

## การจำแนกเจตนาการสนทนาเพื่อการสร้างแชทบอท Intent Classification for Building Chatbot

ณัฐพล เดชประมวณพล และ วริศรา พิสุทธิเจียร

หลักสูตรวิทยาการคอมพิวเตอร์ สาขาวิทยาศาสตร์การคำนวณ คณะวิทยาศาสตร์ วิทยาลัยสงขลานครินทร์

6110210129@email.psu.ac.th, 6110210373@email.psu.ac.th

### บทคัดย่อ

ในปัจจุบัน การสื่อสารจากองค์กรมักมีแชทบอทเข้ามาเป็นตัวช่วยอยู่บ่อยครั้ง แต่ในการทำแชทบอทนั้นมีส่วนที่ยากและสำคัญที่สุดคือการจำแนกเจตนาการสนทนา ในโครงการนี้ศึกษาเรื่องการจัดกลุ่มของข้อความจำนวนมาก โดยใช้การเรียนรู้ของเครื่องแบบ Unsupervised Learning เพื่อสร้าง intent สำหรับสร้างแชทบอทเวอร์ชันแรก และใช้การเรียนรู้ของเครื่องแบบ Supervised Learning เพื่อกำหนด intent ให้ข้อความใหม่ที่จะได้รับเข้ามาสำหรับอัปเดตแชทบอท จากการทดลองด้วยชุดข้อมูลมาตรฐาน 3 ชุด และชุดข้อมูลที่สร้างขึ้นเอง ซึ่งมีการจำลองจากกลุ่มแชทในหัวข้อการเรียนรู้หลักสูตรวิทยาการคอมพิวเตอร์ของมหาวิทยาลัยสงขลานครินทร์ พบว่าวิธีการแบบ deep K-means ที่มีการลดมิติแบบ PCA เป็นวิธีการจัดกลุ่มข้อความที่ดีที่สุด เมื่อพิจารณาจากค่าตัวชี้วัด Purity และ MSE และวิธีการแบบ MLP Neural Network สามารถจำแนกข้อความไปตามกลุ่มเจตนาได้ดีที่สุด เมื่อพิจารณาจากตัวชี้วัดประสิทธิภาพ F1-Score และ Accuracy ดังนั้น ในแชทบอทเวอร์ชันแรกจะนำผลการจัดกลุ่มด้วยวิธี deep K-means ที่นำมากำหนดชื่อ intent เป็นข้อมูลสอน และทำการสอนโมเดลด้วย MLP Neural Network เพื่อระบุ intent ให้กับข้อความเพื่ออัปเดตแชทบอท ผู้จัดทำสร้างแชทบอทผ่าน Dialogflow และใช้งานจริงบนแอปพลิเคชัน Line

**คำสำคัญ :** ทำนาย, การจัดกลุ่ม, การจำแนก

### Abstract

Nowadays, communication with organizations have often chatbot to help. However in chatbot, the hardest and importance part is to identify intent from conversation. In this project, we studied the intent classification of the message using machine learning techniques including Unsupervised Learning to generate intents for the first version of chatbot and Supervised Learning to generate intents for

the updated version of chatbot when the unseen messages are coming. By the experiments on three benchmark data sets and our generated data set with the simulated group chat about the study in computer science curriculum at Prince of Songkla University, our deep K-means with PCA is the best method to cluster intent from messages based on Purity and MSE measures and MLP neural networks is the best method to classify messages to intent groups based on F1-Score and Accuracy measures. Therefore, the clustering result with deep K-means was used to identify the intents of messages for the first version of chatbot and the trained models with MLP neural networks were used to identified intents of messages for updating the chatbot. Our chatbot was build via Dialogflow and deployed on Line application.

**Keywords :** Prediction, Clustering, Classification

### 1 บทนำ

การตอบข้อความของลูกค้าหรือผู้ที่เข้ามาสอบถามเป็นส่วนหนึ่งของงานที่มีเจ้าหน้าที่คอยให้บริการอยู่เสมอ แต่ในการตอบกลับเหล่านั้นยังมีข้อจำกัดในการบริการ เช่น คำถามอาจไม่ได้คำตอบในทันทีทันใด และอาจทำให้ลูกค้าหรือผู้ใช้เกิดความไม่พึงพอใจ จึงมีตัวช่วยที่เรียกกันว่าแชทบอทเข้ามาเป็นส่วนหนึ่งในการให้บริการทางด้านนี้โดยเฉพาะ ในปัจจุบันมีการใช้แชทบอทหรือโปรแกรมคอมพิวเตอร์ในการตอบกลับการสนทนาแทนบุคคลจริง เพื่อลดปัญหาการถามคำถามนอกเวลาทำการ แต่ไม่ได้คำตอบทันทีซึ่งอาจทำให้ผู้ใช้เกิดความไม่พึงพอใจได้ โดยการสร้างแชทบอทมีความท้าทายอย่างหนึ่ง คือ การระบุเจตนาจากการสนทนา หรือ intent ซึ่งในแต่ละ intent จะมีข้อความได้หลากหลาย และการเรียนรู้ของเครื่อง (Machine Learning) ที่สามารถนำมาใช้ในการจำแนกข้อมูลได้ ทางคณะผู้จัดทำจึงนำมาศึกษาเพื่อใช้ในการจำแนก intent และสร้างแชทบอท โดยจะใช้กรณีศึกษาจากชุดข้อมูลมาตรฐาน และบทสนทนาถามตอบเกี่ยวกับการศึกษาของนักศึกษาสาขาวิทยาการคอมพิวเตอร์

## 2 หลักการที่เกี่ยวข้อง

### 2.1 องค์ประกอบและเครื่องมือที่ใช้ในการสร้างแชทบอท

**แชทบอท (Chatbot)** คือ แอปพลิเคชันซอฟต์แวร์ที่ใช้ในการสนทนาออนไลน์ ซึ่งถูกออกแบบมาเพื่อตอบคำถามอัตโนมัติให้กับผู้ใช้อย่างเป็นธรรมชาติ ซึ่งแชทบอทจะประกอบด้วยส่วนหลักสำคัญ ดังนี้

