศา ซึ่งคำนวณได้จากสูตร

*เมื่อ เป็นเวกเตอร์แทนข้อความการสอบถาม และ*

*เป็นเวกเตอร์แทนข้อความใดๆ เป็นมุมระหว่างเวกเตอร์ และ*

K *เป็นจำนวนมิติขิงเวกเตอร์ และ*

โดยที่ และ

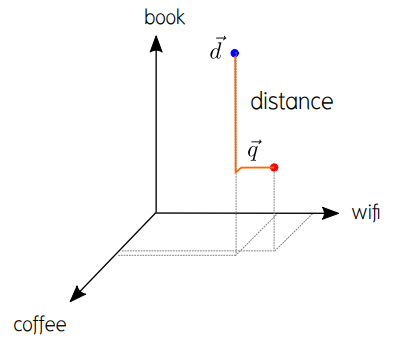
และ มีความคล้ายกันมากที่สุดเมื่อ เข้าใกล้ 0 องศา หรือเข้าใกล้ 1

สามารถสมนัยสมการได้ว่า

โดยที่ และ ซึ่ง และ คือเวกเตอร์มาตราฐานขนาด 1 หนึ่ง

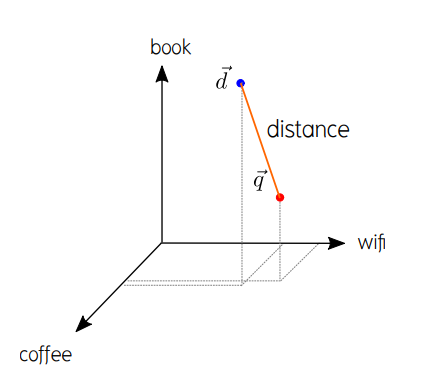
เมื่อค่าของ sim เข้าใกล้ 1 มากเท่าไรยิ่งหมายความว่ามีความคล้ายกันมากเท่านั้น [2]

2.4.2 ความต่างกัน (Distance) เป็นการหาความต่างกันของเวกเตอร์ ซึ่งพิจารณาได้จากทิศทางของเวกเตอร์ ซึ่งพุ่งออกจากจุดกำเนิด ไปยังพิกัดของเวกเตอร์นั้นๆ หากเวกเตอร์ใดมีทิศต่างกัน หมายความได้ว่ามีคุณลักษณะในแต่ละมิติการกระจายตัวต่างกัน โดยขอกล่าวถึง 2 วิธี

1. **Manhattan distance** เป็นการหาระยะทางระหว่างเวกเตอร์หรือจุด ที่ได้จากผลบวกของระยะทางตาม แนวแกนในแต่ละมิติ ดังสมการ

ภาพที่ 2.9 การวัดระยะทางแบบ Manhattan distance

(ที่มา [2])

1. **Euclidean distance** เป็นการหาระยะทางระหว่างเวกเตอร์หรือจุด ซึ่งเป็นระยะแบบเส้นตรง (Straight-line) ดังสมการ

ภาพที่ 2.10 การวัดระยะทางแบบ Euclidean distance

(ที่มา [2])

โดยมีหลักการว่า หากค่าความต่างมีมากเท่าไร ข้อความสอบถามก็จะแตกต่างจากข้อความที่มีมากขึ้นเท่านั้น [2]

## 2.5 การใช้การเรียนรู้ของเครื่อง

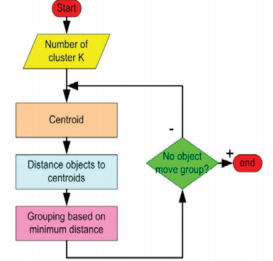
ในการจัดกลุ่มว่าเราจะมี intent กี่กลุ่มนั้นสามารถใช้เทคนิคแบบไม่มีผู้สอน เพื่อจัดกลุ่ม intent เองได้ ซึ่งจะพูดถึงการเรียนรู้แบบไม่มีผู้สอน ได้แก่

2.5.1 การเรียนรู้แบบไม่มีผู้สอน

ในหัวข้อนี้จะกล่าวถึงการเรียนรู้แบบไม่มีผู้สอน 2 วิธี คือ K-Means และ DBSCAN

* + 1. **การแบ่งกลุ่มข้อมูลแบบเคมีน (K-Means clustering)** คือ วิธีการหนึ่งใน Data mining โดยหน้าที่หลักของ K-Means คือการแบ่งกลุ่มแบบ Cluster ซึ่งการแบ่งกลุ่มในลักษณะนี้จะต้องใช้ฟังก์ชั่นอย่างน้อย 2 ฟังก์ชั่นในการคำนวณระยะห่างระหว่างข้อมูลได้แก่ Euclidean distance metric และ Manhattan distance metric ซึ่งหน้าที่ของ Cluster คือการจับกลุ่มของข้อมูลที่มีลักษณะใกล้เคียงกันเป็นกลุ่มเดียวกัน ซึ่งในขั้นตอนการทำงานของ K-Means Clustering สามารถสรุปได้เป็น 4 ขั้นตอนดังนี้

1. กำหนด Cluster (สามารถมีได้มากกว่า 1 ตัว)
2. กำหนดหรือเคลื่อนที่ Centroid (ของพิกัดของ Cluster)
3. คำนวณระยะห่างระว่างตำแหน่งของข้อมูลกับ Centroid
4. จัดกลุ่มข้อมูลที่มีระยะห่างน้อยที่สุด



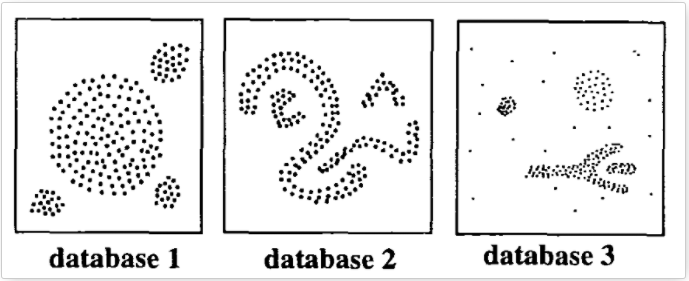
ภาพที่ 2.11 ภาพอธิบายขั้นตอนของ K-Means Clustering

(ที่มา On K-means Data Clustering Algorithm with Genetic Algorithm

สืบค้นจาก : <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7913145>, 2559)

การจัดกลุ่มจะดูจากระยะห่างของข้อมูลกับ Centroid ว่าข้อมูลอยู่ใกล้ Centroid ใดมากกว่า ข้อมูลนั้นก็จะถูกจัดอยู่ใน Cluster นั้น ทำขั้นตอนที่ 2 – 4 วนซ้ำไปเรื่อยๆ จนกว่าค่า Centroid จะอยู่ในตำแหน่งจุดกึงกลางของข้อมูลใน Cluster [4]

* + 1. **การจัดกลุ่มของข้อมูล แบบ DBSCAN** (Density-based spatial clustering of applications with noise) คือ การหาบริเวณข้อมูลที่อยู่รวมกันเป็นกลุ่มโดยจะหากลุ่มข้อมูลได้จากการคำนวณที่เกิดจาก Data point หรือจุดที่ข้อมูลแสดงอยู่



ภาพที่ 2.12 ภาพตัวอย่างชุดข้อมูลที่ใช้กับ DBSCAN

(ที่มา Ester et al. 1996

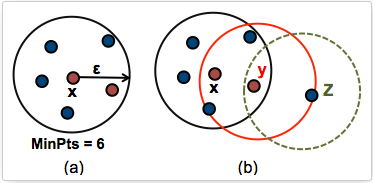
สืบค้นจาก : <http://www.sthda.com/english/wiki/wiki.php?id_contents=7940>)

DBSCAN มักใช้กับชุดข้อมูลที่ไม่สามารถแบ่งแยกกลุ่มก้อนได้อย่างชัดเจน ไม่มี pattern ที่แน่นอน หรือมี outliner ในการทำงานของ DBSCAN นั้นจะใช้ 2 parameter เพื่อหากลุ่มข้อมูล [5] ได้แก่

1. eps คือรัศมีจากจุดศูนย์กลางวงกลม
2. MinPts คือจำนวน Data point ขั้นต่ำในการกำหนด center

