

การปกป้องข้อมูลส่วนบุคคล

PERSONALLY IDENTIFIABLE INFORMATION PROTECTION

ณัฐนิชา ชัยศิริพานิช

NATTANICHA CHAISIRIPANICH

ประวิตรานันท์ บุตรโพธิ์

PRAWITRANUN BUTPHO

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิทยาศาสตรบัณฑิต

สาขาวิชาวิทยาการข้อมูลและการวิเคราะห์เชิงธุรกิจ

คณะเทคโนโลยีสารสนเทศ

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ภาคเรียนที่ 1 ปีการศึกษา 2563

การปกป้องข้อมูลส่วนบุคคล
PERSONALLY IDENTIFIABLE INFORMATION PROTECTION

ณัฐธิดา ชัยศิริพานิช

ประวิตรนันท์ บุตรโพธิ์

อาจารย์ที่ปรึกษา

ดร. นนท์ คณิงสุขเกษม

รศ.ดร. ชีรพงศ์ ลีธานุภาพ

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิทยาศาสตรบัณฑิต

สาขาวิชาวิทยาการข้อมูลและการวิเคราะห์เชิงธุรกิจ

คณะเทคโนโลยีสารสนเทศ

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ภาคเรียนที่ 1 ปีการศึกษา 2563

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 การปกป้องข้อมูลส่วนบุคคล

PERSONALLY IDENTIFIABLE INFORMATION PROTECTION

ผู้จัดทำ

นางสาวณัฐธิดา ชัยศิริพานิช รหัสนักศึกษา 60070135

นางสาวประวีตรานันท์ บุตรโพธิ์ รหัสนักศึกษา 60070148

..... อาจารย์ที่ปรึกษา

(ดร. นนท์ คณิงสุกษม)

..... อาจารย์ที่ปรึกษา

(รศ.ดร. ชีรพงศ์ ลีลาอนุภาพ)

ใบรับรองใบโครงการ (PROJECT)

เรื่อง

การปกป้องข้อมูลที่ระบุตัวบุคคล

PERSONALLY IDENTIFIABLE INFORMATION PROTECTION

นางสาวณัฐธิดา ชัยศิริพานิช รหัสนักศึกษา 60070135

นางสาวประวีตรานันท์ บุตรโพธิ์ รหัสนักศึกษา 60070148

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

.....
(นางสาวณัฐธิดา ชัยศิริพานิช)

.....
(นางสาวประวีตรานันท์ บุตรโพธิ์)

หัวข้อโครงการ	การปกป้องข้อมูลที่ระบุตัวบุคคล		
นักศึกษา	นางสาวณัฐนิชา	ชัยศิริพานิช	รหัสนักศึกษา 60070135
	นางสาวประวีตรานันท์	บุตรโพธิ์	รหัสนักศึกษา 60070148
ปริญญา	วิทยาศาสตร์บัณฑิต		
สาขาวิชา	วิทยาการข้อมูลและการวิเคราะห์เชิงธุรกิจ		
ปีการศึกษา	2563		
อาจารย์ที่ปรึกษา	ดร. นนท์ คณิงสุขเกษม		
	รศ.ดร. ชีรพงศ์ ลีตานภาพ		

บทคัดย่อ

ในปัจจุบันข้อมูลต่าง ๆ ถือเป็นสิ่งที่สำคัญต่อการนำไปวิเคราะห์และพัฒนาประสิทธิภาพขององค์กรจากการประยุกต์ใช้ปัญญาประดิษฐ์ ซึ่งในการนำข้อมูลเหล่านั้นมาประยุกต์ใช้ มักมีข้อจำกัดในเรื่องของสิทธิส่วนบุคคล กล่าวคือ รายละเอียดข้อมูลส่วนใหญ่ที่องค์กรมีอยู่ หรือรวบรวมข้อมูลเพิ่มเติมนั้น มักจะมีข้อมูลส่วนบุคคลปะปนอยู่ด้วยโดยเฉพาะข้อมูลของลูกค้า หากยังมีข้อมูลเหล่านั้นปะปนอยู่ อาจส่งผลให้มีการลักลอบนำข้อมูลไปหาผลประโยชน์ในทางที่ไม่ถูกต้องได้ ดังนั้น ข้อมูลส่วนบุคคลของลูกค้าถือเป็นเรื่องที่ต้องพึงตระหนักเพื่อรักษาสิทธิส่วนบุคคล และรักษาความน่าเชื่อถือขององค์กร นอกจากนี้ ในการรวบรวมข้อมูลขององค์กรต่าง ๆ จากการบันทึกเสียงการสนทนาระหว่างลูกค้ากับศูนย์ให้บริการข้อมูลลูกค้าทางโทรศัพท์ของธนาคาร ก็ถือว่ามีข้อมูลส่วนบุคคลที่ระบุตัวตนของลูกค้าเป็นจำนวนมาก จึงมีความเสี่ยงสูงหากข้อมูลเหล่านั้นยังมีข้อมูลส่วนบุคคลอยู่ในกระบวนการประยุกต์ใช้ปัญญาประดิษฐ์ และไม่สามารถนำไปประยุกต์ใช้ได้หากยังมีข้อมูลเหล่านั้นอยู่

ทางผู้จัดทำได้เล็งเห็นถึงปัญหานี้และมีการประยุกต์ใช้ปัญญาประดิษฐ์ในการดำเนินการปิดคำที่เป็นข้อมูลส่วนบุคคลของลูกค้าจากไฟล์เสียงบทสนทนาระหว่างลูกค้ากับศูนย์บริการข้อมูลลูกค้าทางโทรศัพท์ โดยดำเนินการแปลงไฟล์เสียงพูดให้อยู่ในรูปแบบข้อความ จากนั้นวิเคราะห์คำที่เป็นข้อมูลส่วนบุคคลพร้อมกับเก็บค่าของระยะเวลาที่พูดในไฟล์เสียงนั้น จากนั้นดำเนินการแทนที่คำพูดที่เป็นข้อมูลส่วนบุคคลด้วยเสียงรบกวนในไฟล์เสียงเพื่อให้สามารถนำข้อมูลที่ผ่านมาการปิดข้อมูลส่วนบุคคลแล้วไปใช้ประโยชน์ในการวิเคราะห์ทางด้านอื่น ๆ ได้

Project Title	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 use to analyze and develop the efficiency of organizations from the application of Artificial Intelligence. 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 that information contains personal information in the application of Artificial Intelligence and we cannot apply it to use with that information attainable.

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

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้สำเร็จลุล่วงได้ด้วยความกรุณาและการสนับสนุนจาก
ดร. นนท์ คะนิงสุขเกษม และ รศ.ดร. ชีรพงศ์ ลีตานุกภาพ ที่ได้ช่วยชี้แนะใน
การศึกษาค้นคว้า แนะนำขั้นตอนการปฏิบัติงาน เสนอแนวทางในการแก้ปัญหา
หรืออุปสรรคที่พบเจอในขณะที่ยังผู้จัดทำกำลังพัฒนาโครงงานนี้ และแนะนำวิธี
จัดทำปริญญานิพนธ์จนสำเร็จลุล่วงด้วยดี

ขอขอบพระคุณคณาจารย์คณะเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยี
พระจอมเกล้าเจ้าคุณทหารลาดกระบังทุก ๆ ท่าน ที่ช่วยมอบวิชาความรู้และแนวคิด
ที่สามารถนำไปประยุกต์ใช้ในการปรับปรุงและพัฒนาโครงงานเพื่อให้โครงงานมี
ประสิทธิภาพที่ดีขึ้น สามารถนำไปพัฒนาการดำเนินงานในอนาคตได้

ขอขอบคุณอาจารย์ที่ปรึกษา เพื่อน และรุ่นพี่ในคณะเทคโนโลยีสารสนเทศ
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง และผู้ที่มีส่วนเกี่ยวข้อง
ในการให้คำปรึกษาการพัฒนาโครงงานทุก ๆ ท่าน ที่ได้ให้ความร่วมมือและให้การ
ช่วยเหลือที่ดีตลอดการจัดทำจนสามารถก่อให้เกิดเป็นปริญญานิพนธ์ฉบับนี้ได้

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 - VIII
บทที่ 1.....	1
บทนำ	1
1.1 ที่มาและความสำคัญ.....	1
1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา	3
1.3 ขอบเขตการพัฒนาโครงการ	3
1.4 ขั้นตอนการดำเนินงาน	4
1.5 ประโยชน์ที่คาดว่าจะได้รับ	6
บทที่ 2.....	7
แนวคิด และเทคโนโลยีที่เกี่ยวข้อง	7
2.1 แนวคิดที่เกี่ยวข้อง	7
2.2 เทคโนโลยีเกี่ยวข้อง	8

สารบัญ (ต่อ)

	หน้า
บทที่ 3.....	23
ขั้นตอน และวิธีการดำเนินงานวิจัย.....	23
3.1 การเตรียมข้อมูล (Data Preparation).....	23
3.2 การแปลงเสียงพูดให้อยู่ในรูปแบบข้อความ	30
3.3 การตรวจจับคำที่เป็นข้อมูลส่วนบุคคลจากข้อมูลรูปแบบข้อความ.....	35
3.4 การแทนที่เสียงที่เป็นข้อมูลส่วนบุคคลด้วยเสียงรบกวน	43
3.5 การประเมินผล (Evaluation)	43
บทที่ 4.....	44
ผลการดำเนินงานเบื้องต้น	44
4.1 การแปลงเสียงพูดให้อยู่ในรูปแบบข้อความ	44
4.2 การตรวจจับคำที่เป็นข้อมูลส่วนบุคคลจากข้อมูลรูปแบบข้อความ.....	47
4.3 การแทนที่เสียงที่เป็นข้อมูลส่วนบุคคลด้วยเสียงรบกวน	54
บทที่ 5.....	55
บทสรุป.....	55
5.1 สรุปผลโครงการ	55
5.2 ปัญหาในการทำโครงการและสรุปผล.....	56
5.3 แนวทางในการพัฒนาต่อ.....	57
รายการอ้างอิง.....	58

สารบัญรูปภาพ

หน้า

รูปที่ 1.1	ขั้นตอนการดำเนินงาน	4
รูปที่ 2.1	กระบวนการทำงานของ Google Cloud Speech API	9
รูปที่ 2.2	กระบวนการทำงานทั่วไปของการประมวลผลภาษาธรรมชาติ	10
รูปที่ 2.3	Pre-Trained Part-of-Speech Classification Model.....	12
รูปที่ 2.4	ผลลัพธ์ของการประมวลผลประโยคทั้งหมด	12
รูปที่ 2.5	รูปประโยคหลังการทำ Lemmatization.....	13
รูปที่ 2.6	การระบุ Stop words.....	14
รูปที่ 2.7	การแยกการวิเคราะห์การพึ่งพา	14
รูปที่ 2.8	การคาดเดาประเภทของความสัมพันธ์	15
รูปที่ 2.9	รูปประโยคก่อนการทำการจับกลุ่มคำนาม	15
รูปที่ 2.10	รูปประโยคหลังจากการจับกลุ่มคำนาม.....	16
รูปที่ 2.11	คำนามของประโยค.....	16
รูปที่ 2.12	ประโยคจากการใช้ NER Tagging Model	16
รูปที่ 2.13	การทำ Coreference Resolution.....	17
รูปที่ 3.1	แผนงานการปกป้องข้อมูลที่ระบุตัวบุคคล	23
รูปที่ 3.2	ตัวอย่างบทสนทนาระหว่างลูกค้ากับศูนย์บริการข้อมูลลูกค้าทางโทรศัพท์	24
รูปที่ 3.3	รายละเอียดการวิเคราะห์ประโยค.....	24
รูปที่ 3.4	รายละเอียดการวิเคราะห์คำ.....	25
รูปที่ 3.5	รายการของเครื่องหมายวรรคตอนและ Stop words	25
รูปที่ 3.6	รายละเอียดการวิเคราะห์ความถี่ของคำที่ไม่ซ้ำกัน	26
รูปที่ 3.7	การแจกแจงความถี่ของคำที่ซ้ำกันของบทสนทนาจริง	26
รูปที่ 3.8	การแจกแจงความถี่ของคำที่ซ้ำกันของบทสนทนาที่ผ่านการทำความสะอาด	27
รูปที่ 3.9	รายการชื่อเสียงพูดที่ใช้ในการบันทึกเสียงบทสนทนา	28

สารบัญรูปภาพ (ต่อ)

หน้า

รูปที่ 3.10 ตัวอย่างไฟล์เสียงที่บันทึกจากสมาร์ทโฟน	29
รูปที่ 3.11 เปิดใช้งาน Cloud Storage	30
รูปที่ 3.12 เปิดใช้งาน Cloud Speech To Text	31
รูปที่ 3.13 อัปโหลดไฟล์เสียงขึ้นบน Cloud Storage.....	32
รูปที่ 3.14 การนำเข้าข้อมูลเสียงและกำหนดค่าต่าง ๆ	32
รูปที่ 3.15 ผลลัพธ์จากการทำฟังก์ชัน print_word_offsets	33
รูปที่ 3.16 ผลลัพธ์จากการทำฟังก์ชัน print_sentences.....	33
รูปที่ 3.17 ฟังก์ชันการแปลงเสียงพูดให้อยู่ในรูปแบบของข้อความตัวอักษร.....	34
รูปที่ 3.18 การบันทึกไฟล์ข้อความเป็นไฟล์ประเภท .json.....	34
รูปที่ 3.19 ฟังก์ชันการทำนายนิพจน์ระบุนามของ Stanford NER	36
รูปที่ 3.20 ฟังก์ชันการทำนายนิพจน์ระบุนามของ NLTK.....	38
รูปที่ 3.21 ฟังก์ชันการทำนายนิพจน์ระบุนามของ spaCy	39
รูปที่ 3.22 ฟังก์ชันการเลือกการทำนายประเภทนิพจน์ระบุนามที่เหมือนกัน 2 ใน 3	40
รูปที่ 3.23 การสร้างนิพจน์ระบุนามใหม่โดยใช้ Regular Expressions	41
รูปที่ 3.24 รวมการทำนาย Regular Expression และค่าทำนายจริงเข้าด้วยกัน	42
รูปที่ 3.25 ฟังก์ชันการแทนที่เสียงที่เป็นข้อมูลส่วนบุคคลด้วยเสียงรบกวน	43
รูปที่ 4.1 ตัวอย่างการแปลงข้อมูลเสียงให้อยู่ในรูปแบบข้อความโดยใช้ Cloud Speech to Text.....	44
รูปที่ 4.2 ข้อมูลบทสนทนาจริง	45
รูปที่ 4.3 บทสนทนาที่แบบจำลองทำนาย.....	45
รูปที่ 4.4 ค่าของความแม่นยำในการทำนาย	45
รูปที่ 4.5 บทสนทนาจริงที่ผ่านการทำความสะอาด	46
รูปที่ 4.6 บทสนทนาที่แบบจำลองทำนายที่ผ่านการทำความสะอาด	46
รูปที่ 4.7 ค่าของความแม่นยำในการทำนาย (ใหม่)	47

