

การปกปิดข้อมูลเสียงพูดเพื่อปกป้องข้อมูลที่ระบุตัวบุคคล
SPEECH MASKING FOR PERSONALLY IDENTIFIABLE
INFORMATION PROTECTION

ณัฐณิชา ชัยศิริพานิช
NATTANICHA CHAISIRIPANICH
ประวิตรานันท์ บุตรโพธิ์
PRAWITRANUN BUTPHO

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิทยาศาสตรบัณฑิต
สาขาวิชาวิทยาการข้อมูลและการวิเคราะห์เชิงธุรกิจ
คณะเทคโนโลยีสารสนเทศ
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ภาคเรียนที่ 1 ปีการศึกษา 2563

การปกปิดข้อมูลเสียงพูดเพื่อปกป้องข้อมูลที่ระบุตัวบุคคล

**SPEECH MASKING FOR PERSONALLY IDENTIFIABLE
INFORMATION PROTECTION**

ณัฐธิดา ชัยศิริพานิช

ประวิตรนันท์ บุตรโพธิ์

อาจารย์ที่ปรึกษา

ดร. นนท์ คณิงสุขเกษม

รศ.ดร. ชีรพงศ์ ลีลานุภาพ

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิทยาศาสตรบัณฑิต

สาขาวิชาวิทยาการข้อมูลและการวิเคราะห์เชิงธุรกิจ

คณะเทคโนโลยีสารสนเทศ

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ภาคเรียนที่ 1 ปีการศึกษา 2563

**SPEECH MASKING FOR PERSONALLY IDENTIFIABLE
INFORMATION PROTECTION**

NATTANICHA CHAISIRIPANICH

PRAWITRANUN BUTPHO

**A PROJECT SUBMITTED IN PARTIAL FULFILLMENT OF
THE REQUIREMENT FOR THE DEGREE OF BACHELOR OF
SCIENCE PROGRAM IN DATA SCIENCE AND BUSINESS ANALYTICS
FACULTY OF INFORMATION TECHNOLOGY
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG**

1/2020

COPYRIGHT 2020

FACULTY OF INFORMATION TECHNOLOGY

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

ใบรับรองปริญญาโท ประจำปีการศึกษา 2563

คณะเทคโนโลยีสารสนเทศ

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง การปกปิดข้อมูลเสียงพูดเพื่อปกป้องข้อมูลที่ระบุตัวบุคคล

**SPEECH MASKING FOR PERSONALLY IDENTIFIABLE
INFORMATION PROTECTION**

ผู้จัดทำ

นางสาวณัฐธิดา

ชัยศิริพานิช รหัสนักศึกษา 60070135

นางสาวประวีตรานันท์

บุตรโพธิ์ รหัสนักศึกษา 60070148

..... อาจารย์ที่ปรึกษา

(ดร. นนท์ คณิงสุขเกษม)

..... อาจารย์ที่ปรึกษา

(รศ.ดร. ชีรพงศ์ ลีลานุภาพ)

ใบรับรองใบโครงการ (PROJECT)

เรื่อง

การปกปิดข้อมูลเสียงพูดเพื่อปกป้องข้อมูลที่ระบุตัวบุคคล

SPEECH MASKING FOR PERSONALLY IDENTIFIABLE INFORMATION PROTECTION

นางสาวณัฐธินา ชัยศิริพานิช รหัสนักศึกษา 60070135

นางสาวประวีตรานันท์ บุตรโพธิ์ รหัสนักศึกษา 60070148

ขอรับรองว่ารายงานฉบับนี้ ข้าพเจ้าไม่ได้คัดลอกมาจากที่ใด
รายงานฉบับนี้ได้รับการตรวจสอบและอนุมัติให้เป็นส่วนหนึ่งของ
การศึกษาวិชาโครงการ หลักสูตรวิทยาศาสตร์บัณฑิต (เทคโนโลยีสารสนเทศ)
ภาคเรียนที่ 1 ปีการศึกษา 2563

.....
(นางสาวณัฐธินา ชัยศิริพานิช)

.....
(นางสาวประวีตรานันท์ บุตรโพธิ์)

| | | | |
|------------------|--------------------------------------------------------|--------------|-----------------------|
| หัวข้อโครงการ | การปกปิดข้อมูลเสียงพูดเพื่อปกป้องข้อมูลที่ระบุตัวบุคคล | | |
| นักศึกษา | นางสาวณัฏฐนิชา | ชัยศิริพานิช | รหัสนักศึกษา 60070135 |
| | นางสาวประวีตรานันท์ | บุตรโพธิ์ | รหัสนักศึกษา 60070148 |
| ปริญญา | วิทยาศาสตร์บัณฑิต | | |
| สาขาวิชา | วิทยาการข้อมูลและการวิเคราะห์เชิงธุรกิจ | | |
| ปีการศึกษา | 2563 | | |
| อาจารย์ที่ปรึกษา | ดร. นนท์ คณิงสุขเกษม | | |
| | รศ.ดร. ชีรพงศ์ ลีลานภาพ | | |

บทคัดย่อ

ในปัจจุบันข้อมูลต่าง ๆ นับว่าเป็นสิ่งที่สำคัญต่อการนำไปประยุกต์กับปัญญาประดิษฐ์ เพื่อวิเคราะห์และพัฒนาประสิทธิภาพขององค์กร และนำข้อมูลเหล่านั้นมาประยุกต์ใช้ แต่มักมีข้อจำกัดในเรื่องของสิทธิส่วนบุคคล กล่าวคือ รายละเอียดข้อมูลส่วนใหญ่ที่องค์กรมีอยู่ หรือรวบรวมข้อมูลเพิ่มเติมนั้น มักจะมีข้อมูลส่วนบุคคลปะปนอยู่ด้วยโดยเฉพาะข้อมูลของลูกค้า หากยังมีข้อมูลเหล่านั้นปะปนอยู่ อาจส่งผลให้มีการลักลอบนำข้อมูลไปหาผลประโยชน์ในทางที่ไม่ถูกต้องได้ ดังนั้น ข้อมูลส่วนบุคคลของลูกค้าถือเป็นเรื่องที่ต้องพึงตระหนักเพื่อรักษาสิทธิส่วนบุคคล และรักษาความน่าเชื่อถือขององค์กร นอกจากนี้ ในการรวบรวมข้อมูลขององค์กรต่าง ๆ จากการบันทึกเสียงการสนทนาระหว่างลูกค้ากับศูนย์ให้บริการข้อมูลลูกค้าทางโทรศัพท์ของธนาคาร ก็ถือว่ามีข้อมูลส่วนบุคคลที่ระบุตัวตนของลูกค้าเป็นจำนวนมาก จึงมีความเสี่ยงสูงหากข้อมูลเหล่านั้นยังมีข้อมูลส่วนบุคคลอยู่แล้วนำมาใช้ในการวิเคราะห์

ทางผู้จัดทำได้เล็งเห็นถึงปัญหานี้และมีการทดลองสร้างระบบโดยใช้ปัญญาประดิษฐ์ ในการดำเนินการปกปิดคำที่เป็นข้อมูลส่วนบุคคลของลูกค้าจากไฟล์เสียงบทสนทนาระหว่างลูกค้ากับศูนย์บริการข้อมูลลูกค้าทางโทรศัพท์ ภายในระบบจะดำเนินการแปลงไฟล์เสียงพูดให้อยู่ในรูปแบบข้อความ จากนั้นตรวจจับคำที่เป็นข้อมูลส่วนบุคคลพร้อมกับเก็บค่าของระยะเวลาที่พูดในไฟล์เสียงนั้น ระบบจะทำการแทรกเสียงรบกวนแทนที่คำพูดที่เป็นข้อมูลส่วนบุคคล และผลลัพธ์ที่ได้คือไฟล์เสียงที่ผ่านการปกปิดข้อมูลส่วนบุคคลแล้ว ซึ่งสามารถนำไปใช้ประโยชน์ในการวิเคราะห์ทางด้านอื่น ๆ ต่อ

| | | | |
|----------------------|----------------------------------------------------------------------|----------------|---------------------|
| Project Title | SPEECH MASKING FOR PERSONALLY IDENTIFIABLE INFORMATION PROTECTION | | |
| Student | Nattanicha | Chaisiripanich | Student ID 60070135 |
| | Prawitranun | Butpho | Student ID 60070148 |
| Degree | Bachelor of Science | | |
| Program | Data Science and Business Analytics | | |
| Academic Year | 2020 | | |
| Advisor | Nont Kanungsukkasem, Ph.D. | | |
| | Asst. Prof. Teerapong Leelanupab, Ph.D. | | |

ABSTRACT

Nowadays, information is crucial for the application of Artificial Intelligence to analyze and develop the efficiency of organizations. There will often be a limitation of privacy right for the application of that information. Most of information which organizations have or collect normally contain personally identifiable information, especially for customers' information. If there is still that information, it may cause data stealing and misusing of the data. Therefore, we must be vigilant over customers' personally identifiable information to protect privacy right and the credibility of an organization. Moreover, information collecting from many organizations with voice recording between customers and bank call center is considered to have a lot of personally identifiable information which identifies customers. Consequently, it is at high risk if we use that information containing personally identifiable information for analysis.

We spotted this problem and applied the Artificial Intelligence to create a system that conceals words which contained personally identifiable information from conversation files between customers and call center by converting speech files into text and detected words which were personally identifiable information along with collecting timestamp from the audio files, then replaced those words with white noise. Finally, the outputs are the audio files which had been masked personally identifiable information speech, so that we could use that information for other field of analysis.

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้สำเร็จลุล่วงได้ด้วยความกรุณาและการสนับสนุนจาก
ดร. นนท์ คณิงสุขเกษม และ รศ.ดร. ชีรพงศ์ ลีตานุกาพ ที่ได้ช่วยชี้แนะในการศึกษา
ค้นคว้า แนะนำขั้นตอนการปฏิบัติงาน เสนอแนวทางในการแก้ปัญหาหรืออุปสรรค
ที่พบเจอในขณะที่ยังผู้จัดทำกำลังพัฒนาโครงงานนี้ และแนะนำวิธีจัดทำปริญญานิพนธ์จนสำเร็จลุล่วงด้วยดี

ขอขอบพระคุณคณาจารย์คณะเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบังทุก ๆ ท่าน ที่ช่วยมอบวิชาความรู้และแนวคิดที่สามารถนำไปประยุกต์ใช้ในการปรับปรุงและพัฒนาโครงงานเพื่อให้โครงงานมีประสิทธิภาพที่ดีขึ้น สามารถนำไปพัฒนาการดำเนินงานในอนาคตได้

ขอขอบคุณอาจารย์ที่ปรึกษา เพื่อน และรุ่นพี่ในคณะเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง และผู้ที่มีส่วนเกี่ยวข้องในการให้คำปรึกษาการพัฒนาโครงงานทุก ๆ ท่าน ที่ได้ให้ความร่วมมือและให้การช่วยเหลือที่ดีตลอดการจัดทำจนสามารถก่อให้เกิดเป็นปริญญานิพนธ์ฉบับนี้ได้

จึงขอแสดงความขอบคุณเป็นอย่างยิ่งไว้ ณ โอกาสนี้

ณัฐธิดา ชัยศิริพานิช

ประวิตรนันท์ บุตรโพธิ์

สารบัญ

หน้า

| | |
|--------------------------------------------------|-----------|
| บทคัดย่อภาษาไทย..... | I |
| บทคัดย่อภาษาอังกฤษ..... | II |
| กิตติกรรมประกาศ..... | III |
| สารบัญ..... | IV |
| สารบัญ (ต่อ)..... | V |
| สารบัญรูปภาพ..... | VI |
| สารบัญรูปภาพ (ต่อ)..... | VII – VII |
| สารบัญตาราง..... | VIII |
| บทที่ 1..... | 1 |
| บทนำ | 1 |
| 1.1 ที่มาและความสำคัญ..... | 1 |
| 1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา | 3 |
| 1.3 ขอบเขตการพัฒนาโครงการ | 3 |
| 1.4 ขั้นตอนการดำเนินงาน | 5 |
| 1.5 ประโยชน์ที่คาดว่าจะได้รับ | 7 |
| บทที่ 2..... | 8 |
| แนวคิด ทฤษฎีและเทคโนโลยีที่เกี่ยวข้อง..... | 8 |
| 2.1 แนวคิดที่เกี่ยวข้อง | 8 |
| 2.2 ทฤษฎีที่เกี่ยวข้อง..... | 9 |
| 2.3 เทคโนโลยีและเครื่องมือที่เกี่ยวข้อง | 19 |

สารบัญ (ต่อ)

หน้า

| | |
|-------------------------------------------------------------------|-----------|
| บทที่ 3..... | 25 |
| ขั้นตอนการพัฒนาระบบ | 25 |
| 3.1 การแปลงเสียงพูดให้อยู่ในรูปแบบข้อความ | 26 |
| 3.2 การตรวจจับคำที่เป็นข้อมูลส่วนบุคคลจากข้อมูลรูปแบบข้อความ..... | 26 |
| 3.3 การแทนที่เสียงที่เป็นข้อมูลส่วนบุคคลด้วยเสียงรบกวน | 35 |
| 3.4 การประเมินผล (Evaluation) | 36 |
| บทที่ 4..... | 37 |
| การทดลองและผลลัพธ์ที่ได้จากระบบ | 37 |
| 4.1 การวิเคราะห์ข้อมูลที่สร้างขึ้น..... | 37 |
| 4.2 การแปลงเสียงพูดให้อยู่ในรูปแบบข้อความ | 43 |
| 4.3 การตรวจจับคำที่เป็นข้อมูลส่วนบุคคลจากข้อมูลรูปแบบข้อความ..... | 46 |
| 4.4 การแทนที่เสียงที่เป็นข้อมูลส่วนบุคคลด้วยเสียงรบกวน | 53 |
| บทที่ 5..... | 54 |
| บทสรุป..... | 54 |
| 5.1 สรุปผลโครงการ | 54 |
| 5.2 ปัญหาในการพัฒนาโครงการและสรุปผล..... | 56 |
| 5.3 แนวทางในการพัฒนาต่อ..... | 56 |
| เอกสารอ้างอิง..... | 57 |

สารบัญรูปภาพ

หน้า

| | |
|---------------------------------------------------------------------------|----|
| รูปที่ 1.1 ขั้นตอนการดำเนินงาน | 5 |
| รูปที่ 2.1 กระบวนการทำงานทั่วไปของการประมวลผลภาษาธรรมชาติ | 10 |
| รูปที่ 2.2 Pre-Trained Part-of-Speech Classification Model..... | 12 |
| รูปที่ 2.3 ผลลัพธ์ของการประมวลผลประโยคทั้งหมด | 12 |
| รูปที่ 2.4 รูปประโยคหลังการทำ Lemmatization..... | 13 |
| รูปที่ 2.5 การระบุ Stop words..... | 13 |
| รูปที่ 2.6 การแยกการวิเคราะห์การพินิจ | 14 |
| รูปที่ 2.7 การคาดเดาประเภทของความสัมพันธ์ | 14 |
| รูปที่ 2.8 รูปประโยคก่อนการทำการจับกลุ่มคำนาม | 15 |
| รูปที่ 2.9 รูปประโยคหลังจากการจับกลุ่มคำนาม | 15 |
| รูปที่ 2.10 คำนามของประโยค..... | 16 |
| รูปที่ 2.11 ประโยคจากการใช้ NER Tagging Model | 16 |
| รูปที่ 2.12 การทำ Coreference Resolution..... | 17 |
| รูปที่ 2.13 กระบวนการทำงานของ Google Cloud Speech API | 19 |
| รูปที่ 3.1 แผนงานการปกป้องข้อมูลที่ระบุตัวบุคคล | 25 |
| รูปที่ 3.2 เปิดใช้งาน Cloud Storage | 26 |
| รูปที่ 3.3 เปิดใช้งาน Cloud Speech To Text | 27 |
| รูปที่ 3.4 อัปโหลดไฟล์เสียงขึ้นบน Cloud Storage..... | 27 |
| รูปที่ 3.5 การนำเข้าข้อมูลเสียงและกำหนดค่าต่าง ๆ | 27 |
| รูปที่ 3.6 ฟังก์ชันการแปลงเสียงพูดให้อยู่ในรูปแบบของข้อความตัวอักษร | 28 |
| รูปที่ 3.7 การบันทึกไฟล์ข้อความเป็นไฟล์ประเภท .json..... | 29 |
| รูปที่ 3.8 ฟังก์ชันการทำนายนิพจน์ระบุนามของ Stanford NER | 30 |
| รูปที่ 3.9 ฟังก์ชันการทำนายนิพจน์ระบุนามของ NLTK..... | 31 |

สารบัญรูปภาพ (ต่อ)

หน้า

| | |
|-----------------------------------------------------------------------------------------------------------|----|
| รูปที่ 3.10 ฟังก์ชันการทำนายนิพจน์ระบุนามของ spaCy | 32 |
| รูปที่ 3.11 ฟังก์ชันการเลือกการทำนายประเภทนิพจน์ระบุนามที่เหมือนกัน 2 ใน 3 | 33 |
| รูปที่ 3.12 การสร้างนิพจน์ระบุนามใหม่โดยใช้ Regular Expressions | 34 |
| รูปที่ 3.13 รวมการทำนาย Regular Expression และค่าทำนายจริงเข้าด้วยกัน | 35 |
| รูปที่ 3.14 ฟังก์ชันการแทนที่เสียงที่เป็นข้อมูลส่วนบุคคลด้วยเสียงรบกวน | 36 |
| รูปที่ 4.1 ตัวอย่างบทสนทนาระหว่างลูกค้ากับศูนย์บริการข้อมูลลูกค้าทางโทรศัพท์ | 37 |
| รูปที่ 4.2 รายละเอียดการวิเคราะห์ประโยค..... | 38 |
| รูปที่ 4.3 รายละเอียดการวิเคราะห์คำ..... | 38 |
| รูปที่ 4.4 รายการของเครื่องหมายวรรคตอนและ Stop words | 39 |
| รูปที่ 4.5 รายละเอียดการวิเคราะห์ความถี่ของคำที่ไม่ซ้ำกัน | 39 |
| รูปที่ 4.6 การแจกแจงความถี่ของคำที่ซ้ำกันของบทสนทนาจริง | 40 |
| รูปที่ 4.7 การแจกแจงความถี่ของคำที่ซ้ำกันของบทสนทนาที่ผ่านการทำความสะอาด | 40 |
| รูปที่ 4.8 รายการชื่อเสียงพูดที่ใช้ในการบันทึกเสียงบทสนทนา..... | 42 |
| รูปที่ 4.9 ตัวอย่างไฟล์เสียงที่บันทึกจากสมาร์ตโฟน | 42 |
| รูปที่ 4.10 ตัวอย่างการแปลงข้อมูลเสียงให้อยู่ในรูปแบบข้อความโดยใช้ Cloud Speech to Text..... | 43 |
| รูปที่ 4.11 ข้อมูลบทสนทนาจริง | 44 |
| รูปที่ 4.12 บทสนทนาที่แบบจำลองทำนาย..... | 44 |
| รูปที่ 4.13 บทสนทนาจริงที่ผ่านการทำความสะอาด | 45 |
| รูปที่ 4.14 บทสนทนาที่แบบจำลองทำนายที่ผ่านการทำความสะอาด | 46 |
| รูปที่ 4.15 ตารางการทำนายประเภทของนิพจน์ระบุนาม | 47 |
| รูปที่ 4.16 ตารางค่าทำนายจริงเฉพาะที่มีการคิดแต่นิพจน์ระบุนาม | 52 |
| รูปที่ 4.17 กราฟเปรียบเทียบค่าความแม่นยำของการทำนายนิพจน์ระบุนามทุกประเภท. ผลิตพลาด! ไม่ได้กำหนดบู๊กมาร์ก | |

| | |
|-------------------------------------------------------------------------------------------|----|
| รูปที่ 4.18 การบันทึกเสียงที่ผ่านการแทนที่เสียงที่เป็นข้อมูลส่วนบุคคลด้วยเสียงรบกวน | 53 |
|-------------------------------------------------------------------------------------------|----|

สารบัญตาราง

หน้า

| | |
|-------------------------------------------------------------------------------------------------------------------|-----------------------------------|
| ตารางที่ 4.1 ตารางแสดงค่าความแม่นยำของการแปลงเสียงพูดให้อยู่ในรูปของข้อความ | 45 |
| ตารางที่ 4.2 ตารางแสดงค่าความแม่นยำของการแปลงเสียงพูดให้อยู่ในรูปของข้อความ (ใหม่) | 46 |
| ตารางที่ 4.3 ตารางการแสดงผลค่าความแม่นยำของการทำนายนิพจน์ระบุนามทุกประเภท | 53 |
| ตารางที่ 4.4 ตารางการแสดงผลค่าความแม่นยำของการติดแท็กคำว่า “PERSON” ..ผิดพลาด! ไม่ได้กำหนด บุ๊กมาร์ก | |
| ตารางที่ 4.5 ตารางการแสดงผลค่าความแม่นยำของการติดแท็กคำว่า “ORGANIZATION”ผิดพลาด! ไม่ได้ กำหนดบุ๊กมาร์ก | |
| ตารางที่ 4.6 ตารางการแสดงผลค่าความแม่นยำของการติดแท็กคำว่า “LOCATION” | ผิดพลาด! ไม่ได้ กำหนดบุ๊กมาร์ก |
| ตารางที่ 4.7 ตารางการแสดงผลค่าความแม่นยำของการติดแท็กคำว่า “DATE” ..ผิดพลาด! ไม่ได้กำหนดบุ๊ก มาร์ก | |
| ตารางที่ 4.8 ตารางการแสดงผลค่าความแม่นยำของการติดแท็กคำว่า “DATE” ..ผิดพลาด! ไม่ได้กำหนดบุ๊ก มาร์ก | |
| ตารางที่ 4.9 ตารางการแสดงผลค่าความแม่นยำของการติดแท็กคำของ PII Number ทุกประเภท..ผิดพลาด! ไม่ได้กำหนดบุ๊กมาร์ก | |

บทที่ 1

บทนำ

1.1 ที่มาและความสำคัญ

ความเป็นส่วนตัว (Privacy) คือ การที่บุคคลมีสิทธิอันชอบธรรมที่จะอยู่อย่างสันโดษ ปราศจากการรบกวน จากบุคคลอื่นที่ไม่ได้รับอนุญาตในการเข้าถึงข้อมูล หรือ การนำข้อมูลไปแสวงหาผลประโยชน์ จึงนำมาซึ่งความเสียหายแก่บุคคลนั้น ความเป็นส่วนตัวสามารถแบ่งออกเป็น 2 ประเภท โดยประเภทแรก คือ ความเป็นส่วนตัวทางกายภาพ (Physical Privacy) ซึ่งหมายถึง สิทธิในสถานที่ เวลา และสินทรัพย์ที่บุคคลพึงมี เพื่อหลีกเลี่ยงจากการถูกละเมิดหรือถูกรบกวนจากบุคคลอื่น ประเภทที่สอง คือ ความเป็นส่วนตัวด้านสารสนเทศ (Information Privacy) ซึ่งหมายถึง ข้อมูลทั่วไปเกี่ยวกับตัวบุคคล เช่น ชื่อ-นามสกุล ที่อยู่ หมายเลขโทรศัพท์ หมายเลขบัตรเครดิต เลขที่บัญชีธนาคาร หรือ หมายเลขบัตรประจำตัวประชาชน ที่บุคคลอื่นห้ามนำมาเปิดเผย หากไม่ได้รับอนุญาต [1]