ซึ่งในขั้นตอนการทำงานของ DBSCAN สามารถสรุปเป็น 4 ขั้นตอน [5] ได้ดังนี้

1. Data point ใดๆ ในชุดข้อมูลที่มีข้อมูลอื่นๆ อยู่รอบๆตัวมันในรัศมี eps มากกว่าหรือเท่ากับค่า MinPts จะถูกเรียกว่า core point
2. Data point ใดๆ ในชุดข้อมูลที่มีข้อมูลอื่นๆ อยู่รอบๆ ตัวมันในรัศมี eps น้อยกว่าค่า MinPts แต่อย้ในรัศมี eps ของ core point จะเรียกว่า border point
3. Data point ใดๆ ในชุดข้อมูลที่มีข้อมูลอื่นๆ อยู่รอบๆ ตัวมันในรัศมี eps น้อยกว่าค่า MinPts และไม่อยู่ในรัศมี eps ของ core point แต่อยู่ในรัศมี eps ของ border point จะเรียกว่า border point เช่นกัน
4. Data point ใดๆ ที่มีอยู่ในขั้นตอนที่ 1-3 จะถูกเรียกว่า noise



ภาพที่ 2.13 ตัวอย่าง DBSCAN

(ที่มา DBSCAN: density-based clustering for discovering clusters in large datasets with noise - Unsupervised Machine Learning

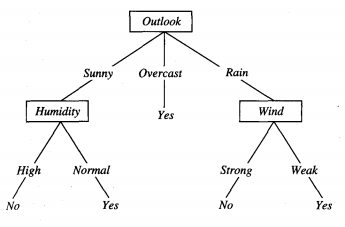
สืบค้นจาก : <http://www.sthda.com/english/wiki/wiki.php?id_contents=7940>)

จากภาพ 2.13 สามารถบอกลักษณะของ data point ได้ว่า x คือ core point ส่วน y และ z คือ border point

2.5.2 การเรียนรู้แบบมีผู้สอน

ในกรณีที่เราทราบแล้วว่าจะมี intent ใดบ้าง แล้วเราต้องการจำแนกข้อความที่เข้ามาว่าจัดอยู่ใน intent ใด สามารถทำได้ โดยใช้การเรียนรู้แบบมีผู้สอน ในที่นี้จะกล่าวถึงการเรียนรู้แบบมีผู้สอน 2 เทคนิค คือ Decision Tree และ Neural Network

1. **ต้นไม้ตัดสินใจ (Decision Tree)** ซึ่งเป็นวิธีการเรียนรู้ของเครื่องที่นิยืมใช้เป็นอย่างมากรู้แบบหนึ่ง ใช้หรับหรือการจำแนก (Classification) ข้อมูลและคลาส (class) ต่างๆ โดยใช้คุณสมบัติ (attribute) ของข้อมูล ในการจำแนกข้อมูลจากคุณสมบัติของข้อมูลจะต้องดูว่า คุณสมบัติใดของข้อมูลที่ใช้ในการจำแนกคลาส และคุณสมบัตินั้นมีความสำคัญอย่างไร ซึ่งสามารถสรุปส่วนประกอบของ Decision tree ได้ 3 ส่วน [6] ดังนี้
2. โหนดภายใน (internal node) คือ คุณสมบัติต่างๆ ของข้อมูลที่ใช้ในการจำแนกว่าข้อมูลจะไปอยุ่ในคลาสไหน โดยโหนดภายในที่เป็นโหนดเริ่มต้นเรียกว่า โหนดราก (root)
3. กิ่ง (branch, link) เป็นคุณสมบัติหรือเงื่อนไขของคุณสมบัติของโหนดที่ใช้ในการจำแนกข้อมูล ซึ่งโหนดภายในจะแตกกิ่งเท่ากับจำนวนคุณสมบัติของโหนดภายในนั้น
4. โหนดใบ (leaf node) คือคลาสต่างๆ ซึ่งเป็นผลลัพธ์ในการจำแนกข้อมูล



ภาพที่ 2.14 ตัวอย่าง Decision tree

(ที่มา Machine Learning

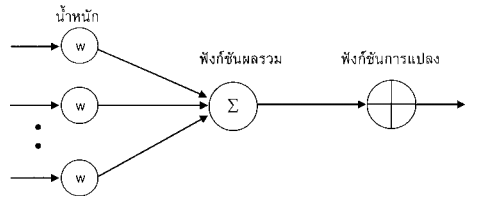
สืบค้นจาก : <http://www.cs.ubbcluj.ro/~gabis/ml/ML-books/McGrawHill%20-%20Machine%20Learning%20-Tom%20Mitchell.pdf>)

โดยคุณสมบัติและลักษณะการเรียนรู้ของ Decision tree สามารถจำแนกได้เป็น 5 ข้อ [6] ดังนี้

1. ผลของการเรียนรู้ของ Decision tree สามารถเข้าใจได้ง่ายเมื่อเทียบกับวิธีที่ใช้ในการจำแนกข้อมูลแบบอื่น
2. เส้นทางทุกเส้นจากโหนดรากถึงโหนดใบ สามารถแสดงให้อยู่ในรูปของ IF-THEN ได้
3. สามารถต้านทานข้อมูลรบกวน (noisy data) ได้
4. มีความเร็วในการเรียนรู้สูง เมื่อเทียบกับวิธีการที่ใช้ในการจำแนกข้อมูลแบบอื่น
5. เหมาะสำหรับการนำไปใช้ในการวิเคราะห์งานทางด้านธุรกิจ
6. **โครงข่ายประสาท (Neural Network)** คือ แบบจำลองทางคณิตศาสตร์ที่พัฒนาขึ้นเพื่อจำลองการทำงานของโครงข่ายประสาทในสมองมนุษย์ Neural Network มีลักษณะของการส่งผ่านสัญญาณประสาทในสมองของมนุษย์ กล่าวคือ มีความสามารถในการรวบรวมความรู้ (knowledge) โดยผ่านการเรียนรู้ (learning process) และความรู้เหล่านั้นจะจัดเก็บอยู่ในโครงข่ายในรูปแบบค่าน้ำหนัก (weight) ซึ่งสามารถปรับเปลี่ยนค่าได้เมื่อมีการเรียนรู้ใหม่ๆ เข้าไป

โหนด (node) เป็นการจำลองลักษณะการทำงานของเซลล์การส่งสัญญาณ (signal) ระหว่างโหนดที่เชื่อมต่อกัน (connection) จำลองมาจากการเชื่อมต่อกัน ภายในโหนดมีฟังก์ชันกำหนดสัญญาณส่งออกเรียกว่า ฟังก์ชันกระตุ้น (activation function) หรือฟังก์ชันการแปลง (transfer function) โดยในโครงสร้างของ Neural Network ประกอบด้วย 5 องค์ประกอบ [7] ดังนี้

1. ข้อมูลนำเข้า (input) คือ ข้อมูลที่เป็นตัวเลข หากเป็นข้อมูลเชิงคุณภาพ ต้องแปลงให้อยู่ในรูปเชิงปริมาณที่ Neural Network ยอมรับได้
2. ข้อมูลส่งออก (output) คือ ผลลัพธ์ที่เกิดขึ้นจริง (actual output) จากกระบวนการเรียนรู้ของ Neural Network
3. ค่าน้ำหนัก (weight) คือ สิ่งที่ได้จากการเรียนรู้ของ Neural Network เรียกอีกอย่างว่า ค่าความรู้ (knowledge) ถูกเก็บเป็นทักษะที่ใช้ในการจดจำข้อมูลอื่นๆ ที่อยู่ในรูปแบบเดียวกัน
4. ฟังก์ชันผลรวม (Summation function) เป็นผลรวมของข้อมูลป้อนเข้า (ai) และค่าน้ำหนัก (wi)
5. ฟังก์ชันการแปลง (transfer function) เป็นการคำนวณการจำลองการทำงานของ Neural Network



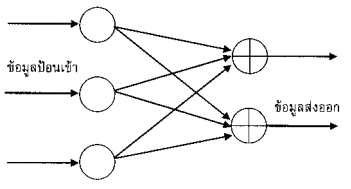
ภาพที่ 2.15 ตัวอย่างของฟังก์ชันการแปลง

(ที่มา โครงขายประสาทเทียม Artificial Neural Networks

สืบค้นจาก : <http://journal.hcu.ac.th/pdffile/5.%20%E0%B9%82%E0%B8%84%E0%B8%A3%E0%B8%87%E0%B8%82%E0%B9%88%E0%B8%B2%E0%B8%A2%2073-87.pdf>)

ลักษณะของ Neural Network จะประกอบไปด้วยโหนดจำนวนมากเชื่อมต่อกันกันกลุ่มย่อย เรียกว่า ชั้น (layer) ชั้นแรกเป็นชั้นข้อมูลเข้า เรียกว่า ขั้นรับข้อมูลป้อนเข้า (input layer) ส่วนชั้นสุดท้าย เรียกว่า ชั้นส่งข้อมูลส่งออก (output layer) และชั้นที่อยู่ระหว่างกลางของทั้ง 2 ชั้น เรียกว่า ชั้นแอบแฝง (hidden layer) ซึ่งโดยทั่วไปชั้นแอบแฝงอาจมีมากกว่า 1 ชั้นก็ได้ ทำให้สามารถจำแนกประเภทของ Neural Network ได้ 2 แบบ [7]