สารบัญรูปภาพ (ต่อ)

หน้า

รูปที่ 4.8 ตารางการทำนายประเภทของนิพจน์ระบุนาม	48
รูปที่ 4.9 ตารางค่าทำนายจริงเฉพาะที่มีการติดแท็กนิพจน์ระบุนาม	49
รูปที่ 4.10 การประเมินผลความแม่นยำของแต่ละแบบจำลอง	50
รูปที่ 4.11 การประเมินผลความแม่นยำในการติดแท็กคำว่า “PERSON”	51
รูปที่ 4.12 การประเมินผลความแม่นยำในการติดแท็กคำว่า “ORGANIZATION”	52
รูปที่ 4.13 การประเมินผลความแม่นยำในการติดแท็กคำว่า “LOCATION”	52
รูปที่ 4.14 การประเมินผลความแม่นยำในการติดแท็กคำว่า “DATE”	53
รูปที่ 4.15 รประเมินผลความแม่นยำในการติดแท็กคำว่า “MONEY”	53
รูปที่ 4.16 การประเมินผลความแม่นยำในการติดแท็กประเภทของ PII Number ทุกประเภท	54
รูปที่ 4.17 การบันทึกเสียงที่ผ่านการแทนที่เสียงที่เป็นข้อมูลส่วนบุคคลด้วยเสียงรบกวน	54

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 สำเนียง 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 ไม่ว่าจะเป็นข้อมูลที่ถูกค้าทำการกรอกลงในเว็บไซต์ ข้อมูลตำแหน่งที่อยู่ ก็ถือเป็นข้อมูลส่วนบุคคลที่ทางองค์กรธุรกิจต่าง ๆ สามารถนำไปซื้อและขายกันได้เช่นกัน

ในบางครั้ง การสนทนาเกี่ยวกับเรื่องความเป็นส่วนตัวในพื้นที่เปิด เช่น การสนทนาพูดคุยกันในคลินิกเล็ก ๆ ข้าง ๆ ห้องรอคิว การประชุมแลกเปลี่ยนความเห็นทางด้านภาษี ต่าง ๆ ในสำนักงาน การประชุมหาแนวทางปฏิบัติในการสอนในโรงเรียน ก็ถือว่ามีความเสี่ยงที่ข้อมูลเหล่านั้นจะรั่วไหลออกไปจากการที่มีบุคคลในห้องข้าง 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 มีความเสี่ยงที่จะถูกรุกล้ำความเป็นส่วนตัวของบุคคล การลักลอบนำข้อมูลไปแสวงหาผลประโยชน์โดยที่ไม่ได้รับอนุญาตจากเจ้าของข้อมูล และการรุกรล้ำความเป็นส่วนตัวของบุคคลจากการเก็บรวบรวมข้อมูลส่วนบุคคลของลูกค้าผ่านการสนทนากับทางศูนย์บริการข้อมูลลูกค้าทางโทรศัพท์ (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) โดยจะมีการทำการตรวจจับการสนทนาบางส่วนกับทางศูนย์บริการข้อมูลลูกค้าทางโทรศัพท์ (Call Center) โดยเฉพาะส่วนที่เป็นข้อมูลส่วนบุคคลของลูกค้า เช่น ชื่อ – นามสกุล วันเกิด เบอร์โทรศัพท์ เลขที่บัญชี และเลขหน้า

บัตรเครดิต หรือเดบิต ก่อนจะนำข้อมูลการสนทนาเหล่านั้นส่งต่อไปสู่กระบวนการวิเคราะห์เพื่อใช้ในกระบวนการทางธุรกิจอื่น ๆ โดยทางผู้จัดทำจะดำเนินการแปลงการสนทนานั้นให้อยู่ในรูปแบบข้อความ ตรวจสอบเนื้อหาของข้อความว่าคำใดมีรูปแบบที่เป็นข้อมูลส่วนบุคคล จากนั้นดำเนินการจับคู่คำกับเวลาในไฟล์บันทึกเสียง และดำเนินการปกปิดข้อความในส่วนนั้นออกไป

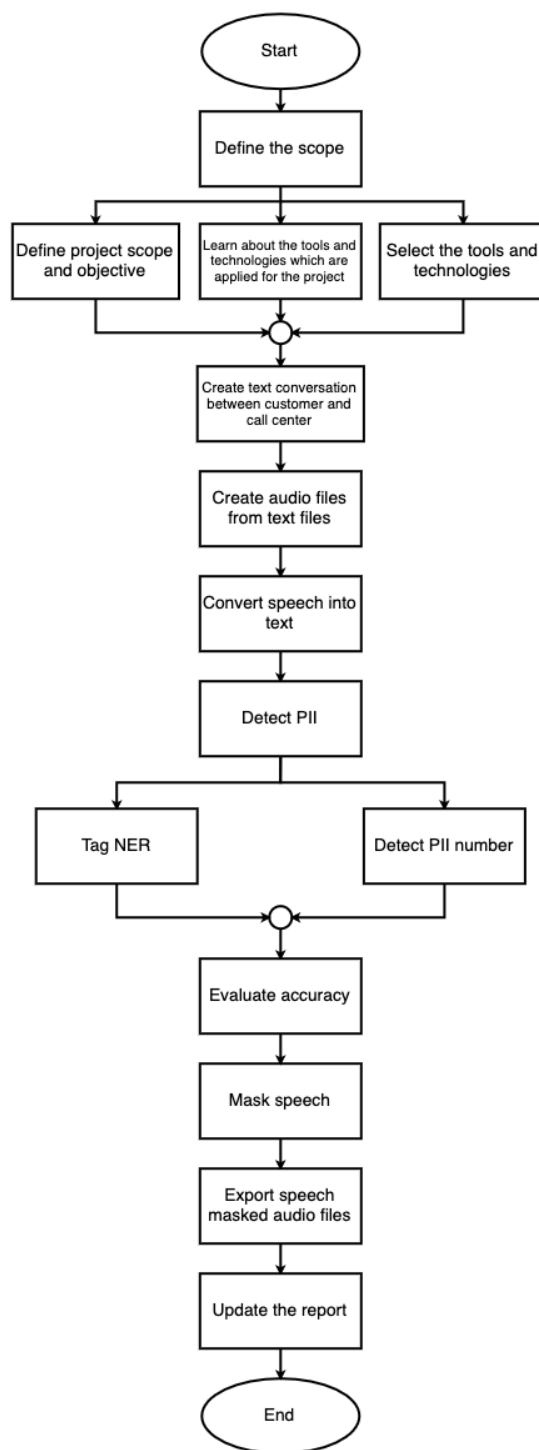
1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา

1. เพื่อศึกษากระบวนการประมวลผลภาษาธรรมชาติ (Natural Language Processing)
2. เพื่อศึกษารูปแบบของการรู้จำเสียงพูด (Speech Recognition)
3. เพื่อศึกษารายละเอียดของข้อมูลส่วนบุคคล
4. ดำเนินการปกปิดข้อมูลที่สำคัญในรูปแบบเสียงเพื่อเพิ่มความปลอดภัยในการนำข้อมูลเหล่านั้นไปดำเนินการวิเคราะห์ต่อ

1.3 ขอบเขตการพัฒนาโครงการ

1. ขอบเขตของชุดข้อมูล
 - 1) ดำเนินการสร้างชุดข้อมูลบทสนทนาระหว่างลูกค้ากับศูนย์บริการลูกค้าทางโทรศัพท์ในรูปแบบข้อความเป็นภาษาอังกฤษ
 - 2) สร้างไฟล์เสียงจากการนำชุดข้อมูลที่ได้สร้างขึ้นมานับันทึกเสียงโดยใช้รายชื่อเสียงของระบบปฏิบัติการ iOS ในการอ่านบทสนทนา 3 เสียง และ 2 เพศ ได้แก่ เพศหญิง 2 เสียง และเพศชาย 1 เสียง
2. ขอบเขตของการพัฒนาระบบ
 - 1) ผู้จัดทำได้เลือกใช้ The Cloud Speech To Text ในการแปลงเสียงพูดให้อยู่ในรูปแบบข้อความ (Speech-to-Text)
 - 2) นำ Stanford Named Entity Recognizer, Natural Language Toolkit (NLTK) และ spaCy มาประยุกต์ใช้ในการวิเคราะห์และประมวลผลข้อความเพื่อติดแท็กคำที่เป็นข้อมูลส่วนบุคคล
 - 3) สร้างเงื่อนไขในการตรวจจับข้อมูลส่วนบุคคลที่เป็นตัวเลขในบทสนทนาเพิ่มเติมโดยใช้ Regular Expressions
 - 4) ดำเนินการแทนที่เสียงที่เป็นข้อมูลส่วนบุคคลด้วยเสียงรบกวนโดยใช้ไลบรารีของ Pydub

1.4 ขั้นตอนการดำเนินงาน



รูปที่ 1.1 ขั้นตอนการดำเนินงาน

1.4.1 กำหนดขอบเขต

- กำหนดขอบเขตของโครงการและวัตถุประสงค์ของโครงการ
- ศึกษาเครื่องมือและเทคโนโลยีที่สามารถนำมาประยุกต์กับโครงการ
- เลือกเครื่องมือและเทคโนโลยีที่เหมาะสมกับโครงการ

1.4.2 สร้างข้อมูลบทสนทนาในรูปแบบข้อความ

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

1.4.3 นำข้อมูลบทสนทนาในรูปแบบข้อความมาดำเนินการบันทึกเสียงและบันทึกเป็นไฟล์เสียง

1.4.4 ดำเนินการพัฒนาแบบจำลองสำหรับการแปลงข้อมูลเสียงให้อยู่ในรูปแบบข้อความ

- หลังจากดำเนินการบันทึกเสียงข้อมูลที่สร้างขึ้นมาเองแล้ว จึงนำข้อมูลเสียงนั้นมาประยุกต์ใช้กับพัฒนาแบบจำลอง โดยการแปลงเสียงพูดให้อยู่ในรูปแบบข้อความ และสังเกตว่าแบบจำลองที่ทดลองมาสัมฤทธิ์ผลหรือไม่

1.4.5 ดำเนินการพัฒนาเครื่องมือของการตรวจจับข้อมูลส่วนบุคคล

- หลังจากแปลงข้อมูลเสียงให้อยู่ในรูปแบบของข้อความแล้ว จึงนำข้อความบทสนทนา นั้น ๆ มาวิเคราะห์และคิดแท็กคำที่เป็นข้อมูลส่วนบุคคล
- ดำเนินการสร้างเงื่อนไขเพิ่มเติมเพื่อตรวจจับเลขที่เป็นข้อมูลส่วนบุคคล

1.4.6 ประเมินผลความแม่นยำของเครื่องมือที่ใช้

- ประเมินผลความแม่นยำของการแปลงข้อมูลเสียงให้อยู่ในรูปแบบข้อความโดยเปรียบเทียบจากข้อมูลจริงในรูปแบบข้อความที่มีการสร้างขึ้นมาก่อนหน้านี้
- ประเมินผลความแม่นยำของเครื่องมือที่ใช้ในการตรวจจับข้อมูลที่ระบุตัวบุคคล

1.4.7 ดำเนินการแทนที่คำพูดที่เป็นข้อมูลส่วนบุคคลด้วยเสียงรบกวนจากไฟล์บันทึกเสียง

- หลังจากตรวจจับข้อมูลส่วนบุคคลในรูปแบบข้อความได้แล้ว จึงดำเนินการจับช่วงเวลาของคำนั้นในไฟล์เสียง และดำเนินการแทนที่เสียงด้วยเสียงรบกวน

1.4.8 บันทึกไฟล์เสียงที่ผ่านการแทนที่เสียงด้วยเสียงรบกวน

- เพื่อให้สามารถนำผลลัพธ์ที่ได้จากระบบไปดำเนินการวิเคราะห์ทางธุรกิจ หรือด้านอื่น ๆ ต่อไป

1.4.9 นำวิธีการดำเนินงานและผลการดำเนินงานที่ได้มาจัดทำรายงาน

1.5 ประโยชน์ที่คาดว่าจะได้รับ

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

บทที่ 2

แนวคิด และเทคโนโลยีที่เกี่ยวข้อง

2.1 แนวคิดที่เกี่ยวข้อง

2.1.1 สิทธิความเป็นอยู่ส่วนบุคคล

สิทธิความเป็นอยู่ส่วนบุคคล (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 เพื่อรักษาสิทธิความเป็นส่วนตัวของบุคคล เนื่องจากทุกครั้งที่เราทำธุรกรรมผ่านศูนย์บริการข้อมูลลูกค้าทางโทรศัพท์ ทางองค์กรจะทำการบันทึกการสนทนา ระหว่างเจ้าหน้าที่ศูนย์บริการข้อมูลลูกค้าทางโทรศัพท์ กับลูกค้า เพื่อนำข้อมูลที่ทางลูกค้าแจ้งไปวิเคราะห์ เพื่อแก้ไขปัญหา หรือ ประเมินศักยภาพขององค์กร