การพูด (Speech) เป็นหนึ่งในรูปแบบการสื่อสารส่วนบุคคลที่มีความเป็นส่วนบุคคลมากที่สุด เนื่องจากในคำพูดนั้น ๆ มักจะประกอบไปด้วยข้อมูลต่าง ๆ เกี่ยวกับ เพศ ลำเนียง จริยธรรม สภาพอารมณ์ของผู้พูดนอกเหนือจากเนื้อหาของข้อความ [2] ดังนั้น ความเป็นส่วนตัวของคำพูด (The privacy of speech) ก็ถือเป็นสิ่งที่ควรพึงตระหนักเช่นกัน หากมีผู้นำการสนทนาเหล่านั้นไปใช้ในทางที่ไม่ถูกต้องตามกฎหมาย ซึ่งนั่นหมายความว่า มีผู้นำข้อมูลส่วนบุคคลนั้นไปใช้โดยที่ไม่ได้รับความยินยอมจากผู้ให้ข้อมูลนั่นเอง

โดยโครงงานฉบับนี้ จะมุ่งไปยังการสนทนาต่าง ๆ เกี่ยวกับความเป็นส่วนตัวด้านสารสนเทศ (Information Privacy) เนื่องจากในปัจจุบันการละเมิดความเป็นส่วนตัวนั้นเกิดขึ้นเป็นจำนวนมาก และสามารถเกิดขึ้นได้ในหลายรูปแบบ เพราะเทคโนโลยีการสื่อสารมีประสิทธิภาพสูง ข้อมูลส่วนบุคคลต่าง ๆ ของบุคคลกลายเป็นที่ต้องการอย่างมากเพื่อนำไปประกอบธุรกิจส่วนบุคคล โดยไม่คำนึงว่าได้มาโดยวิธีใด ไม่ว่าจะเป็นข้อมูลที่ถูกค้าทำการกรอกลงในเว็บไซด์ ข้อมูลตำแหน่งที่อยู่ ก็ถือเป็นข้อมูลส่วนบุคคลที่ทางองค์กรธุรกิจต่าง ๆ สามารถนำไปซื้อและขายกันได้เช่นกัน

ในบางครั้ง การสนทนาเกี่ยวกับเรื่องความเป็นส่วนตัวในพื้นที่เปิด เช่น การสนทนาพูดคุยกันในคลินิกเล็ก ๆ ช้าง ๆ ห้องรอคิว การประชุมแลกเปลี่ยนความเห็นทางด้านภาษี ต่าง ๆ ในสำนักงาน การประชุมหาแนวทางปฏิบัติในการสอนในโรงเรียน ก็ถือว่ามีความเสี่ยงที่ข้อมูลเหล่านั้นจะรั่วไหลออกไปจากการที่มีบุคคลในห้องช้าง ๆ ได้ยิน ได้รับฟังไปด้วย จึงมีการแก้ปัญหาโดยการสร้างเสียง

รบกวนที่มีความมั่นคงพอที่จะปิดบังเสียงของคำพูดที่มีความเป็นส่วนบุคคลไม่ให้ผู้อื่นสามารถรับรู้ หรือได้ยินข้อมูลเหล่านั้นได้ จากการวัดเสียงพูดต่าง ๆ เพื่อหาจุดที่ดังที่สุดของเสียงนั้น จากนั้นทำการดูความสัมพันธ์ของคลื่นเสียง และทำการหาจุดที่ดีที่สุดในการสร้างเสียงรบกวนที่มั่นคงพอเพื่อทำการปิดบังเนื้อหาของการสนทนาเหล่านั้นเพื่อความปลอดภัยของการรักษาข้อมูลส่วนบุคคล [3]

การปกป้องข้อมูลที่สำคัญในการให้บริการของศูนย์บริการข้อมูลลูกค้าทางโทรศัพท์ (Call Center) ก็ถือเป็นเรื่องที่มีความละเอียดอ่อนมากเช่นกัน เนื่องจากข้อมูลของลูกค้าจำนวนมากมีการเก็บไว้ในรูปแบบของการบันทึกเสียง จึงมีการแก้ไขปัญหาการปกป้องข้อมูลที่สำคัญของลูกค้าในการบันทึกเสียงโดยการสร้างวิธีการควบคุมเพื่อจำลองข้อมูลที่มีความละเอียดอ่อน ซึ่งสร้างขึ้นโดยอัตโนมัติจากการแยกแยะเสียงที่มาจากการทำงานของระบบการรู้จำเสียงพูดอัตโนมัติ (Automatic Speech Recognition: ASR) โดยวิธีการดำเนินงานนี้มักจะใช้กับปัญหาการตรวจจับและค้นหาธุรกรรมบัตรเครดิตในการสนทนาจริงระหว่างตัวแทนศูนย์บริการข้อมูลลูกค้าทางโทรศัพท์ (Call Center) และลูกค้าของศูนย์บริการ [4]

ทางผู้จัดทำได้พิจารณาถึงความสำคัญของการรักษาข้อมูลส่วนบุคคล โดยมีการมุ่งเน้นไปที่ปัญหาของการทำธุรกรรมต่าง ๆ กับทางธนาคาร การทำธุรกรรมกับทางธนาคารนั้น มีความเสี่ยงที่จะถูกรุกฉ้อความเป็นส่วนตัวของบุคคล การลักลอบนำข้อมูลไปแสวงหาผลประโยชน์โดยที่ไม่ได้รับอนุญาตจากเจ้าของข้อมูล และการรุกรฉ้อความเป็นส่วนตัวของข้อมูลจากการเก็บรวบรวมข้อมูลส่วนบุคคลของลูกค้าผ่านการสนทนากับทางศูนย์บริการข้อมูลลูกค้าทางโทรศัพท์ (Call Center) ของธนาคารนั้น ก็ถือเป็นความเสี่ยงที่ต้องพึงตระหนักเช่นกัน เนื่องจากการทำงานขององค์กรทางการเงิน จำเป็นต้องนำข้อมูลต่าง ๆ มาทำการวิเคราะห์เพื่อสนับสนุนการตัดสินใจในการทำกิจกรรมต่าง ๆ เช่น วิเคราะห์ความพึงพอใจของลูกค้า วิเคราะห์ความต้องการของลูกค้า และวิเคราะห์ปัญหาต่าง ๆ ที่เกิดขึ้นในระหว่างการค้าบริการกับทางธนาคาร เพื่อนำไปปรับปรุงและแก้ไข แต่ในกระบวนการวิเคราะห์นั้น มักจะมีข้อมูลส่วนบุคคลของลูกค้ารวมอยู่ในกระบวนการการทำธุรกรรมกับทางธนาคารผ่านการสนทนากับทางศูนย์บริการข้อมูลลูกค้าทางโทรศัพท์ (Call Center) ส่งผลให้โอกาสที่ข้อมูลส่วนบุคคลของลูกค้าจะถูกนำไปใช้แสวงหาผลประโยชน์โดยไม่ได้รับอนุญาตสูงขึ้นอีกด้วย

ดังนั้น ทางผู้จัดทำได้เล็งเห็นถึงความสำคัญของการรักษาข้อมูลส่วนบุคคลของลูกค้าในการทำธุรกรรมกับทางธนาคารผ่านศูนย์บริการข้อมูลลูกค้าทางโทรศัพท์ (Call Center) โดยมีการทดลองสร้างระบบในการปกปิดข้อมูลที่ระบุตัวบุคคล ซึ่งภายในระบบจะดำเนินการแปลงไฟล์เสียงพูดให้อยู่ในรูปแบบข้อความ จากนั้นตรวจจับคำที่เป็นข้อมูลส่วนบุคคลพร้อมกับเก็บค่าของระยะเวลาที่พูดในไฟล์

เสียงนั้น ระบบจะทำการแทรกเสียงรบกวนแทนที่คำพูดที่เป็นข้อมูลส่วนบุคคล และผลลัพธ์ที่ได้คือไฟล์เสียงที่ผ่านการปกปิดข้อมูลส่วนบุคคลแล้ว ซึ่งสามารถนำไปใช้ประโยชน์ในการวิเคราะห์ทางด้านอื่น ๆ ต่อไป

1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา

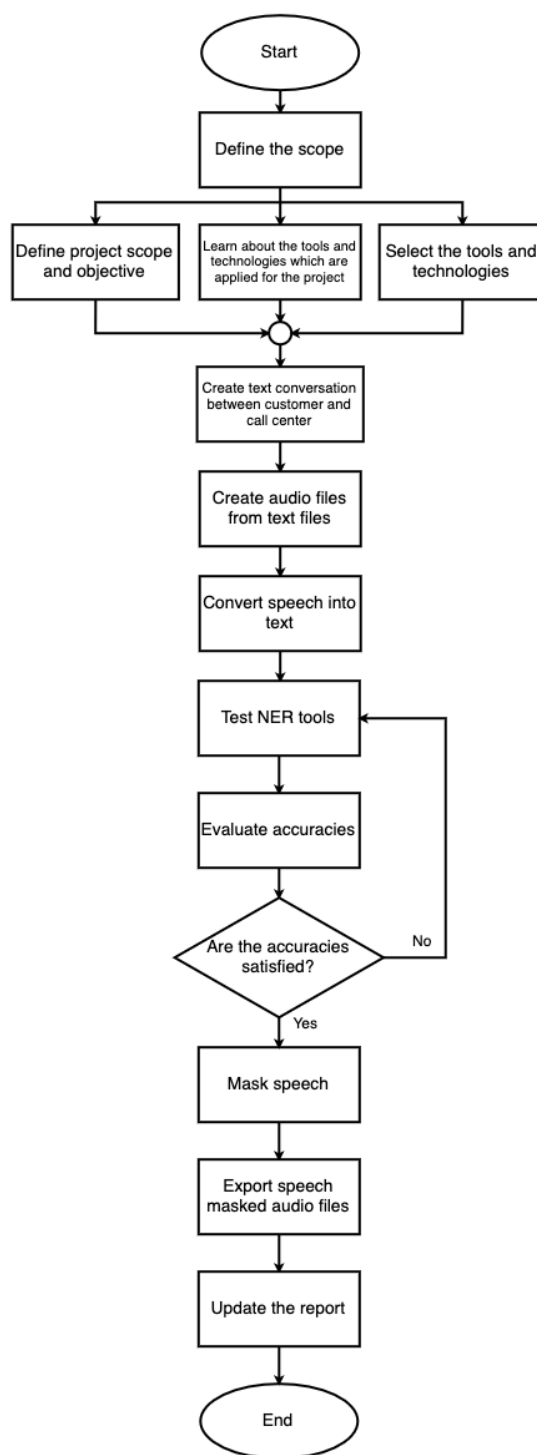
1. เพื่อศึกษากระบวนการประมวลผลภาษาธรรมชาติ (Natural Language Processing)
2. เพื่อศึกษารูปแบบของการรู้จำเสียงพูด (Speech Recognition)
3. เพื่อศึกษารายละเอียดของข้อมูลส่วนบุคคล
4. ดำเนินการปกปิดข้อมูลที่สำคัญในรูปแบบเสียงเพื่อเพิ่มความปลอดภัยในการนำข้อมูลเหล่านั้นไปวิเคราะห์ในด้านอื่น ๆ ต่อไป

1.3 ขอบเขตการพัฒนาโครงการ

1. ขอบเขตการทดลองสร้างชุดข้อมูล
 - 1) ดำเนินการสร้างชุดข้อมูลบทสนทนาระหว่างลูกค้ากับศูนย์บริการลูกค้าทางโทรศัพท์ในรูปแบบข้อความเป็นภาษาอังกฤษ
 - 2) สร้างไฟล์เสียงจากการนำชุดข้อมูลที่ได้สร้างขึ้นมานับพันทีกลเสียงโดยใช้ใช้เครื่องมือสำเร็จรูปในการแปลงข้อความให้อยู่ในรูปของเสียง (Text-to-Speech)
2. ขอบเขตของการพัฒนาระบบ
 - 1) ผู้จัดทำเลือกใช้ The Cloud Speech To Text ในการแปลงเสียงพูดให้อยู่ในรูปแบบข้อความ (Speech-to-Text)
 - 2) ทดลองนำ Stanford Named Entity Recognizer, Natural Language Toolkit (NLTK) และ spaCy มาประยุกต์ใช้ในการวิเคราะห์และประมวลผลข้อความเพื่อทำนายคำที่เป็นข้อมูลส่วนบุคคล
 - 3) สร้างเงื่อนไขในการตรวจจับข้อมูลส่วนบุคคลที่เป็นตัวเลขในบทสนทนาเพิ่มเติมโดยการใช้ Regular Expressions
 - 4) ดำเนินการแทนที่เสียงที่เป็นข้อมูลส่วนบุคคลด้วยเสียงรบกวน โดยใช้ไลบรารีของ Pydub
 - 5) ประเมินผลการแปลงเสียงพูดให้อยู่ในรูปของข้อความ โดยใช้แนวคิดของ Jaccard's Coefficient Similarity

- 6) ประเมินผลความแม่นยำของการตรวจจับข้อมูลส่วนบุคคลโดยวัดจากค่า Recall เท่านั้น จากการสร้างผลเฉลยของข้อมูลเหล่านั้น และเปรียบเทียบความแม่นยำกับข้อมูลที่ทำนาย

1.4 ขั้นตอนการดำเนินงาน



รูปที่ 1.1 ขั้นตอนการดำเนินงาน

รูป 1.1 ผู้จัดทำออกแบบขั้นตอนการดำเนินงานโดยมีกระบวนการ ดังนี้

1.4.1 กำหนดขอบเขต

- กำหนดขอบเขตของโครงการและวัตถุประสงค์ของโครงการ
- ศึกษาเครื่องมือและเทคโนโลยีที่สามารถนำมาใช้กับโครงการได้
- เลือกเครื่องมือและเทคโนโลยีที่เหมาะสมกับโครงการ

1.4.2 การสร้างชุดข้อมูล

- สร้างข้อมูลบทสนทนาในรูปแบบข้อความขึ้นเอง ซึ่งเนื้อหาข้อมูลส่วนบุคคลของบทสนทนาจะประกอบด้วย ชื่อ - นามสกุล เลขที่บัญชี เลขบัตรเดบิต หรือ เครดิต เลขบัตรประชาชน วันเกิด ที่อยู่ และเบอร์โทรศัพท์
- นำข้อมูลบทสนทนาในรูปแบบข้อความมาบันทึกเสียงและบันทึกเป็นไฟล์เสียง

1.4.3 ดำเนินการพัฒนาแบบจำลองและเครื่องมือในการพัฒนาระบบ

- นำข้อมูลเสียงที่สร้างขึ้นมาใช้ในการพัฒนาแบบจำลอง โดยการแปลงเสียงพูดให้อยู่ในรูปแบบข้อความ
- นำผลลัพธ์บทสนทนาที่ได้จากกระบวนการก่อนหน้านี้มาวิเคราะห์และทดลองใช้โดยวิธีต่าง ๆ ที่ใช้ในการระบุพจน์ระบุนาม (Named Entities)

1.4.4 ประเมินผลความแม่นยำของเครื่องมือที่ใช้

- ประเมินผลความแม่นยำของการแปลงข้อมูลเสียงให้อยู่ในรูปแบบข้อความโดยเปรียบเทียบจากข้อมูลจริงในรูปแบบข้อความที่มีการสร้างขึ้นมาก่อนหน้านี้
- ประเมินผลความแม่นยำของเครื่องมือที่ใช้ในการตรวจจับข้อมูลที่ระบุตัวบุคคลโดยวัดจากค่า Recall เท่านั้น โดยเทียบกับข้อมูลที่มีการเฉลยผลการทำนายไว้

1.4.5 ดำเนินการแทนที่คำพูดที่เป็นข้อมูลส่วนบุคคลด้วยเสียงรบกวนจากไฟล์บันทึกเสียง

- หลังจากตรวจจับข้อมูลส่วนบุคคลในรูปแบบข้อความ จึงจับคู่เวลาของคำนั้นในไฟล์เสียง และดำเนินการแทนที่เสียงด้วยเสียงรบกวน
- บันทึกไฟล์เสียงที่ผ่านการแทนที่เสียงด้วยเสียงรบกวน เพื่อให้สามารถนำผลลัพธ์ที่ได้จากระบบไปดำเนินการวิเคราะห์ทางธุรกิจ หรือด้านอื่น ๆ ต่อไป

1.5 ประโยชน์ที่คาดว่าจะได้รับ

- 1) มีกระบวนการนำข้อมูลเสียงเข้าระบบและทำการปิดบังข้อมูลส่วนบุคคลเพื่อรักษาสิทธิส่วนบุคคลของลูกค้า และสร้างความน่าเชื่อถือให้กับองค์กร
- 2) มีความเข้าใจในการจัดการคุณภาพของเสียงมากขึ้น เพื่อนำไปพัฒนาในงานวิจัยด้านอื่น ๆ ที่เกี่ยวข้องต่อ
- 3) มีการแปลงข้อมูลเสียงให้อยู่ในรูปแบบของข้อความเพื่อความสะดวกต่อการนำไปวิเคราะห์ข้อมูลรูปแบบข้อความในอนาคต
- 4) มีความเข้าใจกระบวนการทำงานของการประมวลผลภาษาธรรมชาติมากขึ้น และสามารถนำไปประยุกต์ใช้กับการพัฒนาอื่น ๆ ที่เกี่ยวข้องกับเทคโนโลยีนี้ต่อไป
- 5) มีระบบที่สามารถปกปิดข้อความเสียงในส่วนที่เป็นข้อมูลส่วนบุคคล

บทที่ 2

แนวคิด ทฤษฎี เทคโนโลยีและเครื่องมือที่เกี่ยวข้อง

ผู้จัดทำได้แบ่งรายละเอียดของแนวคิด ทฤษฎี เทคโนโลยีและเครื่องมือที่เกี่ยวข้องที่จะนำมาใช้กับการพัฒนาระบบเป็น 3 ส่วนหลัก ๆ ดังนี้

2.1 แนวคิดที่เกี่ยวข้อง

2.1.1 สิทธิความเป็นส่วนตัวส่วนบุคคล

สิทธิความเป็นส่วนตัวส่วนบุคคล (Privacy Right) มีการบัญญัติรับรองสิทธิดังกล่าวมาแล้วในรัฐธรรมนูญ ถึง 3 ฉบับ ฉบับแรกคือ รัฐธรรมนูญแห่งราชอาณาจักรไทย พ.ศ. 2540 มาตรา 34 บัญญัติว่า “สิทธิของบุคคลในครอบครัว เกียรติยศ ชื่อเสียง ตลอดจนความเป็นอยู่ส่วนบุคคล ย่อมได้รับความคุ้มครอง” ฉบับที่สองคือ รัฐธรรมนูญแห่งราชอาณาจักรไทย พ.ศ. 2550 มาตรา 35 บัญญัติว่า “สิทธิของบุคคลในครอบครัว เกียรติยศ ชื่อเสียง ตลอดจนความเป็นอยู่ส่วนบุคคล ย่อมได้รับความคุ้มครอง การกล่าวหรือไขข่าวแพร่หลายซึ่งข้อความหรือภาพไม่ว่าด้วยวิธีใดไปยังสาธารณชนอันเป็นการละเมิดหรือกระทบถึงสิทธิของบุคคลในครอบครัว เกียรติยศ ชื่อเสียง หรือความเป็นอยู่ส่วนบุคคล จะกระทำได้เว้นแต่กรณีที่เป็น ประโยชน์ต่อสาธารณะ บุคคลย่อมมีสิทธิได้รับความคุ้มครองจากการแสวงประโยชน์โดยมิชอบจากข้อมูลส่วนบุคคลที่เกี่ยวกับตน ทั้งนี้ ตามที่กฎหมายบัญญัติ” และรัฐธรรมนูญฉบับปัจจุบัน คือรัฐธรรมนูญแห่งราชอาณาจักรไทย พ.ศ. 2560 มาตรา 32 ก็รับรองสิทธิดังกล่าวเช่นเดียวกัน [5]

ผู้จัดทำได้เล็งเห็นถึงความสำคัญของข้อกำหนดบังคับใช้และเคารพในสิทธิของผู้อื่น จึงได้จัดทำหัวข้อนี้ เพื่อรักษาสิทธิความเป็นส่วนตัวส่วนบุคคล เนื่องจากทุกครั้งที่เราทำธุรกรรมผ่านศูนย์บริการข้อมูลลูกค้าทางโทรศัพท์ ทางองค์กรจะทำการบันทึกการสนทนา ระหว่างเจ้าหน้าที่ศูนย์บริการข้อมูลลูกค้าทางโทรศัพท์ กับลูกค้า เพื่อนำข้อมูลที่ทางลูกค้าแจ้งไปวิเคราะห์ เพื่อแก้ไขปัญหา หรือ ประเมินศักยภาพขององค์กร

2.2 ทฤษฎีที่เกี่ยวข้อง

2.2.1 การรู้จำเสียงพูด (Speech Recognition)

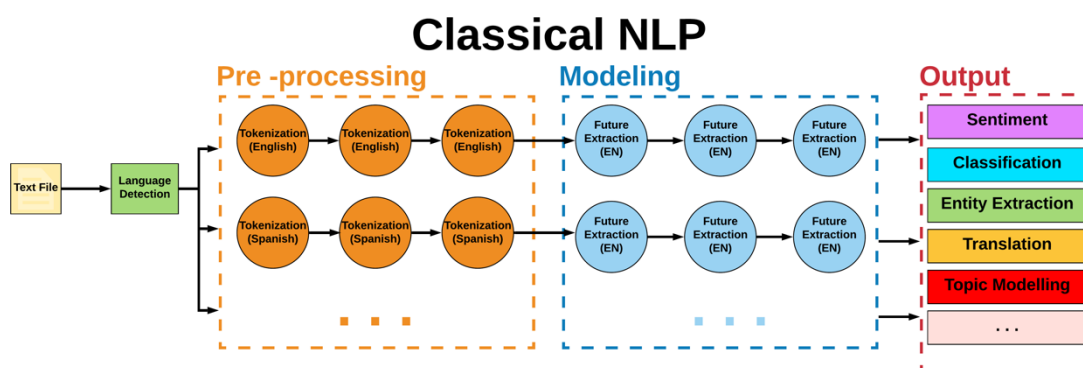
Speech Recognition หรือที่เรียกว่า Automatic Speech Recognition (ASR) หรือ Speech-to-text เป็นสิ่งที่ช่วยให้โปรแกรมสามารถประมวลผลคำพูดของมนุษย์ให้อยู่ในรูปแบบลายลักษณ์อักษร แม้ว่าโดยทั่วไปมักจะถูกสับสนกับการจดจำเสียง (Voice Recognition) แต่การรู้จำเสียงพูด (Speech Recognition) จะเน้นที่การแปลงเสียงพูดจากรูปแบบคำพูดเป็นข้อความ ในขณะที่การจดจำเสียง (Voice Recognition) เป็นเพียงแค่การพยายามระบุเสียงของผู้ใช้แต่ละคน ซึ่งอัลกอริทึมการรู้จำเสียงพูด (Speech recognition algorithms) มีวิธีการที่นิยมใช้หลัก ๆ ดังนี้ [6]