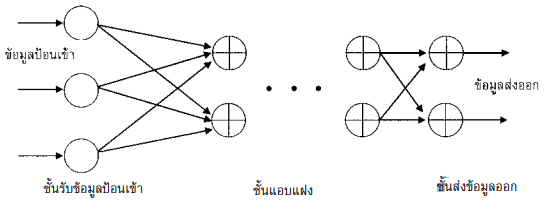
1. Neural Network แบบชั้นเดียว มีเพียงชั้นรับข้อมูลป้อนเข้าและชั้นส่งข้อมูลออกเท่านั้น
2. Neural Network แบบหลายชั้น ประกอบไปด้วยชั้นรับข้อมูลป้อนเข้า ชั้นส่งข้อมูลออกเท่านั้น และชั้นแอบแฝงตั้งแต่ 1 ชั้นขึ้นไป



ภาพที่ 2.16 Neural Network แบบชั้นเดียว

(ที่มา โครงขายประสาทเทียม Artificial Neural Networks

สืบค้นจาก : <http://journal.hcu.ac.th/pdffile/5.%20%E0%B9%82%E0%B8%84%E0%B8%A3%E0%B8%87%E0%B8%82%E0%B9%88%E0%B8%B2%E0%B8%A2%2073-87.pdf>)



ภาพที่ 2.17 Neural Network แบบหลายชั้น

(ที่มา โครงขายประสาทเทียม Artificial Neural Networks

สืบค้นจาก : <http://journal.hcu.ac.th/pdffile/5.%20%E0%B9%82%E0%B8%84%E0%B8%A3%E0%B8%87%E0%B8%82%E0%B9%88%E0%B8%B2%E0%B8%A2%2073-87.pdf>)

Neural Network สามารถเรียนรู้ได้หลายประเภท ในที่นี้จะพูดถึงการเรียนรู้แบบมีผู้สอน (supervised learning) ข้อมูลสำหรับการสอนประกอบไปด้วยตัวอย่างข้อมูลที่ต้องการสอน และผลลัพธ์ที่ต้องการสอนให้ Network สร้างขึ้น เมื่อมีการนำข้อมูลในลักษณะเดียวกันมาเป็นข้อมูลป้อนเข้า Network จะกำหนดค่าผลลัพธ์ที่เป็นเป่าหมายให้กับข้อมูลป้อนเข้าแต่ละตัว Network จะนำค่าผิดพลาดระหว่างค่าเป้าหมายกับค่าผลลัพธ์ที่ได้ มาใช้ในการปรับค่าน้ำหนัก เพื่อให้ผลลัพธ์มีความใกล้เคียงกับเป้าหมายมากที่สุด [7]

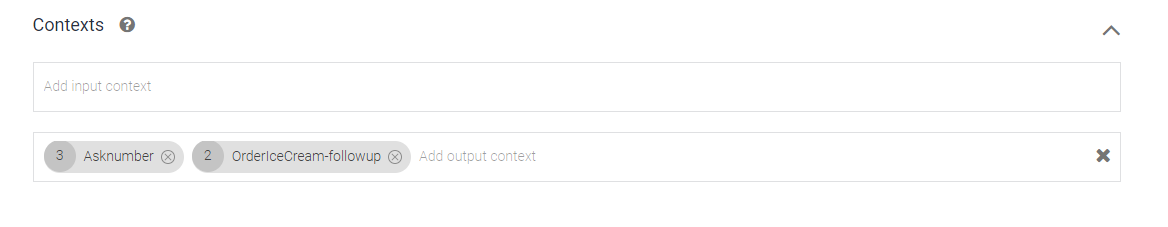
## 2.6 เครื่องมือที่ใช้ในการสร้างแชทบอท

ในการสร้างแชทบอทโดยทั่วไป จะมี framework ให้เลือกใช้หลายตัวขึ้นกับความสามารถของ framework แต่ละแบบว่าสามารถตอบโจทย์ความต้องการในการทำงานมากเพียงใด และบาง framework สามารถเชื่อมต่อไปยังแอปพลิเคชัน เพื่อเพิ่มความสะดวกให้แก่ผู้ใช้งาน แต่บาง framework ไม่สามารถทำได้ ซึ่งผู้จัดทำโครงานได้สืบค้น framework ต่างๆ และขอยกตัวอย่างมาดังนี้

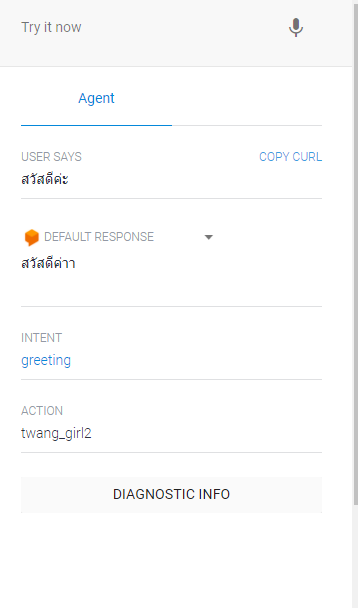
2.6.1 Dialogflow คือ framework ที่สามารถสร้างแชทบอทได้หลายภาษารวมทั้งภาษาไทย โดยไม่ต้องมีการเขียนโค้ดใดๆ และมีฟังก์ชันการใช้งานที่หลากหลาย รวมทั้งสามารเชื่อมต่อไปยัง platform ได้หลากหลาย ซึ่งมีจุดเด่น คือการรองรับการทำ Natural Language Understanding โดยที่ไม่ต้องเขียนโปรแกรมอะไรเพิ่มเติม เพื่อช่วยในเรื่องการพิมพ์ผิดหรือเขียนมาไม่ตรงกับประโยคที่สอนไปเบื้องต้น แต่สามารถหาข้อมูลที่ถูกต้องได้ ซึ่งสามารถใช้งานผ่าน web browser ได้ โดยไม่ต้องติดตั้งโปรแกรมบนเครื่องคอมพิวเตอร์

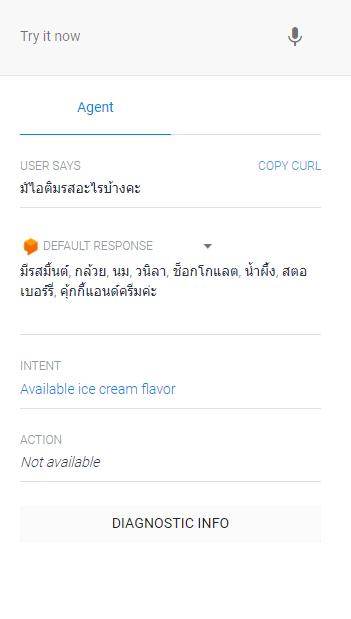
การใช้งาน Dialogflow จะมีส่วนสำคัญคือการสอนคำถามให้กับแชทบอท ซึ่งจะมีประโยคอื่นๆ เพื่อสอนให้แชทบอททราบว่าคำเหล่านี้อยู่ใน intent ใด อันส่งผลให้แชทบอทเรียนรู้ และแยกประโยคเองได้

เมื่อมีการสอนประโยคให้สื่อถึง intent แล้ว ต่อมาจะต้องสอนให้แชทบอทรู้ว่าต้องตอบประโยคว่าอย่างไร โดยสามารถตอบกลับด้วยการเขียนข้อความหรือโค้ดก็ได้ และหากเชื่อมไปยังแอปพลิเคชันอื่น ผู้สร้างสามารถสร้างเป็นภาษาเฉพาะของแอปพลิเคชันได้ ซึ่งหากมีมากกว่า 1 คำตอบ ในส่วนเดียวกัน แชทบอทจะสุ่มเลือกขึ้นมาเอง และสามารถตั้งให้ถามคำถามเพื่อขอคำตอบเพิ่มเติมได้ โดยต้องมีการกำหนดว่าต้องการคำตอบชนิดใด