2.2 เทคโนโลยีเกี่ยวข้อง

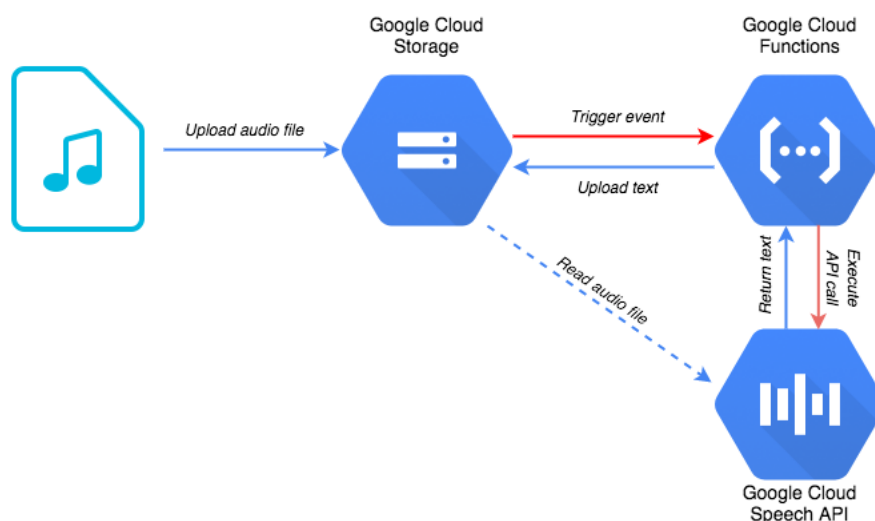
2.2.1 การรู้จำเสียงพูด (Speech Recognition)

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 สิริ (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) มีการประมวลผลข้อมูลที่มีการฝึกฝน (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

2.2.2 Cloud Speech to Text by Google Cloud



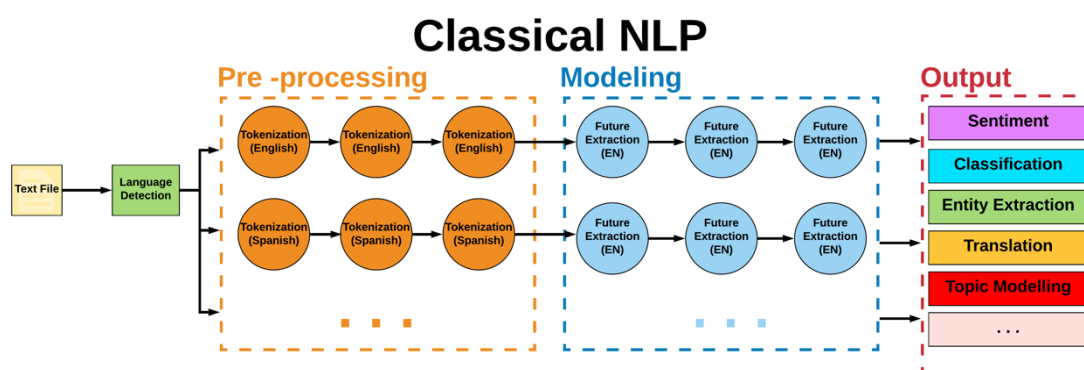
รูปที่ 2.1 กระบวนการทำงานของ Google Cloud Speech API

กูเกิลคลาวด์แพลตฟอร์มเป็นเว็บเซิร์ฟเวอร์ที่ให้บริการคลาวด์แพลตฟอร์มที่ถูกพัฒนาขึ้นโดยกูเกิล ซึ่งภายในกูเกิลคลาวด์แพลตฟอร์มนั้นมีบริการที่แยกย่อยอีกมากมายให้ตรงตามลักษณะการใช้งาน

เช่น Cloud Speech to Text, Cloud Storage, Compute Engine, Machine Learning และอื่น ๆ อีกมากมาย ทั้งนี้การใช้งานกูเกิลคลาวด์แพลตฟอร์มจะคิดค่าใช้จ่ายตามจำนวนการใช้งาน [7]

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

2.2.3 การประมวลผลภาษาธรรมชาติ (Natural Language Processing: NLP)



รูปที่ 2.2 กระบวนการทำงานทั่วไปของการประมวลผลภาษาธรรมชาติ

การประมวลผลภาษาธรรมชาติ คือ หนึ่งในสาขาของวิทยาศาสตร์คอมพิวเตอร์ที่เกี่ยวข้องกับปัญญาประดิษฐ์ (Artificial Intelligence) และภาษาศาสตร์คอมพิวเตอร์ (Computational Linguistics) เป็นศาสตร์ที่ศึกษาเกี่ยวกับการทำให้คอมพิวเตอร์สามารถสื่อสารโต้ตอบด้วยภาษาของมนุษย์ และทำให้คอมพิวเตอร์เข้าใจภาษามนุษย์มากขึ้น เช่น Siri, Google Assistant และ Alexa [8]

การประมวลผลภาษาธรรมชาติ เริ่มแรกเมื่อปลายปี ค.ศ. 1940 จากการใช้เครื่องมือการแปลเพื่อทำการถอดรหัสศัตรูในช่วงสงครามโลกครั้งที่ 2 เป็นครั้งแรก แต่อย่างไรก็ตาม งานวิจัยเกี่ยวกับการประมวลผลภาษาธรรมชาติก็ไม่ได้มีการสร้างขึ้นมาจนถึงปี ค.ศ. 1980 โดยการประมวลผลภาษาธรรมชาตินั้น มีสาขาวิชาหลากหลายด้านที่มีการนำเทคโนโลยีไปประยุกต์ใช้ เช่น การค้นคืนสารสนเทศ (Information Retrieval) การสกัดสารสนเทศ (Information Extraction) และการตั้งคำถาม – ตอบคำถาม (Question - Answering) [9]

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.”

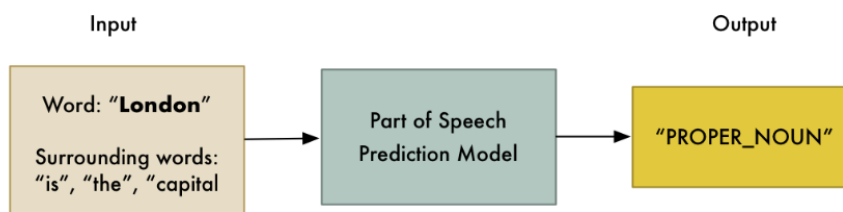
เมื่อทำการแยกคำแล้วจะได้ผลลัพธ์ ดังนี้

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

การทำ Tokenization ในภาษาอังกฤษนั้นสามารถทำได้ง่ายเนื่องจากจะมีการแยกคำทุกครั้งที่มีช่องว่างระหว่างคำเหล่านั้น โดยจะถือว่าเครื่องหมายวรรคตอนเป็นโทเคนแยก เนื่องจากเครื่องหมายวรรคตอนก็มีความหมายเช่นกัน

3) การทำนายส่วนต่าง ๆ ของคำพูดสำหรับในแต่ละโทเคน

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



รูปที่ 2.3 Pre-Trained Part-of-Speech Classification Model

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

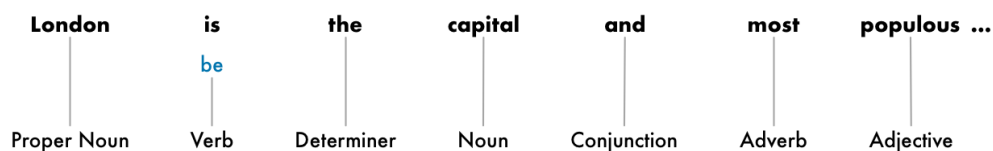
London	is	the	capital	and	most	populous ...
Proper Noun	Verb	Determiner	Noun	Conjunction	Adverb	Adjective

รูปที่ 2.4 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 การรู้รูปแบบพื้นฐานของคำแต่ละคำในประโยคนั้นมีประโยชน์อย่างมาก เพราะจะช่วยให้ทราบได้ว่าทั้งสองประโยคนั้นกำลังกล่าวถึงสิ่งที่เป็นแนว ๆ เดียวกัน มิฉะนั้นคำว่า “Pony” และ “Ponies” จะมีความหมายแตกต่างกันโดยสิ้นเชิงต่อคอมพิวเตอร์ สรุปได้ว่าในกระบวนการนี้จะเป็นการหารูปแบบที่เป็นพื้นฐานมากที่สุดในประโยค หลังจากทำการ Lemmatization เพิ่มในรูปแบบรากของคำกริยาจะมีลักษณะดังนี้

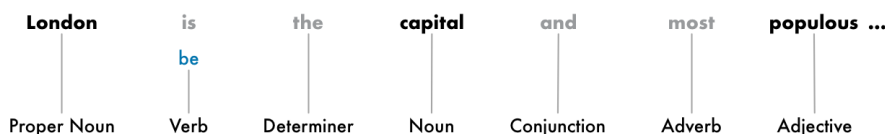


รูปที่ 2.5 รูปประโยคหลังการทำ Lemmatization

จากรูปที่ 2.5 จะสังเกตได้ว่าการเปลี่ยนแปลงเพียงที่เดียวคือ “is” เปลี่ยนเป็น “be”

5) การระบุ Stop words

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

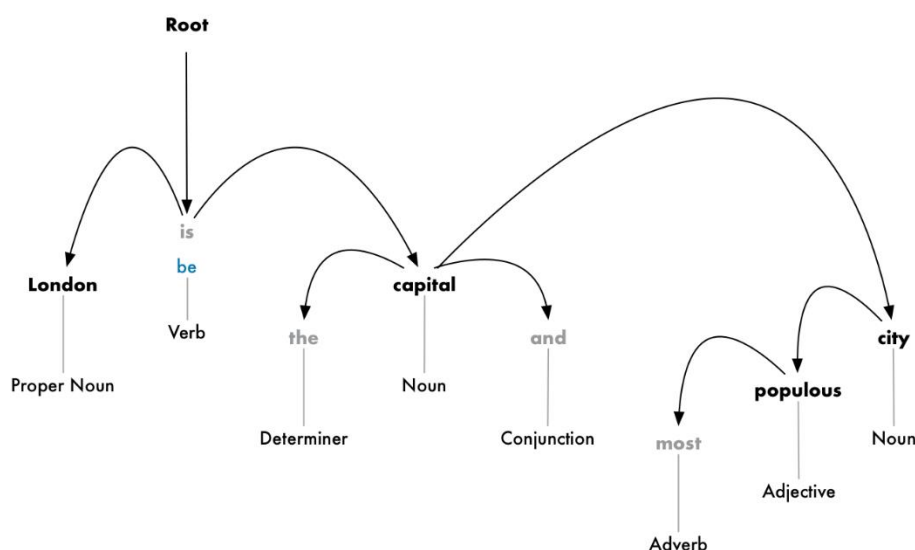


รูปที่ 2.6 การระบุ Stop words

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

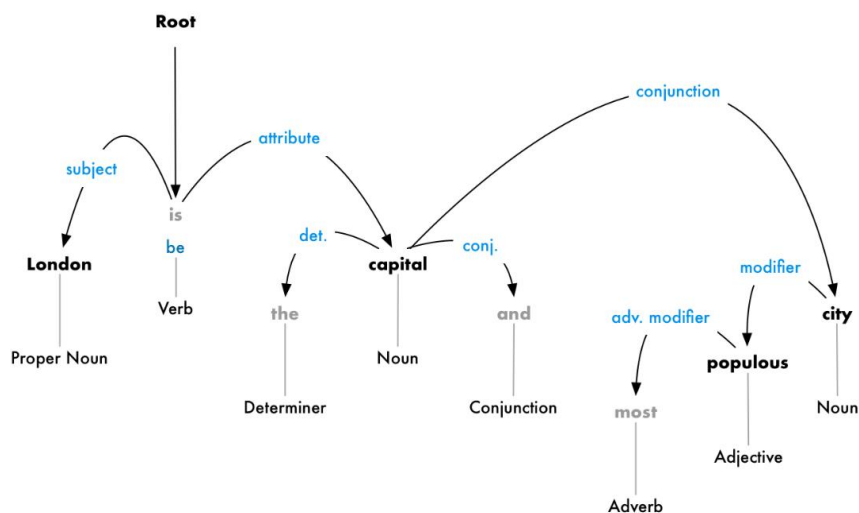
6) การแยกการวิเคราะห์การพึ่งพา (Dependency Parsing)

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



รูปที่ 2.7 การแยกการวิเคราะห์การพึ่งพา

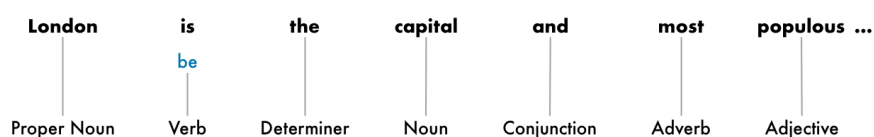
นอกจากนี้ ยังสามารถคาดเดาประเภทของความสัมพันธ์ที่มีอยู่ระหว่างสองคำนี้ได้ ดังรูปที่ 2.8