- Natural Language Processing (NLP): NLP นั้นอาจจะไม่ใช่อัลกอริทึมเฉพาะที่ใช้ในการรู้จำเสียงพูด แต่ก็ถือเป็นหนึ่งในปัญญาประดิษฐ์ (Artificial Intelligence) มุ่งเน้นไปที่การโต้ตอบระหว่างมนุษย์และเครื่องจักรผ่านภาษาพูดและข้อความ เช่น Siri (Siri)
- Hidden Markov Models (HMM): HMM ช่วยให้สามารถรวมเหตุการณ์ที่ซ่อนอยู่ เช่น การติดแท็กส่วนของคำพูด (Part-of-speech tags) ลงในแบบจำลองที่มีความเป็นไปได้ และสามารถประยุกต์ใช้เป็นแบบจำลองที่มีลำดับชั้นในการทำการรู้จำเสียงพูด (Speech Recognition) กำหนดประเภทให้แต่ละหน่วย เช่น วลี พยางค์ และประโยค ตามลำดับโดยที่ประเภทเหล่านี้จะสร้างการจับคู่ด้วยข้อมูลที่จัดเตรียมไว้ ทำให้สามารถกำหนดลำดับของประเภทต่าง ๆ ได้อย่างเหมาะสมที่สุด
- N-grams: เป็นรูปแบบของแบบจำลองทางภาษา (Language model: LM) ที่ง่ายที่สุด ซึ่งมีการกำหนดความน่าจะเป็นให้กับประโยคหรือวลีต่าง ๆ โดยที่ N-gram คือลำดับชั้นของ N-words ตัวอย่างเช่น “Order the pizza” คือ 3-gram และ “Please order the pizza” คือ 4-gram ซึ่งไวยากรณ์และความน่าจะเป็นของลำดับชั้นคำ ๆ นั้นจะถูกนำไปใช้เพื่อเพิ่มประสิทธิภาพของการจดจำ (Recognition) และความแม่นยำ (Accuracy)

- Neural networks: มีการใช้ประโยชน์จากอัลกอริทึมการเรียนรู้เชิงลึก (Deep Learning) เป็นหลัก โดยที่โครงข่ายประสาทเทียม (Neural networks) มีการประมวลข้อมูลที่มีการฝึกฝน (Training data) โดยเลียนแบบการเชื่อมต่อระหว่างกันของสมองมนุษย์ผ่านชั้นของ Node โดยที่แต่ละ Node ถูกสร้างมาจาก ข้อมูลนำเข้า (Inputs), น้ำหนัก (Weights), ความโน้มเอียงหรือเกณฑ์ (A bias or threshold), และผลลัพธ์ (Output) หากค่าผลลัพธ์นั้นเกินเกณฑ์ที่กำหนด Neural networks จะทำการกระตุ้น Node ให้ส่งข้อมูลไปยังชั้นถัดไปในเครือข่าย (Network) เนื่องจากวิธีนี้เป็นการเรียนรู้แบบ Supervised learning ซึ่งมีความแม่นยำกว่าและสามารถรับข้อมูลได้มากขึ้น แต่ก็ส่งผลให้ประสิทธิภาพการทำงานช้าลงเมื่อเทียบกับแบบจำลองทางภาษารูปแบบเดิม
- Speaker Diarization (SD): อัลกอริทึมนี้จะทำการระบุและแบ่งเสียงพูดตามเอกลักษณ์ของผู้พูด วิธีนี้ช่วยให้โปรแกรมสามารถแยกแยะบุคคลในการสนทนาได้ดีขึ้นและมักใช้กับศูนย์บริการข้อมูลลูกค้าทางโทรศัพท์ (Call Center) เพื่อทำการแยกแยะลูกค้าและตัวแทนขาย

ผู้จัดทำนำทฤษฎีนี้มาใช้ในการนำเทคโนโลยีเหล่านั้นมาแปลงเสียงพูดให้อยู่ในรูปแบบข้อความซึ่งอยู่ในขั้นตอนแรกของการพัฒนาระบบ เพื่อนำผลลัพธ์ที่ได้ไปดำเนินการพัฒนาระบบในขั้นตอนถัดไป

2.2.2 การประมวลผลภาษาธรรมชาติ (Natural Language Processing: NLP)



รูปที่ 2.1 กระบวนการทำงานทั่วไปของการประมวลผลภาษาธรรมชาติ

การประมวลผลภาษาธรรมชาติ คือ หนึ่งในสาขาของวิทยาศาสตร์คอมพิวเตอร์ที่เกี่ยวข้องกับปัญญาประดิษฐ์ (Artificial Intelligence) และภาษาศาสตร์คอมพิวเตอร์ (Computational Linguistics)

เป็นศาสตร์ที่ศึกษาเกี่ยวกับการทำให้คอมพิวเตอร์สามารถสื่อสารโต้ตอบด้วยภาษาของมนุษย์ และทำให้คอมพิวเตอร์เข้าใจภาษามนุษย์มากขึ้น เช่น Siri, Google Assistant และ Alexa [7]

การประมวลผลภาษาธรรมชาติ เริ่มแรกเมื่อปลายปี ค.ศ. 1940 จากการใช้เครื่องมือการแปลเพื่อทำการถอดรหัสศัตรูในช่วงสงครามโลกครั้งที่ 2 เป็นครั้งแรก แต่อย่างไรก็ตาม งานวิจัยที่เกี่ยวกับการประมวลผลภาษาธรรมชาติก็ไม่ได้มีการสร้างขึ้นมาจนถึงปี ค.ศ. 1980 โดยการประมวลผลภาษาธรรมชาตินั้น มีสาขาวิชาหลากหลายด้านที่มีการนำเทคโนโลยีไปประยุกต์ใช้ เช่น การค้นคืนสารสนเทศ (Information Retrieval) การสกัดสารสนเทศ (Information Extraction) และการตั้งคำถาม – ตอบคำถาม (Question - Answering) [8]

กระบวนการทำงานของการประมวลผลภาษาธรรมชาติ (NLP Pipelines) มีขั้นตอนดังนี้

1) การแบ่งส่วนประโยค (Sentence Segmentation)

ขั้นตอนแรกคือการแบ่งข้อความให้อยู่ในรูปของประโยคแต่ละประโยค เช่น

“London is the capital and most populous city of England and the United Kingdom.”

“Standing on the River Thames in the south east of the island of Great Britain, London has been a major settlement for two millennia.”

2) Word Tokenization

ขั้นตอนต่อไปหลังจากทำการแบ่งประโยคแล้ว ก็จะเป็นการแบ่งคำในประโยคนั้น ๆ ออกจากกัน หรือเรียกอีกชื่อหนึ่งว่า Tokenization เช่น

“London is the capital and most populous city of England and the United Kingdom.”

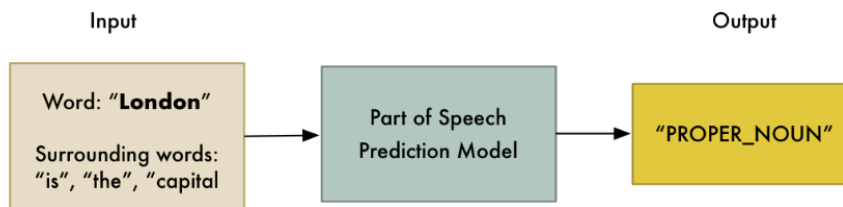
เมื่อทำการแยกคำแล้วจะได้ผลลัพธ์ ดังนี้

“London”, “is”, “ the”, “capital”, “and”, “most”, “populous”, “city”, “of”, “England”, “and”, “the”, “United”, “Kingdom”, “.”

การทำ Tokenization ในภาษาอังกฤษนั้นสามารถทำได้ง่ายเนื่องจากจะมีการแยกคำทุกครั้งที่มีช่องว่างระหว่างคำเหล่านั้น โดยจะถือว่าเครื่องหมายวรรคตอนเป็นโทเคนแยก เนื่องจากเครื่องหมายวรรคตอนก็มีความหมายเช่นกัน

3) การทำนายส่วนต่าง ๆ ของคำพูดสำหรับในแต่ละโทเคน

ขั้นตอนต่อไปคือการสำรวจแต่ละโทเค็นและพยายามคาดเดาส่วนของคำพูด ไม่ว่าจะเป็นคำนาม คำกริยา คำคุณศัพท์ และอื่น ๆ ซึ่งการรู้บริบทของแต่ละคำจะสามารถทำให้เข้าใจได้ว่าประโยคนั้นกล่าวถึงอะไร สามารถทำได้โดยการป้อนคำแต่ละคำเข้าไปในแบบจำลองการจำแนกส่วนหนึ่งของคำพูดที่ยังไม่ผ่านการฝึกฝน (Pre-Trained Part-of-Speech Classification Model)



รูปที่ 2.2 Pre-Trained Part-of-Speech Classification Model

Pre-Trained Part-of-Speech Classification Model ได้รับการฝึกฝนมาจากการเติมประโยคภาษาอังกฤษเป็นล้าน ๆ ประโยคด้วยการใช้ส่วนหนึ่งของคำพูดแต่ละคำที่ติดแท็กแล้ว และเรียนรู้ที่จะจำลองพฤติกรรมนั้นแต่แบบจำลองก็ยังมีข้อจำกัดเนื่องจากการอิงตามสถิติอย่างสมบูรณ์ ไม่สามารถเข้าใจความหมายจริง ๆ เพียงแค่ทราบวิธีการคาดเดาส่วนหนึ่งของคำพูดตามประโยคและคำที่คล้ายกันที่เคยเห็นมาก่อน หลังจากประมวลผลประโยคทั้งหมดจะได้ผลลัพธ์ ดังนี้

| | | | | | | |
|---------------|-----------|------------|----------------|-------------|-------------|---------------------|
| London | is | the | capital | and | most | populous ... |
| Proper Noun | Verb | Determiner | Noun | Conjunction | Adverb | Adjective |

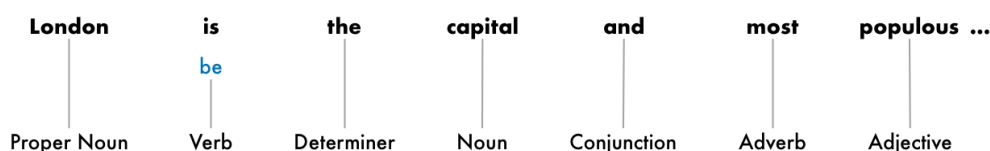
รูปที่ 2.3 ผลลัพธ์ของการประมวลผลประโยคทั้งหมด

จากรูปที่ 2.4 แบบจำลองสามารถเริ่มรวบรวมความหมายพื้นฐานบางประการได้แล้ว ยกตัวอย่างเช่น คำนามในประโยคนี้ประกอบไปด้วยคำว่า “London” และ “Capital” ดังนั้นจึงสรุปได้ว่าประโยคนั้นอาจกล่าวถึงเรื่องเกี่ยวกับ London

4) Text Lemmatization

ในภาษาอังกฤษ และภาษาส่วนใหญ่คำจะปรากฏในรูปแบบที่แตกต่างกัน เช่น “I had a pony.”, “I had two ponies.” จะสังเกตได้ว่าประโยคทั้งคู่กล่าวถึงคำนามที่เป็น Pony แต่มีการใช้รูปคำที่ไม่เหมือนกัน เมื่อมีการทำงานกับข้อความในคอมพิวเตอร์ การรู้รูปแบบพื้นฐานของ

คำแต่ละคำในประโยคนั้นมีประโยชน์อย่างมาก เพราะจะช่วยให้ทราบได้ว่าทั้งสองประโยคนั้นกำลังกล่าวถึงสิ่งที่เป็นแนว ๆ เดียวกัน มิฉะนั้นคำว่า “Pony” และ “Ponies” จะมีความหมายแตกต่างกันโดยสิ้นเชิงต่อคอมพิวเตอร์ สรุปได้ว่าในกระบวนการนี้จะเป็นการหารูปแบบที่เป็นพื้นฐานมากที่สุดในประโยค หลังจากทำการ Lemmatization เพิ่มในรูปแบบรากของคำกริยาจะมีลักษณะดังนี้

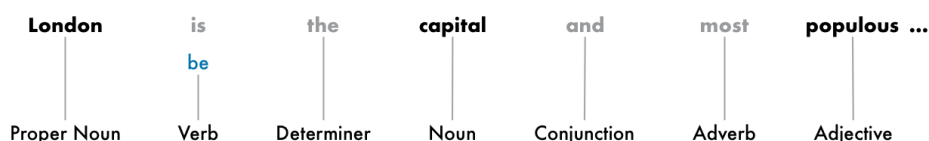


รูปที่ 2.4 รูปประโยคหลังการทำ Lemmatization

จากรูปที่ 2.5 จะสังเกตได้ว่าการเปลี่ยนแปลงเพียงที่เดียวคือ “is” เปลี่ยนเป็น “be”

5) การระบุ Stop words

ขั้นตอนต่อไปเป็นการพิจารณาความสำคัญของแต่ละคำในประโยค เนื่องจากในภาษาอังกฤษมีคำเพิ่มเติมค่อนข้างมากเช่น “and”, “the” และ “a” เมื่อทำสถิติกับข้อความ คำเหล่านี้จะมีการรบกวนต่อแบบจำลองมากหากมีการปรากฏมากกว่าคำอื่น ๆ ดังนั้นในการประมวลผลภาษาธรรมชาติจึงจัดให้คำกลุ่มนี้เป็น Stop words นั่นคือคำที่จำเป็นต้องทำการตัดออกก่อนนำไปทำการวิเคราะห์ทางสถิติ

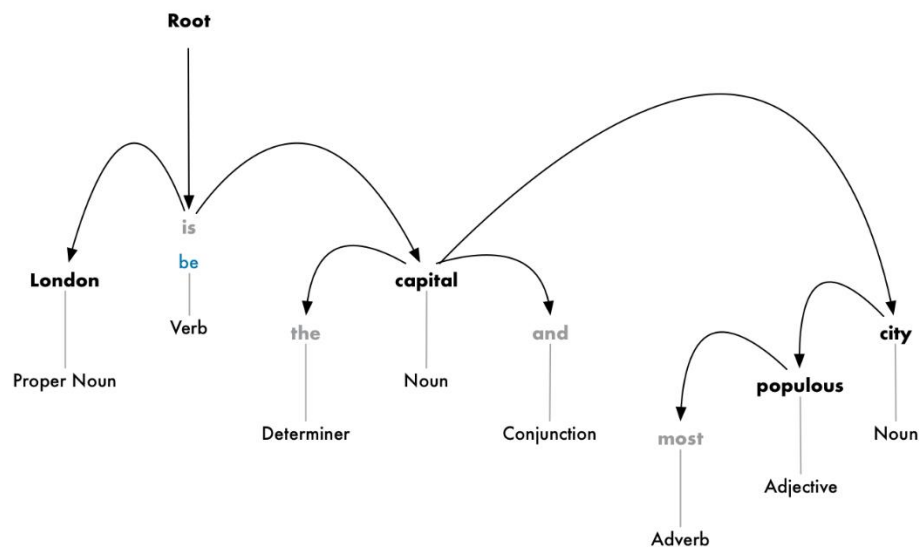


รูปที่ 2.5 การระบุ Stop words

การทำการกำหนด Stop words นั้น ไม่มีมาตรฐานที่ตายตัวในการประยุกต์ใช้ การตัดคำบางคำออกไปนั้นขึ้นอยู่กับจุดประสงค์ของการประยุกต์ใช้ด้วย เช่น การทำเครื่องมือค้นหาวงดนตรีร็อก (Rock Band Search Engine) ผู้ทำจะต้องไม่ทำการตัดคำว่า “The” ออก เนื่องจากบางวงดนตรีอาจมีการใช้ชื่อวงที่มีคำว่า “The” นำหน้า

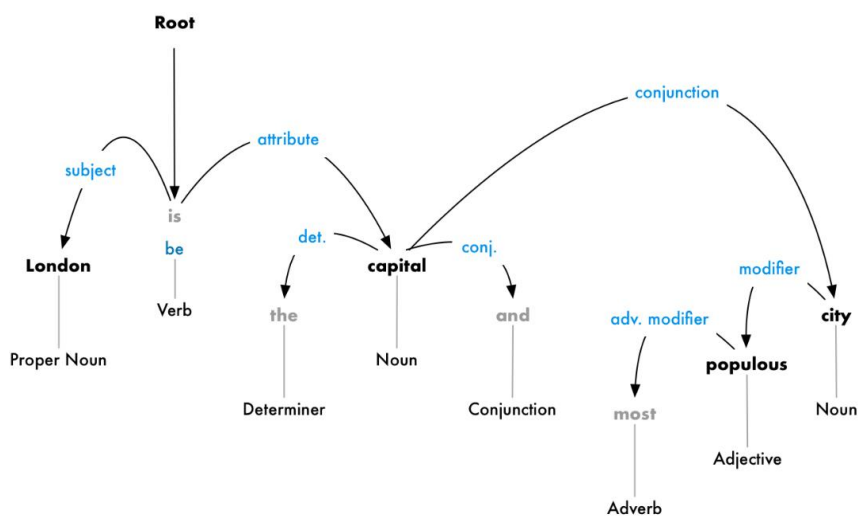
6) การแยกการวิเคราะห์การพึ่งพา (Dependency Parsing)

ขั้นตอนนี้เป็นกระบวนการค้นหาความเกี่ยวข้องกันของคำทั้งหมดในประโยค โดยมีจุดประสงค์คือการสร้างต้นไม้ที่มีพ่อแม่ (Parent) เป็นคำเดียวให้กับแต่ละคำในประโยค โดยราก (Root) ของต้นไม้จะเป็นกริยาหลัก (Main Verb) ของประโยค เมื่อทำการแยกการวิเคราะห์ (Parsing) ผลลัพธ์จะเป็นดังรูปที่ 2.7



รูปที่ 2.6 การแยกการวิเคราะห์การพึ่งพา

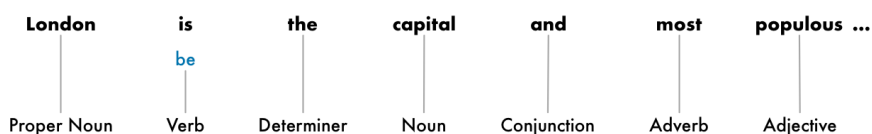
นอกจากนี้ ยังสามารถคาดเดาประเภทของความสัมพันธ์ที่มีอยู่ระหว่างสองคำนี้ได้ ดังรูปที่ 2.8



รูปที่ 2.7 การคาดเดาประเภทของความสัมพันธ์

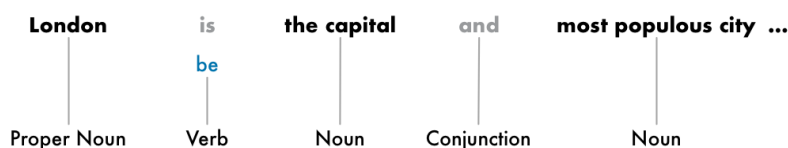
ต้นไม้แสดงให้เห็นว่าหัวข้อของประโยคนั้นเป็นคำนามว่า “London” และมีความสัมพันธ์แบบ “be” กับ “Capital” ทำให้ทราบได้ว่า “ลอนดอนเป็นเมืองหลวง” ขั้นตอนที่มีการใช้ในบางครั้ง คือ การค้นหาคำนาม (Finding Noun Phrases)

นอกจากการทำ Dependency Parsing อย่างเดียวแล้ว ยังสามารถใช้ข้อมูลจาก Dependency Parse Tree ในการจับกลุ่มคำที่กำลังกล่าวถึงสิ่งเดียวกันได้โดยอัตโนมัติ ตัวอย่างเช่น แทนที่จะทำการแบ่งตามรูปที่ 2.8



รูปที่ 2.8 รูปประโยคก่อนการทำการจับกลุ่มคำนาม

สามารถจับกลุ่มคำนามเพื่อจำแนกตามรูปที่ 2.9 ดังนี้

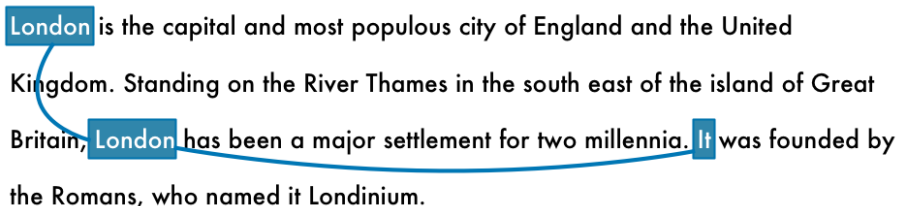


รูปที่ 2.9 รูปประโยคหลังจากการจับกลุ่มคำนาม

8) Coreference Resolution

ในกระบวนการนี้จะทำให้ทราบถึงส่วนต่าง ๆ ของคำสำหรับแต่ละคำว่าคำเหล่านี้มีความเกี่ยวข้องกันอย่างไรและคำใดมีการกล่าวถึงนิพจน์ระบุนาม (Named-Entity) แต่อย่างไรก็ตามภาษาอังกฤษก็ยังประกอบไปด้วยคำสรรพนามค่อนข้างมาก เช่นคำว่า He, She และ It โดยคำเหล่านี้มนุษย์สามารถเข้าใจบริบทของคำว่าใช้แทนสิ่งใด แต่แบบจำลองของการระบุคำที่เป็นนิพจน์ระบุนามนั้นไม่สามารถทราบได้ว่าคำสรรพนามเหล่านั้นหมายถึงสิ่งใดเนื่องจากมีการตรวจสอบเพียงหนึ่งประโยคในแต่ละครั้ง เมื่อมนุษย์อ่านประโยคที่เคยกล่าวถึงไปข้างต้นมนุษย์จะสามารถเข้าใจได้ว่าคำว่า “it” นั้นหมายถึง “London” ดังนั้น จุดประสงค์ของการทำ Coreference Resolution คือการจับคู่คำ ๆ เดียวกันโดยการติดตามจากคำสรรพนาม,ข้ามประโยค ดังรูปที่ 2.13 [9]

London is the capital and most populous city of England and the United Kingdom. Standing on the River Thames in the south east of the island of Great Britain, London has been a major settlement for two millennia. It was founded by the Romans, who named it Londinium.