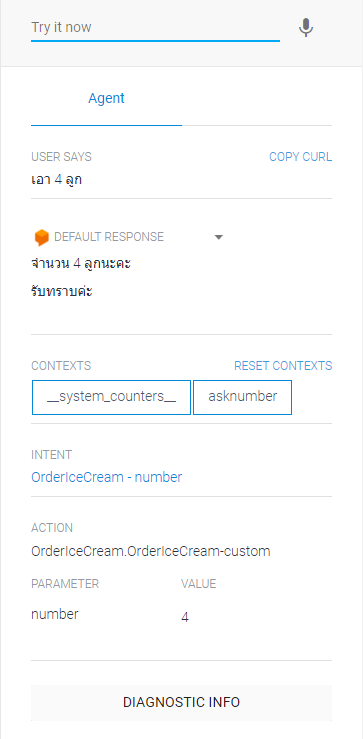
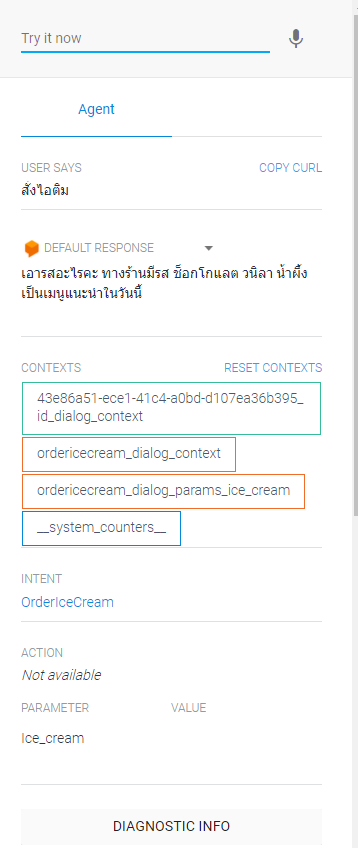
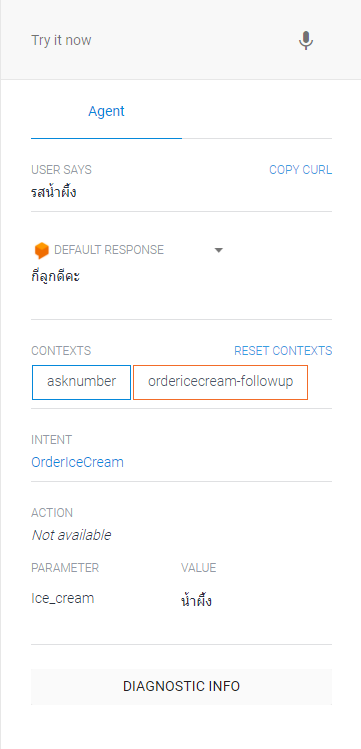
 Dialogflow สามารถสร้าง flow ให้มีการพูดคุยในหัวข้อนั้นได้ ด้วยการส่งต่อหัวข้อไปยัง intent ที่เกี่ยวข้องต่อไป โดยจะมีตัวเลขเป็นการนับว่าจะยังคงอยู่ในหัวข้อดังกล่าวต่อไปอีกนานแค่ไหน

ภาพที่ 2.18 หน้าจอ intent ส่วนของ contexts

ขั้นตอนในการทดสอบแชทบอท ก่อนนำไปใช้งานจริง ซึ่งในส่วนของ Dialogflow สามารถทำการทดสอบได้ตลอดเวลาในฝั่งด้านขวาของหน้าเว็บ [8] จากการทดสอบจะได้ผลออกมาดังนี้



ภาพที่ 2.19 หน้าจอแสดงผลการทดสอบแชทบอท Dialogflow (1)

ภาพที่ 2.20 หน้าจอแสดงผลการทดสอบแชทบอท Dialogflow (2)

ข้อดีของ Dialogflow

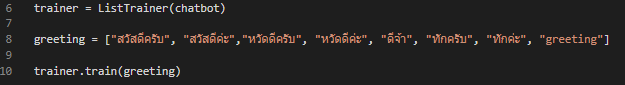
1. สามารถสร้างแชทบอท โดยไม่ต้องเขียนโค้ด
2. ทำแชทบอทครั้งเดียว แต่สามารถใช้ได้กับหลาย platform เช่น Line, Facebook messenger และเว็บแชทของ Dialogflow เป็นต้น
3. สามารถทำ pattern การสนทนาได้ คือ ต้องได้รับคำตอบครบทั้งหมดก่อน จึงสามารถประมวลผลได้ ยกตัวอย่างกรณีการคำนวณ BMI ที่ต้องได้รับค่าน้ำหนักและส่วนสูงครบก่อน จึงสามารถคำนวณค่า BMI ออกมาได้
4. สามารถใช้งานได้ฟรี และไม่ต้องมี server เป็นของตัวเอง
5. มีการรองรับ Natural Language understanding ทำให้ผู้ใช้เขียนข้อความนอกเหนือจากที่เคยสอนไปแชทบอทก็ยังสามารถเข้าใจได้

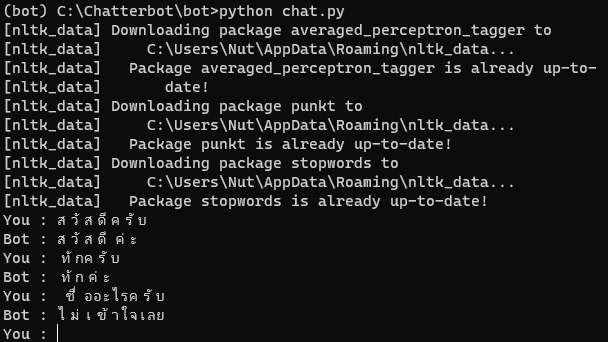
ข้อเสียของ Dialogflow

1. อินเตอร์เฟสใช้งานยากสำหรับคนที่ไม่ใช่สายงาน developer
2. ไม่มีการเก็บข้อมูลลงฐานข้อมูล
3. ไม่สามารถดึงข้อมูลที่ผู้ใช้เขียนเข้ามาภายในแชทบอทออกมาได้
4. เมื่อเขียนประโยคเพิ่มเข้าไปภายในแชทบอทผ่านหัวข้อ Training ด้านซ้าย ต้องจัดการแต่ละประโยคเองว่าเหมาะสมกับ intent ใด
5. ในการเชื่อมต่อไปยังแอปพลิเคชันอื่นมีขั้นตอนเยอะเกินไปสำหรับคนใช้งานทั่วไป

แนวทางที่นำมาใช้กับงาน

ใช้งาน API ที่เชื่อมกับระบบอื่นๆ เช่น database เป็นต้น

2.6.2 chatterbot คือ Library ภาษา Python ที่ใช้ทำแชทบอทแบบที่สามารถพิมพ์ผิดได้ เพราะ chatterbot มีการใช้ Machine Learning แบบ Supervised Learning ในการหาค่า Confidence เพื่อดูความคล้ายคลึงของประโยค และมี LogicAdapter คือวิธีที่จะใช้เลือกคำตอบที่อยู่ใน Data set นอกจากจะมีให้ใช้แล้ว ยังสามารถเขียนเพิ่มเข้าไปเองได้ ซึ่ง chatterbot นั้นต้องมีการเขียนข้อความ train ในรูปแบบ list [9]