**Intent** คือ เจตนาหรือความต้องการที่อยู่ในข้อความที่ได้รับจากผู้ใช้งาน หรือข้อความที่ผู้ใช้ต้องการกล่าวถึง

**Response** คือ การตอบกลับข้อความ ซึ่งจะเกิดขึ้นเมื่อมีการระบุ intent จากข้อความที่ผู้ใช้ส่งเข้ามา

ในการสร้างแชทบอทโดยทั่วไปจะมี framework ให้เลือกใช้หลายตัว ซึ่งเครื่องมือที่ทางคณะผู้จัดทำสนใจคือ Dialogflow ซึ่งเป็น framework ที่สามารถสร้างแชทบอทได้หลายภาษาโดยไม่ต้องมีการเขียนโค้ดใดๆ รวมทั้งสามารถเชื่อมต่อไปยัง platform ได้หลากหลาย มีจุดเด่นคือการรองรับ Natural Language Understanding โดยไม่ต้องเขียนโปรแกรมอะไรเพิ่มเติม และสามารถใช้งานผ่าน web browser

### 2.2 คำนวณ TF-IDF

TF-IDF เป็นคุณลักษณะแทนข้อความที่มีจำนวนมิติเท่ากับจำนวนคำศัพท์ทั้งหมด แต่ละมิติจะแทนค่าน้ำหนักของคำนั้น ๆ การหา TF-IDF [1] จะมาจากสูตร

$$tf = 1 + \log(tf^{org})$$

$$idf_t = \log_{10} \frac{N}{df_t}$$

$$TF-IDF_{t,d} = tf \times idf_t$$

โดย  $df_t$  คือ จำนวนข้อความที่มีคำ  $t$  ปรากฏอยู่

$N$  คือ จำนวนข้อความที่มีอยู่

$tf^{(org)}$  คือ ความถี่ของคำที่ปรากฏ

### 2.3 การเรียนรู้แบบไม่มีผู้สอน

ในงานนี้ใช้การเรียนรู้แบบไม่มีผู้สอน 2 วิธี คือ K-Means และ DBSCAN

- 1) การจัดกลุ่มของข้อมูลแบบเคมีน (K-Means clustering) คือ วิธีการหนึ่งที่มีหน้าที่หลักในการแบ่งกลุ่มแบบ Cluster ซึ่งมีหน้าที่ในการจับกลุ่มของข้อมูลที่มีลักษณะใกล้เคียงกันเป็นกลุ่มเดียวกัน โดยการจับกลุ่มจะดูจากระยะห่างของข้อมูลกับ Centroid ว่าข้อมูลอยู่ใกล้ Centroid ไດ ข้อมูลก็จะถูกจัดอยู่ใน Cluster นั้น และทำขั้นตอนที่ 2 - 4 วนซ้ำไป จนกว่าค่า Centroid จะอยู่ในตำแหน่งจุดกึ่งกลางของข้อมูลใน Cluster [2]
- 2) การจัดกลุ่มของข้อมูลแบบ DBSCAN (Density-based spatial clustering of applications with noise) คือ การหาบริเวณข้อมูลที่อยู่รวมกันเป็นกลุ่ม โดยหาได้จากการคำนวณ Data point หรือจุดที่ข้อมูลแสดงอยู่ และ

DBSCAN มักใช้กับชุดข้อมูลที่ไม่สามารถแบ่งแยกกลุ่มก่อนได้อย่างชัดเจน

### 2.4 การเรียนรู้แบบมีผู้สอน

ในงานนี้ มีการใช้การเรียนรู้แบบมีผู้สอน 2 เทคนิค คือ Decision Tree และ Neural Network

- 1) ต้นไม้ตัดสินใจ (Decision Tree) ซึ่งเป็นวิธีการเรียนรู้ของเครื่องที่นิยมใช้รูปแบบหนึ่ง ใช้สำหรับจำแนก (Classification) ข้อมูลและคลาส (class) โดยใช้คุณสมบัติ (attribute) ของข้อมูล [3]
- 2) โครงข่ายประสาท (Neural Networks) คือ แบบจำลองการทำงานของโครงข่ายประสาทในสมองมนุษย์ โดยมีลักษณะเหมือนการส่งผ่านสัญญาณประสาทในสมองของมนุษย์ กล่าวคือ มีความสามารถในการรวบรวมความรู้ (knowledge) ผ่านการเรียนรู้ (learning process) และความรู้เหล่านั้นจะจัดเก็บอยู่ในโครงข่ายในรูปค่าน้ำหนัก (weight) ซึ่งสามารถปรับค่าได้เมื่อมีการเรียนรู้ใหม่ๆ

โหนด (node) เป็นการจำลองการทำงานของเซลล์ส่งสัญญาณ (signal) ระหว่างโหนดที่เชื่อมต่อกัน (connection) ภายในโหนดมีฟังก์ชันกำหนดสัญญาณส่งออกเรียกว่า ฟังก์ชันกระตุ้น (activation function) หรือ ฟังก์ชันการแปลง (transfer function) [4]

## 3 การวิเคราะห์และขั้นตอนวิธี

### 3.1 การจัดกลุ่มข้อความเพื่อกำหนด intent

ในขั้นตอนนี้จะเป็นการจัดกลุ่มของข้อความ โดยใช้การเรียนรู้ของเครื่องแบบไม่มีผู้สอน เมื่อจัดกลุ่มเรียบร้อยแล้วจะให้มนุษย์มาระบุ intent ให้กับแต่ละกลุ่ม ขั้นตอนจะเป็นดังภาพที่ 1 ประกอบด้วย การทำความสะอาดข้อความ [1] การสร้างชุดคำศัพท์ การสกัดคุณลักษณะ การจัดกลุ่ม และ การกำหนด intent label