รูปที่ 2.12 การทำ Coreference Resolution

2.2.3 การรู้จำนิพจน์ระบุนาม (Named Entity Recognition)

นิพจน์ระบุนาม (Named Entity) คือ นิพจน์ที่ทำหน้าที่ระบุชี้เฉพาะถึงสิ่งต่าง ๆ เช่น ชื่อบุคคล ชื่อองค์กร ชื่อสถานที่ รวมไปถึงนิพจน์แสดงวันเวลา ค่าเงิน และเปอร์เซ็นต์

การรู้จำนิพจน์ระบุนาม (Named Entity Recognition) เป็นขั้นตอนที่สำคัญสำหรับการพัฒนาระบบประมวลผลเอกสาร โดยเฉพาะอย่างยิ่งสำหรับระบบที่เกี่ยวข้องกับการเข้าถึงข้อมูล เช่น ระบบสกัดสารสนเทศ (Information Extraction) หรือในระบบค้นคืนเอกสาร (Information Retrieval) [10]

ผู้จัดทำได้นำทฤษฎีนี้มาใช้ในการตรวจจับคำที่เป็นข้อมูลส่วนบุคคลโดยการตรวจจับ ชื่อ – นามสกุล องค์กร สถานที่ วัน และค่าเงิน และนำเทคโนโลยีที่เกี่ยวข้องกับทฤษฎีนี้มาใช้ในการพัฒนาระบบดังที่จะกล่าวถึงในหัวข้อเทคโนโลยีที่เกี่ยวข้อง

2.2.4 Jaccard's Coefficient Similarity

เป็นสถิติประยุกต์แนวคิดในทฤษฎีเซตเพื่อนำมาใช้เปรียบเทียบความคล้ายคลึงและความหลากหลายของกลุ่มตัวอย่าง เมื่อแรกเริ่มค่าสัมประสิทธิ์ Jaccard's Coefficient Similarity ถูกเสนอขึ้นเพื่อเปรียบเทียบความหลากหลายในเชิงพฤกษศาสตร์ ต่อมาจึงแพร่หลายไปสู่วงการอื่น ๆ โดยเฉพาะอย่างยิ่ง ในงานค้นคืนสารสนเทศ (Information Retrieval)

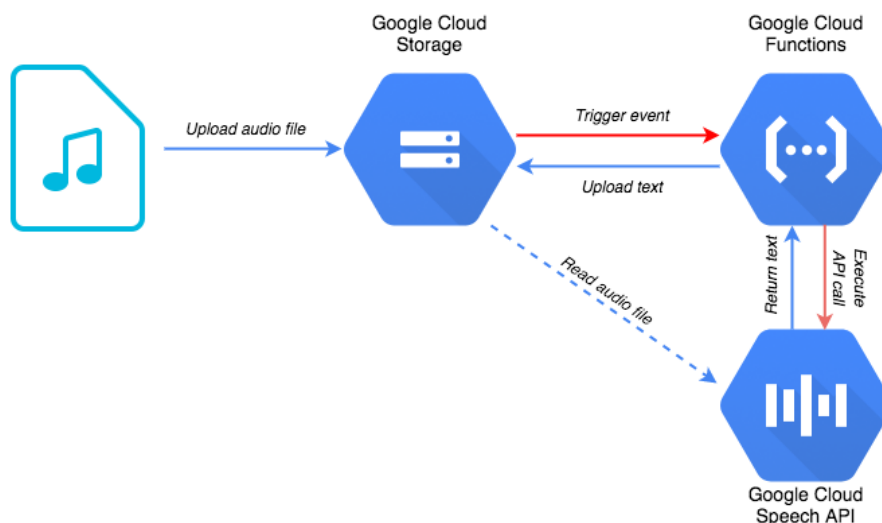
แนวคิดของค่าสัมประสิทธิ์ Jaccard's Coefficient Similarity คือ การวัดค่าความคล้ายคลึงระหว่างกลุ่มประชากร 2 กลุ่ม โดยคำนวณจากขนาดของประชากรที่ทั้งสองกลุ่มมีตัวอย่างร่วมกัน (อินเตอร์เซกชันในทฤษฎีเซต) หารด้วยขนาดของประชากรทั้งหมดจากทั้งสองกลุ่มตัวอย่าง (ยูเนียนในทฤษฎีเซต) [11] ดังสมการที่ 2.1

$$Jaccard(X, Y) = \frac{|X \cap Y|}{|X \cup Y|} \quad (2.1)$$

ซึ่งทางผู้จัดทำได้นำ Jaccard's Coefficient Similarity มาใช้ในการประเมินผลความแม่นยำของการแปลงข้อมูลเสียงให้อยู่ในรูปแบบข้อความ

2.3 เทคโนโลยีและเครื่องมือที่เกี่ยวข้อง

2.3.1 Cloud Speech to Text by Google Cloud



รูปที่ 2.13 กระบวนการทำงานของ Google Cloud Speech API

กูเกิลคลาวด์แพลตฟอร์มเป็นเว็บเซิร์ฟเวอร์ที่ให้บริการคลาวด์แพลตฟอร์มที่ถูกพัฒนาขึ้นโดยกูเกิล ซึ่งภายในกูเกิลคลาวด์แพลตฟอร์มนั้นมีบริการที่แยกย่อยอีกมากมายให้ตรงตามลักษณะการใช้งาน เช่น Cloud Speech to Text, Cloud Storage, Compute Engine, Machine Learning และอื่น ๆ อีกมากมาย ทั้งนี้การใช้งานกูเกิลคลาวด์แพลตฟอร์มจะคิดค่าใช้จ่ายตามจำนวนการใช้งาน [12]

ทางผู้จัดทำเลือกบริการ Cloud Storage ในการเก็บไฟล์เสียงที่ทางผู้จัดทำสร้างบทสนทนา ระหว่างลูกค้ากับศูนย์ให้บริการข้อมูลลูกค้าทางโทรศัพท์ และใช้ Cloud Speech to Text ในการแปลงเสียงพูดให้อยู่ในรูปแบบข้อความ ซึ่งเทคโนโลยีนี้มีไลบรารีที่ชื่อว่า Speech ภายในไลบรารีนี้มีแบบจำลองในการแปลงเสียงพูดให้อยู่ในรูปแบบข้อความให้เลือกใช้ตามความเหมาะสมของงาน และสามารถกำหนดค่าต่าง ๆ ได้ตามความต้องการเพื่อให้เหมาะสมกับงานที่ทำ

2.3.2 Stanford Named Entity Recognizer (Stanford NER)

เป็นเครื่องมือการประยุกต์ใช้จากภาษาจาวา (Java) สำหรับการรู้จำนิพจน์ระบุนาม (Named Entity Recognizer: NER) ซึ่งเป็นการจัดประเภทของคำในข้อความ เช่น ชื่อสิ่งของ ชื่อบุคคล และบริษัท เป็นการกำหนด โครงสร้างการสัณฐานสมบัติที่เหมาะสมสำหรับการรู้จำนิพจน์ระบุนาม (Named Entity Recognition: NER) [13] [14] ซึ่ง Stanford NER แบ่งแบบจำลองออกเป็น 3 ประเภท ดังนี้

- 1) แบบจำลองสำหรับติดแท็กนิพจน์ระบุนาม 3 ประเภท ได้แก่ PERSON, ORGANIZATION และ LOCATION
- 2) แบบจำลองสำหรับติดแท็กนิพจน์ระบุนาม 4 ประเภท ได้แก่ PERSON, ORGANIZATION, LOCATION และ MISCELLANEOUS ENTITIES
- 3) แบบจำลองสำหรับแบ่งนิพจน์ระบุนาม 7 ประเภท ได้แก่ PERSON, ORGANIZATION, LOCATION, DATE, TIME, MONEY และ PERCENT

ผู้จัดทำตัดสินใจเลือกแบบจำลองสำหรับติดแท็กนิพจน์ระบุนาม 7 ประเภท และดำเนินการเลือกการติดแท็กในบทสนทนาทั้งหมดเป็นจำนวน 5 ประเภท ได้แก่ PERSON, ORGANIZATION, LOCATION, DATE และ MONEY

2.3.3 Natural Language Toolkit (NLTK)

เป็นแพลตฟอร์มที่นิยมในโปรแกรมภาษาไพทอน (Python) เพื่อทำงานกับข้อมูลภาษาของมนุษย์ พร้อมกับชุดของไลบรารีที่ช่วยในการประมวลข้อความ แบ่งประเภทของคำ (Classification) การแบ่งโทเค็นของคำ (Tokenization) การตัดคำ (Stemming) การติดแท็กคำ (Tagging) และการแยกวิเคราะห์คำ (Parsing) [15]

NLTK สามารถติดแท็กนิพจน์ระบุนาม (Named Entities) ได้ทั้งหมด 9 ประเภท ดังนี้

- ORGANIZATION เช่น Georgia-Pacific Corp., WHO
- PERSON เช่น Eddy Bonte, President Obama
- LOCATION เช่น Murray River, Mount Everest
- GPE เช่น South East Asia, Midlothian
- DATE เช่น June, 2008-06-29
- TIME เช่น two fifty a m, 1:30 p.m.

- MONEY เช่น 175 million Canadian Dollars, GBP 10.40
- PERCENT เช่น twenty pct, 18.75 %
- FACILITY เช่น Washington Monument, Stonehenge

จากประเภทที่กล่าวมาด้านบนนั้น ทางผู้จัดทำได้ดำเนินการเลือกการติดแท็กในบทสนทนาเป็นจำนวนทั้งหมด 6 ประเภท ได้แก่ ORGANIZATION, PERSON, LOCATION, GPE, DATE และ MONEY

2.3.4 spaCy

เป็นไลบรารีสำหรับการทำการประมวลผลภาษาธรรมชาติขั้นสูงในภาษาไพทอน (Python) โดยที่ spaCy ถูกออกแบบมาสำหรับการประยุกต์ใช้งานจริง และช่วยสร้างแอปพลิเคชันที่สามารถประมวลผลและทำความเข้าใจข้อความจำนวนมาก สามารถใช้ในการดำเนินการสกัดข้อมูล (Information Extraction) หรือระบบการทำความเข้าใจภาษาธรรมชาติเพื่อดำเนินการประมวลผลข้อความล่วงหน้าสำหรับการเรียนรู้เชิงลึก (Deep Learning) ซึ่งคุณสมบัติของ spaCy มีดังต่อไปนี้ [16]

- Tokenization: การแบ่งข้อความให้อยู่ในรูปของคำโดด ๆ หรือ เครื่องหมายวรรคตอน
- Part-of-speech (POS) Tagging: การกำหนดประเภทคำให้กับโทเค็นนั้น ๆ เช่น กริยา หรือ คำนาม
- Dependency Parsing: การกำหนดประเภทของการพึ่งพาในการสร้างประโยค และอธิบายความสัมพันธ์ระหว่างโทเค็นแต่ละตัว เช่น ประธาน หรือ กรรม
- Lemmatization: การกำหนดรูปฐานเดิมของคำนั้น ๆ ตัวอย่างเช่น lemma ของคำว่า “was” คือ “be” และ lemma ของคำว่า “rats” คือ “rat”
- Sentence Boundary Detection (SBD): การค้นหาและแบ่งส่วนประโยคของแต่ละประโยค
- Named Entity Recognition (NER): การกำหนดประเภทให้กับวัตถุ (Object) ที่อยู่ในโลกความจริง เช่น บุคคล องค์กร หรือสถานที่
- Entity Linking (EL): การลบความคลุมเครือของข้อความเอนทิตี เพื่อให้มีตัวบ่งชี้เฉพาะหนึ่งเดียวของคำนั้น ๆ ในฐานความรู้
- Similarity: การเปรียบเทียบคำ ช่วงของข้อความ และเอกสารต่าง ๆ ว่ามีความคล้ายคลึงกันอย่างไร

- Text Classification: กำหนดหมวดหมู่หรือประเภทในเอกสารทั้งหมด หรือส่วนใดส่วนหนึ่งในเอกสาร
- Rule-based Matching: การค้นหาลำดับของโทเค็นในข้อความเดิม และคำอธิบายทางภาษา (Linguistic Annotations) ซึ่งคล้ายกับ Regular Expressions
- Training: การแก้ไข และเพิ่มประสิทธิภาพการทำนายแบบจำลองทางสถิติ (Statistical Model's Predictions)
- Serialization: ดำเนินการบันทึกลงไฟล์ต่าง ๆ

spaCy สามารถติดแท็กนิพจน์ระบุนาม (Named Entities) ได้ทั้งหมด 18 ประเภท ดังนี้

- PERSON คือ บุคคล รวมถึงตัวละครต่าง ๆ
- NORP คือ สัญชาติ หรือศาสนา หรือพรรคการเมือง
- FAC คือ อาคาร สนามบิน ทางด่วน และสะพาน
- ORG คือ บริษัท หน่วยงาน และสถาบัน
- GPE คือ ประเทศ เมือง และรัฐ
- LOC คือ สถานที่ที่ไม่ใช่ GPE เทือกเขา และแหล่งน้ำ
- PRODUCT คือ วัตถุต่าง ๆ ยานพาหนะ อาหาร และสิ่งที่ไม่ใช่การบริการ
- EVENT คือ ชื่อพายุเฮอริเคน การแข่งขัน สงคราม และการแข่งขันกีฬา
- WORK_OF_ART คือ ชื่อหนังสือ และเพลง
- LAW คือ เอกสารต่าง ๆ ที่มีการจดลิขสิทธิ์
- LANGUAGE คือ ภาษาต่าง ๆ
- DATE คือ วันที่แน่นอน หรือช่วงเวลาที่ไม่เฉพาะเจาะจง
- TIME คือ เวลาที่เฉพาะเจาะจงกว่า DATE
- PERCENT คือ เปอร์เซ็นต์ และตัวเลขที่มีเครื่องหมาย “%”
- MONEY คือ ค่าของเงิน รวมถึงหน่วยของเงิน
- QUANTITY คือ मात्रาวัดต่าง ๆ เช่น น้ำหนัก หรือระยะทาง
- ORDINAL คือ เลขลำดับ เช่น “first”, “second” และ “third” เป็นต้น
- CARDINAL คือ ตัวเลขที่ไม่ได้อยู่ในประเภทอื่น ๆ

จากประเภทที่กล่าวมาด้านบนนั้น ทางผู้จัดทำได้ดำเนินการเลือกการคิดแท็กในบทความเป็นจำนวนทั้งหมด 6 ประเภท ได้แก่ PERSON, ORG, GPE, LOC, DATE และ MONEY

2.3.5 Regular Expressions

เป็นสัญลักษณ์ที่ใช้ระบุชุดของอักขระตัวอักษร เมื่อชุดของอักขระตัวอักษรที่เฉพาะเจาะจงนั้นอยู่ในชุดอักขระตัวอักษรที่มีการกำหนดให้เป็น Regular Expressions โดยทั่วไปแล้วจะใช้สัญลักษณ์ "*", "+", "?", "()" และ "|" ในการกำหนดเงื่อนไขของชุดตัวอักษร [Regexp_matching_can_be_simple] [17] [18]

ตัวอย่างประเภทของ Basic Regular Expression Meta-Characters มีดังนี้

- "." คือ สัญลักษณ์ตัวแทน หมายความว่าจับคู่อักขระตัวอักษรใดก็ได้
- "^abc" คือ จับคู่รูปแบบที่มีอักขระตัวอักษร "abc" ขึ้นต้นประโยค
- "abc\$" คือ จับคู่รูปแบบที่มีอักขระตัวอักษร "abc" อยู่ท้ายประโยค
- "[abc]" คือ จับคู่ชุดอักขระตัวอักษรที่อยู่ 1 ใน 3 ของชุดอักขระตัวอักษรนั้น
- "[A-Z0-9]" คือ จับคู่ 1 ในช่วงของชุดอักขระตัวอักษรนั้น ๆ
- "ed|ing|s" คือ จับคู่หนึ่งในตัวอักษรที่กำหนดเฉพาะเจาะจง จากตัวอย่าง คือจับคู่คำที่ลงท้ายด้วย "ed" หรือ "ing" หรือ "s"
- "*" คือ อักขระอักขระที่จะไม่มี หรือซ้ำกันมากกว่า 2 ตัวอักษรขึ้นไป เช่น "a*" คือ ไม่มีตัวอักษร "a" หรือมีตัวอักษร "a" ซ้ำกันมากกว่า 2 ตัวขึ้นไป ("aa", "aaaa")
- "+" คือ มีอักขระอักขระนั้นตั้งแต่ 1 ตัวขึ้นไป เช่น "a+" คือ มีตัวอักษร "a" เป็นจำนวน 1 ตัวอักษร หรือมากกว่า 1 ตัวอักษร ("a", "aaaa")
- "?" คือ ไม่มีตัวอักษรนั้น ๆ หรือมีเพียงแค่ 1 ตัวอักษร เช่น "e-?mail" คือ ถ้าเป็นคำว่า "email" หรือ "e-mail" ก็สามารถเข้าเงื่อนไขนั้นได้เช่นกัน
- "{n}" คือ กำหนดจำนวนตัวอักษรนั้น ๆ โดยที่ n ไม่สามารถเป็นค่าลบได้ เช่น "a{9}" คือ กำหนดให้มีอักขระ "a" ซ้ำกัน 9 ตัว จึงจะเข้าเงื่อนไข
- "{n,}" คือ กำหนดขั้นต่ำตัวอักษรที่ซ้ำกันเป็น n จำนวน
- "{,n}" คือ ต้องมีตัวอักษรที่ซ้ำกันไม่เกิน n จำนวน
- "{m,n}" คือ กำหนดตัวอักษรขั้นต่ำ m จำนวน แต่ไม่เกิน n จำนวน

- “a(b|c)+” คือ ต้องประกอบด้วยตัวอักษร “a” นำหน้า ส่วนตัวอักษรที่ 2 จะเป็นคำว่า “b” หรือ “c” ตั้งแต่ 1 ตัวอักษรหรือมากกว่าก็ได้เช่นกัน

2.3.6 Pydub

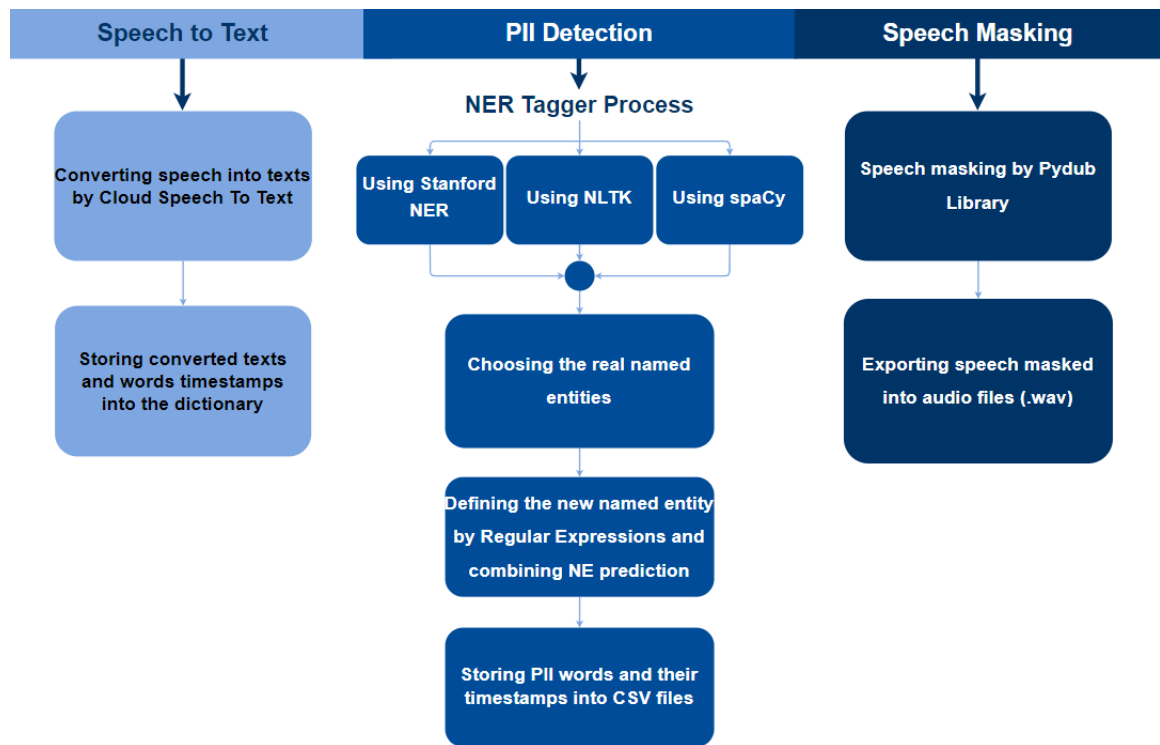
Pydub เป็นไลบรารีหนึ่งของภาษาไพทอนที่ใช้ในการจัดการกับข้อมูลที่เป็นไฟล์เสียง ซึ่งไลบรารีนี้มีคำสั่งต่าง ๆ ในการจัดการข้อมูล ยกตัวอย่างเช่น

- การโหลดไฟล์เสียง
- การบันทึกไฟล์เสียงให้อยู่ในรูปแบบไฟล์เสียงอื่น ๆ ได้
- การตัดแยก หรือ เพิ่ม เสียงในแต่ละส่วนของไฟล์เสียง
- การใส่ฟิลเตอร์เข้าไปในเสียง
- การเพิ่มหรือแทนที่เสียงรบกวน

บทที่ 3

ขั้นตอนการพัฒนาระบบ

ผู้จัดทำได้นำเทคโนโลยีและเครื่องมือมาประยุกต์ใช้โดยมีการพัฒนาตามโครงสร้างของระบบ ดังรูปที่ 3.1



รูปที่ 3.1 โครงสร้างของระบบ

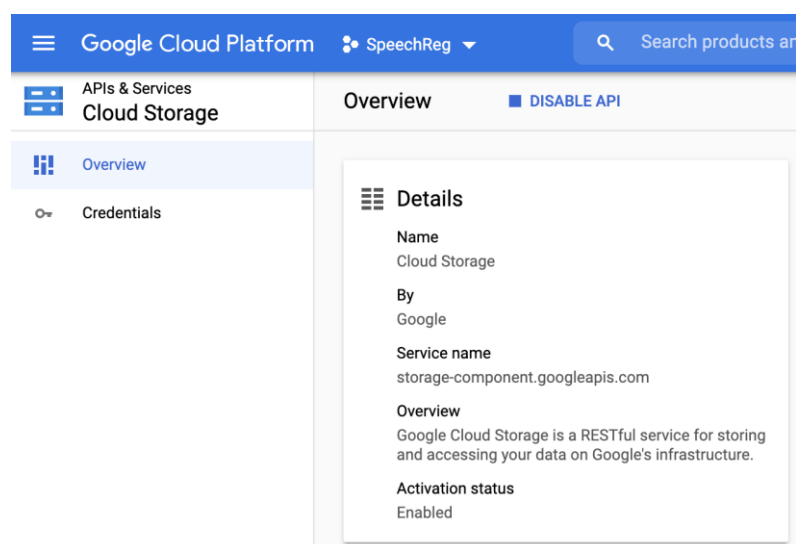
ภายในระบบมีการดำเนินงาน 3 ส่วน ดังรูปที่ 3.1 คือ การแปลงเสียงพูดให้อยู่ในรูปแบบข้อความ การตรวจจับคำที่เป็นข้อมูลส่วนบุคคลจากข้อมูลรูปแบบข้อความ และการแทนที่เสียงที่เป็นข้อมูลส่วนบุคคลด้วยเสียงรบกวน ซึ่งผู้จัดทำได้มุ่งเน้นไปที่การพัฒนาระบบการที่ 2 และ 3 ของระบบ (การตรวจจับคำที่เป็นข้อมูลส่วนบุคคลจากข้อมูลรูปแบบข้อความ และการแทนที่เสียงที่เป็นข้อมูลส่วนบุคคลด้วยเสียงรบกวน) มีรายละเอียดการดำเนินงาน ดังนี้

3.1 การแปลงเสียงพูดให้อยู่ในรูปแบบข้อความ

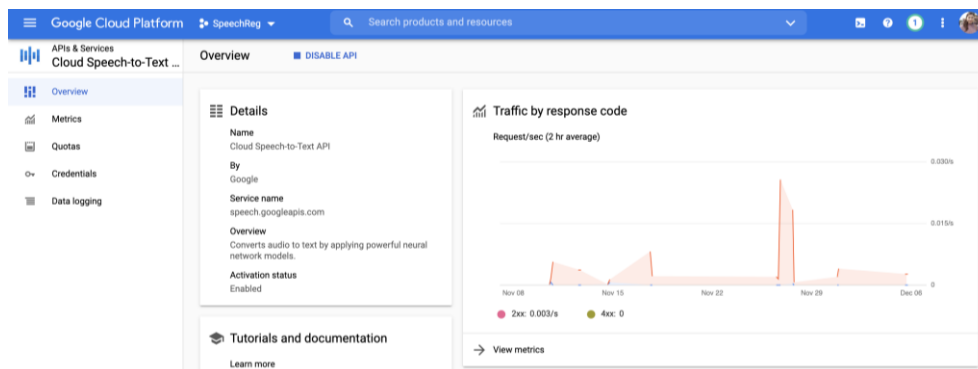
3.1.1 กระบวนการแปลงเสียงให้อยู่ในรูปแบบข้อความโดยใช้ Cloud Speech to Text