รูปที่ 2.8 การคาดเดาประเภทของความสัมพันธ์

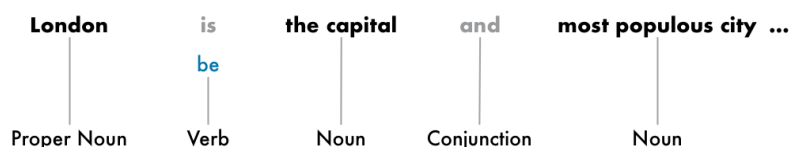
ต้นไม้นี้แสดงให้เห็นว่าหัวข้อของประโยคนั้นเป็นคำนามว่า “London” และมีความสัมพันธ์แบบ “be” กับ “Capital” ทำให้ทราบได้ว่า “ลอนดอนเป็นเมืองหลวง” ขั้นตอนที่มีการใช้ในบางครั้ง คือ การค้นหาคำนาม (Finding Noun Phrases)

นอกจากการทำ Dependency Parsing อย่างเดียวแล้ว ยังสามารถใช้ข้อมูลจาก Dependency Parse Tree ในการจับกลุ่มคำที่กำลังกล่าวถึงสิ่งเดียวกันได้โดยอัตโนมัติ ตัวอย่างเช่น แทนที่จะทำการแบ่งตามรูปที่ 2.9



รูปที่ 2.9 รูปประโยคก่อนการทำการจับกลุ่มคำนาม

สามารถจับกลุ่มคำนามเพื่อจำแนกตามรูปที่ 2.10 ดังนี้



รูปที่ 2.10 รูปประโยคหลังจากการจับกลุ่มคำนาม

7) การระบุคำที่เป็นนิพจน์ระบุนาม (Named Entity Recognition: NER)

ในประโยคจากรูปที่ 2.10 นั้นมีคำนามดังต่อไปนี้

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

รูปที่ 2.11 คำนามของประโยค

เป้าหมายของการระบุคำที่เป็นนิพจน์ระบุนาม คือ การตรวจจับและระบุชื่อคำนามเหล่านี้ โดยที่รูปที่ 2.12 คือลักษณะประโยคหลังจากที่มีการเรียกใช้โทเค็นแต่ละตัวผ่านการใช้ NER Tagging Model

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

Geographic Entity Geographic Entity Geographic Entity

รูปที่ 2.12 ประโยคจากการใช้ NER Tagging Model

แต่ระบบการระบุคำที่เป็นนิพจน์ระบุนามจะไม่ทำการค้นหาพจนานุกรมทั่ว ๆ ไป แต่จะใช้บริบทของคำที่ปรากฏในประโยคและแบบจำลองทางสถิติเพื่อคาดเดาคำนามชนิดนั้น

ชนิดของวัตถุ (Objects) ที่ระบบ การระบุคำที่เป็นนิพจน์ระบุนามทั่วไปสามารถคิดแท็กได้ ดังนี้

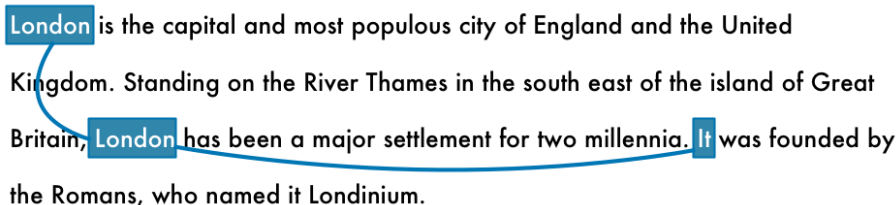
- ชื่อบุคคล (People's Names)
- ชื่อบริษัท (Company Names)
- สถานที่ทางภูมิศาสตร์ (Geographic Locations)
- ชื่อสินค้า (Product Names)
- วันที่และเวลา (Dates and Times)

- จำนวนเงิน (Amounts of Money)
- ชื่อเหตุการณ์ต่าง ๆ (Names of Events)

การระบุคำที่เป็นนิพจน์ระบุนามมีการใช้งานที่หลากหลายเนื่องจากง่ายต่อการดึงข้อมูลที่มีโครงสร้างออกจากข้อความ

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 [10]



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 2.13 การทำ Coreference Resolution

2.2.4 Stanford Named Entity Recognizer (Stanford NER)

เป็นการประยุกต์ใช้จากภาษาจาวา (Java) สำหรับการรู้จำนิพจน์ระบุนาม (Named Entity Recognizer: NER) ซึ่งเป็นการจัดประเภทของคำในข้อความ เช่น ชื่อสิ่งของ ชื่อบุคคล และบริษัท เป็นการกำหนดโครงสร้างการสัณฐานสมบัติที่เหมาะสมสำหรับการรู้จำนิพจน์ระบุนาม (Named Entity Recognition: NER) [11] [12] 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

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

2.2.5 Natural Language Toolkit (NLTK)

เป็นแพลตฟอร์มที่นิยมในโปรแกรมภาษาไพทอน (Python) เพื่อทำงานกับข้อมูลภาษาของมนุษย์ พร้อมกับชุดของไลบรารีที่ช่วยในการประมวลข้อความ แบ่งประเภทของคำ (Classification) การแบ่งโทเค็นของคำ (Tokenization) การตัดคำ (Stemming) การติดแท็กคำ (Tagging) และการแยกวิเคราะห์คำ (Parsing) [13]

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

2.2.6 spaCy

เป็นไลบรารีสำหรับการทำการประมวลผลภาษาธรรมชาติขั้นสูงในภาษาไพทอน (Python) โดยที่ spaCy ถูกออกแบบมาสำหรับการประยุกต์ใช้งานจริง และช่วยสร้างแอปพลิเคชันที่สามารถประมวลผล และทำความเข้าใจข้อความจำนวนมาก สามารถใช้ในการดำเนินการสกัดข้อมูล (Information Extraction) หรือระบบการทำความเข้าใจภาษาธรรมชาติเพื่อดำเนินการประมวลผลข้อความล่วงหน้า สำหรับการเรียนรู้เชิงลึก (Deep Learning) ซึ่งคุณสมบัติของ spaCy มีดังต่อไปนี้ [14]

- 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 เพื่อภูเขา 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 หรือช่วงเวลาที่เฉพาะเจาะจง
- 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 ทางผู้จัดทำได้ดำเนินการเลือกการติดแท็กในบทสนทนาเป็นจำนวนทั้งหมด 6 ประเภท ได้แก่ PERSON, ORG, GPE, LOC, DATE และ MONEY

2.2.7 Regular Expressions

เป็นสัญลักษณ์ที่ใช้ระบุชุดของอักขระตัวอักษร เมื่อชุดของอักขระตัวอักษรที่เฉพาะเจาะจงนั้น อยู่ในชุดอักขระตัวอักษรที่มีการกำหนดให้เป็น Regular Expressions โดยทั่วไปแล้วจะใช้สัญลักษณ์ “*”, “+”, “?”, “()” และ “|” ในการกำหนดเงื่อนไขของชุดตัวอักษร [Regexp_matching_can_be_simple] [15] [16]

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” ก็สามารรถเข้าเงื่อนไขนั้นได้เช่นกัน
- “{n}” คือ กำหนดจำนวนตัวอักษรนั้น ๆ โดยที่ n ไม่สามารถเป็นค่าลบได้ เช่น “a{9}” คือ กำหนดให้มีอักษร “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

2.2.8 Pydub

Pydub เป็นไลบรารีหนึ่งของภาษาไพทอนที่ใช้ในการจัดการกับข้อมูลที่เป็นไฟล์เสียง ซึ่งไลบรารีนี้มีคำสั่งต่าง ๆ ในการจัดการข้อมูล ยกตัวอย่างเช่น

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

2.2.9 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) [17] ดังสมการที่ 2.1

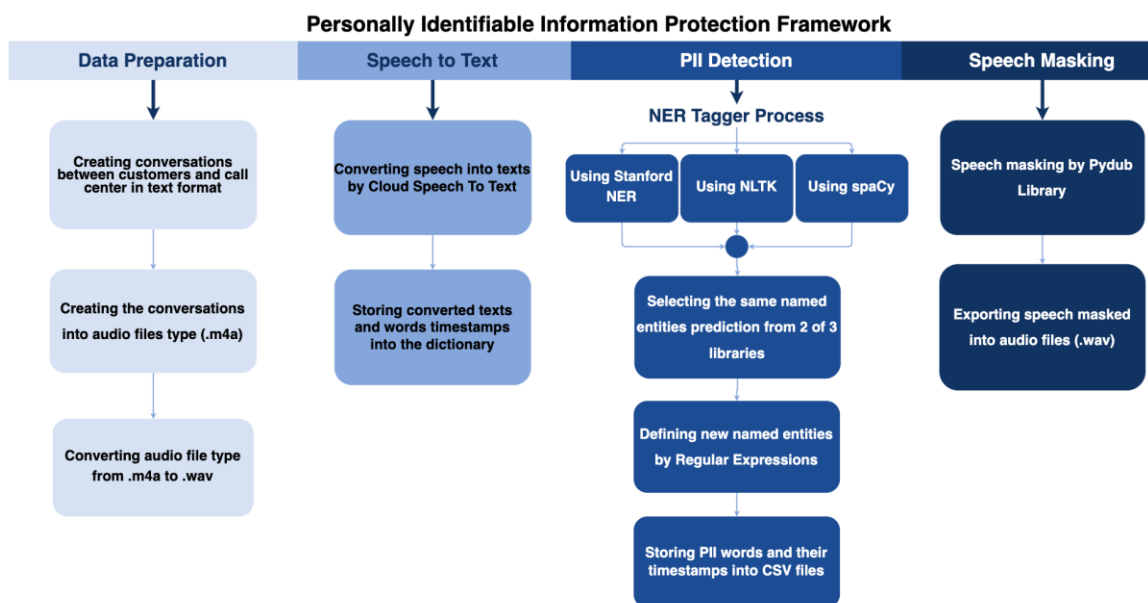
$$Jaccard(X, Y) = \frac{|X \cap Y|}{|X \cup Y|} \quad (2.1)$$

ซึ่งทางผู้จัดทำได้นำ Jaccard's Coefficient Similarity มาใช้ในการประเมินผลความแม่นยำของการแปลงข้อมูลเสียงให้อยู่ในรูปแบบข้อความ

บทที่ 3

ขั้นตอน และวิธีการดำเนินงานวิจัย

หลังจากที่ทางผู้จัดทำได้ดำเนินการศึกษาค้นคว้าและทำความเข้าใจกระบวนการทำงานของเทคโนโลยีที่เกี่ยวข้องต่าง ๆ ดังที่ได้กล่าวมาในบทที่ 2 นั้น ผู้จัดทำจะทำการอธิบายรายละเอียดของขั้นตอนการดำเนินงานที่ได้นำเทคโนโลยีที่ศึกษามาประยุกต์ใช้งานในบทที่ 3 โดยมีการดำเนินงานตามแผนงานการปกป้องข้อมูลที่ระบุตัวบุคคล (Personally Identifiable Information Protection Framework) ที่ทางผู้จัดทำได้ออกแบบขึ้น ดังรูปที่ 3.1



รูปที่ 3.1 แผนงานการปกป้องข้อมูลที่ระบุตัวบุคคล

จากรูป 3.1 สามารถอธิบายได้ว่าในแผนงานการปกป้องข้อมูลที่ระบุตัวบุคคล ทางผู้จัดทำได้ดำเนินการแบ่งส่วนของการดำเนินงานออกเป็น 4 ส่วนหลัก ๆ คือ การเตรียมข้อมูล การแปลงเสียงพูดให้อยู่ในรูปแบบข้อความ การตรวจจับคำที่เป็นข้อมูลส่วนบุคคลจากข้อมูลรูปแบบข้อความ และการแทนที่เสียงที่เป็นข้อมูลส่วนบุคคลด้วยเสียงรบกวน มีรายละเอียดการดำเนินงาน ดังนี้

3.1 การเตรียมข้อมูล (Data Preparation)

ในขั้นตอนนี้ ทางผู้จัดทำจะนำเสนอรายละเอียดเกี่ยวกับการเตรียมข้อมูล ซึ่งเป็นกระบวนการแรกในการปกป้องข้อมูลที่ระบุตัวบุคคล มีรายละเอียด ดังนี้

ทางผู้จัดทำได้ดำเนินการสร้างชุดข้อมูลขึ้นเองเพื่อนำไปประยุกต์ใช้กับการพัฒนาระบบในขั้นตอนถัดไป ซึ่งมีวิธีการดำเนินงาน ดังนี้

3.1.1 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 3.2 ตัวอย่างบทสนทนาระหว่างลูกค้ากับศูนย์บริการข้อมูลลูกค้าทางโทรศัพท์

จากรูปที่ 3.2 ทางผู้จัดทำได้ยกตัวอย่างข้อมูลบทสนทนาเป็นจำนวน 1 บทสนทนา ซึ่งทางผู้จัดทำได้ดำเนินการสร้างชุดข้อมูลในรูปแบบข้อความเป็นจำนวนทั้งหมด 23 บทสนทนา (Conversations) เพื่อใช้ในการพัฒนาและประเมินผลระบบ

ทางผู้จัดทำได้ดำเนินการวิเคราะห์และสำรวจข้อมูล (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 3.3 รายละเอียดการวิเคราะห์ประโยค

จากรูปที่ .. สามารถอธิบายได้ว่าในบทสนทนาจำนวนทั้งหมดนั้น มีประโยคทั้งหมด 566 ประโยค ซึ่งทางผู้จัดทำได้ดำเนินการแบ่งประโยคโดยใช้ไลบรารีของ NLTK และใน 1 บทสนทนา จะมีประโยคเฉลี่ยทั้งหมดประมาณ 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 3.4 รายละเอียดการวิเคราะห์คำ

จากรูปที่ 3.4 ทางผู้จัดทำได้ดำเนินการแบ่งการวิเคราะห์คำออกเป็น 2 ประเภท คือ วิเคราะห์คำจากบทสนทนาจริง และวิเคราะห์คำจากบทสนทนาที่ดำเนินการทำความสะอาดข้อมูล (Data Cleaning) จากการตัดเครื่องหมายวรรคตอนและ Stop words ที่ทางไลบรารีของ NLTK ได้จัดสรรให้ ดังรูปที่ 3.5

```
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', 'agains t', 't', 'mustn', 'whom', 'wouldn', 'for', 'no', 'him', 'between', 'at'}
```