เมื่อได้เป็นเวกเตอร์ TF-IDF จากนั้นทำการลดจำนวนมิติของเวกเตอร์ลง (Feature reduction) โดยใช้หลักการ Principal Components Analysis (PCA) [5] ข้อความที่ใช้สร้างแชทบอทถูกสกัดคุณลักษณะได้เป็นเวกเตอร์ แล้วนำมาจัดกลุ่ม โดยในโครงงานนี้ทำการทดลองจัดกลุ่มข้อความด้วย 2 วิธี ได้แก่ K-means และ DBSCAN และผลจากการจัดกลุ่มทั้ง 2 วิธี จะถูกนำมาเปรียบเทียบประสิทธิภาพด้วยตัวชี้วัดประสิทธิภาพ เพื่อเลือกวิธีการจัดกลุ่มที่เหมาะสม ก่อนที่จะนำไประบุ intent เป็นกลุ่ม

### 3.2 การจัดกลุ่มโดยใช้ deep K-means

ในโครงงานนี้ ได้เสนอ deep K-means เป็นวิธีการในการแบ่งกลุ่มแบบ K-means ลึกลงไป 3 ชั้น โดยมีขั้นตอน ดังนี้

- 1) *level1*: จัดกลุ่มข้อมูลโดยใช้วิธี K-means แบบ optimal
- 2) สำหรับแต่ละกลุ่มที่  $j$  คำนวณหาระยะทางเฉลี่ยจากข้อมูลในกลุ่มไปยังจุดศูนย์กลางของกลุ่ม แทนด้วย  $d_j$

$$d_j = \frac{\sum_{i=1}^{n_j} |centroid_j - x_i|}{n_j}$$

- 3) หาค่าเฉลี่ยของค่าที่ได้ในข้อที่ 2)

$$D_{avg} = \frac{\sum_{j=1}^N d_j}{N}$$

- 4) *level2*: สำหรับแต่ละกลุ่มที่  $j$  ถ้า  $d_j > D_{avg}$  จะนำข้อมูลที่อยู่ในกลุ่มที่  $j$  ไปจัดกลุ่มอีกครั้ง
- 5) พิจารณาผลการจัดกลุ่มใหม่ที่ได้ในข้อ 4) ตามหลักการในข้อ 2) - 3)
- 6) *level3*: สำหรับแต่ละกลุ่มที่  $j$  ถ้า  $d_j > D_{avg}$  จะนำข้อมูลที่อยู่ในกลุ่มที่  $j$  ไปจัดกลุ่มอีกครั้ง

โดยที่  $n$  คือ จำนวนข้อมูลภายในกลุ่มที่  $j$

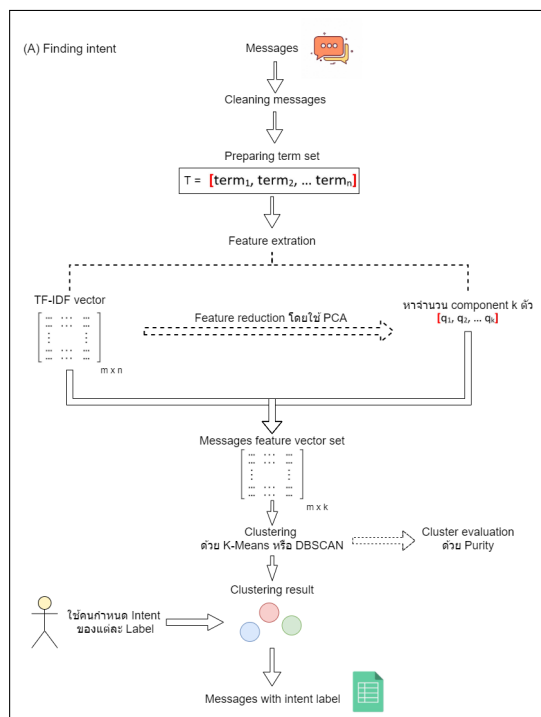
$i$  คือ ตำแหน่งของข้อมูลในกลุ่มที่  $j$