มีวิธีการดำเนินงาน ดังนี้

- หากมีบัญชีผู้ใช้ของ Google อยู่แล้วให้เข้าสู่ระบบเพื่อใช้งานกุญแจคลาวด์ ในการใช้งานครั้งแรกทางกูเกิลจะให้เครดิต 300 ดอลลาร์ หรือคิดเป็นค่าเงินบาทประมาณ 9,000 บาท เนื่องจากการใช้งานกุญแจคลาวด์นั้นจะเสียค่าใช้จ่ายตามจำนวนที่ใช้จริง
- เริ่มการสร้างโปรเจกต์บนกูเกิลคลาวด์ และเปิด API ที่ต้องการใช้งาน ผู้จัดทำเลือกใช้บริการ API 2 ประเภท ได้แก่ Cloud Storage ดังรูปที่ 3.2 และ Cloud Speech to Text ดังรูปที่ 3.3



รูปที่ 3.2 เปิดใช้งาน Cloud Storage



รูปที่ 3.3 เปิดใช้งาน Cloud Speech To Text

- ดำเนินการอัปโหลดไฟล์เสียงที่ทางผู้จัดทำสร้างบทสนทนาระหว่างลูกค้ากับศูนย์ให้บริการ ข้อมูลลูกค้าทางโทรศัพท์ขึ้นบน Cloud Storage ดังรูปที่ 3.4

voicedata_speech

OBJECTS CONFIGURATION PERMISSIONS RETENTION LIFECYCLE

Buckets > voicedata_speech > Voice

UPLOAD FILES UPLOAD FOLDER CREATE FOLDER MANAGE HOLDS DOWNLOAD DELETE

Filter Filter by object or folder name prefix

| <input type="checkbox"/> | Name | Size | Type | Created time | Storage class | Last modified | Public access | Encryption | Retention expiration date | |
|--------------------------|----------|--------|-----------|-------------------|---------------|---------------|---------------|--------------------|---------------------------|-------|
| <input type="checkbox"/> | Amanda | 3 MB | audio/wav | Nov 10, 2020, ... | Standard | Nov 10, 20... | Not public | Google-managed key | — | ⬇ ⬆ ⬇ |
| <input type="checkbox"/> | Caleb.w | 2.8 MB | audio/wav | Nov 10, 2020, ... | Standard | Nov 10, 20... | Not public | Google-managed key | — | ⬇ ⬆ ⬇ |
| <input type="checkbox"/> | Date.w | 1.3 MB | audio/wav | Nov 14, 2020, ... | Standard | Nov 14, 20... | Not public | Google-managed key | — | ⬇ ⬆ ⬇ |
| <input type="checkbox"/> | Laura.w | 4 MB | audio/wav | Nov 10, 2020, ... | Standard | Nov 10, 20... | Not public | Google-managed key | — | ⬇ ⬆ ⬇ |
| <input type="checkbox"/> | Michael | 1.3 MB | audio/wav | Nov 10, 2020, ... | Standard | Nov 10, 20... | Not public | Google-managed key | — | ⬇ ⬆ ⬇ |
| <input type="checkbox"/> | Nancy-S | 3.2 MB | audio/wav | Nov 27, 2020, ... | Standard | Nov 27, 20... | Not public | Google-managed key | — | ⬇ ⬆ ⬇ |
| <input type="checkbox"/> | Nelson.v | 2.7 MB | audio/wav | Nov 10, 2020, ... | Standard | Nov 10, 20... | Not public | Google-managed key | — | ⬇ ⬆ ⬇ |
| <input type="checkbox"/> | Robert.v | 3.6 MB | audio/wav | Nov 10, 2020, ... | Standard | Nov 10, 20... | Not public | Google-managed key | — | ⬇ ⬆ ⬇ |
| <input type="checkbox"/> | Sandra.v | 3.4 MB | audio/wav | Nov 10, 2020, ... | Standard | Nov 10, 20... | Not public | Google-managed key | — | ⬇ ⬆ ⬇ |
| <input type="checkbox"/> | convers | 2.5 MB | audio/wav | Nov 30, 2020, ... | Standard | Nov 30, 20... | Not public | Google-managed key | — | ⬇ ⬆ ⬇ |
| <input type="checkbox"/> | convers | 2 MB | audio/wav | Nov 30, 2020, ... | Standard | Nov 30, 20... | Not public | Google-managed key | — | ⬇ ⬆ ⬇ |
| <input type="checkbox"/> | convers | 2.9 MB | audio/wav | Nov 30, 2020, ... | Standard | Nov 30, 20... | Not public | Google-managed key | — | ⬇ ⬆ ⬇ |
| <input type="checkbox"/> | convers | 1.7 MB | audio/wav | Nov 30, 2020, ... | Standard | Nov 30, 20... | Not public | Google-managed key | — | ⬇ ⬆ ⬇ |
| <input type="checkbox"/> | convers | 2.7 MB | audio/wav | Nov 30, 2020, ... | Standard | Nov 30, 20... | Not public | Google-managed key | — | ⬇ ⬆ ⬇ |
| <input type="checkbox"/> | convers | 1.3 MB | audio/wav | Nov 30, 2020, ... | Standard | Nov 30, 20... | Not public | Google-managed key | — | ⬇ ⬆ ⬇ |

รูปที่ 3.4 อัปโหลดไฟล์เสียงขึ้นบน Cloud Storage

- นำเข้าข้อมูลเสียงจาก Cloud Storage และกำหนดค่าต่าง ๆ เพื่อนำไปใช้ในการแปลงเสียงพูดให้อยู่ในรูปแบบข้อความ ดังรูปที่ 3.5

```
from google.cloud import speech

audio = speech.RecognitionAudio(uri="gs://voicedata_speech/Voice/conversation 2.wav")
config = speech.RecognitionConfig(
    sample_rate_hertz=16000,
    language_code="en-US",
    enable_automatic_punctuation=True,
    enable_word_time_offsets=True,
    model="phone_call")
```

รูปที่ 3.5 การนำเข้าข้อมูลเสียงและกำหนดค่าต่าง ๆ

- แปลงเสียงพูดให้อยู่ในรูปแบบข้อความโดยการสร้างฟังก์ชันต่าง ๆ ได้แก่ ฟังก์ชัน `print_word_offsets` ระบุเวลากับคำในบทสนทนา (Timestamp) โดยเวลาที่เริ่มต้นในแต่ละคำ และเวลาที่สิ้นสุดของคำ ๆ นั้น โดยหน่วยเวลาเป็นวินาทีเพื่อให้่ายต่อการเข้าใจ ฟังก์ชัน `print_sentences` แสดงผลประโยคที่ผ่านการแปลงเสียงพูดให้อยู่ในรูปแบบข้อความและแสดงผลค่าความเชื่อมั่น (Confidence) ส่วนสุดท้ายในขั้นตอนการแปลงเสียงพูดให้อยู่ในรูปแบบของข้อความในฟังก์ชัน `speech_to_text` ใช้โมดูลของไลบรารี `Speech` ในการแปลงข้อมูลเสียงพูดให้อยู่ในรูปแบบข้อความ และระบุไฟล์

```
data = []
start_all = []
end_all = []
word_all = []

def speech_to_text(config, audio):
    client = speech.SpeechClient()
    operation = client.long_running_recognize(config=config, audio=audio)
    response = operation.result(timeout=90)
    return print_sentences(response)

def print_sentences(response):
    for result in response.results:
        best_alternative = result.alternatives[0]
        transcript = best_alternative.transcript
        confidence = best_alternative.confidence
        data.append(transcript)
        print("-" * 80)
        print(f"Transcript: {transcript}")
        print(f"Confidence: {confidence:.0%}")
        print_word_offsets(best_alternative)

def print_word_offsets(alternative):
    start, end, words = [], [], []
    for word in alternative.words:
        start_s = word.start_time.total_seconds()
        start.append(start_s)
        end_s = word.end_time.total_seconds()
        end.append(end_s)
        word = word.word
        words.append(word)
        print(f"{start_s:>7.3f} | {end_s:>7.3f} | {word}")

    start_all.append(start_s)
    end_all.append(end_s)
    word_all.append(word)
    return resultdict
speech_to_text(config, audio)
```

รูปที่ 3.6 ฟังก์ชันการแปลงเสียงพูดให้อยู่ในรูปแบบของข้อความตัวอักษร

3.1.2 บันทึกไฟล์ที่ผ่านการแปลงเสียงพูดให้อยู่ในรูปแบบข้อความให้อยู่ในรูปแบบของ Dictionary

บันทึกเป็นประเภทไฟล์ JSON เพื่อทำการตรวจจับคำที่เป็นข้อมูลส่วนบุคคลในขั้นตอนถัดไป ดังรูปที่ 3.7

```
import json
with open('data/GG_Speech/conversation 2.json', 'w') as outfile:
    json.dump(resultdict, outfile)
```

รูปที่ 3.7 การบันทึกไฟล์ข้อความเป็นไฟล์ประเภท .json

3.2 การตรวจจับคำที่เป็นข้อมูลส่วนบุคคลจากข้อมูลรูปแบบข้อความ

นำเข้าข้อมูลที่ Cloud Speech to Text ดำเนินการแปลงให้อยู่ในรูปแบบข้อความ ซึ่งเป็นไฟล์ JSON ในรูปแบบของ Dictionary และนำข้อมูลเหล่านั้นมาพัฒนาเครื่องมือ โดยรายละเอียดของกระบวนการทั้งหมด มีดังนี้

3.2.1 กระบวนการตรวจจับนิพจน์ระบุนาม (Named Entities Tagger Process)

ในขั้นตอนนี้มีการใช้ไลบรารีทั้งหมด 3 ไลบรารี เพื่อเพิ่มความแม่นยำในการตรวจจับนิพจน์ระบุนาม หรือข้อมูลส่วนบุคคล โดยรายละเอียดการดำเนินงาน มีดังนี้

- 1) พัฒนาไลบรารีของ Stanford Named Entities Recognizer โดยเลือกรูปแบบแบบจำลองสำหรับติดแท็กนิพจน์ระบุนาม 7 ประเภท และดำเนินการเลือกการติดแท็กในบทสนทนาทั้งหมดเป็นจำนวน 5 ประเภท ได้แก่ PERSON, ORGANIZATION, LOCATION, DATE และ MONEY โดยสร้างกระบวนการวิเคราะห์ข้อความจำนวน 1 ฟังก์ชัน ในฟังก์ชันมีการทำ Word Tokenization เพื่อแยกโทเค็นของคำในข้อความเป็นอันดับแรก ต่อมาติดแท็กนิพจน์ระบุนาม (Named Entities) โดยใช้อัลกอริทึมของ Stanford NER จากนั้นสร้างเงื่อนไขเก็บเฉพาะโทเค็นที่เป็นนิพจน์ระบุนามเท่านั้น และตรวจสอบโทเค็นคำที่ไลบรารีแบ่งออกมาเทียบกับโทเค็นที่ Cloud Speech to Text แบ่งไว้ให้ เพื่อให้แน่ใจว่าโทเค็นที่ Stanford NER ติดแท็กได้นั้นตรงกับระยะเวลาที่ Cloud Speech to Text ทำนายออกมา และเก็บค่าของคำที่ติดแท็กได้ พร้อมกับประเภทของนิพจน์ระบุนาม ดังรูปที่ 3.8

```

def Stanford_pred(dictt, df):

    java_path = ("C:/Program Files/Java/jdk-15.0.1/bin/java.exe")
    os.environ['JAVAHOME'] = java_path
    jar = ('D:/Program/stanford-ner-4.0.0/stanford-ner.jar')
    model = ('D:/Program/stanford-ner-4.0.0/classifiers/english.muc.7class.distsim.crf')
    st = StanfordNERTagger(model, jar, encoding = 'utf-8')

    word_token = word_tokenize(dictt)
    classified_text = st.tag(word_token)

    wordlst = []
    ne_lst = []

    for i in range(len(classified_text)):
        if str(classified_text[i][1]) != 'O':
            if str(classified_text[i][1]) == 'PERSON' or str(classified_text[i][1]) == 'LOCATION':
                wordlst.append(str(classified_text[i][0]))
                ne_lst.append(str(classified_text[i][1]))

    st_pred = []
    check = 0

    for ww in df['word']:
        check = 0
        for w, n in zip(wordlst, ne_lst):
            if ww.__contains__(w):
                check = 1
                st_pred.append(str(n))
                break
        if check == 0:
            st_pred.append('O')

    df['stanford_pred'] = st_pred

    return st_pred, df

```

รูปที่ 3.8 ฟังก์ชันการทำนายนิพจน์ระบุนามของ Stanford NER

- 2) พัฒนาไลบรารีของ NLTK โดยเลือกการติดแท็กในบทสนทนาเป็นจำนวนทั้งหมด 6 ประเภท ได้แก่ ORGANIZATION, PERSON, LOCATION, GPE, DATE และ MONEY มีการสร้างกระบวนการวิเคราะห์ข้อความเป็นจำนวน 1 ฟังก์ชัน ในฟังก์ชันนั้นมีการทำ Word Tokenization เพื่อแยกโทเค็นของคำในข้อความ จากนั้นติดแท็กนิพจน์ระบุนาม (Named Entities) โดยใช้อัลกอริทึมของ NLTK ซึ่งต้องมีการติดแท็กส่วนของประโยค (Part-of-Speech) ก่อน จึงจะสามารถติดแท็กนิพจน์ระบุนามได้ ต่อมาได้สร้างเงื่อนไขเลือกเฉพาะโทเค็นที่มีนิพจน์ระบุนาม และเปลี่ยนประเภทของนิพจน์ระบุนามให้เหมือนกับไลบรารีอื่น ๆ เช่น คำว่า “LOC” เปลี่ยนเป็น “LOCATION” เป็นต้น และนอกจากนี้ ยังได้รวมนิพจน์ระบุนามประเภท LOCATION และ GPE เข้าด้วยกัน โดยการเปลี่ยนชื่อประเภท GPE ให้เป็น LOCATION ทั้งหมด เพื่อให้เป็นประเภทเดียวกันกับ Stanford NER จากนั้นจับคู่โทเค็นคำที่ไลบรารีแบ่งออกมาเทียบกับโทเค็นที่ Cloud Speech to Text แบ่งไว้ให้ เพื่อให้แน่ใจว่าโทเค็นที่ NLTK ติดแท็กได้นั้นตรงกับระยะเวลาที่ Cloud Speech to Text ทำนายออกมา และเก็บค่าของคำที่ติดแท็กได้ พร้อมกับประเภทของนิพจน์ระบุนาม ดังรูปที่ 3.9

```
def NLTK_pred(dictt, df):

    word_token = word_tokenize(dictt)
    tagged_words = pos_tag(word_token)
    ne_tagged = ne_chunk(tagged_words, binary = False)

    lst_word = []
    lst_ne = []

    for chunk in ne_tagged:
        if hasattr(chunk, 'label'):
            if chunk.label() == 'PERSON' or chunk.label() == 'LOCATION' or chunk.label() == 'ORG':
                if chunk.label() == 'ORG':
                    lst_word.append(chunk[0][0])
                    lst_ne.append('ORGANIZATION')
                if chunk.label() == 'LOC' or chunk.label() == 'GPE':
                    lst_word.append(chunk[0][0])
                    lst_ne.append('LOCATION')
            else:
                lst_word.append(chunk[0][0])
                lst_ne.append(chunk.label())

    nltk_pred = []
    check = 0

    for ww in df['word']:
        check = 0
        for w, n in zip(lst_word, lst_ne):
            if ww.__contains__(w):
                check = 1
                nltk_pred.append(str(n))
                break
        if check == 0:
            nltk_pred.append('0')

    df['nltk_pred'] = nltk_pred
```

รูปที่ 3.9 ฟังก์ชันการทำนายนิพจน์ระบุนามของ NLTK

- 3) พัฒนาไลบรารีของ spaCy โดยเลือกการติดแท็กในบทสนทนาเป็นจำนวนทั้งหมด 6 ประเภท ได้แก่ ORGANIZATION, PERSON, LOCATION, GPE, DATE และ MONEY และสร้างกระบวนการวิเคราะห์ข้อความเป็นจำนวน 1 ฟังก์ชัน ในฟังก์ชันมีการใช้อัลกอริทึมของ spaCy ซึ่งในอัลกอริทึมจะดำเนินการวิเคราะห์ข้อความต่าง ๆ อัตโนมัติ ส่งผลให้ทางผู้จัดทำสามารถเรียกค่าได้จากอัลกอริทึมของไลบรารีได้ทันที จากนั้นสร้างเงื่อนไขเลือกเฉพาะโทเค็นคำที่มีนิพจน์ระบุนาม (Named Entities) และเปลี่ยนชื่อประเภทของนิพจน์ระบุนามให้ตรงกับไลบรารีอื่น ๆ เช่นเดียวกันกับ Stanford NER และ NLTK ต่อมาทำการจับคู่โทเค็นคำที่ไลบรารีแบ่งออกมาเทียบกับโทเค็นที่ Cloud Speech to Text แบ่งไว้ให้ เพื่อให้แน่ใจว่าโทเค็นที่ spaCy ติดแท็กได้นั้นตรงกับระยะเวลาที่ Cloud Speech to Text ทำนายออกมา และเก็บค่าของคำที่ติดแท็กได้ พร้อมกับประเภทของนิพจน์ระบุนาม ดังรูปที่ 3.10

```
def spaCy_pred(dictt, df):
    nlp = en_core_web_sm.load()
    # list of words that have named entities
    text = ([str(X) for X in nlp(dictt)
              if (X.ent_type_ != '' and X.ent_type_ != 'CARDINAL') & (str(X) != 'a') & (str(X) != '0')])
    # list of named entities
    ne = ([X.ent_type_ for X in nlp(dictt)
           if (X.ent_type_ != '' and X.ent_type_ != 'CARDINAL') & (str(X) != 'a') & (str(X) != '0')])

    sp_pred = []

    for n, i in enumerate(ne):
        if i == 'LOC':
            ne[n] = 'LOCATION'
        if i == 'GPE':
            ne[n] = 'LOCATION'
        if i == 'ORG':
            ne[n] = 'ORGANIZATION'

    check = 0

    for ww in df['word']:
        check = 0
        for w, n in zip(text, ne):
            if ww.__contains__(w):
                check = 1
                sp_pred.append(str(n))
                break
        if check == 0:
            sp_pred.append('0')

    df['spacy_pred'] = sp_pred

    return sp_pred, df
```

รูปที่ 3.10 ฟังก์ชันการทำนายนิพจน์ระบุนามของ spaCy

3.2.2 กระบวนการเลือกคำทำนายจริง

สร้างฟังก์ชันเพื่อเลือกโทเค็นของคำที่ไลบรารีทำนายประเภทของนิพจน์ระบุนามเหมือนกัน ตั้งแต่ 2 ไลบรารีขึ้นไป และ เลือกคำทำนายจริงประเภท PERSON และ DATE ที่ spaCy ทำนายทันทีในกรณีที่ Stanford NER และ NLTK ทำนายประเภทไม่เหมือนกัน ซึ่งเงื่อนไขนี้ได้จากการทดลองที่ 2 ในบทของการทดลอง และผลลัพธ์ที่ได้จากระบบ เนื่องจากการใช้ไลบรารีเดียวอาจไม่แม่นยำมากพอที่จะทำนายประเภทของโทเค็นคำได้อย่างถูกต้อง ผู้จัดทำจึงสร้างเงื่อนไขนี้เพื่อเพิ่มประสิทธิภาพของการทำนาย และเก็บค่าของโทเค็นคำกับประเภทของนิพจน์ระบุนาม เพื่อนำไปวิเคราะห์ในขั้นตอนถัดไปในที่นี้ จะแทนผลลัพธ์ที่ได้จากระบบการนี้ว่า “คำทำนายจริง” ดังรูปที่ 3.11

```
# ----- Selecting same named entity predictions 2 of 3 models -----

i_twooth = []
ne_twooth = []

for i, st, nl, sp in zip(df.index, df['stanford_pred'], df['nltk_pred'], df['spacy_pred']):
    # check if spacy predict 2 NE
    if (st == '0' or nl == '0') and (str(sp) == 'DATE' or str(sp) == 'PERSON'):
        i_twooth.append(i)
        ne_twooth.append(str(sp))
    # check if stanford and nltk are same named entities
    elif (st != '0' and nl != '0') and (str(st) == str(nl)):
        i_twooth.append(i)
        ne_twooth.append(str(st))
    # check if stanford and spacy are same named entities
    elif (st != '0' and sp != '0') and (str(st) == str(sp)):
        i_twooth.append(i)
        ne_twooth.append(str(sp))
    # check if nltk and spacy are same named entities
    elif (nl != '0' and sp != '0') and (str(nl) == str(sp)):
        i_twooth.append(i)
        ne_twooth.append(str(sp))

combined = []
combined_check = 0

for i in df.index:
    combined_check = 0
    for ii, n in zip(i_twooth, ne_twooth):
        if i == ii:
            combined_check = 1
            combined.append(str(n))
            break
    if combined_check == 0:
        combined.append('0')
```

รูปที่ 3.11 ฟังก์ชันการเลือกการทำนายประเภทนิพจน์ระบุนามที่เหมือนกัน 2 ใน 3

3.2.3 สร้างประเภทของนิพจน์ระบุนามเพิ่มเพื่อติดแท็กเลขที่เป็นข้อมูลส่วนบุคคลโดยใช้ Regular Expressions และรวมกับคำทำนายจริง

นำคำโทเค็นคำของ Cloud Speech to Text มาวิเคราะห์โดยสร้างเงื่อนไขเพื่อติดแท็กเฉพาะโทเค็นที่เป็นตัวเลขโดยใช้ Regular Expressions ในการตรวจสอบ และกำหนดประเภทของเลขที่เป็นข้อมูลส่วนบุคคลให้เป็นชื่อประเภท “PIINUM” หรือ Personally Identifiable Information

Number เพื่อตรวจจับเลขข้อมูลส่วนบุคคลต่าง ๆ ดังนี้ เลขบัตรประชาชน (13 หลัก) เบอร์โทรศัพท์ (10 หลัก) เลขที่บัญชี (9 หลัก) เลขบัตรเดบิต หรือบัตรเครดิต (16 หลัก) และ เลขอื่น ๆ ที่มีจำนวนตั้งแต่ 9 หลักขึ้นไป ในกรณีที่ Cloud Speech to Text แปลงเป็นข้อความออกมาได้ไม่แม่นยำ ดังรูปที่ 3.12

```
# ----- Regular Expression checking -----

pii_index = []
pii_type = []
date_check = 0

for i, num in zip(df.index, df['word']):
    date_check = 0
    for ii in i.twooth:
        if i == ii:
            date_check = 1
            break
    if date_check == 0:
        # ID card e.g. +666-666-666-6666
        if re.search('(\\+?[0-9]{3,}-?[0-9]{3,}-?[0-9]{3,}-?[0-9]{4,})', num):
            pii_index.append(i)
            pii_type.append('PIINUM')
        # phone number e.g. 666-666-6666
        elif re.search('(\\+?[0-9]{3,}-?[0-9]{3,}-?[0-9]{4,})', num):
            pii_index.append(i)
            pii_type.append('PIINUM')
        # account number e.g. 666-666-666
        elif re.search('(\\+?[0-9]{3,}-?[0-9]{3,}-?[0-9]{3,})', num):
            pii_index.append(i)
            pii_type.append('PIINUM')
        # card number
        elif re.search('(\\+?[0-9]{2,}-?[0-9]{3,}-?[0-9]{3,}-?[0-9]{3,}-?[0-9]{+}\\+?[0-9]{+})', num):
            pii_index.append(i)
            pii_type.append('PIINUM')
        # if not has punctuation
        elif re.search('(\\+?[0-9]{9,})', num):
            pii_index.append(i)
            pii_type.append('PIINUM')
```