รูปที่ 3.5 รายการของเครื่องหมายวรรคตอนและ Stop words

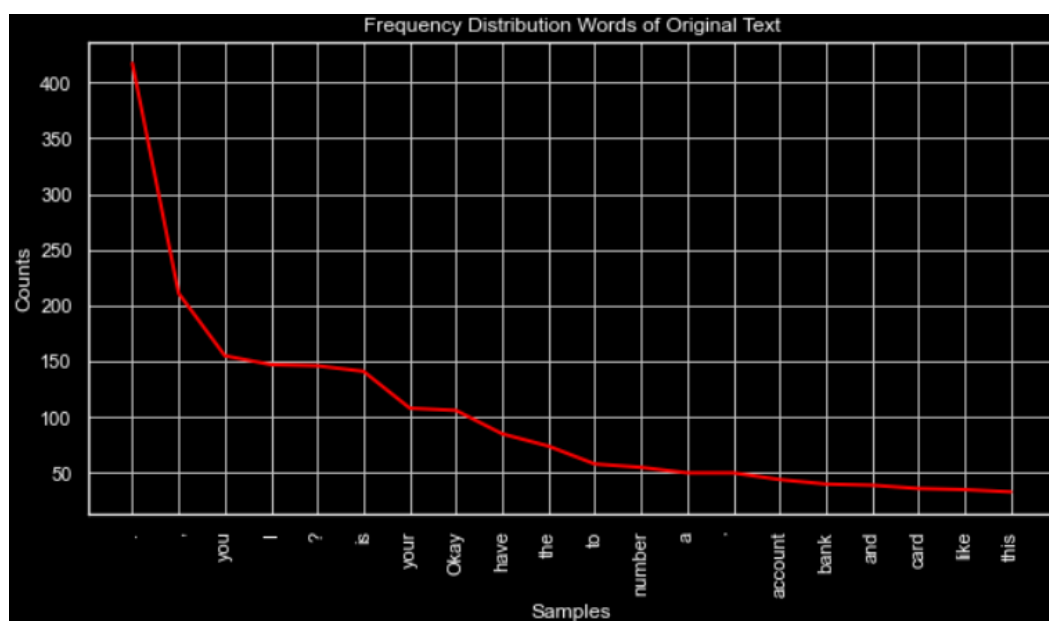
เมื่อดำเนินการตัดคำในรายการเหล่านั้นออกแล้ว ดังรูปที่ 3.5 สามารถอธิบายได้ว่า จากบทสนทนาจริง มีคำในบทสนทนาทั้งหมด 4095 คำ และใน 1 บทสนทนามีจำนวนคำเฉลี่ย 178.04 คำ และจากบทสนทนาที่ผ่านการทำความสะอาดข้อมูลแล้ว มีคำในบทสนทนาทั้งหมด 1732 คำ และใน 1 บทสนทนามีจำนวนคำเฉลี่ย 75.30 คำ

- 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

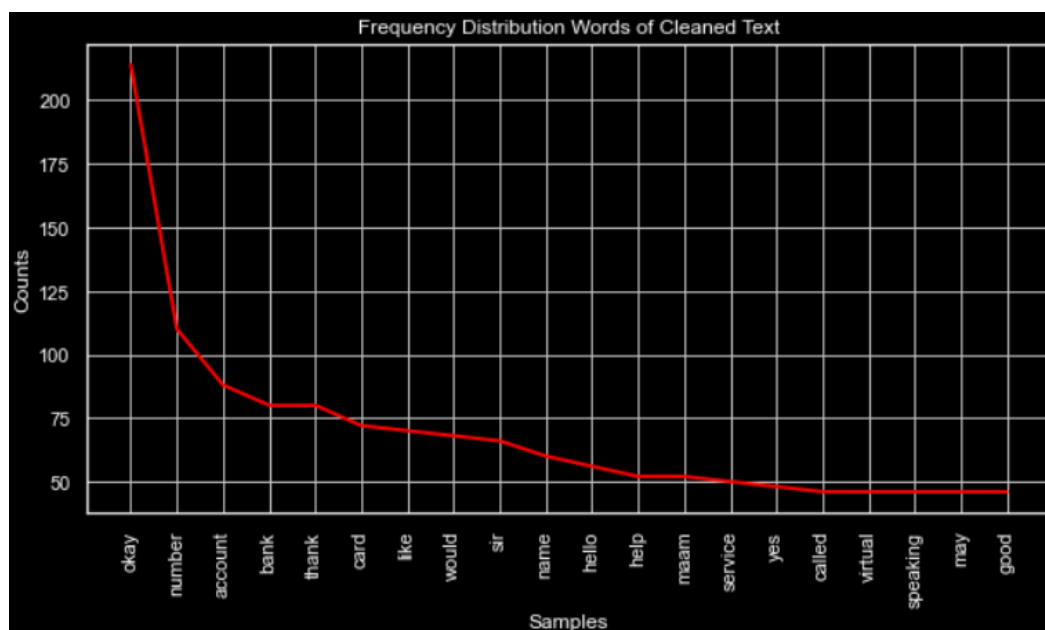
รูปที่ 3.6 รายละเอียดการวิเคราะห์ความถี่ของคำที่ไม่ซ้ำกัน

จากรูปที่ 3.6 ทางผู้จัดทำได้ดำเนินการแบ่งการวิเคราะห์คำเป็น 2 ประเภท เช่นเดียวกับขั้นตอนการวิเคราะห์คำ (Words Analysis) ก่อนหน้านี้ สามารถอธิบายได้ว่าในบทสนทนาจริงมีจำนวนคำที่ไม่ซ้ำกันเป็นจำนวน 510 คำ และบทสนทนาที่ผ่านการทำความสะอาดข้อมูลแล้ว มีจำนวนคำที่ไม่ซ้ำกันเป็นจำนวน 385 คำ ซึ่งทางผู้จัดทำได้ทำการแจกแจงความถี่ของคำที่ซ้ำกันมากที่สุด 20 คำแรกของบทสนทนาจริง ดังรูปที่ .. และแจกแจงความถี่ของคำที่ซ้ำกันมากที่สุด 20 คำแรกของบทสนทนาที่ผ่านการทำความสะอาดข้อมูลแล้ว 20 คำแรก ดังรูปที่ 3.7



รูปที่ 3.7 การแจกแจงความถี่ของคำที่ซ้ำกันของบทสนทนาจริง

จากรูปที่ 3.7 ทางผู้จัดทำยกตัวอย่างการอ่านกราฟคร่าว ๆ 3 อันดับแรกที่มีความถี่มากที่สุด คือ “.” มีความถี่ทั้งหมด 417 คำ รองลงมาคือ “,” มีความถี่ทั้งหมด 211 คำ และสุดท้ายคือ “you” มีความถี่ทั้งหมด 155 คำ เป็นต้น



รูปที่ 3.8 การแจกแจงความถี่ของคำที่ซ้ำกันของบทสนทนาที่ผ่านการทำความสะอาด

จากรูปที่ 3.8 ทางผู้จัดทำยกตัวอย่างการอ่านกราฟคร่าว ๆ 3 อันดับแรกที่มีความถี่มากที่สุด คือ “okay” มีความถี่ทั้งหมด 214 คำ รองลงมาคือ “number” มีความถี่ทั้งหมด 110 คำ และสุดท้ายคือ “account” มีความถี่ทั้งหมด 88 คำ เป็นต้น

3.1.2 นำชุดข้อมูลบทสนทนาในรูปแบบข้อความที่ได้ดำเนินการสร้างขึ้นมาดำเนินการบันทึกเสียง

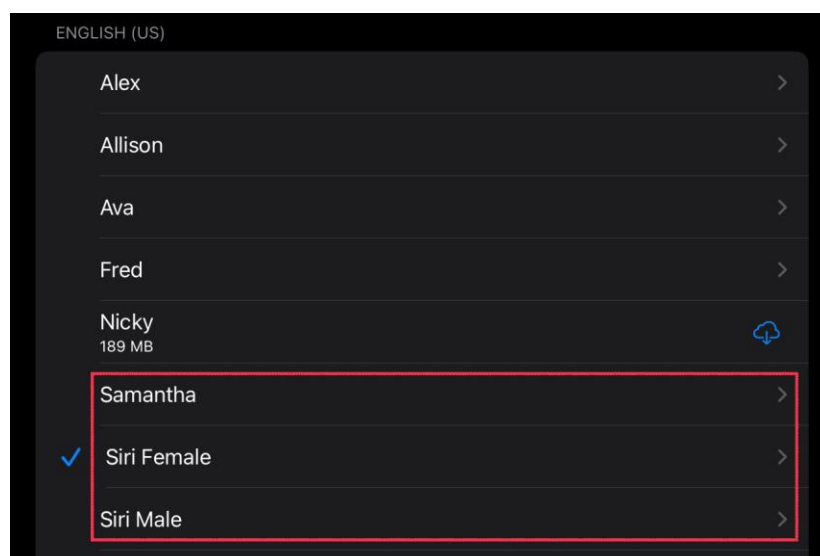
เนื่องจากบทสนทนาที่ทางผู้จัดทำสร้างขึ้นเป็นบทสนทนาภาษาอังกฤษ ทางผู้จัดทำได้มีการนำประโยคบทสนทนาไปบันทึกเสียงโดยใช้ระบบสังการด้วยเสียงของระบบปฏิบัติการ iOS หรือที่เป็นที่รู้จักกันในนามของ “สิริ” (Siri) ในการช่วยอ่านบทสนทนาเหล่านั้น ใน 1 บทสนทนาจะประกอบไปด้วยเสียงของพนักงานและลูกค้า โดยแบ่งตามเพศได้ ดังนี้

- เสียงพนักงานที่ให้บริการในศูนย์บริการข้อมูลลูกค้าทางโทรศัพท์ (Call Center)

ทางผู้จัดทำกำหนดให้เสียงพนักงานมีเพียงเพศเดียว คือ เพศหญิง ซึ่งเสียงของพนักงานทางผู้จัดทำได้กำหนดให้ใช้เสียงที่มีชื่อว่า “Siri Female” และใช้สำเนียงของประเทศสหรัฐอเมริกา (The United States of America) ในการอ่านข้อความเพื่อบันทึกเสียง

- เสียงของลูกค้า

เสียงของลูกค้ามี 2 เพศ คือ เพศชาย และเพศหญิง โดยเพศชายทางผู้จัดทำได้กำหนดให้ใช้เสียงที่มีชื่อว่า “Siri Male” และใช้สำเนียงของประเทศสหรัฐอเมริกา (The United States of America) ในการอ่านข้อความเพื่อบันทึกเสียง และในส่วนของเพศหญิงนั้น ทางผู้จัดทำได้กำหนดให้ใช้เสียงที่มีชื่อว่า “Samantha” และใช้สำเนียงของประเทศสหรัฐอเมริกา (The United States of America) ในการอ่านข้อความเพื่อบันทึกเสียง ดังรูปที่ 3.9



รูปที่ 3.9 รายการชื่อเสียงพูดที่ใช้ในการบันทึกเสียงบทสนทนา

เมื่อดำเนินการใช้เสียงพูดจากรายชื่อที่กล่าวมาในด้านบนแล้ว ก็ดำเนินการบันทึกเสียงโดยมีการบันทึกเสียงจากสมาร์ตโฟน ประเภทของไฟล์คือ “.m4a” ซึ่งระยะเวลาในแต่ละไฟล์เสียงของบทสนทนานั้นโดยเฉลี่ยคิดเป็นความยาวประมาณ 1 นาที ส่วนใหญ่แล้วมักจะไม่เกิน 2 นาทีจากบทสนทนาทั้งหมด ดังรูปที่ 3.10

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 3.10 ตัวอย่างไฟล์เสียงที่บันทึกจากสมาร์ทโฟน

3.1.3 ดำเนินการแปลงประเภทของไฟล์เสียงบทสนทนา

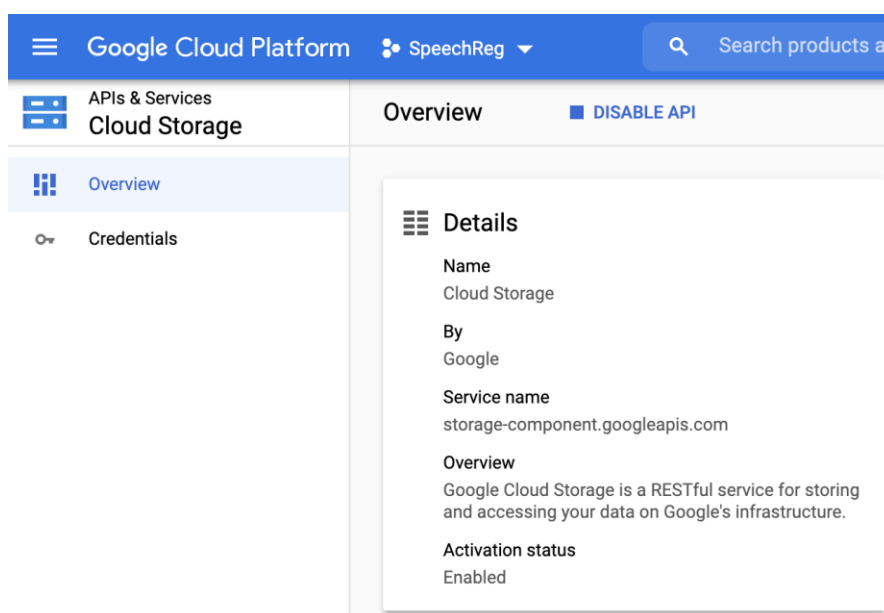
เนื่องจากทางผู้จัดทำได้ใช้แบบจำลองที่ชื่อว่า Cloud Speech to Text ในการดำเนินการแปลงข้อมูลเสียงให้อยู่ในรูปแบบข้อความ แต่ข้อจำกัดของแบบจำลองคือสามารถประมวลผลข้อมูลเสียงที่เป็นประเภทไฟล์ที่ชื่อว่า “.wav” และ “.mp3” เท่านั้น ทางผู้จัดทำจึงต้องดำเนินการแปลงประเภทไฟล์เสียงจาก “.m4a” ให้อยู่ในประเภทไฟล์ “.wav” เพราะ ประเภทไฟล์ “.wav” นั้นไม่ทำให้ไฟล์เสียงสูญเสียคุณภาพ [18] โดยได้ดำเนินการแปลงบนเว็บไซต์ที่ชื่อว่า “Convert MP4 to WAV” [19]