$x_i$  คือ ตำแหน่งข้อมูลตัวที่  $i$  ของกลุ่ม

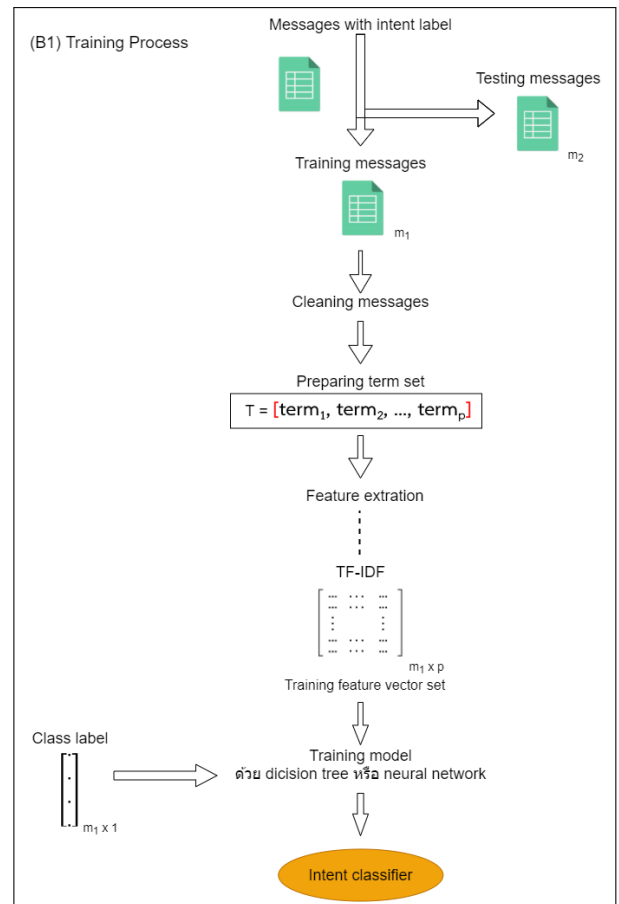
$centroid$  คือ จุดศูนย์กลางของกลุ่ม

$N$  คือ จำนวนกลุ่มทั้งหมดภายในการจัดกลุ่ม

$j$  คือ ตำแหน่งกลุ่มภายในการจัดกลุ่ม



รูปที่ 1: ขั้นตอนการจัดกลุ่มข้อความเพื่อกำหนด intent



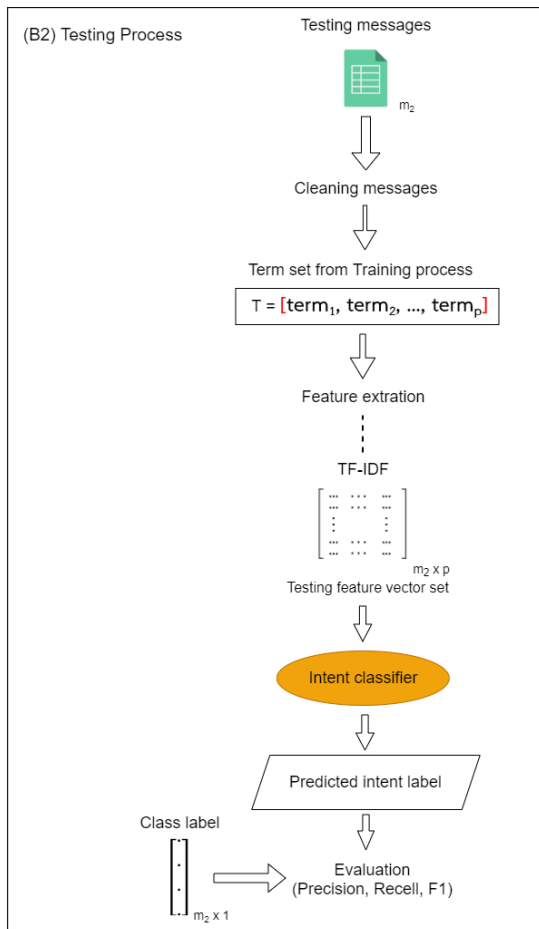
รูปที่ 2: ขั้นตอนการสร้างโมเดล

### 3.3 การสร้างและทดสอบโมเดลในการจำแนก intent

ขั้นตอนนี้เป็นการนำชุดข้อความที่ผ่านการแบ่งกลุ่ม และระบุ intent แล้ว มาสร้างโมเดลทำนาย (Predict) intent ให้กับข้อความโดยใช้การเรียนรู้ของเครื่องแบบมีผู้สอน ซึ่งข้อมูลทั้งหมดจะถูกแบ่งเป็น 2 ส่วน คือ ชุดข้อมูลสอน (Training Data) และชุดข้อมูลทดสอบ (Testing Data) โดยจะนำข้อมูลสอนเข้าสู่กระบวนการเรียนรู้ (Training Process) จนได้เป็นโมเดลขั้นต้น ตอนทำงานในภาพที่ 2 สำหรับกระบวนการทดสอบโมเดลจะนำข้อความจากชุดทดสอบที่ถูกแบ่งไว้ก่อนหน้านี้ มาทำนาย intent โดยใช้โมเดลที่สร้างขึ้น โดยจะมีลำดับขั้นตอนดังภาพที่ 3 โดยบทความฉบับนี้จะทำการทดลองสร้างโมเดลด้วย 2 เทคนิค คือ Decision Tree และ Neural Networks จากนั้นจะนำผลการทำนายไปเปรียบเทียบกับประสิทธิภาพ

### 3.4 การสร้างแชทบอท

ขั้นตอนนี้เป็นการนำชุดข้อความที่ยังไม่มีการระบุ intent มาระบุ intent โดยใช้การจัดกลุ่มจากวิธีการที่เลือก โดยข้อความมีการทำความสะอาด และสกัดคุณลักษณะ ก่อนนำเข้ามาจัดกลุ่ม และจะนำผลลัพธ์ที่ได้เข้าสู่ Dialogflow โดยจะมีการกำหนดคำตอบที่เหมาะสมตาม intent เพื่อสอนแชทบอท โดย



รูปที่ 3: ขั้นตอนการทดสอบโมเดลในการจำแนก intent

จะมีลำดับขั้นตอนดังภาพที่ 4

### 3.5 การอัปเดตแชทบอท

เมื่อมีคำถามที่ไม่อยู่ภายในแชทบอท ข้อความเหล่านี้จะถูกนำเข้าในกระบวนการเพื่ออัปเดตแชทบอทอีกครั้ง ดังขั้นตอนในภาพที่ 5 ซึ่งจะระบุ intent ด้วยการนำเข้าสู่โมเดล และนำผลลัพธ์ที่เรียกว่า เซตของข้อความใหม่ มารวมกับข้อมูลเดิม ก่อนนำเข้า Dialogflow เพื่ออัปเดตแชทบอทให้มีข้อมูลเพิ่มขึ้น