รูปที่ 3.12 การสร้างนิพจน์ระบุนามใหม่โดยใช้ Regular Expressions

จากนั้นรวมค่าทำนายจริงกับค่าของเลขที่เป็นข้อมูลส่วนบุคคลเข้าด้วยกัน และเก็บค่านั้นไว้ในตาราง ซึ่งมีการเก็บค่าของตัวเลขที่เป็นข้อมูลส่วนบุคคลรวมกับค่าทำนายจริงเท่านั้น ไม่ได้มีการเก็บค่าเหล่านี้รวมกับไลบรารี 3 ไลบรารีที่ใช้ในการพัฒนา ดังรูปที่ 3.13


```
# ----- Combining real ents and regex -----

cb_rg = []

for ent, regex in zip(combined, regex_lst):
    if ent != '0' and regex == '0':
        cb_rg.append(ent)
    elif regex != '0' and ent == '0':
        cb_rg.append(regex)
    else:
        cb_rg.append('0')

df['real_ents'] = cb_rg
```

รูปที่ 3.13 รวมการทำนาย Regular Expressions และคำทำนายจริงเข้าด้วยกัน

3.2.4 เก็บค่าต่าง ๆ ให้อยู่ในรูปของไฟล์ CSV

หลังจากได้ผลลัพธ์การทำนายนิพจน์ระบุนาม จึงจัดเก็บค่าเหล่านั้นให้อยู่ในรูปแบบตารางและบันทึกเป็นไฟล์ CSV เป็นจำนวนทั้งหมด 5 คอลัมน์ ได้แก่ ลำดับโทเค็น (indx) โทเค็นคำ (word) เวลาที่เริ่มพูดโทเค็นนั้นในไฟล์เสียง (start_time) เวลาที่สิ้นสุด (end_time) และประเภทของนิพจน์ระบุนาม (real_ents) ดังรูปที่ 3.14

```
def filter_ents(df):

    formal_ents = df.drop(['stanford_pred', 'nltk_pred', 'spacy_pred'], axis = 1)
    formal_ents = formal_ents[formal_ents['real_ents'] != '0']

    return formal_ents
```

รูปที่ 3.14 การเก็บค่าทำนายจริงและค่าของเลขที่เป็นข้อมูลส่วนบุคคล

3.3 การแทนที่เสียงที่เป็นข้อมูลส่วนบุคคลด้วยเสียงรบกวน

นำค่าที่ได้จากไฟล์ CSV ในขั้นตอนก่อนหน้านี้ โดยเลือกใช้เพียงคอลัมน์เวลาเริ่มต้น (start_time) และเวลาสิ้นสุดของคำนั้น (end_time) จากนั้นแบ่งช่วงของเวลาเริ่มต้นและสิ้นสุด แปลงค่าของเวลาให้อยู่ในหน่วยของมิลลิวินาที และแทนที่เสียงรบกวนในช่วงเวลาที่ได้คำนวณไว้ พร้อมปรับเดซิเบลของเสียงรบกวนให้อยู่ในระดับที่เหมาะสม และบันทึกไฟล์เสียงที่มีการปิดบังข้อมูลส่วนบุคคลเป็นไฟล์เสียงประเภท .wav ดังรูปที่ 3.15

```

file = pd.read_csv('data/ner_pred/ner_pred_con2.csv', usecols=['start_time', 'end_time'])

def white_noise(file, input_path, output_path):
    file = pd.read_csv(file, usecols = ['start_time', 'end_time']) #ไฟล์ csv ที่ต้องตัด word ออก พร้อมระบุเวลา
    sound = AudioSegment.from_file(input_path) #ไฟล์เสียงต้นฉบับ
    begin = 0
    new_sound = sound[0:0]
    for start, stop in zip(file['start_time'], file['end_time']):
        normal = sound[begin * 1000:start * 1000] # * 1000 เพื่อให้ทำให้เป็น millisec
        begin = stop
        new_sound += (normal + WhiteNoise().to_audio_segment(duration = (stop - start) * 1000).apply_gain(-35))
    new_sound.export(out_f = output_path, format = "wav")
    return new_sound

new_sound2 = white_noise("data/ner_pred/ner_pred_con2.csv", "data/Voice files/conversation_2.wav", "Output/conversation_2.wa

```

รูปที่ 3.14 ฟังก์ชันการแทนที่เสียงที่เป็นข้อมูลส่วนบุคคลด้วยเสียงรบกวน

3.4 การประเมินผล (Evaluation)

มีการประเมินผลกระบวนการทั้งหมด 2 กระบวนการ คือ ประเมินผลความแม่นยำของการแปลงเสียงพูดให้อยู่ในรูปแบบข้อความจากการเทียบจากข้อมูลจริงโดยประยุกต์ใช้แนวคิด Jaccard's Coefficient Similarity และประเมินผลความแม่นยำของการตรวจจับคำที่เป็นข้อมูลส่วนบุคคลจากข้อมูลรูปแบบข้อความจากการสร้างผลเฉลยของการทำนายข้อความและโทเค็นต่าง ๆ เพื่อใช้ตรวจสอบความแม่นยำในการทำนายของไลบรารีทั้งหมดโดยวัดจากค่า Recall เท่านั้น

บทที่ 4

การทดลองและผลลัพธ์ที่ได้จากระบบ

ผู้จัดทำได้แบ่งการแสดงผลการทดลองและผลลัพธ์ที่ได้เป็น 4 หัวข้อหลัก มีรายละเอียด ดังนี้

4.1 การวิเคราะห์ข้อมูลที่สร้างขึ้น

ในการเตรียมข้อมูลเกิดปัญหาที่ไม่สามารถหาชุดข้อมูลจากแหล่งข้อมูลสาธารณะมาพัฒนาระบบได้เนื่องจากข้อมูลเหล่านี้มีข้อจำกัดในเรื่องของข้อมูลส่วนบุคคล จึงต้องทดลองสร้างชุดข้อมูลขึ้นมาเอง โดยมีรายละเอียดการสร้างข้อมูล ดังนี้

4.1.1 สร้างบทสนทนาระหว่างลูกค้าและศูนย์บริการข้อมูลลูกค้าทางโทรศัพท์ (Call Center)

```
Hello, you have called Virtual bank, this is Linda speaking. How may I help you?
Hi Linda. I was just at your Ville branch and I think I left my Debit card in the ATM machine.
Okay. Do you have your Debit card number?
I don't have.
Okay, well do you have the checking account number associated with the Debit card?
That I do have. Are you ready? I will give you what I have got. 765456789.
Okay. That's 765456789.
Correct.
What is your identification number?
7745896589665.
Okay, I have 7745896589665 and what is your name sir?
It is Robert Applebaum.
Okay. I have Robert Applebaum.
Yes.
And what is your date of birth Mr. Applebaum?
July 7th, 1974.
Okay. July 7th, 1974.
Yes.
And your phone number?
It is 6102651715.
Okay. I have 6102651715.
Yes.
Okay Mr. Applebaum. I have just suspended your card. If it is in the machine, we will contact you and lift the suspension.
Oh, thank you.
Sure. Thank you.
```

รูปที่ 4.1 ตัวอย่างบทสนทนาระหว่างลูกค้ากับศูนย์บริการข้อมูลลูกค้าทางโทรศัพท์

สร้างชุดข้อมูลในรูปแบบข้อความเป็นจำนวนทั้งหมด 23 บทสนทนา (Conversations) เพื่อใช้ในการพัฒนาและประเมินผลระบบ

ดำเนินการวิเคราะห์และสำรวจข้อมูล (Exploratory Data Analysis: EDA) ดังนี้

- วิเคราะห์ประโยค (Sentences Analysis)

```
----- All Conversations -----
Amount of all sentences: 566 sentences
----- Average Sentences per one conversation -----
Average Sentences: 24.61 sentences
-----
```

รูปที่ 4.2 รายละเอียดการวิเคราะห์ประโยค

รูปที่ 4.2 แสดงให้เห็นว่ามีประโยคทั้งหมด 566 ประโยค ซึ่งถูกแบ่งประโยคโดยใช้ไลบรารีของ NLTK และใน 1 บทสนทนา มีจำนวนเฉลี่ยทั้งหมด 24.61 ประโยค

- วิเคราะห์คำ (Words Analysis)

```
----- Original Text -----
Total words amount: 4095 words
Average words in one conversation: 178.04 words
----- Cleaned Text -----
Words amount after remove punctuation and stop words: 1732 words
Average words in one conversation after remove punctuation and stop words: 75.30 words
-----
```

รูปที่ 4.3 รายละเอียดการวิเคราะห์คำ

จากรูปที่ 4.3 แบ่งการวิเคราะห์คำออกเป็น 2 ประเภท คือ วิเคราะห์คำจากบทสนทนาจริง และวิเคราะห์คำจากบทสนทนาที่ดำเนินการทำความสะอาดข้อมูล (Data Cleaning) จากการตัดเครื่องหมายวรรคตอนและ Stop words ที่ทางไลบรารีของ NLTK ได้จัดสรรให้ ดังรูปที่ 4.4

```
Stoplist that has to remove: {'up', 'just', 'now', "you'll", 've', "she's", 'ain', "mustn't", 'before', '%', "haven't", 'under', 'about', 'was', 'yourselves', 'couldn't', '"', 'du ring', 'its', 'over', 'ma', "you're", 'o', 'until', 'had', ',', '{', ';', 'himself', 'the ir', "should've", "you'd", 'while', 'myself', 'same', '\\', 'to', "it's", 'by', 'they', "mightn't", 'that', 'i', 'out', 'who', ')', ']', 'hadn', 'we', 'have', 'or', "couldn't", 'didn', 'll', 'nor', 'weren', '+', 'if', 'there', "didn't", 'me', 'our', '/', 'needn', "s han't", 'through', "hasn't", 'don', 'you', "weren't", 'here', 'can', '|', "isn't", 'itsel f', 'should', 'm', 'my', 'this', 'are', 'ours', 'been', '#', '[', 'such', 'shouldn', 'he r', 'it', 'what', 'did', 'all', 'some', 'doesn', '!', ':', "wasn't", 'only', 'off', "are n't", 'won', 'so', 'an', 'own', 'on', 'aren', "needn't", 'am', 'doing', 'too', 'again', 'more', 'not', "shouldn't", '&', 'where', 'in', '}', 'both', '<', 'she', 'as', 'from', 'b elow', 'above', 'down', '$', '~', 'after', 'will', 'most', 'your', 'once', '_', 'has', '=', 'being', 'of', 'his', 'those', 'few', 'isn', '-', 'further', 'with', 'he', "would n't", 'having', 'haven', 'does', 're', 'these', 'themselves', '>', 'a', "hadn't", 'oursel ves', '*', 'because', 'd', 'mightn', 'which', 'why', 'yourself', 'shan', 'y', 'were', 'th an', '-', 'hers', 'wasn', "you've", 'is', 'be', 'do', 'the', 'then', '^', 's', '?', "does n't", 'and', 'herself', 'any', 'each', 'very', '(', '"', 'yours', 'theirs', '.', "won't", 'but', 'how', "don't", 'them', 'into', '@', 'hasn', 'other', 'when', "that'll", 'again s t', 't', 'mustn', 'whom', 'wouldn', 'for', 'no', 'him', 'between', 'at'}
```

รูปที่ 4.4 รายการของเครื่องหมายวรรคตอนและ Stop words

เมื่อตัดคำในรายการเหล่านั้นออกแล้ว ดังรูปที่ 4.4 อธิบายได้ว่าจากบทสนทนาจริง มีคำในบทสนทนาทั้งหมด 4095 คำ และใน 1 บทสนทนามีจำนวนเฉลี่ย 178.04 คำ และจากบทสนทนาที่ผ่านการทำความสะอาดข้อมูลแล้ว มีคำในบทสนทนาทั้งหมด 1732 คำ และใน 1 บทสนทนามีจำนวนเฉลี่ย 75.30 คำ

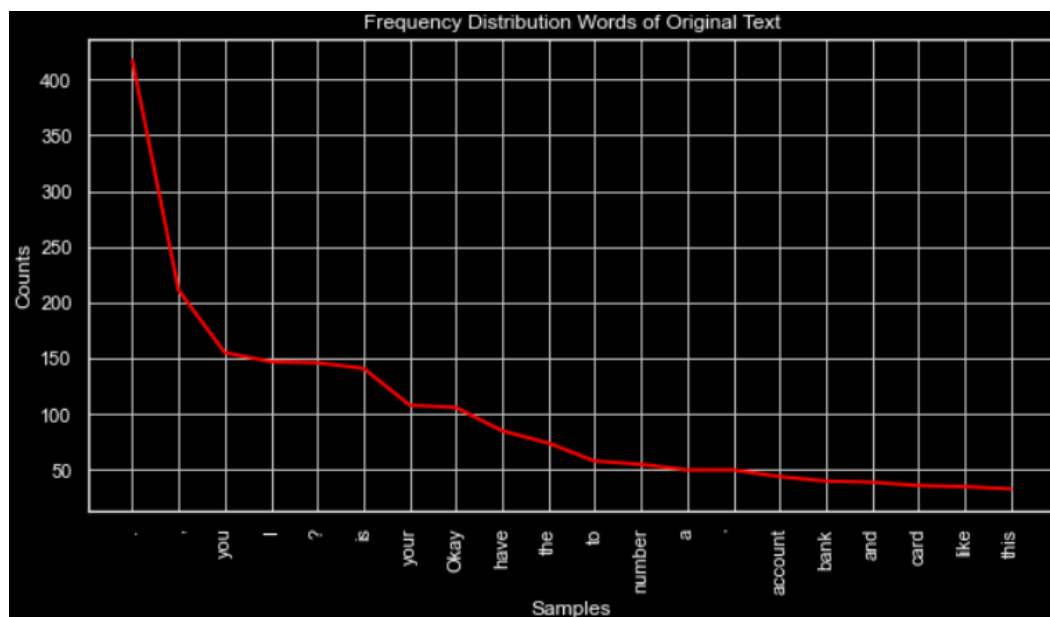
- วิเคราะห์ความถี่ของคำที่ไม่ซ้ำกัน (Distinct Word Frequencies)

```
----- Frequency Distribution of Original Text -----
Amount of distinct words: 510 words

----- Frequency Distribution of Cleaned Text -----
Amount of distinct words: 385 words
```

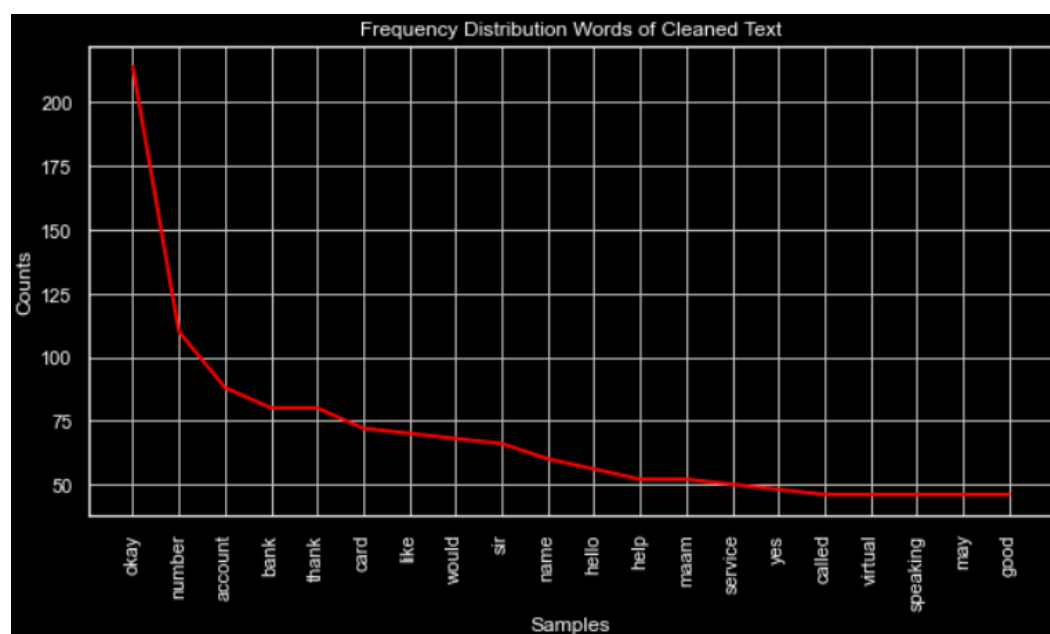
รูปที่ 4.5 รายละเอียดการวิเคราะห์ความถี่ของคำที่ไม่ซ้ำกัน

รูปที่ 4.5 แบ่งการวิเคราะห์คำเป็น 2 ประเภทเช่นเดียวกับขั้นตอนการวิเคราะห์คำ (Words Analysis) ก่อนหน้านี้ อธิบายได้ว่าในบทสนทนาจริงมีจำนวนคำที่ไม่ซ้ำกันเป็นจำนวน 510 คำ และบทสนทนาที่ผ่านการทำความสะอาดข้อมูลแล้ว มีคำที่ไม่ซ้ำกันเป็นจำนวน 385 คำ มีการแจกแจงความถี่ของคำที่ซ้ำกันมากที่สุด 20 คำแรกของบทสนทนาจริง ดังรูปที่ 4.6 และแจกแจงความถี่ของคำที่ซ้ำกันมากที่สุด 20 คำแรกของบทสนทนาที่ผ่านการทำความสะอาดข้อมูลแล้ว 20 คำแรก ดังรูปที่ 4.7



รูปที่ 4.6 การแจกแจงความถี่ของคำที่ซ้ำกันของบทสนทนาจริง

รูปที่ 4.6 ตัวอย่างการอ่านกราฟ 3 อันดับแรกที่มีความถี่มากที่สุด คือ “.” มีความถี่ทั้งหมด 417 คำ รองลงมาคือ “,” มีความถี่ทั้งหมด 211 คำ และสุดท้ายคือ “you” มีความถี่ทั้งหมด 155 คำ เป็นต้น



รูปที่ 4.7 การแจกแจงความถี่ของคำที่ซ้ำกันของบทสนทนาที่ผ่านการทำความสะอาด

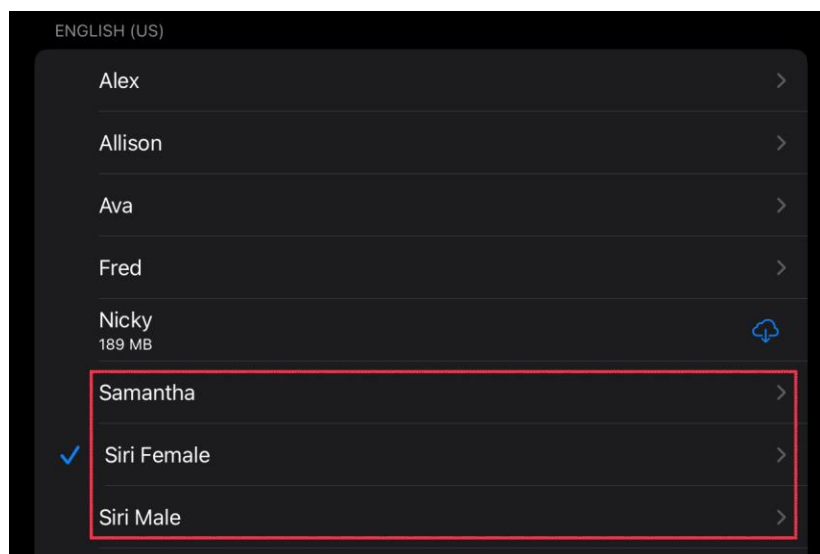
รูปที่ 4.7 ตัวอย่างการอ่านกราฟ 3 อันดับแรกที่มีความถี่มากที่สุด คือ “okay” มีความถี่ทั้งหมด 214 คำ รองลงมาคือ “number” มีความถี่ทั้งหมด 110 คำ และสุดท้ายคือ “account” มีความถี่ทั้งหมด 88 คำ เป็นต้น

4.1.2 นวัตกรรมในรูปแบบข้อความที่สร้างขึ้นมาดำเนินการบันทึกเสียง

เนื่องจากบทสนทนาที่สร้างขึ้นเป็นบทสนทนาภาษาอังกฤษ จึงต้องนำประโยคบทสนทนามาบันทึกเสียงโดยใช้เครื่องมือสำเร็จรูปในการแปลงข้อความให้อยู่ในรูปของเสียง (Text-to-Speech) โดยใช้ Siri (Siri) ในการช่วยอ่านบทสนทนาเหล่านั้น ใน 1 บทสนทนาประกอบไปด้วยเสียงของพนักงานและลูกค้า โดยแบ่งตามเพศ ดังนี้

- เสียงพนักงานที่ให้บริการในศูนย์บริการข้อมูลลูกค้าทางโทรศัพท์ (Call Center)
ผู้จัดทำกำหนดให้เสียงพนักงานมีเพียงเพศเดียว คือ เพศหญิง ซึ่งเสียงของพนักงานได้ใช้เสียงที่มีชื่อว่า “Siri Female” และใช้สำเนียงของประเทศสหรัฐอเมริกา (The United States of America) ในการอ่านข้อความเพื่อบันทึกเสียง
- เสียงของลูกค้า

เสียงของลูกค้ามี 2 เพศ คือ เพศชาย และเพศหญิง โดยเพศชายได้ใช้เสียงที่มีชื่อว่า “Siri Male” และใช้สำเนียงของประเทศสหรัฐอเมริกา (The United States of America) ในการอ่านข้อความเพื่อบันทึกเสียง และในส่วนของเพศหญิงนั้น ได้ใช้เสียงที่มีชื่อว่า “Samantha” และใช้สำเนียงของประเทศสหรัฐอเมริกา (The United States of America) ในการอ่านข้อความเพื่อบันทึกเสียง ดังรูปที่ 4.8



รูปที่ 4.8 รายการชื่อเสียงพูดที่ใช้ในการบันทึกเสียงบทสนทนา

ดำเนินการบันทึกเสียงจากสมาร์ทโฟน ประเภทของไฟล์คือ “.m4a” ซึ่งระยะเวลาในแต่ละไฟล์เสียงของบทสนทนาโดยเฉลี่ยคิดเป็นความยาวประมาณ 1 นาที ส่วนใหญ่แล้วมักจะไม่เกิน 2 นาทีจากบทสนทนาทั้งหมด ดังรูปที่ 4.9

| | |
|------------------------|-------|
| conversation 3 | 01:16 |
| Monday | |
| conversation 2 | 01:35 |
| Monday | |
| conversation 15 | 01:21 |
| Monday | |
| conversation 20 | 00:57 |
| Monday | |
| conversation 18 | 01:04 |
| Monday | |
| conversation 22 | 00:42 |
| Monday | |
| conversation 21 | 01:30 |
| Monday | |

รูปที่ 4.9 ตัวอย่างไฟล์เสียงที่บันทึกจากสมาร์ทโฟน

4.1.3 แปลงประเภทของไฟล์เสียงบทสนทนา

ในกระบวนการใช้แบบจำลองแปลงข้อมูลเสียงให้อยู่ในรูปแบบข้อความมีข้อจำกัดคือแบบจำลองสามารถประมวลผลเฉพาะข้อมูลเสียงที่เป็นประเภทไฟล์ “.wav” และ “.mp3” เท่านั้น จึงต้องแปลงประเภทไฟล์เสียงจาก “.m4a” ให้อยู่ในประเภทไฟล์ “.wav” เพราะ ประเภทไฟล์ “.wav” นั้นไม่ทำให้ไฟล์เสียงสูญเสียคุณภาพ [19] โดยแปลงประเภทของไฟล์บนเว็บไซต์ที่ชื่อว่า “Convert MP4 to WAV” [20]