3.2 การแปลงเสียงพูดให้อยู่ในรูปแบบข้อความ

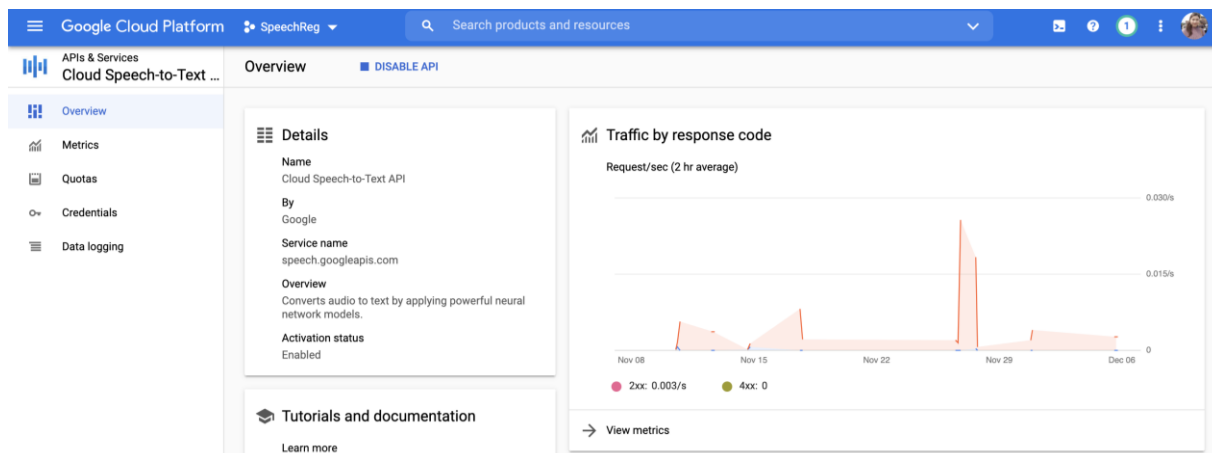
3.2.1 กระบวนการแปลงเสียงให้อยู่ในรูปแบบข้อความโดยใช้ Cloud Speech to Text

มีวิธีการดำเนินงาน ดังนี้

- หากมีบัญชีผู้ใช้ของ Google อยู่แล้วให้ทำการเข้าสู่ระบบเพื่อใช้งานกุญแจคลาวด์ ซึ่งในการใช้งานครั้งแรกทางกูเกิลจะให้เครดิต 300 ดอลลาร์ หรือคิดเป็นค่าเงินบาทประมาณ 9,000 บาท เนื่องจากการใช้งานกุญแจคลาวด์นั้นจะเสียค่าใช้จ่ายตามจำนวนที่ใช้จริง
- เริ่มการสร้าง Project บนกุญแจคลาวด์ และเปิด API ที่ต้องการใช้งาน ในที่นี้ทางผู้จัดทำเลือกใช้ บริการ API สองตัว ได้แก่ Cloud Storage ดังรูปที่ 3.11 และ Cloud Speech to Text ดังรูปที่ 3.12



รูปที่ 3.11 เปิดใช้งาน Cloud Storage



รูปที่ 3.12 เปิดใช้งาน Cloud Speech To Text

- ดำเนินการการอัปโหลดไฟล์เสียงที่ทางผู้จัดทำสร้างบทสนทนาระหว่างลูกค้ากับศูนย์ให้บริการข้อมูลลูกค้าทางโทรศัพท์ขึ้นบน Cloud Storage ดังรูปที่ 3.13

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.wi	2.8 MB	audio/wav	Nov 10, 2020, ...	Standard	Nov 10, 20...	Not public	Google-managed key	—	⬇ ⋮
<input type="checkbox"/>	Date.wa	1.3 MB	audio/wav	Nov 14, 2020, ...	Standard	Nov 14, 20...	Not public	Google-managed key	—	⬇ ⋮
<input type="checkbox"/>	Laura.wi	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"/>	conversi	2.5 MB	audio/wav	Nov 30, 2020, ...	Standard	Nov 30, 20...	Not public	Google-managed key	—	⬇ ⋮
<input type="checkbox"/>	conversi	2 MB	audio/wav	Nov 30, 2020, ...	Standard	Nov 30, 20...	Not public	Google-managed key	—	⬇ ⋮
<input type="checkbox"/>	conversi	2.9 MB	audio/wav	Nov 30, 2020, ...	Standard	Nov 30, 20...	Not public	Google-managed key	—	⬇ ⋮
<input type="checkbox"/>	conversi	1.7 MB	audio/wav	Nov 30, 2020, ...	Standard	Nov 30, 20...	Not public	Google-managed key	—	⬇ ⋮
<input type="checkbox"/>	conversi	2.7 MB	audio/wav	Nov 30, 2020, ...	Standard	Nov 30, 20...	Not public	Google-managed key	—	⬇ ⋮
<input type="checkbox"/>	conversi	1.3 MB	audio/wav	Nov 30, 2020, ...	Standard	Nov 30, 20...	Not public	Google-managed key	—	⬇ ⋮

รูปที่ 3.13 อัปโหลดไฟล์เสียงขึ้นบน Cloud Storage

- นำเข้าข้อมูลเสียงจาก Cloud Storage และทำการกำหนดค่าต่าง ๆ เพื่อนำไปใช้ในการแปลงเสียงพูดให้อยู่ในรูปแบบข้อความ ดังรูปที่ 3.14

```
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 3.14 การนำเข้าข้อมูลเสียงและกำหนดค่าต่าง ๆ

- ดำเนินการแปลงเสียงพูดให้อยู่ในรูปแบบข้อความ ในส่วนของฟังก์ชัน `print_word_offsets` จะทำการระบุเวลากับคำในบทสนทนา (Timestamp) โดยทำการระบุเวลาที่เริ่มต้นในแต่ละคำ และเวลาที่สิ้นสุดของคำ ๆ นั้น โดยหน่วยของเวลาเป็นวินาที เพื่อให้ง่ายต่อการเข้าใจ ผู้จัดทำจึงแสดงผลในรูปแบบของตาราง ดังรูปที่ 3.15 ต่อมาเป็นฟังก์ชัน `print_sentences` จะทำการแสดงผลประโยคที่ผ่านการแปลงเสียงพูดให้อยู่ในรูปแบบข้อความตัวอักษรและทำการแสดงผลค่าความเชื่อมั่น (Confidence) ดังรูปที่ 3.16 และส่วนสุดท้ายในขั้นตอนการแปลงเสียงพูดให้อยู่ในรูปแบบของข้อความในฟังก์ชัน `speech_to_text` ใช้โมดูลของไลบรารี `Speech` ในการแปลงข้อมูลเสียงพูดให้อยู่ในรูปแบบข้อความ และระบุไฟล์ ตามหัวข้อข้างต้น

	word	start_times	end_times
0	Hello,	0.0	0.4
1	you	0.4	1.2
2	have	1.2	1.3
3	called	1.3	1.8
4	virtual	1.8	2.2
...
164	Thank	91.7	92.4
165	you,	92.4	92.5
166	sir.	92.5	93.4
167	Thank	93.4	94.5
168	you.	94.5	94.7

รูปที่ 3.15 ผลลัพธ์จากการทำฟังก์ชัน `print_word_offsets`

```

-----
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 ca
rd, but I do have are you ready? I will give you what I have got 760-545-6789. O
kay. 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. Appel board. Okay.
Confidence: 72%

```

รูปที่ 3.16 ผลลัพธ์จากการทำฟังก์ชัน `print_sentences`

```

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 3.17 ฟังก์ชันการแปลงเสียงพูดให้อยู่ในรูปแบบของข้อความตัวอักษร

3.2.2 บันทึกไฟล์ที่ผ่านการแปลงเสียงพูดให้อยู่ในรูปแบบข้อความให้อยู่ในรูปของ Dictionary

บันทึกเป็นประเภทไฟล์ JSON เพื่อทำการตรวจจับคำที่เป็นข้อมูลส่วนบุคคลในขั้นตอนต่อไป

ดังรูปที่ 3.18

```

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

```

รูปที่ 3.18 การบันทึกไฟล์ข้อความเป็นไฟล์ประเภท .json

3.3 การตรวจจับคำที่เป็นข้อมูลส่วนบุคคลจากข้อมูลรูปแบบข้อความ

ทางผู้จัดทำได้มีการดึงข้อมูลที่ Cloud Speech to Text ดำเนินการแปลงให้อยู่ในรูปแบบข้อความ ซึ่งเป็นไฟล์ JSON ในรูปของ Dictionary และนำข้อมูลเหล่านั้นไปวิเคราะห์ต่อ โดยรายละเอียดของกระบวนการทั้งหมด มีดังนี้

3.3.1 กระบวนการตรวจจับนิพจน์ระบุนาม (Named Entities Tagger Process)

ในขั้นตอนนี้ทางผู้จัดทำได้ใช้ไลบรารีทั้งหมด 3 ไลบรารี เพื่อเพิ่มความแม่นยำในการตรวจจับนิพจน์ระบุนาม หรือข้อมูลส่วนบุคคล ซึ่งทางผู้จัดทำจะดำเนินการอธิบายรายละเอียดของไลบรารีแต่ละแบบที่ได้ใช้ตามกระบวนการ ดังนี้

- 1) ดำเนินการพัฒนาไลบรารีของ Stanford Named Entities Recognizer ทางผู้จัดทำได้ตัดสินใจเลือกไลบรารีสำหรับติดแท็กนิพจน์ระบุนาม 7 ประเภท และดำเนินการเลือกการติดแท็กในบทสนทนาทั้งหมดเป็นจำนวน 5 ประเภท ได้แก่ PERSON, ORGANIZATION, LOCATION, DATE และ MONEY ดังที่กล่าวไว้ในบทแนวคิด และเทคโนโลยีที่เกี่ยวข้อง โดยมีการสร้างกระบวนการวิเคราะห์ข้อความต่าง ๆ ไว้หนึ่งฟังก์ชัน และในฟังก์ชันนั้นมีการทำ Word Tokenization เพื่อแยกโทเค็นของคำในข้อความเป็นอันดับแรก ต่อมา มีการติดแท็กนิพจน์ระบุนาม (Named Entities) โดยใช้อัลกอริทึมของ Stanford NER จากนั้นสร้างเงื่อนไขเก็บเฉพาะโทเค็นที่เป็นนิพจน์ระบุนามเท่านั้น จากนั้นดำเนินการแก้ไขประเภทของนิพจน์ระบุนามที่ถูกติดแท็ก เพื่อให้ประเภทของนิพจน์ระบุนามตรงกับไลบรารีอื่น ๆ เช่น คำว่า “ORG” ที่ทางไลบรารีติดแท็กไว้ ทางผู้จัดทำจะดำเนินการเปลี่ยนเป็นคำว่า “ORGANIZATION” เพื่อให้ตรงกับไลบรารีทั้ง 2 แบบ และสะดวกต่อการนำไปประเมินผล จากนั้นทำการตรวจสอบโทเค็นคำที่ไลบรารีแบ่งออกมาเทียบกับโทเค็นที่ Cloud Speech to Text แบ่งไว้ให้ เพื่อให้แน่ใจว่าโทเค็นที่ Stanford NER ติดแท็กได้นั้นตรงกับระยะเวลาที่ Cloud Speech to Text ทำนายออกมา และเก็บค่าของคำที่ติดแท็กได้ พร้อมกับประเภทของนิพจน์ระบุนาม ดังรูปที่ 3.19

```

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.c
    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]) != '0':
            if str(classified_text[i][1]) == 'PERSON' or str(classified_text[i][1])
                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('0')

    df['stanford_pred'] = st_pred

    return st_pred, df

```

Stanford NER importing

Word tokenization

Words tagger

Rename named entities conditions

Stanford NER tokenized words and
GG Speech Recognition words matching

รูปที่ 3.19 ฟังก์ชันการทำนายนิพจน์ระบุนามของ Stanford NER

ดำเนินการพัฒนาไลบรารีของ NLTK ทางผู้จัดทำได้ดำเนินการเลือกการติดแท็กในบทสนทนาเป็นจำนวนทั้งหมด 6 ประเภท ได้แก่ ORGANIZATION, PERSON, LOCATION, GPE, DATE และ MONEY ดังที่กล่าวไว้ในบทแนวคิด และเทคโนโลยีที่เกี่ยวข้อง โดยมีการสร้างกระบวนการวิเคราะห์ข้อความต่าง ๆ ไว้หนึ่งฟังก์ชัน และในฟังก์ชันนั้นมีการทำ Word Tokenization เพื่อแยกโทเค็นของคำในข้อความ จากนั้นทำการติดแท็กนิพจน์ระบุนาม (Named Entities) โดยใช้อัลกอริทึมของ NLTK ซึ่งต้องทำการติดแท็กส่วนของประโยค (Part-of-Speech) ก่อน จึงจะดำเนินการติดแท็กนิพจน์ระบุนามได้ ต่อมาได้สร้างเงื่อนไขเลือกเฉพาะโทเค็นที่มีนิพจน์ระบุนาม และเปลี่ยนประเภทของนิพจน์ระบุนามให้เหมือนกับไลบรารีอื่น ๆ เช่น คำว่า “LOC” เปลี่ยนเป็น “LOCATION” เป็นต้น และนอกจากนี้ ทางผู้จัดทำได้รวมนิพจน์ระบุนามประเภท LOCATION และ GPE เข้าด้วยกัน โดยการเปลี่ยนชื่อประเภท GPE ให้เป็น LOCATION ทั้งหมด เพื่อให้เป็นประเภทเดียวกันกับ Stanford NER จากนั้นทำการจับคู่โทเค็นคำที่ไลบรารีแบ่งออกมาเทียบกับโทเค็นที่ Cloud Speech to Text แบ่งไว้ให้ เพื่อให้แน่ใจว่า