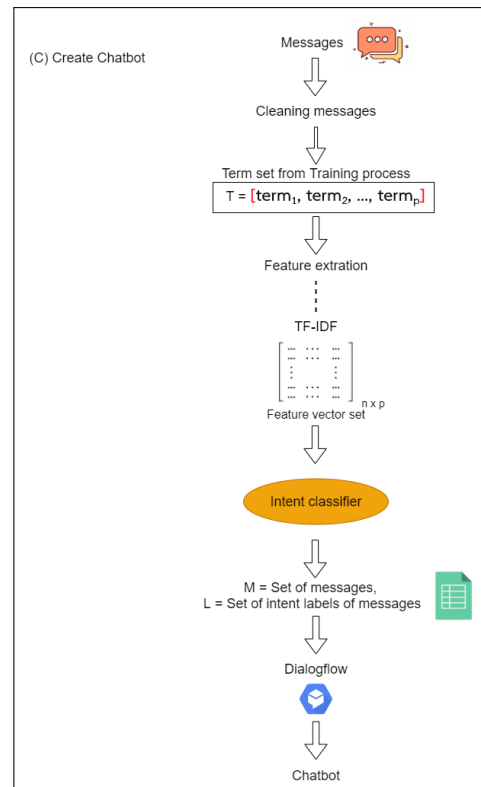
## 4 การทดลองและผลการทดลอง

### 4.1 ชุดข้อมูลที่ใช้

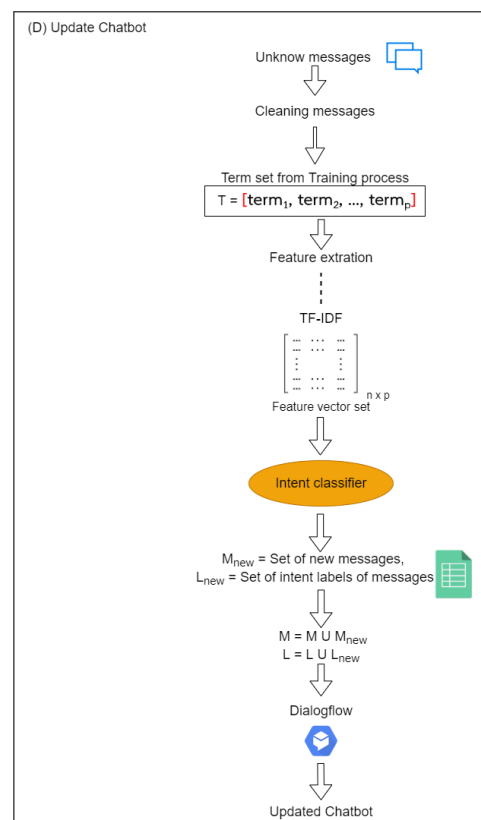
#### ชุดข้อมูลมาตรฐาน

ในโครงงานฉบับนี้ได้ใช้ชุดข้อมูลมาตรฐานสำหรับทดสอบข้อมูลจำนวน 3 กลุ่ม ได้แก่

- 1) ชุดข้อมูล ATIS หรือ Airline Travel Information System เป็นชุดข้อมูลที่ถูกจัดทำขึ้นเพื่อฝึกฝนในการจำแนก intent โดยคุณ Yun-Nung (Vivian) Chen ซึ่งประกอบด้วยข้อมูลจำนวน 4978 ตัว



รูปที่ 4: ขั้นตอนการพัฒนาต้นแบบแชทบอท



รูปที่ 5: ขั้นตอนการอัปเดตแชทบอท

2) ชุดข้อมูล Corona Dataset ชุดข้อมูลที่จัดทำขึ้นเพื่อสร้างแชทบอทเกี่ยวกับคำถาม COVID-19 โดย บริษัท Danish Conversational AI and chatbot company ซึ่งประกอบด้วยข้อมูลจำนวน 1053 ตัว

3) ชุดข้อมูล case routing intent เป็นตัวอย่างชุดข้อมูลจากเว็บไซต์ Salesforce ซึ่งประกอบด้วยข้อมูลจำนวน 150 ตัว

### ชุดข้อมูลที่สร้างขึ้นเอง

ชุดข้อมูลจำลองเพื่อใช้สำหรับสร้างแชทบอท ซึ่งถูกจัดทำขึ้นภายใต้หัวข้อ "แชทบอทสำหรับสถาบันการศึกษา" โดยจะมีความไม่ต่ำกว่า 500 ข้อความ

## 4.2 การจัดกลุ่มข้อความ

ในการทดลองนี้ จะทำการศึกษาประสิทธิภาพการจัดกลุ่มเมื่อการแปลงเวกเตอร์ TF-IDF ด้วย PCA การลดจำนวนมิติของเวกเตอร์ TF-IDF ด้วย PCA เป็นการทำให้ข้อมูลมีขนาดเล็กลง ซึ่งในการทดลองลดขนาดมิติของเวกเตอร์ TF-IDF กับชุดข้อมูลทั้ง 4 ตัว ดังแสดงในตารางที่ 1 จะเห็นว่า เมื่อพิจารณาจากค่า purity ในกรณีชุดข้อมูล case\_routing\_intent เมื่อไม่มีการลดจำนวนมิติด้วย PCA ค่า purity ที่ได้ออกมาจะมีค่ามากที่สุดที่ 0.6667 แต่ในกรณีชุดข้อมูล corona และ atis\_intents เมื่อมีการลดจำนวนมิติด้วย PCA ค่า purity ที่ได้ออกมาจะมีค่ามากที่สุดที่ 0.613 และ 0.8763 ตามลำดับ และในทุกชุดข้อมูลที่มีการลดจำนวนมิติด้วย PCA ค่า MSE ที่ได้จะมีค่าน้อยที่สุดเสมอ จึงสรุปได้ว่าในโครงการนี้การลดจำนวนมิติด้วย PCA จะมีประสิทธิภาพมากที่สุด เมื่อพิจารณาโดยรวม พบว่า มิติของเวกเตอร์ TF-IDF มีขนาดมีขนาดลดลงโดยเฉลี่ย 73.71% และทำการทดสอบการจัด