ภาพที่ 2.21 หน้าจอโปรแกรมเขียน train ให้กับ chatbot

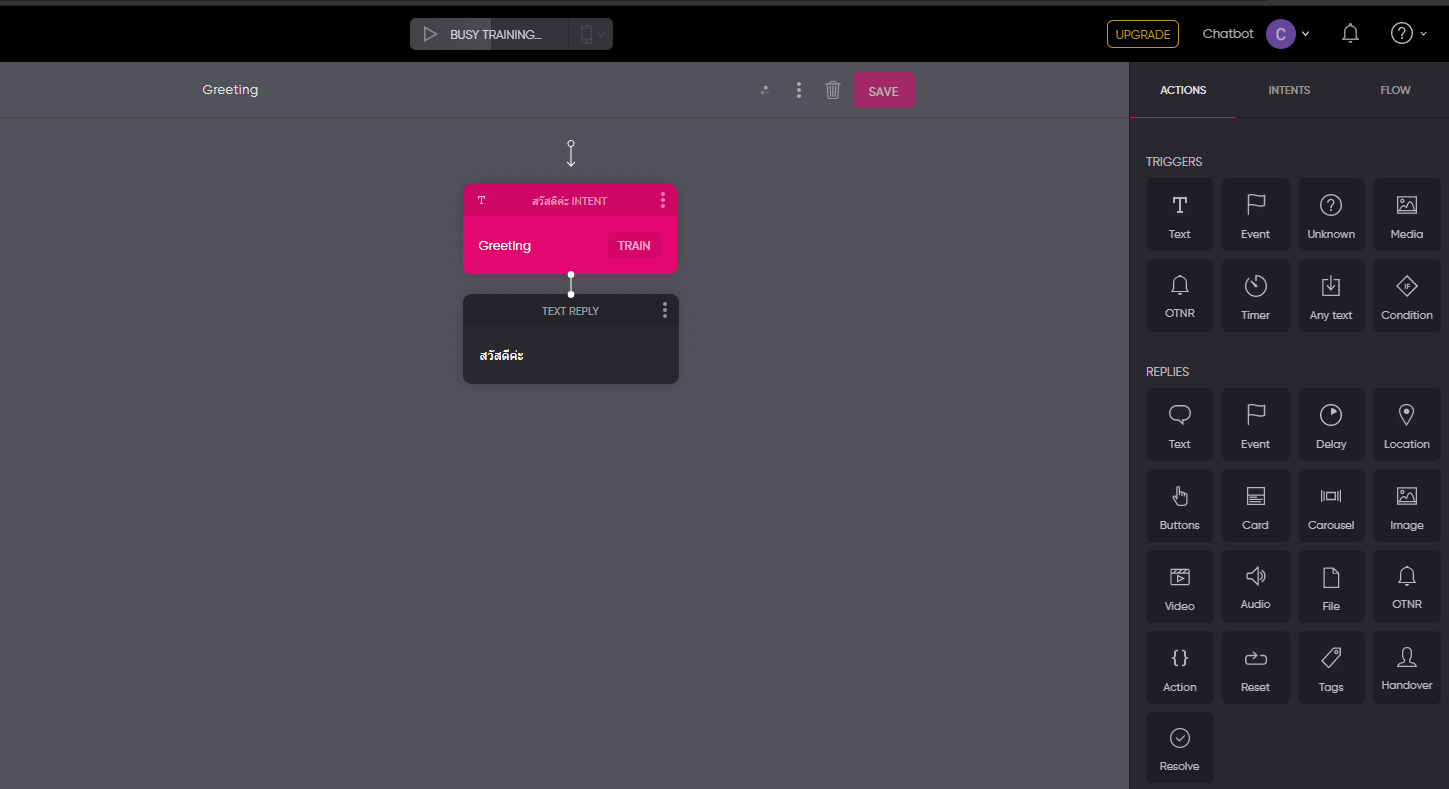
ภาพที่ 2.22 หน้าจอผลลัพธ์จากการรันโปรแกรมที่เขียนด้วย chatterbot

ข้อดีของ chatterbot

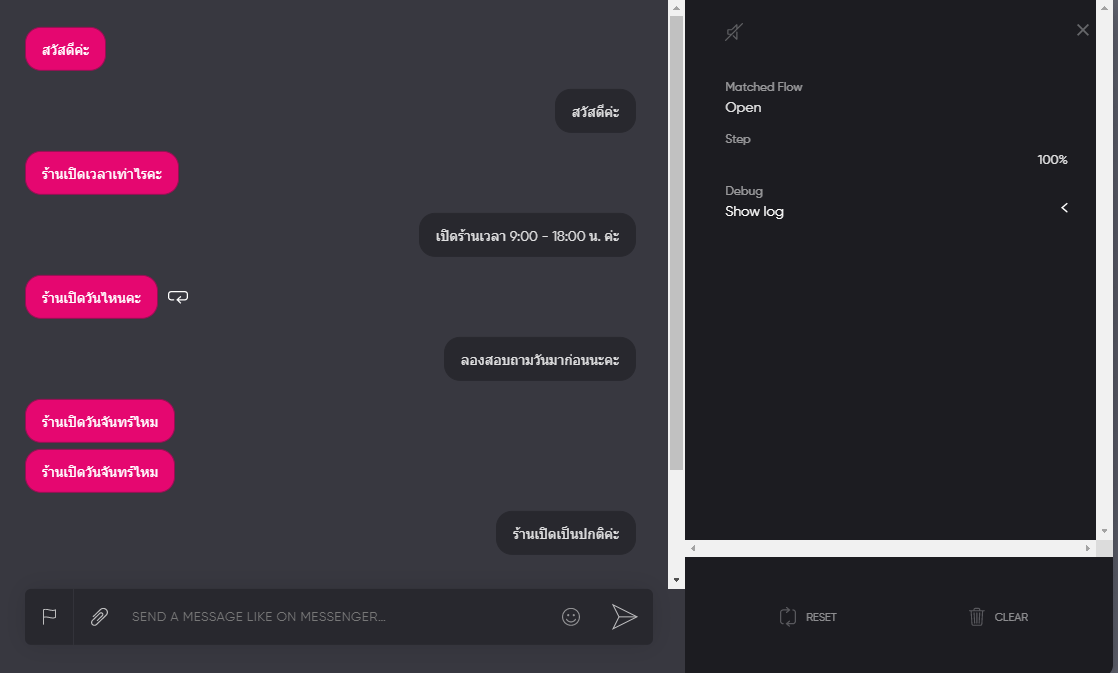
1. เมื่อมีการพิมพ์ต่างจากที่สอนไป แชทบอทจะยังสามารถเข้าใจได้ด้วยการแยกคำ
2. มีการจดจำบริบทก่อนหน้าไว้ในไฟล์ ทำให้มีความสนใจกับคำที่ถูกพูดถึงบ่อยๆ เป็นพิเศษ

ข้อเสียของ chatterbot

1. มีคำตอบตามรูปแบบที่กำหนดเท่านั้น
2. การสร้าง story ทำได้ยาก

2.6.3 flow.ai คือ โปรแกรมสำหรับสร้างแชทบอท ที่สามารถตอบกลับด้วยข้อความ เสียง วิดิโอและฯลฯ ซึ่งตัวโปรแกรมนี้ถูกออกแบบมาเพื่อการทำงานเป็นทีมแบบวันต่อวันผ่าน web browser และมีการรองรับภาษาที่หลากหลาย ซึ่งโปรแกรม flow.ai จะเน้นไปที่ flow ในการพูดคุยเป็นหลัก จึงทำให้หน้าต่างหลักจะแสดง flow ของข้อความเพื่อให้เข้าใจได้ง่าย และตรงจุดประสงค์หลักของการใช้งาน โดยในการสร้างแชทบอทจะให้เลือกว่าจะเชื่อมต่อไปยังแอปพลิเคชันใด และใช้งานในด้านใดเป็นหลัก ซึ่งจะส่งผลต่อข้อมูลต่อมาที่ระบบต้องการรับ และหน้าต่างแสดงผลเมื่อมีการรัน [10]

ภาพที่ 2.23 หน้าจอหลักของ flow.ai

ภาพที่ 2.24 หน้าจอแสดงผลการรันของ flow.ai

ข้อดีของ flow.ai

1. มีการเรียก api ไปบริการบนเว็บ
2. สามารถสร้างแชทบอทโดยไม่ต้องเขียนโค้ด
3. สามารถใช้งานได้ฟรี และไม่ต้องมี server เป็นของตัวเอง
4. มีการเข้าใจภาษาด้วย NLP ทำให้เมื่อเขียนข้อความที่ไม่ได้สอนก็ยังสามารถหาได้

ข้อเสียของ flow.ai

1. ไม่เหมาะกับบุคคลทั่วไปที่ไม่ใช่ developer
2. ไม่มี entity เบื้องต้นมาให้ จึงต้องทำเองทั้งหมด

# บทที่ 3 การวิเคราะห์และขั้นตอนวิธี

ในการสร้างแชทบอทนั้น เราจะต้องมีการเตรียมชุดข้อมูลที่ประกอบด้วยข้อความ (message) และการระบุเจตนา (intent มาจากคำว่า intention) โดยทั่วไปการระบุ intent ให้กับข้อความจำเป็นจะต้องใช้มนุษย์เป็นผู้พิจารณาว่าข้อความนั้นๆ ตรงกับ intent ใด ซึ่งหากมีข้อความจำนวนมาก จะทำให้ต้องใช้เวลานาน การจัดกลุ่มข้อความที่มีความคล้ายกันให้อยู่ในกลุ่มเดียวกันแล้วระบุ intent ในคราวเดียวกันจะทำให้การเตรียมข้อมูลมีความรวดเร็วมากขึ้น นอกจากนั้นแชทบอทที่มีการสร้างขึ้น เมื่อนำไปใช้งานสักระยะจะมีข้อความใหม่ที่ควรนำมาสอนแชทบอทเพิ่มเติม การระบุ intent ให้กับข้อความใหม่นี้หากมีจำนวนมาก และใช้มนุษย์ทำก็จะสิ้นเปลืองเวลาเช่นกัน จึงต้องมีโมเดลสำหรับจำแนกข้อความที่เกิดขึ้นใหม่เหล่านี้ว่าเป็น intent ใด

คณะผู้จัดทำโครงงานจึงได้วิเคราะห์และออกแบบขั้นตอนในการพัฒนางาน โดยประกอบไปด้วย 4 หัวข้อหลัก ดังนี้