4.2 การแปลงเสียงพูดให้อยู่ในรูปแบบข้อความ

```
{'transcript': "Hello, you have called virtual bank. This is Linda speaking. How may I help you? Hi Linda. I was just at your bill branch and I think I left my debit card in the ATM machine. Okay. Do you have your debit card number? I don't know. Okay. Well, do you have the checking account number associated with the debit card, but I do have are you ready? I will give you what I have got 760-545-6789. Okay. That's +765-450-600-7089. Correct? What is your identification number? 774-589-6589 665 okay. I have +774-580-960-5896 65 and what is your name sir? It is Robert. Appelbaum. Okay. I have Robert Applebaum yet. And what is your date of birth Mr. Appelbaum, July 7th, 1974. Okay, July 7th, 1974. Yes, and your phone number. It is 610-265-1715. Okay, I have 610-265-1715. Yes. Okay, Mr. Appelbaum. I have just this pended your card. If it is in the machine, we will contact you as lift the suspension 00. Thank you, sir. Thank you.", 'values': {'start': [0.0, 0.4, 1.2, 1.3, 1.8, 2.2, 2.4, 3.2, 3.4, 3.8, 4.3, 5.3, 5.3, 5.5, 5.7, 6.2, 6.8, 7.2, 8.0, 8.2, 8.3, 8.7, 8.8, 9.0, 9.5, 9.8, 9.8, 10.0, 10.2, 10.4, 10.7, 11.1, 11.2, 11.6, 11.7, 11.8, 12.3, 13.1, 14.2, 14.2, 14.4, 14.6, 15.0, 15.1, 15.4, 16.4, 16.5, 16.7, 18.2, 18.9, 19.2, 19.3, 19.4, 19.6, 19.9, 20.5, 20.8, 21.1, 21.8, 21.9, 22.3, 22.4, 23.1, 23.3, 23.4, 23.6, 24.6, 24.8, 25.1, 25.9, 26.1, 26.2, 26.5, 26.6, 26.7, 26.8, 27.2, 30.6, 31.8, 32.7, 36.0, 37.1, 37.2, 37.3, 37.5, 38.1, 38.9, 42.7, 43.7, 44.5, 45.2, 45.4, 49.0, 49.5, 50.2, 50.3, 50.4, 50.6, 50.7, 51.1, 51.8, 51.9, 52.3, 52.7, 53.0, 54.4, 54.4, 55.0, 55.4, 56.0, 57.1, 58.3, 58.4, 58.5, 58.7, 58.9, 59.1, 59.3, 59.8, 60.3, 61.6, 62.1, 63.8, 64.9, 66.0, 66.6, 68.3, 69.3, 70.3, 70.4, 70.7, 71.1, 71.9, 71.9, 75.4, 76.0, 76.4, 77.4, 81.0, 82.4, 82.6, 83.1, 83.6, 84.5, 84.8, 85.2, 85.3, 85.8, 85.9, 86.4, 87.2, 87.4, 87.5, 87.6, 87.7, 87.9, 88.8, 89.0, 89.4, 89.8, 89.9, 90.3, 90.4, 90.5, 91.7, 92.4, 92.5, 93.4, 94.5], 'end': [0.4, 1.2, 1.3, 1.8, 2.2, 2.4, 3.2, 3.4, 3.8, 4.3, 5.3, 5.3, 5.5, 5.7, 5.9, 6.8, 7.2, 8.0, 8.2, 8.3, 8.7, 8.8, 9.0, 9.5, 9.8, 9.8, 10.0, 10.2, 10.4, 10.7, 11.1, 11.2, 11.6, 11.7, 11.8, 12.3, 13.1, 14.2, 14.2, 14.4, 14.6, 15.0, 15.1, 15.4, 16.4, 16.5, 16.7, 18.2, 18.9, 19.2, 19.3, 19.4, 19.6, 19.9, 20.5, 20.8, 21.1, 21.8, 21.9, 22.3, 22.4, 23.1, 23.3, 23.4, 23.6, 24.6, 24.8, 25.1, 25.9, 26.1, 26.2, 26.5, 26.6, 26.7, 26.8, 27.2, 30.6, 31.8, 32.7, 35.7, 36.7, 37.2, 37.3, 37.5, 38.1, 38.9, 42.7, 43.7, 44.5, 45.2, 45.4, 49.0, 49.5, 50.2, 50.3, 50.4, 50.6, 50.7, 51.1, 51.8, 51.9, 52.3, 52.7, 53.0, 53.8, 54.4, 55.0, 55.4, 56.0, 57.1, 58.3, 58.4, 58.5, 58.7, 58.9, 59.1, 59.3, 59.8, 60.3, 61.6, 62.1, 63.5, 64.9, 66.0, 66.6, 68.3, 69.3, 70.3, 70.4, 70.7, 71.1, 71.9, 71.9, 75.4, 76.0, 76.4, 77.4, 80.7, 81.4, 82.6, 83.1, 83.6, 84.5, 84.8, 85.2, 85.3, 85.8, 85.9, 86.4, 87.2, 87.4, 87.5, 87.6, 87.7, 87.9, 88.8, 89.0, 89.4, 89.8, 89.9, 90.3, 90.4, 90.5, 91.7, 92.4, 92.5, 93.4, 94.5, 94.7], 'word': ['Hello,', 'you', 'have', 'called', 'virtual', 'bank.', 'This', 'is', 'Linda', 'speaking.', 'How', 'may', 'I', 'help', 'you?', 'Hi', 'Linda.', 'I', 'was', 'just', 'at', 'your', 'bill', 'branch', 'and', 'I', 'think', 'I', 'left', 'my', 'debit', 'card', 'in', 'the', 'ATM', 'machine.', 'Okay.', 'Well,', 'do', 'you', 'have', 'the', 'checking', 'account', 'number', 'associated', 'with', 'the', 'debit', 'card,', 'but', 'I', 'do', 'have', 'are', 'you', 'ready?', 'I', 'will', 'give', 'you', 'what', 'I', 'have', 'got', '760-545-6789.', 'Okay.', 'That's', '+765-450-600-7089.', 'Correct?', 'What', 'is', 'your', 'identification', 'number?', '774-589-6589', '665', 'okay.', 'I', 'have', '+774-580-960-5896', '65', 'and', 'what', 'is', 'your', 'name', 'sir?', 'It', 'is', 'Robert.', 'Appelbaum.', 'Okay.', 'I', 'have', 'Robert', 'Applebaum', 'yet.', 'And', 'what', 'is', 'your', 'date', 'of', 'birth', 'Mr.', 'Appelbaum,', 'July', '7th,', '1974.', 'Okay,', 'July', '7th,', '1974.', 'Yes,', 'and', 'your', 'phone', 'number.', 'It', 'is', '610-265-1715.', 'Okay,', 'I', 'have', '610-265-1715.', 'Yes.', 'Okay,', 'Mr.', 'Appelbaum.', 'I', 'have', 'just', 'this', 'pended', 'your', 'card.', 'If', 'it', 'is', 'in', 'the', 'machine,', 'we', 'will', 'contact', 'you', 'as', 'lift', 'the', 'suspension', '00.', 'Thank', 'you,', 'sir.', 'Thank', 'you.', '']
```

รูปที่ 4.10 ตัวอย่างการแปลงข้อมูลเสียงให้อยู่ในรูปแบบข้อความโดยใช้ Cloud Speech to Text

รูปที่ 4.10 แปลงให้อยู่ในรูปแบบของ Dictionary และสร้างคีย์ที่ชื่อว่า transcript เพื่อเก็บข้อความในบทสนทนาทั้งหมด ในส่วนของโทเคนคำมีการสร้างคีย์ที่ชื่อว่า values เพื่อเก็บค่าของเวลาที่เริ่มพูดโทเคน (start) เวลาสิ้นสุด (end) และโทเคน (word)

นอกจากนี้ มีการประเมินผลความแม่นยำในการทำนายของแบบจำลอง โดยนำข้อมูลบทสนทนาจริงเทียบกับข้อมูลที่แบบจำลองทำนายโดยใช้ Jaccard's Coefficient Similarity ดังนี้

```
'Hello, you have called virtual bank, this is Linda speaking. How may I help you? Hi Linda. I was just at your Vill e branch and I think I left my Debit c ard in the ATM machine. Okay. Do you h ave your Debit card number? I don't ha ve. Okay, well do you have the checkin g account number associated with the D ebit card? That I do have. Are you rea dy? I will give you what I have got. 7 65-456-789. Okay. That's 765-456-789. Correct. What is your identification n umber? 774-589-658-9665. Okay, I have 774-589-658-9665 and what is your name sir? It is Robert Applebaum. Okay. I h ave Robert Applebaum. Yes. And what is your date of birth Mr. Applebaum? July 7th, 1974. Okay. July 7th, 1974. Yes. And your phone number? It is 610-265-1 715. Okay. I have 610-2651715. Yes. Ok ay Mr. Applebaum. I have just suspende d your card. If it is in the machine, we will contact you and lift the suspe nsion. Oh, thank you, Sure. Thank yo u.'
```

รูปที่ 4.11 ข้อมูลบทสนทนาจริง

```
"Hello, you have called virtual bank. This is Linda speaking. How may I help you? Hi Linda. I was just at your bill branch and I think I left my debit card in the ATM machine. Okay. Do you have your debit card number? I don't k now. Okay. Well, do you have the checking acc ount number associated with the debit card, b ut I do have are you ready? I will give you w hat I have got 760-545-6789. Okay. That's +76 5-450-600-7089. Correct? What is your identif ication number? 774-589-6589 665 okay. I have +774-580-960-5896 65 and what is your name si r? It is Robert. Appel board. Okay.I have Rob ert Applebaum yet. And what is your date of b irth Mr. Appelbaum, July 7th, 1974. Okay, Jul y 7th, 1974. Yes, and your phone number. It i s 610-265-1715. Okay, I have 610-265-1715. Ye s. Okay, Mr. Appelbaum. I have just this pend ed your card. If it is in the machine, we wil l contact you as lift the suspension 00. Than k you, sir. Thank you."
```

รูปที่ 4.12 บทสนทนาที่แบบจำลองทำนาย

| Algorithm | Accuracy (%) |
|----------------------|--------------|
| Cloud Speech to Text | 57.02 |

ตารางที่ 4.1 ตารางแสดงค่าความแม่นยำของการแปลงเสียงพูดให้อยู่ในรูปของข้อความ

จากตารางที่ 4.1 ความแม่นยำในการทำนายคำพูดของแบบจำลองคิดเป็นร้อยละ 57.02 ซึ่งเป็นค่าความแม่นยำที่ไม่สูงนัก แต่เมื่อเปรียบเทียบจากข้อมูลบทสนทนาจริง และข้อมูลบทสนทนาที่ทำการทำนายออกมาจากรูปที่ 4.11 และรูปที่ 4.12 สังเกตได้ว่า สิ่งที่ส่งผลให้ค่าความแม่นยำของแบบจำลองไม่สูงนั้นส่วนใหญ่ขึ้นอยู่กับเครื่องหมายวรรคตอนของข้อมูลบทสนทนาจริงและข้อมูลบทสนทนาที่แบบจำลองทำนายออกมา ดังนั้น ทางผู้จัดทำจึงสร้างฟังก์ชันตัดเครื่องหมายวรรคตอนทั้งในข้อมูลบทสนทนาจริงและบทสนทนาที่แบบจำลองทำนาย เพื่อประเมินผลค่าความแม่นยำใหม่ ดังรูปที่ 4.13 และ 4.14 และ ตารางที่ 4.2

```
'Hello you have called virtual bank this is Linda speaking How may I help you? Hi Linda I was just at your Ville branch and I think I left my Debit card in the ATM machine Okay Do you have your Debit card number? I dont have Okay well do you have the checking account number associated with the Debit card? That I do have Are you ready? I will give you what I have got 765456789 Okay Thats 765456789 Correct What is your identification number? 7745896589665 Okay I have 7745896589665 and what is your name sir? It is Robert Applebaum Okay I have Robert Applebaum Yes And what is your date of birth Mr Applebaum? July 7th 1974 Okay July 7th 1974 Yes And your phone number? It is 6102651715 Okay I have 6102651715 Yes Okay Mr Applebaum I have just suspended your card If it is in the machine we will contact you and lift the suspension Oh thank you Sure Thank you '
```

รูปที่ 4.13 บทสนทนาจริงที่ผ่านการทำความสะอาด

```
'Hello you have called virtual bank Th
is is Linda speaking How may I help yo
u? Hi Linda I was just at your bill br
anch and I think I left my debit card i
n the ATM machine Okay Do you have yo
ur debit card number? I dont know Okay
Well do you have the checking account n
umber associated with the debit card bu
t I do have are you ready? I will give
you what I have got 7605456789 Okay T
hats 7654506007089 Correct? What is yo
ur identification number? 7745896589 66
5 okay I have 7745809605896 65 and wha
t is your name sir? It is Robert Appel
board Okay I have Robert Applebaum yet
And what is your date of birth Mr Appe
lbaum July 7th 1974 Okay July 7th 1974
Yes and your phone number It is 610265
1715 Okay I have 6102651715 Yes Okay
Mr Appelbaum I have just this pended
your card If it is in the machine we w
ill contact you as lift the suspension
00 Thank you sir Thank you '
```

รูปที่ 4.14 บทสนทนาที่แบบจำลองทำนายที่ผ่านการทำความสะอาด

| Algorithm | Accuracy (%) |
|----------------------|--------------|
| Cloud Speech to Text | 71.43 |

ตารางที่ 4.2 ตารางแสดงค่าความแม่นยำของการแปลงเสียงพูดให้อยู่ในรูปของข้อความ (ใหม่)

จากตารางที่ 4.2 ความแม่นยำในการทำนายคำพูดของแบบจำลองคิดเป็นร้อยละ 71.43 สามารถเห็นได้ชัดว่าค่าความแม่นยำสูงขึ้นอย่างชัดเจน เมื่อดำเนินการตัดเครื่องหมายวรรคตอนออกเบื้องต้น

4.3 การตรวจจับคำที่เป็นข้อมูลส่วนบุคคลจากข้อมูลรูปแบบข้อความ

4.3.1 การทดลองทำนายคำที่เป็นนิพจน์ระบุนามของทั้ง 3 ไลบรารีและเลือกคำทำนายจริง

ผู้จัดทำได้ทดลองนำข้อมูลในรูปแบบข้อความที่เป็นผลลัพธ์จากการทำ Cloud Speech to Text เข้าสู่กระบวนการตรวจจับนิพจน์ระบุนามทั้ง 3 ฟังก์ชัน ของไลบรารี Stanford NER, NLTK และ spaCy จากนั้นแสดงผลลัพธ์ในรูปแบบตาราง ดังรูปที่ 4.15

| word | start_time | end_time | stanford_pred | nltk_pred | spacy_pred |
|-----------|------------|----------|---------------|-----------|------------|
| Hello, | 0.0 | 0.4 | DATE | LOCATION | O |
| you | 0.4 | 1.2 | O | O | O |
| have | 1.2 | 1.3 | O | O | O |
| called | 1.3 | 1.8 | O | O | O |
| virtual | 1.8 | 2.2 | O | O | O |
| bank. | 2.2 | 2.4 | O | O | O |
| This | 2.4 | 3.2 | O | O | O |
| is | 3.2 | 3.4 | O | O | O |
| Linda | 3.4 | 3.8 | PERSON | PERSON | PERSON |
| speaking. | 3.8 | 4.3 | O | O | O |
| How | 4.3 | 5.3 | O | O | O |
| may | 5.3 | 5.3 | O | O | O |
| I | 5.3 | 5.5 | O | O | O |
| help | 5.5 | 5.7 | O | O | O |
| you? | 5.7 | 5.9 | O | O | O |
| Hi | 6.2 | 6.8 | O | O | O |
| Linda. | 6.8 | 7.2 | PERSON | PERSON | PERSON |
| I | 7.2 | 8.0 | O | O | O |
| was | 8.0 | 8.2 | O | O | O |
| just | 8.2 | 8.3 | O | O | O |

รูปที่ 4.15 ตารางการทำนายประเภทของนิพจน์ระบุนามของ 3 ไบรารี

รูปที่ 4.15 เก็บค่าการทำนายของโทเค็นทุก ๆ คำ ไว้ในตารางเดียวกันตามประเภทของนิพจน์ระบุนาม หากในแถวใดมีการทำนายเป็นคำว่า “O” หมายความว่าโทเค็นนั้นไม่ได้เป็นนิพจน์ระบุนาม ซึ่งมีการเก็บค่าการทำนายทั้งหมด 3 คอลัมน์ ได้แก่ stanford_pred คือ ค่าที่ Stanford NER ทำนาย nltk_pred คือ ค่าที่ NLTK ทำนาย และ spacy_pred คือ ค่าที่ spaCy ทำนาย

นอกจากนี้ ผู้จัดทำได้มีการประเมินผลการทำนายนิพจน์ระบุนามแต่ละประเภท เพื่อใช้ในการพิจารณาวิธีเลือกค่าทำนายจริง โดยมีการพิจารณาจากค่า Recall เท่านั้น แบ่งตามนิพจน์ระบุนาม 5 ประเภท ดังตารางต่อไปนี้

| Library | Precision (%) | Recall (%) | F1-score (%) | Accuracy (%) |
|--------------|---------------|------------|--------------|--------------|
| Stanford NER | 100 | 76.7 | 86.8 | 99.2 |
| NLTK | 33.3 | 60 | 42.9 | 98.7 |
| spaCy | 93.8 | 100 | 96.8 | 100 |

ตารางที่ .. ตารางการแสดงความแม่นยำของการติดแท็กคำว่า “PERSON”

จากตารางที่ 4.1 spaCy มีค่า Recall สูงที่สุด คิดเป็นร้อยละ 100 หมายความว่าไลบรารีนี้สามารถทำนายนิพจน์ระบุนามประเภท PERSON ได้ดีที่สุด รองลงมาคือ Stanford NER มีค่า Recall คิดเป็นร้อยละ 76.7 และสุดท้ายคือ NLTK มีค่า Recall คิดเป็นร้อยละ 60

| Library | Precision (%) | Recall (%) | F1-score (%) | Accuracy (%) |
|--------------|---------------|------------|--------------|--------------|
| Stanford NER | 28.6 | 100 | 44.4 | 100 |
| NLTK | 0 | 0 | 0 | 99.8 |
| spaCy | 33.3 | 100 | 50 | 100 |

ตารางที่ .. ตารางการแสดงความแม่นยำของการติดแท็กคำว่า “ORGANIZATION”

จากตารางที่ 4.2 Stanford NER และ spaCy มีค่า Recall สูงที่สุด คิดเป็นร้อยละ 100 หมายความว่าไลบรารีนี้สามารถทำนายนิพจน์ระบุนามประเภท ORGANIZATION ได้ดีที่สุด แต่ NLTK ไม่สามารถทำนายนิพจน์ระบุนามประเภทนี้ได้เลย

| Library | Precision (%) | Recall (%) | F1-score (%) | Accuracy (%) |
|--------------|---------------|------------|--------------|--------------|
| Stanford NER | 40 | 33.3 | 36.4 | 99.6 |
| NLTK | 20 | 33 | 25 | 99.6 |
| spaCy | 0 | 0 | 0 | 99.3 |

ตารางที่ .. ตารางการแสดงความแม่นยำของการติดแท็กคำว่า “LOCATION”

จากตารางที่ 4.3 ทั้ง 3 ไบรารีนี้ยังไม่สามารถทำนายนิพจน์ระบุนามประเภท LOCATION ได้อย่างแม่นยำ แต่เมื่อเรียงลำดับค่า Recall ของแต่ละไบรารี สรุปได้ว่า Stanford NER มีค่า Recall สูงที่สุด คิดเป็นร้อยละ 33.3 รองลงมาคือ NLTK มีค่า Recall คิดเป็นร้อยละ 33 แต่ spaCy ไม่สามารถทำนายนิพจน์ระบุนามประเภทนี้ได้เลย

| Library | Precision (%) | Recall (%) | F1-score (%) | Accuracy (%) |
|--------------|---------------|------------|--------------|--------------|
| Stanford NER | 47.1 | 88.9 | 61.5 | 99.9 |
| NLTK | 0 | 0 | 0 | 99 |
| spaCy | 52.9 | 100 | 69.2 | 100 |

ตารางที่ .. ตารางการแสดงความแม่นยำของการติดแท็กคำว่า “DATE”

จากตารางที่ 4.4 spaCy มีค่า Recall สูงที่สุด คิดเป็นร้อยละ 100 รองลงมาคือ Stanford NER มีค่า Recall คิดเป็นร้อยละ 88.9 แต่ NLTK ไม่สามารถทำนายนิพจน์ระบุนามประเภทนี้ได้เลย

| Library | Precision (%) | Recall (%) | F1-score (%) | Accuracy (%) |
|--------------|---------------|------------|--------------|--------------|
| Stanford NER | 100 | 100 | 100 | 100 |
| NLTK | 0 | 0 | 0 | 99.6 |
| spaCy | 100 | 100 | 100 | 100 |

ตารางที่ .. ตารางการแสดงความแม่นยำของการติดแท็กคำว่า “MONEY”

จากตารางที่ 4.5 Stanford NER และ spaCy มีค่า Recall สูงที่สุด คิดเป็นร้อยละ 100 แต่ NLTK ไม่สามารถทำนายนิพจน์ระบุนามประเภทนี้ได้เลย

จากตารางทั้งหมดที่ได้นำเสนอ พบว่าแต่ละไลบรารีมีประสิทธิภาพในการทำนายนิพจน์ระบุนามต่ำ และสูงแตกต่างกัน สังเกตได้ว่าการทำนายนิพจน์ระบุนามแต่ละประเภท ค่า Recall ที่ได้จากการทำนายนั้นมักจะมีค่าในระดับที่เท่า ๆ กันประมาณ 2 ไบรารีเสมอ ดังนั้น ผู้จัดทำจึงทดลองพัฒนาประสิทธิภาพของการทำนายประเภทของนิพจน์ระบุนามโดยใช้แนวคิดของ Majority Voting คือ เมื่อมีผลการทำนายของแต่ละไลบรารีที่เหมือนกันตั้งแต่ 2 ไบรารีขึ้นไปนับว่าเป็นค่าทำนายจริง (แต่ในการทดลองนี้ยังไม่ได้นำประเภทของเลขที่เป็นข้อมูลส่วนบุคคลมาใส่ในค่าทำนายจริง เนื่องจากต้องการ

ทดลองให้การทำนายนิพจน์ระบุนามทั้ง 5 ประเภทแรกนั้นมีค่า Recall ที่สูงขึ้นก่อน จากนั้นจึงนำค่าทำนายจริงที่ดีที่สุดรวมกับค่าของเลขที่เป็นข้อมูลส่วนบุคคล) ดังการทดลองที่ 1

- การทดลองที่ 1

การทดลองที่ 1 คือ การทดลองการเลือกค่าทำนายจริงจากการใช้แนวคิด Majority Voting) ได้ผลลัพธ์ดังตารางที่ 4.6

| Library | Precision (%) | Recall (%) | F1-score (%) | Accuracy (%) |
|--------------|---------------|------------|--------------|--------------|
| Stanford NER | 78 | 52 | 62.4 | 94.7 |
| NLTK | 31.2 | 26.7 | 28.8 | 88.9 |
| spaCy | 81.8 | 60 | 69.2 | 95.5 |
| Combined | 95.7 | 58.7 | 72.7 | 96.3 |