ตารางที่ 1: ผลการเปรียบเทียบการจัดกลุ่มใช้ PCA และไม่ใช่ PCA บนชุดข้อมูลมาตรฐาน

ข้อมูล	วิธีการ	ใช้ PCA	MSE	purity	เวลาจัดกลุ่ม
case_routing_intent	K-means	no	11.9901	<b>0.6667</b>	0.1180
case_routing_intent	K-means	yes	<b>7.1318</b>	0.6333	<b>0.1090</b>
case_routing_intent	DBSCAN	no	140.1929	0.3133	0.0618
case_routing_intent	DBSCAN	yes	9.7105	0.2933	0.0608
corona	K-means	no	6.5163	0.5745	1.8632
corona	K-means	yes	<b>3.7144</b>	<b>0.6135</b>	<b>0.8775</b>
corona	DBSCAN	no	239.5598	0.0997	0.3211
corona	DBSCAN	yes	7.4579	0.0969	0.3012
atis_intents	K-means	no	4.2154	0.6135	28.8498
atis_intents	K-means	yes	<b>2.3498</b>	<b>0.8763</b>	<b>10.0008</b>
atis_intents	DBSCAN	no	57.8026	0.7704	1.7692
atis_intents	DBSCAN	yes	6.5732	0.7808	1.6541
ข้อมูลที่สร้างขึ้นเอง	K-means	no	1.9024	-	0.6299
ข้อมูลที่สร้างขึ้นเอง	K-means	yes	<b>0.4558</b>	-	0.5657
ข้อมูลที่สร้างขึ้นเอง	DBSCAN	no	3.4051	-	0.3950
ข้อมูลที่สร้างขึ้นเอง	DBSCAN	yes	3.9046	-	0.3603

ตารางที่ 2: ผลการเปรียบเทียบประสิทธิภาพการจัดกลุ่มระหว่าง K-mean, DBSCAN และ deep K-mean

ข้อมูล	วิธีการ	MSE	purity
case_routing_intent	K-means	7.1318	0.6333
case_routing_intent	deep K-means	<b>2.6980</b>	<b>0.86</b>
case_routing_intent	DBSCAN(minPts=3)	9.7105	0.2933
case_routing_intent	DBSCAN(minPts=5)	9.7105	0.2933
corona	K-means	3.7144	0.6135
corona	deep K-means	<b>1.4682</b>	<b>0.8072</b>
corona	DBSCAN(minPts=3)	7.4579	0.0969
corona	DBSCAN(minPts=5)	7.9554	0.0551
atis_intents	K-means	2.3498	0.8763
atis_intents	deep K-means	<b>1.0107</b>	<b>0.9249</b>
atis_intents	DBSCAN(minPts=3)	6.5732	0.7802
atis_intents	DBSCAN(minPts=5)	7.1913	0.7616
ข้อมูลที่สร้างขึ้นเอง	K-means	0.4558	NaN
ข้อมูลที่สร้างขึ้นเอง	deep K-means	<b>0.0380</b>	NaN
ข้อมูลที่สร้างขึ้นเอง	dbscan(minPts = 3)	33.5717	NaN
ข้อมูลที่สร้างขึ้นเอง	dbscan(minPts = 5)	41.6536	NaN

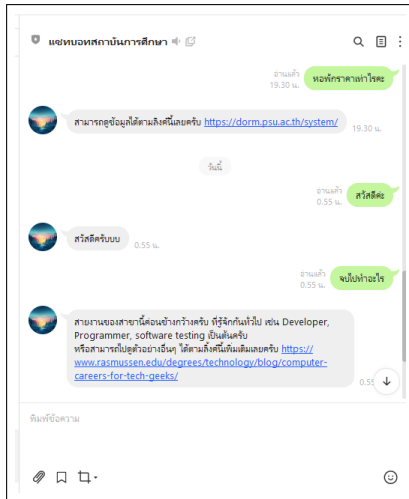
กลุ่มด้วย K-means และ DBSCAN ด้วยชุดข้อมูลทั้ง 4 พบว่า การลดมิติของเวกเตอร์ TF-IDF ด้วย PCA ให้ประสิทธิภาพที่มากกว่าทั้งในชุดข้อมูลมาตรฐานและชุดข้อมูลที่สร้างขึ้นเอง

หลังจากได้ผลการทดลองในตารางที่ 1 พบว่าในทุกๆ ชุดข้อมูลที่ใช้ K-means มีค่า MSE น้อยกว่า DBSCAN หมายความว่าระยะห่างของข้อมูลมีความใกล้ชิดกับจุดศูนย์กลางของกลุ่มมาก ดังนั้นจึงสามารถสรุปได้ว่าประสิทธิภาพของ K-means มีมากกว่า DBSCAN แต่ประสิทธิภาพของ K-means ก็ยังไม่เป็นที่น่าพอใจสำหรับนำไปใช้งานจริงจึงมีการพัฒนา K-means ให้มีประสิทธิภาพมากขึ้น จนได้มาเป็นวิธีการจัดกลุ่มแบบ deep K-means ได้ผลการเปรียบเทียบประสิทธิภาพ ดังตารางที่ 2 จะเห็นได้ว่าค่า MSE ของ deep K-means มีค่าน้อยที่สุด และ ค่า purity มีค่ามากที่สุดและมากกว่า 80% ในทุกชุดข้อมูล ดังนั้นจึงเลือกวิธีการ deep K-means ในการจัดกลุ่มข้อความ

หลังจากทำการจัดกลุ่มข้อความเรียบร้อยแล้ว จำเป็นจะต้องกำหนดชื่อ intent ของแต่ละกลุ่ม เพื่อให้ง่ายต่อความเข้าใจ และสามารถระบุคำตอบได้ตรงตามเป้าหมายของแต่ละกลุ่มข้อมูล

## 4.3 การสร้างแชทบอท เวอร์ชัน 1

เมื่อทำการระบุ intent ครบทุกกลุ่ม จำเป็นต้องแยกข้อมูลเป็นกลุ่มตาม intent เพื่อกำหนด response ให้กับกลุ่มนั้นๆ แต่เนื่องจากการสอน Dialogflow จำเป็นจะต้องมีรูปแบบการเขียนข้อมูลนำเข้าตามที่ถูกกำหนดไว้ จึงมีการเขียนอัลกอริทึมเพื่อจัด