* การจัดกลุ่มข้อความเพื่อกำหนด intent
* การสร้างโมเดล และทดสอบโมเดลในการจำแนก intent
* ชุดข้อมูลที่ใช้ทำการทดสอบ และตัวชี้วัดประสิทธิภาพ
* การประยุกต์ใช้ เพื่อพัฒนาต้นแบบแชทบอท และการอัพเดตแชทบอท

โดยรายละเอียดในการดำเนินการแต่ละหัวข้อ จะกล่าวดังต่อไปนี้

## 3.1 การจัดกลุ่มข้อความเพื่อกำหนด intent

ในขั้นตอนแรกจะเป็นการจัดกลุ่มของข้อความ โดยใช้การเรียนรู้ของเครื่อง แบบไม่มีผู้สอน เมื่อสามารถแบ่งกลุ่มเรียบร้อยแล้ว จะให้มนุษย์มาระบุ intent ให้กับข้อความแต่ละกลุ่ม ข้อความทั้งหมดที่อยู่ในกลุ่มนั้นๆ จะมี intent เดียวกัน โดยจะมีลำดับขั้นตอนดังภาพที่ 3.1 ประกอบด้วย

* การทำความสะอาดข้อความ
* การเตรียมชุดคำศัพท์
* การสกัดคุณลักษณะ
* การจัดกลุ่ม
* การกำหนด intent label

ดังรายละเอียดที่จะกล่าวต่อไปนี้

Diagram

Description automatically generatedภาพที่ 3.1 ขั้นตอนการทำงาน A: การจัดกลุ่มข้อความเพื่อกำหนด intent

3.1.1 การทำความสะอาดข้อความ (Cleaning messages)

ในขั้นตอนนี้จะเป็นการเตรียมข้อมูลให้เหมาะสมสำหรับการนำไปวิเคราะห์ [2] มีขั้นตอนดังนี้

1. ปรับอักขระให้อยู่ในรูปแบบเดียวกัน คือ พิมพ์เล็ก ดังตัวอย่างต่อไปนี้

Where is the Songkhla city? -> where is the songkhla city?

1. ลบเครื่องหมาย หรืออักขระพิเศษ และตัวเลข ด้วยการแทนค่าเหล่านั้นด้วยช่องว่าง ถ้าช่องว่างติดกันมากกว่า 1 ช่อง ให้ทำการลบออกจนเหลือเพียง 1 ช่อง ดังตัวอย่างต่อไปนี้

hi! how are you? -> hi how are you

i am 22 year old -> i am 22 year old

1. ตัดแบ่งแต่ละคำด้วยช่องว่าง ดังตัวอย่างต่อไปนี้

hi how are you -> “hi”, “how”, “are”, “you”

1. ลบ stop word เนื่องจาก stop word เป็นคำที่ปรากฎในข้อความบ่อยครั้ง ซึ่งมีผลน้อยมากในการแบ่งกลุ่ม หรือจำแนกข้อความ เช่น a, an, do, to, or และ is ดังตัวอย่างต่อไปนี้

“hi”, “how”, “are”, “you” -> “hi”

1. เปลี่ยนรูปคำ/ลดรูปคำ (Stemming) ที่มีความหมายเหมือนกันให้อยู่ในรูปของรากศัพท์ ดังตัวอย่างต่อไปนี้

is, am, are -> be

loads, loaded, loading -> load

3.1.2 การสร้างชุดคำศัพท์ (Preparing term set)

ให้ m แทน จำนวนข้อความที่พิจารณา และ

**M1, M2, …, Mm** แทน เซตของคำศัพท์ ในข้อความที่ 1, 2, …, m ที่ได้ผ่านการทำความสะอาดตามกระบวนการในข้อที่ 3.1.1 มาแล้ว

คำศัพท์ทั้งหมดจะถูกนำมายูเนียนกัน เพื่อสร้างเป็นชุดคำศัพท์ หรือดิกชันนารี (Dictionary) จะได้ว่า

เมื่อ n คือ จำนวนคำศัพท์ที่แตกต่างกัน ชุดคำศัพท์ที่ได้จะถูกนำไปใช้เป็นตัวกำหนดจำนวนมิติในการสกัดคุณลักษณะต่อไป

3.1.3 การสกัดคุณลักษณะของข้อมูล (Feature extraction)

ข้อความที่ทำความสะอาดแล้วตามหัวที่ 3.1.1 และสร้างชุดคำศัพท์ที่ได้ตามหัวข้อ 3.1.2 จะถูกนำมาใช้ในการสกัดคุณลักษณะข้อความให้ได้เป็นเวกเตอร์ โดยในโครงงานนี้  
จะทำการแทนข้อความด้วยเวกเตอร์ด้วย TF-IDF ดังวิธีการที่ได้กล่าวไว้ในหัวข้อที่ 2.3 จากนั้นทำการลดจำนวนมิติของเวกเตอร์ลง (Feature reduction) โดยใช้หลักการ Principal Components Analysis (PCA) ซึ่งเราจะได้เวกเตอร์ component และค่า variance ออกมาทั้งหมดเท่ากับจำนวนมิติที่มี และ component ที่มีความสำคัญจะมีค่า variance สูง [11] ทำให้ในการลดจำนวนมิติ เราจึงเก็บ component ที่มีค่า variance สูง เพื่อไม่ให้ส่วนที่มีความสำคัญหายไปจากการลดมิติ และทำการลดมิติ component เมื่อค่า variance ต่ำ อย่างไรก็ตามการลดมิติมากไปก็ไม่ดี ในโครงงานนี้จึงใช้หลักการ ดังนี้

* 1. หาค่าเปอร์เซนต์ของ variance เมื่อเทียบกับผลรวมของ variance ทั้งหมด

โดยที่ *i* = 1, 2, …, *n*

โดย ค่า คือ ค่าความแปรปรวนของ component ที่ *i* โดยที่ หน่วยเป็นเปอร์เซนต์

ค่า คือ ค่าความแปรปรวนของ component ที่ *i*

ค่า คือ ค่าความแปรปรวนรวมทั้งหมด ตั้งแต่ component ที่ 1 จนถึง n

* 1. นำค่า ของ *i* ตั้งแต่ 1 จนถึง *n* ที่ได้มาคำนวณหาค่า component ที่ดีที่สุด ด้วยสมการดังต่อไปนี้

โดยที่จำนวน component ที่ดีที่สุดจะได้จากจำนวนผลรวม k ตัวแรก ของ ที่ทำให้ ซึ่งแทนว่ามีจำนวน component ที่ดีที่สุด k ตัว เขียนได้ว่า [] ซึ่งจะใช้ในการลดมิติของข้อมูล ผลลัพธ์สุดท้ายจะได้ออกมาเป็นเวกเตอร์ข้อความตามขนาดของมิติ คือ ขนาด k

3.1.4 การแบ่งกลุ่ม (Clustering)

ข้อความที่จะนำมาสร้างแชทบอทจะถูกนำมาจัดกลุ่ม เพื่อให้การระบุ intent label เป็นไปอย่างรวดเร็ว และมีประสิทธิภาพ ในโครงงานนี้ทำการทดลองจัดกลุ่มข้อความด้วย 2 วิธี ได้แก่

1. K-Means

เป็นการจัดกลุ่มข้อมูลแบบ center-based ดังหลักการที่ได้กล่าวไว้ในหัวข้อ 2.5.1 หัวข้อที่ 1) โดยจะต้องมีการกำหนดพารามิเตอร์ k ซึ่งแสดงถึงจำนวนกลุ่มที่ต้องการจัดข้อมูล ในการทำโครงงานนี้ทางคณะผู้จัดทำจะหาค่า k ที่ดีที่สุด [12] เพื่อนำมาแบ่งกลุ่มข้อมูล โดยไม่ต้องใช้มนุษย์เข้ามาช่วย ด้วยหลักการ The Silhouette Method

1. DBSCAN

เป็นการจัดกลุ่มแบบ density-based ดังที่ได้กล่าวไว้ในหัวข้อที่ 2.5.1 หัวข้อที่ 2) พารามิเตอร์ที่ต้องใช้ในการจัดกลุ่ม คือ Eps ซึ่งแสดงถึงรัศมีของบริเวณพื้นที่ และ MinPts ซึ่งแสดงถึงจำนวนจุดข้อมูลที่อยู่ในบริเวณที่พิจารณา

ผลจากการจัดกลุ่มทั้ง 2 วิธี จะถูกนำมาประเมิลผลเปรียบเทียบประสิทธิภาพต่อไปในหัวข้อที่ 3.9 เพื่อเลือกเทคนิคการจัดกลุ่มที่เหมาะสมไปใช้จัดกลุ่มข้อความ ก่อนที่จะนำไปพิจารณาระบุ intent ของแต่ละกลุ่ม ซึ่งข้อความที่อยู่ในกลุ่มเดียวกันจะได้รับการระบุว่าเป็น intent เดียวกันทั้งกลุ่ม จากนั้นข้อความที่มีการระบุ intent แล้วจะถูกนำไปใช้ในขั้นตอนต่อไป