ตารางที่ 4.6 ตารางการแสดงความแม่นยำของการทำนายนิพจน์ระบุนาม 5 ประเภทจากการทดลองที่ 1

จากตารางที่ 4.6 ค่า Recall ที่ได้จากการทดลองที่ 1 (Combined) คิดเป็นร้อยละ 58.7 เมื่อเปรียบเทียบกับทุกไลบรารีจะเห็นว่าค่า Recall ของการทดลองนั้นต่ำกว่าไลบรารีของ spaCy (มีค่า Recall คิดเป็นร้อยละ 60) ซึ่งเมื่อลองกลับไปพิจารณาค่า Recall ของ spaCy เป็นรายประเภทพบว่านิพจน์ระบุนามประเภท PERSON และ DATE ไลบรารี spaCy มีค่า Recall สูงที่สุด โดยคิดเป็นร้อยละ 100 ทั้ง 2 ประเภท แต่เมื่อใช้แนวคิดของ Majority Voting กับทุกประเภท เมื่อพิจารณาประเภท PERSON และ DATE ของ Stanford NER ที่มีค่า Recall ประมาณร้อยละ 80 จึงเป็นผลให้ค่า Recall การทำนายนิพจน์ระบุนามทั้ง 5 ประเภทของค่าทำนายจริงต่ำกว่า spaCy ได้ เช่น หาก spaCy ทำนายว่าเป็น PERSON แต่ Stanford NER ไม่ได้ทำนายว่าเป็น PERSON หรือทำนายว่าเป็นประเภทอื่นที่ไม่เหมือนกันกับ NLTK ทำนาย จะส่งผลให้ค่า Recall ในประเภท PERSON ของการทดลองนี้ มีค่าอยู่ที่ประมาณร้อยละ 80 ทั้ง ๆ ที่ spaCy มีค่า Recall ในการทำนายประเภท PERSON คิดเป็นร้อยละ 100 ดังนั้น ผู้จัดทำจึงทดลองอีกครั้งโดยประยุกต์ใช้ทั้ง Majority Voting และ เลือกนิพจน์ระบุนามประเภท PERSON และ DATE เป็นค่าทำนายจริงทันที เมื่อ Stanford NER และ NLTK ไม่มีการทำนายประเภทที่เหมือนกัน แต่ spaCy ทำนายนิพจน์ระบุนามออกมาเป็น 2 ประเภทนั้น ดังการทดลองที่ 2

● การทดลองที่ 2

การทดลองที่ 2 คือ การทดลองจากการใช้ Majority Voting และเลือกค่าทำนายประเภท DATE และ PERSON เป็นค่าทำนายจริงจากการทำนายของ spaCy ในกรณีที่ Stanford NER และ NLTK ทำนายประเภทออกมาไม่เหมือนกัน ได้ผลลัพธ์ดังตารางที่ 4.7

| Library | Precision (%) | Recall (%) | F1-score (%) | Accuracy (%) |
|--------------|---------------|------------|--------------|--------------|
| Stanford NER | 78 | 52 | 62.4 | 94.7 |
| NLTK | 31.2 | 26.7 | 28.8 | 88.9 |
| spaCy | 81.8 | 60 | 69.2 | 95.5 |
| Combined | 82.5 | 62.7 | 71.2 | 95.7 |

ตารางที่ 4.7 ตารางการแสดงความแม่นยำของการทำนายนิพจน์ระบุนาม 5 ประเภทจากการทดลองที่ 2

จากการทดลองที่ 2 ถือว่าการทดลองที่ผู้จัดทำได้คิดค้นเพื่อพัฒนาประสิทธิภาพในการทำนายประเภทของนิพจน์ระบุนามนั้นเป็นผลสัมฤทธิ์ เนื่องจากมีค่า Recall สูงที่สุด ซึ่งคิดเป็นร้อยละ 62.7 ดังนั้น ผู้จัดทำจึงเลือกวิธีการจากการทดลองที่ 2 มาใช้ในการพัฒนาระบบการตรวจสอบนิพจน์ระบุนามในรูปแบบข้อความ

4.3.2 นำค่าทำนายจริงรวมกับค่าทำนายของเลขที่เป็นข้อมูลส่วนบุคคล

นำค่าทำนายจริงที่ได้จากการทดลองที่เป็นผลสัมฤทธิ์แล้วมารวมกับค่าทำนายของเลขที่เป็นข้อมูลส่วนบุคคล จากนั้นเก็บค่าเหล่านั้นให้อยู่ในรูปแบบตารางโดยเลือกเพียงโทเค็นที่มีการทำนายว่าเป็นนิพจน์ระบุนามเท่านั้น ดังรูปที่ 4.16

| | word | start_time | end_time | real_ents |
|------|--------------------|------------|----------|--------------|
| indx | | | | |
| 8 | Linda | 3.4 | 3.8 | PERSON |
| 16 | Linda. | 6.8 | 7.2 | PERSON |
| 34 | ATM | 11.7 | 11.8 | ORGANIZATION |
| 76 | 760-545-6789. | 27.2 | 30.6 | PIINUM |
| 79 | +765-450-600-7089. | 32.7 | 35.7 | PIINUM |
| 86 | 774-589-6589 | 38.9 | 42.7 | PIINUM |
| 91 | +774-580-960-5896 | 45.4 | 49.0 | PIINUM |
| 101 | Robert. | 51.9 | 52.3 | PERSON |
| 107 | Robert | 55.0 | 55.4 | PERSON |
| 108 | Applebaum | 55.4 | 56.0 | PERSON |
| 118 | Appelbaum, | 59.8 | 60.3 | PERSON |
| 119 | July | 60.3 | 61.6 | DATE |
| 120 | 7th, | 61.6 | 62.1 | DATE |
| 121 | 1974. | 62.1 | 63.5 | DATE |
| 123 | July | 64.9 | 66.0 | DATE |
| 124 | 7th, | 66.0 | 66.6 | DATE |
| 125 | 1974. | 66.6 | 68.3 | DATE |
| 133 | 610-265-1715. | 71.9 | 75.4 | PIINUM |
| 137 | 610-265-1715. | 77.4 | 80.7 | PIINUM |
| 141 | Appelbaum. | 83.1 | 83.6 | PERSON |

รูปที่ 4.16 ตารางค่าทำนายจริงทั้งหมดเฉพาะที่มีการติดแท็กนิพจน์ระบุนาม

นอกจากนี้ มีการประเมินผลความแม่นยำในการทำนายนิพจน์ระบุนามทั้ง 5 ประเภทแรก และประเภทของเลขที่เป็นข้อมูลส่วนบุคคล ดังตารางที่ 4.8

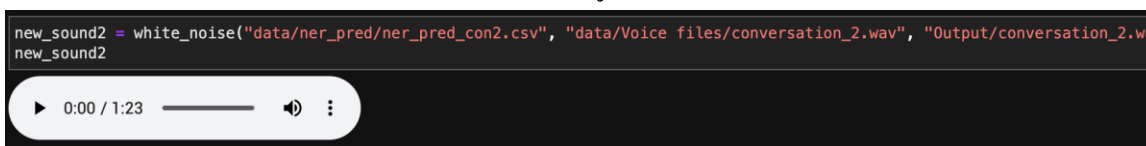
| Library | Precision (%) | Recall (%) | F1-score (%) | Accuracy (%) |
|-----------------------|---------------|------------|--------------|--------------|
| Stanford NER | 78 | 52 | 62.4 | 94.7 |
| NLTK | 31.2 | 26.7 | 28.8 | 88.9 |
| spaCy | 81.8 | 60 | 69.2 | 95.5 |
| Combined | 82.5 | 62.7 | 71.2 | 95.7 |
| Combined and Regex | 87.3 | 92 | 89.6 | 98.2 |

ตารางที่ 4.3 ตารางการแสดงความแม่นยำของการทำนายนิพจน์ระบุนามทุกประเภท

จากตารางที่ 4.8 สังเกตได้ว่าเมื่อรวมค่าทำนายจริง และสร้างเงื่อนไขทำนายเลขที่เป็นข้อมูลส่วนบุคคลจาก Regular Expressions นั้น (Combined and Regex) ส่งผลให้ค่า Recall ในการทำนายนิพจน์ระบุนามสูงที่สุด คิดเป็นร้อยละ 92 ซึ่งถือว่าการทดลองพัฒนาระบบที่ได้คิดค้นขึ้นนั้นเป็นผลสัมฤทธิ์

4.4 การแทนที่เสียงที่เป็นข้อมูลส่วนบุคคลด้วยเสียงรบกวน

นำข้อมูลผลลัพธ์ที่ได้จากกระบวนการก่อนหน้านี้มาแทนที่เสียงด้วยเสียงรบกวน จากนั้นบันทึกไฟล์เสียงผลลัพธ์ที่ได้เป็นไฟล์เสียงประเภท .wav ดังรูปที่ 4.18



รูปที่ 4.17 การบันทึกเสียงที่ผ่านการแทนที่เสียงที่เป็นข้อมูลส่วนบุคคลด้วยเสียงรบกวน

บทที่ 5

บทสรุป

5.1 สรุปผลโครงการ

การสร้างข้อมูลบทสนทนาในรูปแบบข้อความ เนื้อหาข้อมูลส่วนบุคคลของบทสนทนา ประกอบด้วย ชื่อ - นามสกุล เลขที่บัญชี เลขบัตรเดบิต หรือ เครดิต เลขบัตรประชาชน วันเกิด ที่อยู่ และ เบอร์โทรศัพท์ จากนั้นนำข้อมูลบทสนทนาในรูปแบบข้อความมาดำเนินการบันทึกเสียงและบันทึกเป็น ไฟล์เสียง เพื่อนำไปใช้ในการพัฒนาระบบ

ในส่วนของการแปลงเสียงพูดให้อยู่ในรูปแบบข้อความนั้น หากเป็นการประเมินผลโดยไม่คำนึงถึงความถูกต้องของเครื่องหมายวรรคตอน ถือว่าค่าของความแม่นยำอยู่ในระดับที่ดี อาจจะมีการแปลงชื่อบุคคลที่ไม่ตรงกับข้อมูลบทสนทนาจริงเล็กน้อย อาจเป็นสาเหตุมาจากเสียงที่ใช้ในการบันทึกเสียงแต่ละบุคคลมีสำเนียงการพูดไม่เหมือนกัน เช่น นามสกุล Applebaum เมื่อเป็นเสียงของ Siri Male แบบจำลองแปลงได้เป็น 2 โทเค็น คือ “Appel” และ “board.” แต่เมื่อเป็นเสียงของ Siri Female แบบจำลองกลับแปลงคำได้ถูกต้อง จึงสรุปได้ว่าบางครั้งสำเนียงการพูดของแต่ละตัวบุคคลอาจส่งผลต่อความแม่นยำของการแปลงข้อมูลเสียงให้อยู่ในรูปแบบข้อความ นอกจากนี้ ยังมีการแปลงผิดพลาดเล็กน้อย เช่น เมื่อสิริพูดว่า “oh” ในบางครั้งแบบจำลองจะแปลงเป็นเลข “0” ซึ่งส่งผลให้ความแม่นยำของแบบจำลองลดลง

ขั้นตอนต่อมาเป็นการตรวจจับคำที่เป็นข้อมูลส่วนบุคคลจากข้อมูลรูปแบบข้อความ ได้ทำการทดลอง ทั้งหมด 3 ไลบรารี ดังนี้

- Stanford NER สามารถติดแท็กบุคคลได้ค่อนข้างแม่นยำ และค่าเงิน ได้ค่อนข้างแม่นยำ ส่วนนิพจน์ระบุนามประเภทอื่น มีค่าความแม่นยำไม่ต่างจากไลบรารีอื่น ๆ แต่ในการติดแท็กวันที่ ด้วยข้อจำกัดของไลบรารีที่ไม่มีการติดแท็กตัวเลขที่เป็นประเภท Cardinal เหมือนไลบรารีอีก 2 ไลบรารี จึงส่งผลให้มีการติดแท็กตัวเลขธรรมดา เป็นประเภทของวันที่ (Date) ส่งผลให้ความแม่นยำของไลบรารีลดลง
- NLTK แทบจะไม่สามารถติดแท็กประเภทของนิพจน์ระบุนามได้อย่างถูกต้อง แต่ในการติดแท็กสถานที่ NLTK สามารถติดแท็กได้แม่นยำเท่ากับ Stanford NER ซึ่งเมื่อมีการใช้แนวคิดของ Majority Voting จึงสามารถทำให้การติดแท็กสถานที่นั้นมีความ

แม่นยำกว่าเดิม สาเหตุที่ส่งผลให้ NLTK มักมีการติดแท็กที่ผิดพลาด คือ หากโทเค็นนั้น ๆ ขึ้นต้นด้วยตัวอักษรพิมพ์ใหญ่ เช่น คำว่า “Hello” ไลบรารีจะติดแท็กเป็นสถานที่ทันที นอกจากนี้ ไลบรารีนี้สามารถติดแท็กตัวเลขประเภท Cardinal ได้ดีที่สุด แต่เนื่องจากทางผู้จัดทำไม่ได้มุ่งเน้นติดแท็กตัวเลขจากไลบรารี จึงไม่ได้ส่งผลต่อความแม่นยำในส่วนนี้

- spaCy จากผลลัพธ์การประเมินผลความแม่นยำ จะสังเกตได้ว่าส่วนใหญ่แล้ว spaCy มักมีค่าความแม่นยำสูงในการติดแท็กโทเค็น แต่หากให้สรุปเป็นรายประเภท จะสามารถสรุปได้ว่า ไลบรารีนี้สามารถติดแท็กบุคคล วันที่ และค่าเงินได้ดีที่สุด แต่เนื่องจากการติดแท็กของไลบรารีนี้ยังมีความไม่แม่นยำอยู่บ้าง ทางผู้จัดทำจึงมีความเห็นว่าควรมีการเลือกค่าทำนายจริงโดยใช้วิธีการจากการทดลองที่ 2 ในการพัฒนาระบบเพื่อเพิ่มประสิทธิภาพในการติดแท็กนิพจน์ระบุนาม

นอกจากนี้ ในการตรวจจับข้อมูลส่วนบุคคลทั้ง 3 ไลบรารี ยังไม่สามารถตรวจจับข้อมูลของเลขบ้านเลขที่ให้เป็นประเภทของ LOCATION ได้ จึงไม่สามารถปกปิดบ้านเลขที่ในขั้นตอนสุดท้ายได้ แต่ในส่วนของการนำค่าทำนายจริงโดยใช้วิธีการจากการทดลองที่ 2 มารวมกับการทำนายเลขที่เป็นข้อมูลส่วนบุคคลนั้น มีความแม่นยำค่อนข้างสูง ซึ่งค่าของ Recall คิดเป็นร้อยละ 92 ถือเป็นค่าความแม่นยำที่น่าพึงพอใจ

การตรวจจับเลขที่เป็นข้อมูลส่วนบุคคลโดยใช้ Regular Expressions ก็มีความแม่นยำค่อนข้างสูงเช่นกัน แต่ในบางครั้งอาจไม่แม่นยำอย่างสมบูรณ์เนื่องจากรูปแบบการแปลงตัวเลขของ Cloud Speech to Text อาจแบ่งโทเค็นได้ไม่ตรงกับตัวเลขที่ควรจะเป็น เช่น เลขบัตรเดบิต หรือบัตรเครดิต 16 หลัก ทางแบบจำลองอาจมีรูปแบบการแปลงตัวเลขได้เพียงแค่ 13 หลัก แล้วจึงแบ่งเลขอีก 3 หลักหลังเป็นอีกโทเค็น ซึ่งในเงื่อนไขมักจะติดแท็กเลขที่มากกว่า 9 หลักขึ้นไปโดยไม่สนใจเครื่องหมายต่าง ๆ เช่น +111-111-111-1111 หรือ 111-111-1111 เป็นต้น แต่หากพิจารณาถึงภาพรวมของค่าความแม่นยำแล้ว นับว่าเป็นที่น่าพึงพอใจเช่นกัน

ในขั้นตอนสุดท้ายเป็นการแทนที่เสียงที่เป็นข้อมูลส่วนบุคคลด้วยเสียงรบกวน ในบางช่วงที่มีการแทนที่เสียงรบกวนนั้นอาจเกินหรือขาดจากระยะเวลาของคำพูดที่เป็นข้อมูลส่วนบุคคล แต่โดยภาพรวมแล้วถือว่าปิดบังคำพูดที่เป็นข้อมูลส่วนบุคคลได้ดี

5.2 ปัญหาในการพัฒนาโครงการและสรุปผล

5.2.1 ปัญหาในการพัฒนาโครงการ

- 1) ปัญหาในการเตรียมข้อมูล โครงการนี้ไม่สามารถหาชุดข้อมูลสาธารณะในการนำไปพัฒนาระบบได้เนื่องจากข้อมูลที่ใช้ดำเนินงานนั้นเป็นข้อมูลส่วนบุคคล ผู้จัดทำจึงจำเป็นต้องสร้างชุดข้อมูลขึ้นเองเพื่อพัฒนาระบบนี้
- 2) ปัญหาในการพัฒนาระบบหลัก ๆ คือ ความแม่นยำของการแปลงข้อมูลเสียงให้อยู่ในรูปแบบข้อความ หากบางบทสนทนามีค่าความแม่นยำที่ไม่สูงมากพอ เมื่อนำข้อมูลเหล่านั้นเข้าสู่กระบวนการตรวจจับคำที่เป็นข้อมูลส่วนบุคคลจากข้อมูลรูปแบบข้อความ ส่งผลให้โปรแกรมไม่สามารถคิดแท็กประเภทของโทเคนที่ควรจะมีพจน์ระบุนามได้ เช่น ชื่อบุคคล หรือ ส่วนเล็ก ๆ ของเลขที่เป็นข้อมูลสำคัญ ส่งผลให้เป็นปัญหาต่อการปิดบังคำที่เป็นข้อมูลส่วนบุคคลในขั้นตอนสุดท้ายได้
- 3) การแทนที่เสียงที่เป็นข้อมูลส่วนบุคคลด้วยเสียงรบกวน ในบางช่วงของการแทนที่เสียงรบกวน อาจเกินหรือขาดจากระยะเวลาของคำพูดที่เป็นข้อมูลส่วนบุคคล

5.2.2 สรุปผล

จากการพัฒนาโครงการนี้ สามารถตอบโจทย์วัตถุประสงค์ที่ผู้จัดทำได้คาดหวังไว้ ดังนี้

- 1) เข้าใจการทำงานของกระบวนการประมวลผลภาษาธรรมชาติ (Natural Language Processing) ในการประยุกต์ใช้กับการตรวจจับข้อมูลส่วนบุคคล
- 2) เข้าใจการทำงานของระบบการรู้จำเสียงพูด (Speech Recognition) และแบบจำลองที่ได้เลือกใช้
- 3) สามารถปกปิดข้อมูลสำคัญในรูปแบบเสียงโดยการแทรกเสียงรบกวนแทนที่เสียงข้อมูลส่วนบุคคลได้

5.3 แนวทางในการพัฒนาต่อ

- 1) วิทยานิพนธ์ฉบับนี้สามารถต่อยอดโดยการนำระบบที่พัฒนาไปใช้ในด้านของการปกปิดข้อมูลที่เป็นส่วนบุคคลในหน่วยงานและองค์กรที่ต้องการประยุกต์ใช้ระบบได้ทั้งในรูปแบบไฟล์เสียง และข้อมูลที่เป็นข้อความ
- 2) สามารถนำผลลัพธ์ที่ได้จากการพัฒนาระบบไปวิเคราะห์และพัฒนาต่อในด้านอื่น ๆ ได้ โดยไม่ต้องคำนึงถึงสิทธิส่วนบุคคลเนื่องจากมีการปกปิดข้อมูลส่วนบุคคลแล้ว

เอกสารอ้างอิง

- [1] ศ. สวัสดิ์พงษ์ธาดา, “ความเป็นส่วนตัว (Privacy)”, 2015. .
- [2] M. A. Pathak, *Privacy-preserving machine learning for speech processing*. Springer Science & Business Media, 2012.
- [3] T. Tamesue, S. Yamaguchi, และ T. Saeki, “Study on achieving speech privacy using masking noise”, *J. Sound Vib.*, ปี 297, ฉบับที่ 3–5, น. 1088–1096, 2006.
- [4] T. A. Faruque, S. Negi, และ L. V Subramaniam, “Protecting Sensitive Customer Information in Call Center Recordings”, ใน *2009 IEEE International Conference on Services Computing*, 2009, น. 81–88, doi: 10.1109/SCC.2009.51.
- [5] อ. สนั่นศิลป์, “Is the Infringement of Privacy Right and Personal Data of the Offender Treated as Social Sanction to the Offender in Accordance with Theories of Punishment?” .
- [6] “What is Speech Recognition? | IBM”. [ออนไลน์]. Available at: <https://www.ibm.com/cloud/learn/speech-recognition>.
- [7] K. Surapong, “Natural Language Processing (NLP) คืออะไร รวมคำศัพท์เกี่ยวกับ Natural Language Processing (NLP) – NLP ep.1”, 2018.
- [8] R. Alfred, L. C. Leong, C. K. On, และ P. Anthony, “Malay named entity recognition based on rule-based approach”, 2014.
- [9] “Natural Language Processing is Fun! | by Adam Geitgey | Medium”. [ออนไลน์]. Available at: <https://medium.com/@ageitgey/natural-language-processing-is-fun-9a0bff37854e>.
- [10] ต. รัฐภูมิ, “การสกัดความสัมพันธ์ระหว่างนิพจน์ระบุนามภาษาไทย”, 2552.
- [11] ศุภวัจน์, “การตรวจเทียบภายนอกหาลักลอกในงานวิชาการโดยใช้แบบจำลองซัพพอร์ตเวกเตอร์แมชชีนและ การวัดค่าความละม้ายของข้อความ”, 2560.

- [12] “การเปิดใช้งาน Cloud Speech API. Cloud Speech API... | by Flame Sillawat | Medium”. [ออนไลน์]. Available at: <https://medium.com/@flame.zaxaou/การเปิดใช้งาน-cloud-speech-api-6b7a05c72752>.
- [13] “Named Entity Recognition with Stanford NER Tagger”. [ออนไลน์]. Available at: <https://pythonprogramming.net/named-entity-recognition-stanford-ner-tagger/>.
- [14] “The Stanford Natural Language Processing Group”. [ออนไลน์]. Available at: <https://nlp.stanford.edu/software/CRF-NER.html>.
- [15] “Natural Language Toolkit — NLTK 3.5 documentation”. [ออนไลน์]. Available at: <https://www.nltk.org/>.
- [16] “spaCy 101: Everything you need to know · spaCy Usage Documentation”. [ออนไลน์]. Available at: <https://spacy.io/usage/spacy-101>.
- [17] “7. Extracting Information from Text”. [ออนไลน์]. Available at: <https://www.nltk.org/book/ch07.html>.
- [18] “Regular Expression Matching Can Be Simple And Fast”. [ออนไลน์]. Available at: <https://swtch.com/~rsc/regexp/regexpl.html>.
- [19] “How do MP3 and WAV Files Differ?” [ออนไลน์]. Available at: <https://www.dawsons.co.uk/blog/how-do-mp3-and-wav-files-differ>.
- [20] “MP4 to WAV online file converter”. [ออนไลน์]. Available at: <https://audio.online-convert.com/convert/mp4-to-wav>.