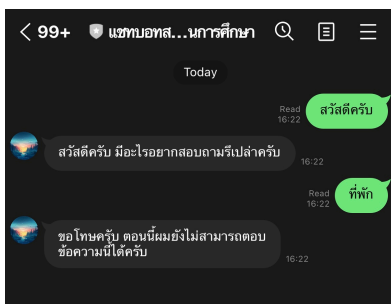


รูปที่ 6: ตัวอย่างผลลัพธ์การทดสอบแชทบอท จากคำที่เคยสอนแชทบอทแล้ว

ข้อมูลให้อยู่ในรูปแบบฟอร์มเฉพาะและนำขึ้น Dialogflow ก่อนเชื่อมต่อไปยัง Line เพื่อใช้งานจริง ได้ผลออกมาดังภาพที่ 6

#### 4.4 การสร้างโมเดลสำหรับจำแนก intent

เนื่องจากแชทบอทเวอร์ชันที่ 1 อาจมีคำถามที่แชทบอทไม่สามารถตอบได้ เช่น ที่พัก ดังรูปที่ 7 หรือมีการนำข้อความใหม่เข้ามาจึงต้องอัปเดตความสามารถให้กับแชทบอท ด้วยโมเดล machine learning เพื่อระบุ intent ให้กับข้อความใหม่ ดังนั้นในหัวข้อนี้จะทำการทดลองเพื่อหาวิธีการสร้างโมเดลที่เหมาะสมที่สุด โดยใช้ทำการทดลองแบบ K-Fold Cross Validation



รูปที่ 7: ตัวอย่างข้อความที่แชทบอทไม่สามารถตอบได้

K-Fold Cross Validation เป็นวิธีการในการประเมินประสิทธิภาพของโมเดลโดยการแบ่งข้อมูลออกเป็น K ส่วนเท่าๆกัน ซึ่งในแต่ละส่วนของข้อมูลที่ถูกรวบรวมจะต้องมาจากการสุ่มเพื่อให้ข้อมูลมีการกระจายตัวอย่างสม่ำเสมอ โดยทั่วไปแล้วจะนิยมใช้การแบ่งข้อมูลเป็น 5 หรือ 10 ส่วน และในกระบวนการนี้ได้แบ่งเป็น 5 ส่วน ( $K=5$ ) เพื่อแบ่งข้อมูลใช้ในการประเมินประสิทธิภาพ โดยคิดเป็นอัตราส่วนฝึกสอนต่อทดสอบ (train:test) อยู่ที่ 80:20 เปอร์เซนต์

ตารางที่ 3: พารามิเตอร์ของ Neural Network

รูปแบบที่	จำนวน hidden layers	จำนวน Node ใน Layer
1	1	จำนวน feature
2	2	จำนวน feature, จำนวน feature
3	2	จำนวน feature, จำนวน feature/2

ตารางที่ 4: พารามิเตอร์ของ Decision tree

รูปแบบที่	ค่าความลึกสูงสุดของ tree
1	จำนวน feature
2	จำนวน feature/2
3	จำนวน feature/4

หลังจากแบ่งข้อมูลเรียบร้อยแล้ว ขั้นตอนต่อไปคือการสร้างโมเดลสำหรับทดสอบ โดยมีรายละเอียดดังนี้

- Neural Network (MLP)  
ในการทดลองนี้จะมีการกำหนดพารามิเตอร์ hidden layers โดยจะกำหนดขนาดและจำนวนของ hidden layers ซึ่งในโครงงานนี้จะกำหนดพารามิเตอร์ไว้ 3 ชุด ดังตารางที่ 3
- Decision tree  
ในการทดลองนี้จะมีการกำหนดพารามิเตอร์ max depth คือค่าความลึกสูงสุดของ tree ที่สามารถกระจายลึกลงไปได้ ซึ่งในโครงงานนี้จะกำหนดพารามิเตอร์ไว้ 3 ชุด ดังตารางที่ 4

โดยวัดประสิทธิภาพของโมเดลด้วย mean F1-score และ mean Accuracy ได้ผลการทดลองดังตารางที่ 5 ซึ่งพบว่า Neural Network จำแนกข้อมูลในชุดข้อมูลมาตรฐาน 2 ชุดแรก และชุดข้อมูลที่สร้างขึ้นเองมีประสิทธิภาพดีกว่า แต่มีเพียงชุดข้อมูล atis\_intent เท่านั้นที่ Decision tree จำแนกผลออกมาได้ดีกว่า และเมื่อสังเกตพารามิเตอร์ของ Neural Network ในชุดข้อมูลมาตรฐาน พารามิเตอร์รูปแบบที่ 2 มีประสิทธิภาพดีกว่า แต่ในชุดข้อมูลที่สร้างขึ้นเองนั้นพารามิเตอร์รูปแบบที่ 1 มีประสิทธิภาพดีกว่า และเมื่อพิจารณาพารามิเตอร์ทั้ง 2 รูปแบบผลลัพธ์ที่ได้ต่างกันไม่มากอย่างมีนัยสำคัญ ซึ่งจากพารามิเตอร์รูปแบบที่ 1 มี layer เพียงชั้นเดียว ทำให้โครงสร้างมีความซับซ้อนน้อยกว่าส่งผลให้ทำงานได้เร็วกว่า ดังนั้น ทางคณะผู้จัดทำจึงเลือกใช้ Neural Network แบบพารามิเตอร์รูปแบบที่ 1 มาใช้สร้างโมเดลเนื่องจากเป็นวิธีการจำแนกข้อความที่เหมาะสมที่สุด

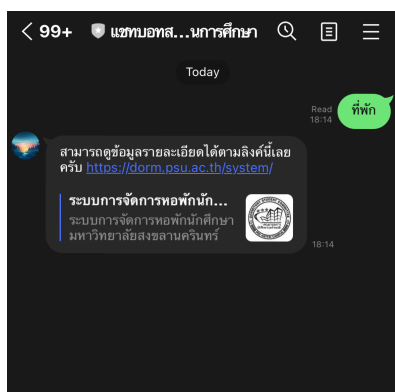
#### 4.5 การสร้างแชทบอท เวอร์ชัน 2

หลังจากสร้างโมเดลได้สำเร็จ จึงนำไปสู่การอัปเดตแชทบอท โดยมุ่งเน้นไปที่ข้อความที่แชทบอทไม่สามารถตอบได้ต่อไปจะเป็นการระบุ intent ให้กับข้อความโดยใช้โมเดลที่สร้างไว้