โทเอนที่ NLTK คิดแยกได้นั้นตรงกับระยะเวลาที่ Cloud Speech to Text ทำนาขออกมา และเก็บค่าของคำที่คิดแยกได้ พร้อมกับประเภทของนิพจน์ระบุนาม ดังรูปที่ 3.20

```
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

    return nltk_pred, df
```

Word tokenization
Words tagger

Rename named entities conditions

NLTK tokenized words and
GG Speech Recognition words matching

รูปที่ 3.20 ฟังก์ชันการทำนาขนิพจน์ระบุนามของ NLTK

- 2) ดำเนินการพัฒนาไลบรารีของ spaCy ทางผู้จัดทำได้ดำเนินการเลือกการคิดแยกในบทสนทนาเป็นจำนวนทั้งหมด 6 ประเภท ได้แก่ ORGANIZATION, PERSON, LOCATION, GPE, DATE และ MONEY ดังที่กล่าวไว้ในบทแนวคิด และเทคโนโลยีที่เกี่ยวข้อง โดยมีการสร้างกระบวนการวิเคราะห์ข้อความต่าง ๆ ไว้หนึ่งฟังก์ชัน และในฟังก์ชันนั้นมีการใช้อัลกอริทึมของ spaCy ซึ่งในอัลกอริทึมจะดำเนินการวิเคราะห์ข้อความต่าง ๆ อัตโนมัติ ส่งผลให้ทางผู้จัดทำสามารถเรียกดูค่าได้จากอัลกอริทึมของไลบรารีได้ทันที จากนั้นสร้างเงื่อนไขเลือก

เฉพาะโทเ็นคำที่มีนิพจน์ระบุนาม (Named Entities) และเปลี่ยนชื่อประเภทของนิพจน์ระบุนามให้ตรงกับไลบรารีอื่น ๆ เช่นเดียวกับ Stanford NER และ NLTK ต่อมาทำการจับคู่โทเ็นคำที่ไลบรารีแบ่งออกมาเทียบกับโทเ็นที่ Cloud Speech to Text แบ่งไว้ให้ เพื่อให้แน่ใจว่าโทเ็นที่ spaCy ดัดแปลงได้นั้นตรงกับระยะเวลาที่ Cloud Speech to Text ทำนายออกมา และเก็บค่าของคำที่ติดแท็กได้ พร้อมกับประเภทของนิพจน์ระบุนาม ดังรูปที่ 3.21

```
def spaCy_pred(dictt, df):

    nlp = en_core_web_sm.load() Text Analysis
    # list of words that have named entities NE tokenized storage
    text = ([str(X) for X in nlp(dictt)
              if (X.ent_type_ != '' and X.ent_type_ != 'CARDINAL') & (str(X) != 'a')])
    # 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')])

    sp_pred = []

    Rename named entities conditions
    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
```

spaCy tokenized words and
GG Speech Recognition words matching

รูปที่ 3.21 ฟังก์ชันการทำนายนิพจน์ระบุนามของ spaCy

3.3.2 กระบวนการเลือกการทำนายประเภทของนิพจน์ระบุนามที่เหมือนกันตั้งแต่ 2 ใน 3 ของไลบรารี

ขั้นตอนนี้ทางผู้จัดทำได้ดำเนินการสร้างฟังก์ชันเพื่อเลือกโทเค็นของคำที่ไลบรารีทำนายประเภทของนิพจน์ระบุนามเหมือนกันตั้งแต่ 2 ไลบรารีขึ้นไป เนื่องจากในบางครั้งการใช้ไลบรารีแค่แบบเดียวอาจไม่แม่นยำมากพอที่จะทำนายประเภทของโทเค็นคำได้อย่างถูกต้อง ทางผู้จัดทำจึงได้สร้างเกณฑ์นี้มาเพื่อเพิ่มประสิทธิภาพของการทำนาย หลังจากดำเนินการเลือกการทำนายที่เหมือนกันตั้งแต่ 2 จาก 3 ของไลบรารีแล้ว ทางผู้จัดทำก็ได้ดำเนินการเก็บค่าของโทเค็นคำ และประเภทของนิพจน์ระบุนาม เพื่อนำไปวิเคราะห์ในขั้นตอนถัดไป ดังรูปที่ 3.22

```
def combined_models(df):

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

    i_twooth = []
    ne_twooth = []

    # Same prediction 2 of 3 models condition
    for i, st, nl, sp in zip(df.index, df['stanford_pred'], df['nltk_pred'], df['spacy_pred']):
        # check if stanford and nltk are same named entities
        if (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(st))
        # 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(nl))

    combined = []
    combined_check = 0

    # Tokenized words and GG Speech Recognition words matching
    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 3.22 ฟังก์ชันการเลือกการทำนายประเภทนิพจน์ระบุนามที่เหมือนกัน 2 ใน 3

3.3.3 สร้างประเภทของนิพจน์ระบุนามเพิ่ม เพื่อติดแท็กเลขที่เป็นข้อมูลส่วนบุคคลโดยใช้ Regular Expressions

ขั้นตอนนี้จะต่อเนื่องจากขั้นตอนก่อนหน้านี้ คือ นำค่าที่ทำนายเหมือนกันตั้งแต่ 2 จาก 3 ไลบรารี ในที่นี้ ทางผู้จัดทำขอแทนว่าเป็นค่าทำนายจริง เพื่อให้สะดวกต่อการนำไปกล่าวในขั้นตอนอื่น ๆ โดยจะนำค่าโทเค็นคำของ Cloud Speech to Text มาวิเคราะห์ก่อน ทางผู้จัดทำได้สร้างเงื่อนไขเพื่อติดแท็กเฉพาะโทเค็นที่เป็นเฉพาะตัวเลขตามเงื่อนไขที่สร้างไว้โดยใช้ Regular Expressions ในการตรวจสอบ ซึ่งทางผู้จัดทำได้ดำเนินการแบ่งประเภทของเลขที่เป็นข้อมูลส่วนบุคคลไว้ 5 ประเภท คือ IDCARD (เลขบัตรประชาชน 13 หลัก) PHONENUM (เบอร์โทรศัพท์ 10 หลัก) ACCNUM (เลขบัญชี 9 หลัก) CARDNUM (เลขบัตรเครดิต หรือบัตรเครดิต 16 หลัก) และ PIINUM (เลขอื่น ๆ ที่ไม่เข้าเงื่อนไขประเภทก่อนหน้านี้ แต่มีตั้งแต่ 9 หลักขึ้นไป มีไว้ในกรณีที่ Cloud Speech to Text แปลงเป็นข้อความออกมาได้ไม่แม่นยำ) ดังรูปที่ 3.23

```
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('IDCARD')
        # 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('PHONENUM')
        # 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('ACCNUM')
        # 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('CARDNUM')
        # if not has punctuation
        elif re.search('(\+[0-9]{9,})', num):
            pii_index.append(i)
            pii_type.append('PIINUM')
```

รูปที่ 3.23 การสร้างนิพจน์ระบุนามใหม่โดยใช้ Regular Expressions

และขั้นตอนสุดท้ายคือการรวมค่าที่ทำนายจริง กับค่าของเลขที่เป็นข้อมูลส่วนบุคคลมารวมกัน และเก็บค่านั้นไว้ในตาราง ดังรูปที่ 3.24

```

regex_lst = []
regex_check = 0
Regular Expressions tagger condition
for i in df.index:
    regex_check = 0
    for ii, pi in zip(pii_index, pii_type):
        if i == ii:
            regex_check = 1
            regex_lst.append(str(pi))
            break
    if regex_check == 0:
        regex_lst.append('0')

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

cb_rg = []
Real entities and regex combination
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

return cb_rg, df

```

รูปที่ 3.24 รวมการทำนาย Regular Expression และค่าทำนายจริงเข้าด้วยกัน

3.3.4 เก็บค่าต่าง ๆ ให้อยู่ในรูปของไฟล์ CSV

หลังจากดำเนินการทำนายนิพจน์ระบุนาม (Named Entities) ทั้งหมดแล้ว ทางผู้จัดทำก็ได้จัดเก็บค่าเหล่านั้นให้อยู่ในรูปแบบตารางและบันทึกเป็นไฟล์ CSV โดยมีจำนวนทั้งหมด 5 คอลัมน์ ได้แก่ ลำดับโทเค็น โทเค็นค่า เวลาที่เริ่มพุดโทเค็นนั้นในไฟล์เสียง เวลาที่พุดโทเค็นนั้นจบ และประเภทของนิพจน์ระบุนาม

3.4 การแทนที่เสียงที่เป็นข้อมูลส่วนบุคคลด้วยเสียงรบกวน

ในขั้นตอนนี้จะดำเนินการการแทนที่เสียงรบกวนที่เป็นข้อมูลส่วนบุคคล โดยผู้จัดทำได้ใช้ค่าต่าง ๆ จากไฟล์ CSV จากขั้นตอนก่อนหน้านี้ซึ่งประกอบไปด้วยค่าที่เป็นข้อมูลส่วนบุคคล เวลาเริ่มต้นของคำนั้น เวลาสิ้นสุดของคำนั้น ในส่วนนี้เราจะใช้เพียงแค่เวลาเริ่มต้นของคำนั้นและเวลาสิ้นสุดของคำนั้น โดยดำเนินการแบ่งช่วงของเวลาเริ่มต้นและสิ้นสุด เนื่องจากในการแทนที่เสียงรบกวนนั้น ค่าของเวลาต้องอยู่ในหน่วยของมิลลิวินาที แต่เวลาเริ่มต้นและสิ้นสุดที่ได้มาจากไฟล์ CSV เป็นหน่วยวินาที จึงต้องดำเนินการแปลงให้อยู่ในหน่วยมิลลิวินาที ต่อมาดำเนินการแทนที่เสียงรบกวนในช่วงเวลาที่ได้อ่านไว้ และปรับระดับของเสียงรบกวนให้พอดี และขั้นตอนสุดท้ายดำเนินการบันทึกไฟล์เสียงที่มีการปิดบังข้อมูลส่วนตัวเป็นไฟล์เสียงประเภท .wav ดังรูปที่ 3.25

```
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 3.25 ฟังก์ชันการแทนที่เสียงที่เป็นข้อมูลส่วนบุคคลด้วยเสียงรบกวน

3.5 การประเมินผล (Evaluation)

ทางผู้จัดทำได้ดำเนินการประเมินผลกระบวนการทั้งหมด 2 กระบวนการหลัก ๆ คือ ประเมินผลความแม่นยำของการแปลงเสียงพูดให้อยู่ในรูปแบบข้อความ และกระบวนการประเมินผลความแม่นยำของการตรวจจับคำที่เป็นข้อมูลส่วนบุคคลจากข้อมูลรูปแบบข้อความ จากการสร้างผลเฉลยของการทำนายข้อความและโทเค็นต่าง ๆ เพื่อใช้ตรวจสอบความแม่นยำในการทำนายของไลบรารีทั้งหมด และในส่วนของการประเมินผลความแม่นยำของการแปลงเสียงพูดให้อยู่ในรูปแบบข้อความนั้น ทางผู้จัดทำได้นำแนวคิดของ Jaccard's Coefficient Similarity มาประยุกต์ใช้ในการประเมินผล

บทที่ 4

ผลการดำเนินงานเบื้องต้น

4.1 การแปลงเสียงพูดให้อยู่ในรูปแบบข้อความ

ทางผู้จัดทำขอยกตัวอย่างส่วนหนึ่งของการแปลงเสียงพูดให้อยู่ในรูปแบบข้อความจากการใช้ Cloud Speech to Text 1 บทสนทนา ดังรูปที่ 4.1

```
{ '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. Appelboard. 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, 68.6, 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, 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], '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.', 'Appelboard.', '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 4.1 ตัวอย่างการแปลงข้อมูลเสียงให้อยู่ในรูปแบบข้อความโดยใช้ Cloud Speech to Text

จากรูปที่ 4.1 ทางผู้จัดทำได้ดำเนินการแปลงให้อยู่ในรูปแบบของ Dictionary และสร้างคีย์ที่ชื่อว่า transcript ไว้เก็บข้อความในบทสนทนาทั้งหมด ในส่วนของโทเค็นคำ ได้มีการสร้างคีย์ที่ชื่อว่า values ไว้เก็บค่าของเวลาที่เริ่มพูดโทเค็นนั้น ๆ (start) เวลาที่พูดจบ (end) และโทเค็นนั้น ๆ (word)

นอกจากนี้ ยังได้มีการประเมินผลความแม่นยำในการทำนายของแบบจำลอง โดยการนำข้อมูลบทสนทนาจริงเทียบกับข้อมูลที่แบบจำลองทำนายโดยใช้ 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 4.2 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 4.3 บทสนทนาที่แบบจำลองทำนาย