## 3.2 การสร้างโมเดลและทดสอบโมเดลในการจำแนก intent

ในขั้นตอนนี้เป็นการนำชุดข้อความที่ผ่านการแบ่งกลุ่ม และระบุ intent เรียบร้อยแล้ว มาสร้างโมเดลสำหรับทำนาย (Predict) intent ให้กับข้อความ โดยใช้การเรียนรู้ของเครื่องแบบมีผู้สอน ข้อมูลทั้งหมดจะถูกแบ่งเป็น 2 ส่วน คือ ชุดข้อมูลสอน (Training Data) และชุดข้อมูลทดสอบ (Testing Data) โดยจะทำการทดลองด้วยวิธีการแบ่งข้อมูลเป็นส่วน ข้อความ และ intent ของแต่ละข้อความ โดยจะนำข้อมูลสอนเข้าสู่กระบวนการเรียนรู้ (Training Process) จนได้เป็นโมเดล โดยจะมีลำดับขั้นตอนดังภาพที่ 3.2 ดังรายละเอียดที่จะกล่าวต่อไปนี้

เริ่มต้นจะต้องมีการนำข้อความในชุดข้อมูลสอน มาทำความสะอาดตามหลักการที่ได้กล่าวไว้ก่อนหน้านี้ในหัวข้อที่ 3.1.1 แล้วทำการสร้างชุดคำศัพท์ตามวิธีในหัวข้อที่ 3.1.2 ข้อความที่ผ่านการทำความสะอาด และ intent label จะใช้เป็นข้อมูลนำเข้าสู่กระบวนการเรียนรู้ เพื่อสร้างโมเดล ในโครงงานนี้จะทำการทดลองสร้างโมเดล โดยใช้ 2 เทคนิค คือ Decision Tree และ Neural Networks ทำให้โมเดลที่ได้จะถูกนำไปใช้ในกระบวนการทดสอบต่อไป

Diagram

Description automatically generated

ภาพที่ 3.2 ขั้นตอนการทำงาน B1: การสร้างโมเดล

สำหรับกระบวนการทดสอบโมเดล จะนำข้อความจากชุดทดสอบที่ถูกแบ่งไว้ก่อนหน้านี้ มาทำนาย intent โดยใช้โมเดลที่สร้างขึ้น โดยจะมีลำดับขั้นตอนดังภาพที่ 3.3

เริ่มต้นจากการทำความสะอาดข้อความ แล้วทำชุดคำศัพท์ที่ได้จากกระบวนการสอนมาใช้ในการสกัดคุณลักษณะให้เป็นเวกเตอร์ TF-IDF เมื่อนำเข้าสู่โมเดลทำนายว่าเป็น intent ใด จากนั้นเราจะนำผลการทำนายไปเปรียบเทียบประสิทธิภาพ โดยใช้ตัวชี้วัดที่กล่าวไว้ในหัวข้อ 3.3.3 ต่อไป

Diagram

Description automatically generated

ภาพที่ 3.3 ขั้นตอนการทำงาน B2: ทดสอบโมเดลในการจำแนก intent

## 3.3 ชุดข้อมูลที่ใช้ทำการทดลองและตัวชี้วัดประสิทธิภาพ

3.3.1 ชุดข้อมูลที่ใช้

ทางคณะผู้จัดทำได้ใช้ชุดข้อมูลมาตรฐานสำหรับทดสอบข้อมูลจำนวน 3 กลุ่ม ได้แก่

1. ชุดข้อมูล ATIS หรือ Airline Travel Information System จาก [https://www.kaggle.com/hassanamin/atis-airlinetravelinformationsystem?select=atis\_intents.csv](https://www.kaggle.com/hassanamin/atis-airlinetravelinformationsystem?select=atis_intents.csv%20) เป็นชุดข้อมูลมาตรฐานของที่ใช้กันอย่างแพร่หลายสำหรับสร้าง chatbot
2. ชุดข้อมูล Corona Dataset จาก [https://github.com/botxo/corona\_dataset](https://github.com/botxo/corona_dataset%20%20)  ชุดข้อมูลนี้สามารถนำไปใช้ในการสอนแชทบอทให้สามารถเข้าใจคำถามเกี่ยว corona virus ได้
3. ชุดข้อมูล case routing intent จาก [language - Einstein intent - training dataset error - Salesforce Stack Exchange](https://salesforce.stackexchange.com/questions/199697/einstein-intent-training-dataset-error) เป็นชุดข้อมูลคำถามเกี่ยวกับการซื้อของ

3.3.2 ข้อมูลจำลองเพื่อสร้างแชทบอทโดยอาศัยวิธีการจัดกลุ่มและจำแนกที่นำเสนอในโครงงานนี้

ทางคณะผู้จัดทำได้สร้างชุดข้อมูลทดสอบจากหัวข้อ “แชทบอทสำหรับสถาบันการศึกษา” โดยประกอบไปด้วยข้อความสำหรับทดสอบจำนวน 100 ข้อความ

3.3.3 การประเมินผล (Evaluation)

เป็นการตรวจสอบประสิทธิภาพของวิธีการในส่วนต่างๆ ดังนี้

1) การประเมินผลการแบ่งกลุ่ม (Clustering evaluation)

ทางคณะผู้จัดทำใช้วิธี Purity ในการตรวจสอบการแบ่งกลุ่ม [13] โดยมีสมการ ดังนี้

เมื่อ *i* = 1, 2, … ,*k* กลุ่ม และ *j* = 1, 2, …, *l* คลาส

โดย คือ จำนวนข้อมูลที่ิอยู่ภายในกลุ่ม *i* ทั้งหมด

คือ จำนวนข้อมูลในคลาส *j* ที่อยู่ในกลุ่ม *i* ทั้งหมด

หลังจากได้ค่า จะนำมาคำนวณหาค่า Purity ของกลุ่มที่ *i* เขียนแทนด้วย มีสูตรดังสมการ

จากนั้นนำค่า ที่ได้ไปหาค่า Purity ของการแบ่งกลุ่มด้วยสมการ

โดย m คือ จำนวนข้อมูลทั้งหมดที่นำมาแบ่งกลุ่ม

2) การประเมินผลการจำแนก (Classification evaluation)

เป็นการตรวจสอบประสิทธิภาพในการจำแนกของวิธีการทั้ง 2 วิธีที่คณะผู้จัดทำสนใจ โดยวิธีที่ทางคณะผู้จัดทำใช้ในการประเมินได้แก่ Precision, Recall และ F1-Score [14] โดยก่อนที่จะทำความเข้าใจ จำเป็นจะต้องรู้คำศัพท์ที่ต้องใช้เสียก่อน

TP (True Positive) คือ จำนวนข้อมูลทดสอบที่จำแนกว่าอยู่ในคลาสที่สนใจ และถูกต้องตามเฉลย

TN (True Negative) คือ จำนวนข้อมูลทดสอบที่จำแนกว่าไม่อยู่ในคลาสที่สนใจ และถูกต้องตามเฉลย

FN (False Negative) คือ จำนวนข้อมูลทดสอบที่จำแนกว่าอยู่ในคลาสที่สนใจ แต่ไม่ถูกต้องตามเฉลย

FP (False Positive) คือ จำนวนข้อมูลทดสอบที่จำแนกว่าไม่อยู่ในคลาสที่สนใจ แต่ไม่ถูกต้องตามเฉลย

จากนั้นนำไปใช้กับวิธีการในการประเมินข้างต้น ดังนี้

1. Precision คือ สัดส่วนความถูกต้องของผลลัพธ์ที่ได้จากการจำแนกคลาสที่สนใจ ต่อจำนวนข้อมูลที่อยู่ในคลาสที่สนใจทั้งหมดตามผลเฉลย เป็นการตรวจสอบความแม่นยำในการจำแนกของโมเดล
2. Recall คือ สัดส่วนความถูกต้องของผลลัพธ์ที่ได้จากการจำแนกคลาสที่สนใจ ต่อจำนวนข้อมูลที่ได้จากการจำแนกจากคลาสที่สนใจทั้งหมด เป็นการตรวจความครบถ้วนในการจำแนกของโมเดล
3. F1-Score คือ การหาค่าเฉลี่ยแบบ harmonic mean ของ Precision และ Recall เพื่อใช้ในการอธิบายประสิทธิภาพของโมเดล.