ตารางที่ 5: ผลการเปรียบเทียบประสิทธิภาพการจำแนก intent ระหว่าง Neural Network และ Decision tree

ชุดข้อมูล	Method	รูปแบบที่	F1-Score	Accuracy
case_routing_intent	Neural Network	1	0.7647	0.7647
case_routing_intent	Neural Network	2	<b>0.7694</b>	<b>0.7694</b>
case_routing_intent	Neural Network	3	0.7656	0.7656
case_routing_intent	Decision tree	1	0.5600	0.5600
case_routing_intent	Decision tree	2	0.5600	0.5600
case_routing_intent	Decision tree	3	0.6000	0.6000
corona	Neural Network	1	0.9630	0.9630
corona	Neural Network	2	<b>0.9646</b>	<b>0.9646</b>
corona	Neural Network	3	0.9624	0.9624
corona	Decision tree	1	0.5481	0.5481
corona	Decision tree	2	0.5605	0.5605
corona	Decision tree	3	0.5529	0.5529
atis_intents	Neural Network	1	0.8200	0.8200
atis_intents	Neural Network	2	0.8200	0.8200
atis_intents	Neural Network	3	0.8067	0.8067
atis_intents	Decision tree	1	<b>0.9405</b>	<b>0.9405</b>
atis_intents	Decision tree	2	0.9381	0.9381
atis_intents	Decision tree	3	0.9385	0.9385
ข้อมูลที่สร้างขึ้นเอง	Neural Network	1	<b>0.9534</b>	<b>0.9534</b>
ข้อมูลที่สร้างขึ้นเอง	Neural Network	2	0.9517	0.9517
ข้อมูลที่สร้างขึ้นเอง	Neural Network	3	0.9517	0.9517
ข้อมูลที่สร้างขึ้นเอง	Decision tree	1	0.6722	0.6722
ข้อมูลที่สร้างขึ้นเอง	Decision tree	2	0.6389	0.6389
ข้อมูลที่สร้างขึ้นเอง	Decision tree	3	0.6278	0.6278

โดยจะแบ่งกลุ่มของข้อความเป็น 2 กลุ่ม ได้แก่ กลุ่มของข้อความที่สามารถจำแนกได้และกลุ่มของข้อความที่ไม่สามารถจำแนกได้ และข้อความที่สามารถจำแนกได้จะถูกประมวลเข้ากับข้อความที่มีการระบุ responses ตาม intent ก่อนหน้า และถูกนำไปใส่แอปเกรตแชทบอท ส่วนข้อความที่ไม่สามารถจำแนกได้สามารถนำไปพิจารณาเพื่อจัดกลุ่มและสร้าง intent ใหม่ตามหัวข้อที่ 4.4 ซึ่งจะเป็นการทำงานแบบนั้นวนไป เพื่อทำการสอนให้ แชทบอทฉลาดขึ้นต่อไป ดังภาพที่ 8



รูปที่ 8: ตัวอย่างแชทบอทหลังการอัปเดต

## 5 บทสรุป

### 5.1 สรุปผลการทดลอง

จากการทดลองและพัฒนาการจำแนกเจตนาการสนทนา เพื่อการสร้างแชทบอท โดยมีขั้นตอนจากการจัดกลุ่ม ซึ่งพิจารณาจากหลักการ 3 วิธี รวมทั้งพิจารณาการลดมิติของข้อมูลโดยใช้ PCA ได้ผลว่าวิธีการแบบ deep K-means ที่มีการลดมิติแบบ PCA มีผลการจัดกลุ่มที่มีประสิทธิภาพที่สุด และในส่วนของการจำแนกข้อความ ซึ่งพิจารณาหลักการ 2 วิธี ได้ว่าวิธีการแบบ Neural Network (MLP) สามารถจำแนกข้อความออกมาได้เหมาะสมที่สุด จึงนำไปสร้างเป็นโมเดล เพื่อจำแนกข้อความที่เข้ามาว่าอยู่ภายใน intent ไต สุดท้ายจึงสามารถนำข้อความที่แชทบอทตอบไม่ได้เข้าสู่โมเดลที่สร้างขึ้นจากวิธีการ Neural Network (MLP) เพื่อจำแนกข้อความว่าอยู่ภายใน intent ไต เพื่อนำไปอัปเดตให้กับ Dialogflow ที่ส่งต่อไปยัง line developer

### เอกสารอ้างอิง

- [1] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. *Introduction to Information Retrieval*. Cambridge University Press, USA, 2008. (Accessed on 01/10/2021).
- [2] Shruti Kapil, Meenu Chawla, and Mohd Dilshad Ansari. On k-means data clustering algorithm with genetic algorithm. In *2016 Fourth International Conference on Parallel, Distributed and Grid Computing (PDGC)*, pages 202--206, 2016. (Accessed on 01/09/2021).
- [3] Prince Yadav. Decision tree in machine learning. <https://towardsdatascience.com/decision-tree-in-machine-learning-e380942a4c96>, November 2018. (Accessed on 01/10/2021).
- [4] ธนาวุฒิ ประกอบผล. โครงข่ายประสาทเทียม artificial neural. *วารสาร มจร.วิชาการ*, 12(24):73--87, 2552. (Accessed on 01/10/2021).
- [5] Pop Phiphat. Principal components analysis (PCA) ต่างจาก factor analysis (FA) ยังไง ? (ตอนที่ 1). <https://medium.com/ingenio/principal-components-analysis-pca-factor-analysis-fa-1-c395e55bdc3>, June 2018. (Accessed on 03/01/2021).