## 3.4 การประยุกต์ใช้เพื่อพัฒนาต้นแบบแชทบอทและการอัพเดตแชทบอท

ในหัวข้อนี้จะกล่าวถึงการประยุกต์ใช้ 2 ส่วน คือ การสร้างแชทบอท และการอัพเดต  
แชทบอท ดังรายละเอียดที่จะกล่าวต่อไปนี้

3.4.1 การสร้างแชทบอท

ในขั้นตอนนี้เป็นการนำชุดข้อความที่ยังไม่มีการระบุ intent ที่เตรียมไว้สำหรับสร้าง  
แชทบอท มาระบุ intent ของข้อความโดยใช้โมเดลที่สร้างขึ้นจากขั้นตอนก่อนหน้า โดยมีการทำความสะอาดข้อความ และการสกัดคุณลักษณะ ซึ่งเป็นหลักการเดียวกันกับตอนสร้างโมเดล หลังจากการนำข้อความเข้าสู่โมเดล ค่าผลลัพธ์ที่ได้จะปรากฎในรูป เซตข้อความ แทนด้วย M และเซต intent ของข้อความ แทนด้วย L ซึ่งจะนำเข้าสู่ Dialogflow โดยจะมีการกำหนดคำตอบที่เหมาะสมตาม intent เพื่อเป็นการสอนแชทบอท โดยจะมีลำดับขั้นตอนดังภาพที่ 3.4

Diagram

Description automatically generated

ภาพที่ 3.4 ขั้นตอนการทำงาน C: พัฒนาต้นแบบแชทบอท

3.4.2 การอัพเดตแชทบอท

เมื่อมีคำถาม ซึ่งไม่มีอยู่ภายในแชทบอท เช่น คำถามจากแชทบอทที่แชทบอทไม่สามารถตอบได้ หรือข้อมูลเพิ่มเติมที่หาได้ ข้อความเหล่านี้จะถูกนำเข้ามาในกระบวนการ เพื่ออัพเดตแชทบอทอีกครั้ง ดังขั้นตอนในภาพที่ 3.5 จะต้องนำมาระบุ intent ด้วยการนำเข้าสู่โมเดล และนำผลลัพธ์ที่เรียกว่า เซตของข้อความใหม่ แทนด้วย Mnew และเซตของ intent ของข้อความใหม่ แทนด้วย Lnew และนำมารวมกับข้อมูลเดิม จะเขียนออกมาได้ว่า

ก่อนนำเข้า Dialogflow เพื่ออัปเดตแชทบอท ให้มีข้อมูลเพิ่มขึ้น และเป็นการอัพเดตแชทบอท ให้มีความสามารถมากขึ้น

Diagram

Description automatically generated

ภาพที่ 3.5 ขั้นตอนการทำงาน D: การอัพเดตแชทบอท

# เอกสารอ้างอิง

[1] ไม่ปรากฎผู้แต่ง, ไม่ปรากฎปีที่พิมพ์, Different types of chatbots: Rule-based vs. NLP,

สืบค้นเมื่อวันที่ 8 มกราคม 2564, สืบค้นจาก <https://flow.ai/blog/kb-different-kinds-of-chatbots>

[2] Christopher D. Manning, Prabhakar Raghavan and Hinrich Schütze, ปี 2008,

Introduction to Information Retrieval, Cambridge UniversityPress, สืบค้นเมื่อวันที่ 10 มกราคม 2564, สืบค้นจาก <http://nlp.stanford.edu/IR-book/information-retrieval-book.html>

[3] ไม่ปรากฎผู้แต่ง, วันที่แก้ไขล่าสุด วันที่ 9 มกราคม 2564, Jaccard index, สืบค้นเมื่อวันที่ 10

มกราคม 2564, สืบค้นจาก <https://en.wikipedia.org/wiki/Jaccard_index>

[4] Shriti Kapil, Meenu Chawla, Mohd Dilshad Ansari, ปี 2559, On K-means Data

Clustering Algorithm with Genetic Algorithm, สืบค้นเมื่อวันที่ 9 มกราคม 2564, สืบค้นจาก <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7913145>

[5] ไม่ปรากฎผู้แต่ง, ไม่ปรากฎปีที่พิมพ์, DBSCAN: density-based clustering for discovering

clusters in large datasets with noise - Unsupervised Machine Learning, สืบค้นเมื่อวันที่ 10 มกราคม 2564, สืบค้นจาก<http://www.sthda.com/english/wiki/wiki.php?id_contents=7940>

[6] Tom M. Mitchell, วันที่ 1 มีนาคม 2540, Machine Learning, สืบค้นเมื่อวันที่ 10 มกราคม

2564, สืบค้นจาก <http://www.cs.ubbcluj.ro/~gabis/ml/ML-books/McGrawHill%20-%20Machine%20Learning%20-Tom%20Mitchell.pdf>

[7] ธนาวุฒิ ประกอบผล, วันที่ 24 มกราคม 2552, โครงขายประสาทเทียม Artificial Neural

Networks, สืบค้นเมื่อวันที่ 10 มกราคม 2564, สืบค้นจาก <http://journal.hcu.ac.th/pdffile/5.%20%E0%B9%82%E0%B8%84%E0%B8%A3%E0%B8%87%E0%B8%82%E0%B9%88%E0%B8%B2%E0%B8%A2%2073-87.pdf>

[8] ไม่ปรากฎผู้แต่ง, วันที่แก้ไขล่าสุด วันที่ 10 ธันวาคม 2564, Dialogflow, สืบค้นวันที่ 3 มกราคม

2564, สืบค้นจาก <https://cloud.google.com/dialogflow/docs>

[9] ไม่ปรากฎผู้แต่ง, ปี 2562, About ChatterBot, สืบค้นวันที่ 4 มกราคม 2564, สืบค้นจาก

<https://chatterbot.readthedocs.io/en/stable/>

[10] ไม่ปรากฎผู้แต่ง, ไม่ปรากฎปีที่พิมพ์, What can Flow.ai do?, สืบค้นวันที่ 9 มกราคม 2564,

สืบค้นจาก <https://flow.ai/docs/what-can-flow-do>

[11] Pop Phiphat, วันที่ 23 มิถุนายน 2561, Principal Components Analysis (PCA) ต่างจาก

Factor Analysis (FA) ยังไง ??? (ตอนที่ 1), สืบค้นเมื่อวันที่ 1 มีนาคม 2564, สืบค้นจาก <https://medium.com/ingenio/principal-components-analysis-pca-%E0%B8%95%E0%B9%88%E0%B8%B2%E0%B8%87%E0%B8%88%E0%B8%B2%E0%B8%81-factor-analysis-fa-%E0%B8%A2%E0%B8%B1%E0%B8%87%E0%B9%84%E0%B8%87-%E0%B8%95%E0%B8%AD%E0%B8%99%E0%B8%97%E0%B8%B5%E0%B9%88-1-c395e55bdc3>

[12] Khyati Mahendru, วันที่ 17 มิถุนายน 2562, How to Determine the Optimal K for K-

Means?, สืบค้นเมื่อวันที่ 3 มีนาคม 2564, สืบค้นจาก [https://medium.com/analytics-](https://medium.com/analytics-vidhya/how-to-determine-the-optimal-k-for-k-means-708505d204eb)

[vidhya/how-to-determine-the-optimal-k-for-k-means-708505d204eb](https://medium.com/analytics-vidhya/how-to-determine-the-optimal-k-for-k-means-708505d204eb)

[13] ไม่ปรากฎผู้แต่ง, ไม่ปรากฎปีที่พิมพ์, Cluster Analysis: Basic Concepts and Algorithms,

สืบค้นเมื่อวันที่ 8 มีนาคม 2564, สืบค้นจาก <https://www-users.cs.umn.edu/~kumar001/dmbook/ch8.pdf?fbclid=IwAR04kwi6bFDWLzmjKtGMaW67J4wBBMISma-tZ4svV6L_ZNpJ7Xmhy_PNS7U>

[14] ไม่ปรากฎผู้แต่ง, ไม่ปรากฎปีที่พิมพ์, 3.3. Metrics and scoring: quantifying the quality of

predictions, สืบค้นเมื่อวันที่ 10 มีนาคม 2564, สืบค้นจาก [https://scikit-learn.org/stable/modules/model\_evaluation.html?fbclid=IwAR2JOz6tVB2PNu0oQUbssP48jiXQbhZvS1lbQXrXMlCTQjws2OCsGajAdi8#the-scoring-parameter-defining-model-evaluation-rules](https://scikit-learn.org/stable/modules/model_evaluation.html?fbclid=IwAR2JOz6tVB2PNu0oQUbssP48jiXQbhZvS1lbQXrXMlCTQjws2OCsGajAdi8%23the-scoring-parameter-defining-model-evaluation-rules)