```
acc = Jaccard_Similarity(dict_, ori_text)
acc = acc*100

print('Accuracy of the conversation:', '%.2f' %acc, '%')
Accuracy of the conversation: 57.02 %
```

รูปที่ 4.4 ค่าของความแม่นยำในการทำนาย

จากรูปที่ 4.4 ความแม่นยำในการทำนายค่าพูดของแบบจำลองคิดเป็นร้อยละ 57.02 ซึ่งเป็นค่าความแม่นยำที่ไม่สูงนัก แต่หากเปรียบเทียบจากข้อมูลบทสนทนาจริง และข้อมูลบทสนทนาที่ทำการทำนายออกมาจากรูปที่ 4.2 และรูปที่ 4.3 จะสังเกตได้ว่า สิ่งที่ส่งผลให้ค่าความแม่นยำของแบบจำลองไม่สูงนั้นส่วนใหญ่แล้วขึ้นอยู่กับเครื่องหมายวรรคตอนของข้อมูลบทสนทนาจริงและข้อมูลบทสนทนาที่แบบจำลองทำนายออกมา ดังนั้น ทางผู้จัดทำจึงดำเนินการสร้างฟังก์ชันตัดเครื่องหมายวรรคตอนทั้งในข้อมูลบทสนทนาจริงและบทสนทนาที่แบบจำลองทำนาย เพื่อประเมินผลค่าความแม่นยำใหม่ ดังรูปที่ 4.5, 4.6 และ 4.7

```
'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 4.5 บทสนทนาจริงที่ผ่านการทำความสะอาด

```
'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 dont 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 7605456789 Okay Thats 765450607089 Correct? What is your identification number? 7745896589665 okay I have 774580960589665 and what is your name sir? It is Robert Appelboard Okay I have Robert Applebaum yet 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 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 4.6 บทสนทนาที่แบบจำลองทำนายที่ผ่านการทำความสะอาด

```
acc = Jaccard_Similarity(clean_text(dict_), clean_text1(ori_text))
acc = acc*100

print('Accuracy of the conversation:', '%.2f' %acc, '%')

Accuracy of the conversation: 71.43 %
```

รูปที่ 4.7 ค่าของความแม่นยำในการทำนาย (ใหม่)

จากรูปที่ 4.7 ความแม่นยำในการทำนายคำพูดของแบบจำลองคิดเป็นร้อยละ 71.43 สามารถเห็นได้ชัดว่าค่าความแม่นยำสูงขึ้นอย่างชัดเจน เมื่อดำเนินการตัดเครื่องหมายวรรคตอนออกเบื้องต้น

4.2 การตรวจจับคำที่เป็นข้อมูลส่วนบุคคลจากข้อมูลรูปแบบข้อความ

เมื่อดำเนินการนำข้อมูลในรูปแบบข้อความที่ได้จาก Cloud Speech to Text มาเข้าฟังก์ชันต่าง ๆ ของไลบรารี Stanford NER, NLTK และ spaCy พร้อมกับนำเข้าฟังก์ชันของการเลือกคำทำนายจริง และสร้างนิพจน์ระบุนาม (Named Entities) เพิ่ม สำหรับเลขที่เป็นข้อมูลส่วนบุคคลโดยใช้ Regular Expressions ดังที่ได้กล่าวไว้ในบทขึ้นตอน และวิธีการดำเนินงานวิจัยแล้ว ทางผู้จัดทำก็ได้ดำเนินการเก็บค่าของการทำนายของทุก ๆ ไลบรารีไว้ในรูปแบบตาราง ดังรูปที่ 4.8

	word	start_time	end_time	stanford_pred	nltk_pred	spacy_pred	real_ents
indx							
0	Hello,	0.0	0.4	DATE	LOCATION	O	O
1	you	0.4	1.2	O	O	O	O
2	have	1.2	1.3	O	O	O	O
3	called	1.3	1.8	O	O	O	O
4	virtual	1.8	2.2	O	O	O	O
5	bank.	2.2	2.4	O	O	O	O
6	This	2.4	3.2	O	O	O	O
7	is	3.2	3.4	O	O	O	O
8	Linda	3.4	3.8	PERSON	PERSON	PERSON	PERSON
9	speaking.	3.8	4.3	O	O	O	O
10	How	4.3	5.3	O	O	O	O
11	may	5.3	5.3	O	O	O	O
12	I	5.3	5.5	O	O	O	O
13	help	5.5	5.7	O	O	O	O
14	you?	5.7	5.9	O	O	O	O
15	Hi	6.2	6.8	O	O	O	O
16	Linda.	6.8	7.2	PERSON	PERSON	PERSON	PERSON
17	I	7.2	8.0	O	O	O	O
18	was	8.0	8.2	O	O	O	O
19	just	8.2	8.3	O	O	O	O

รูปที่ 4.8 ตารางการทำนายประเภทของนิพจน์ระบุนาม

จากรูปที่ 4.8 ทางผู้จัดทำได้ดำเนินการเก็บค่าการทำนายของโทเค็นทุก ๆ คำ ไว้ในตารางเดียวกันตามประเภทของนิพจน์ระบุนาม หากในแถวใดมีการทำนายเป็นคำว่า “O” หรือที่เรียกว่า อักษรตัวโอพิมพ์ใหญ่ในภาษาอังกฤษ หมายความว่าโทเค็นนั้นไม่ได้เป็นนิพจน์ระบุนาม ซึ่งมีการเก็บค่าการทำนายทั้งหมด 4 คอลัมน์ ได้แก่ stanford_pred คือ ค่าที่ Stanford NER ทำนาย nltk_pred คือ ค่าที่ NLTK ทำนาย spacy_pred คือ ค่าที่ spaCy ทำนาย และคอลัมน์สุดท้าย real_ents คือ ค่าทำนายที่แท้จริง (จากการเลือกค่าทำนายที่เหมือนกันตั้งแต่ 2 ใน 3 ของไลบรารี) และการติดแท็กค่าของเลขที่เป็นข้อมูลส่วนบุคคลจากการใช้ Regular Expressions

นอกจากนี้ ทางผู้จัดทำได้ดำเนินการเก็บบันทึกค่าการทำงานจริง เฉพาะโทเค็นที่มีการติดแท็กนิพจน์ระบุนาม (Named Entities) ขึ้นมาอีก 1 ตาราง เพื่อดำเนินการบันทึกให้อยู่ในรูปแบบไฟล์ CSV และนำไปปกปิดเสียงในขั้นตอนถัดไป ดังรูปที่ 4.9

	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	PHONENUM
79	+765-450-600-7089.	32.7	35.7	IDCARD
86	774-589-6589	38.9	42.7	PHONENUM
91	+774-580-960-5896	45.4	49.0	IDCARD
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	PHONENUM
137	610-265-1715.	77.4	80.7	PHONENUM
141	Appelbaum.	83.1	83.6	PERSON

รูปที่ 4.9 ตารางค่าทำงานจริงเฉพาะที่มีการติดแท็กนิพจน์ระบุนาม

ทางผู้จัดทำมีการประเมินผลความแม่นยำในการทำนายนิพจน์ระบุนามของแต่ละไลบรารี โดยการนำโทเค็นที่ Cloud Speech to Text แบ่งออกมา ไปทำการเจตนิพจน์ระบุนามจริง เพื่อที่จะนำไปประเมินผลความแม่นยำของการทำนายนิพจน์ระบุนามในทุก ๆ ไลบรารี



รูปที่ 4.10 การประเมินผลความแม่นยำของแต่ละแบบจำลอง

จากรูปที่ 4.10 สามารถสรุปได้ ดังนี้

- ความแม่นยำของการทำนายนิพจน์ระบุนาม (Named Entities) ของ Stanford NER คิดเป็นร้อยละ 88.17
- ความแม่นยำของการทำนายนิพจน์ระบุนาม (Named Entities) ของ NLTK คิดเป็นร้อยละ 84.62
- ความแม่นยำของการทำนายนิพจน์ระบุนาม (Named Entities) ของ spaCy คิดเป็นร้อยละ 94.67
- ความแม่นยำของการทำนายนิพจน์ระบุนาม (Named Entities) ของการรวมไลบรารีและการทำ Regular Expressions คิดเป็นร้อยละ 97.04

จากรูปที่ 4.10 จะสังเกตได้ว่า เมื่อดำเนินการรวมการทำนายของแต่ละไลบรารีเข้าด้วยกัน และสร้างเงื่อนไขจาก Regular Expressions นั้น ส่งผลให้ค่าความแม่นยำในการทำนายนิพจน์ระบุนามสูงที่สุด

นอกจากนี้ ทางผู้จัดทำได้ประเมินผลความแม่นยำของนิพจน์ระบุนาม (Named Entities) ในแต่ละประเภท เพื่อวิเคราะห์ว่าประเภทใดมีค่าความแม่นยำแตกต่างกันอย่างไร สามารถสรุปได้ ดังนี้

- การประเมินผลความแม่นยำในการติดแท็กคำว่า “PERSON”

```
----- PERSON Prediction Accuracies -----
Stanford Accuracy: 98.82%
NLTK Accuracy: 93.49%
spaCy Accuracy: 100.00%

** Combined Models and using Regular Expressions Accuracy: 100.00% **
```

รูปที่ 4.11 การประเมินผลความแม่นยำในการติดแท็กคำว่า “PERSON”

จากรูปที่ 4.11 ความแม่นยำในการติดแท็กคำว่า “PERSON” ของ Stanford NER คิดเป็นร้อยละ 98.82 ไลบรารี NLTK คิดเป็นร้อยละ 93.49 ไลบรารี spaCy คิดเป็นร้อยละ 100 และเมื่อรวมการทำนายของแต่ละไลบรารีเข้าด้วยกันพร้อมกับสร้างเงื่อนไขจาก

Regular Expressions มีความแม่นยำคิดเป็นร้อยละ 100 ซึ่งหมายความว่าไม่มีการทำนายผิดพลาดเลย

- การประเมินผลความแม่นยำในการติดแท็กคำว่า “ORGANIZATION”

```
----- ORGANIZATION Prediction Accuracies -----
Stanford Accuracy: 99.41%
NLTK Accuracy: 100.00%
spaCy Accuracy: 99.41%
** Combined Models and using Regular Expressions Accuracy: 99.41% **
```

รูปที่ 4.12 การประเมินผลความแม่นยำในการติดแท็กคำว่า “ORGANIZATION”

จากรูปที่ 4.12 ความแม่นยำในการติดแท็กคำว่า “ORGANIZATION” ของไลบรารี Stanford NER คิดเป็นร้อยละ 99.41 ไลบรารี NLTK คิดเป็นร้อยละ 100 ไลบรารี spaCy คิดเป็นร้อยละ 99.41 และเมื่อรวมการทำนายของแต่ละไลบรารีเข้าด้วยกันพร้อมกับสร้างเงื่อนไขจาก Regular Expressions มีความแม่นยำคิดเป็นร้อยละ 99.41 เนื่องจากเงื่อนไขในการรวมแบบจำลองก็จะทำการเลือกค่าทำนายที่เหมือนกันตั้งแต่ 2 จาก 3 ไลบรารีขึ้นไป และสิ่งที่แบบจำลอง NLTK ทำนายเป็นค่าที่แบบจำลองอีก 2 แบบไม่ได้ทำนายตรงกัน จึงส่งผลให้การรวมไลบรารีมีค่าความแม่นยำต่ำกว่า NLTK แต่หากมองในมุมของการทำนายภาพรวม ยังถือว่าการรวมไลบรารีมีค่าความแม่นยำมากที่สุด

- การประเมินผลความแม่นยำในการติดแท็กคำว่า “LOCATION”

```
----- LOCATION Prediction accuracies -----
Stanford Accuracy: 98.82%
NLTK Accuracy: 99.41%
spaCy Accuracy: 100.00%
** Combined Models and using Regular Expressions Accuracy: 100.00% **
```

รูปที่ 4.13 การประเมินผลความแม่นยำในการติดแท็กคำว่า “LOCATION”

จากรูปที่ 4.13 ความแม่นยำในการติดแท็กคำว่า “LOCATION” ของไลบรารี Stanford NER คิดเป็นร้อยละ 98.82 ไลบรารี NLTK คิดเป็นร้อยละ 99.41 ไลบรารี spaCy คิดเป็นร้อยละ 100 และเมื่อรวมการทำนายของแต่ละไลบรารีเข้าด้วยกันพร้อมกับสร้าง

เงื่อนไขจาก Regular Expressions มีความแม่นยำคิดเป็นร้อยละ 100 ซึ่งหมายความว่าไม่มีการทำนายผิดพลาดเลย

- การประเมินผลความแม่นยำในการติดแท็กคำว่า “DATE”

```
----- DATE Prediction accuracies -----
Stanford Accuracy: 94.67%
NLTK Accuracy: 96.45%
spaCy Accuracy: 100.00%

** Combined Models and using Regular Expressions Accuracy: 100.00% **
```

รูปที่ 4.14 การประเมินผลความแม่นยำในการติดแท็กคำว่า “DATE”

จากรูปที่ 4.14 ความแม่นยำในการติดแท็กคำว่า “DATE” ของไลบรารี Stanford NER คิดเป็นร้อยละ 94.67 ไลบรารี NLTK คิดเป็นร้อยละ 96.45 ไลบรารี spaCy คิดเป็นร้อยละ 100 และเมื่อรวมการทำนายของไลบรารีเข้าด้วยกันพร้อมกับสร้างเงื่อนไขจาก Regular Expressions มีความแม่นยำคิดเป็นร้อยละ 100 ซึ่งหมายความว่าไม่มีการทำนายผิดพลาดเลย

- การประเมินผลความแม่นยำในการติดแท็กคำว่า “MONEY”

```
----- MONEY Prediction accuracies -----
Stanford Accuracy: 100.00%
NLTK Accuracy: 100.00%
spaCy Accuracy: 100.00%

** Combined Models and using Regular Expressions Accuracy: 100.00% **
```

รูปที่ 4.15 รประเมินผลความแม่นยำในการติดแท็กคำว่า “MONEY”

จากรูปที่ 4.15 ความแม่นยำในการติดแท็กคำว่า “MONEY” ของไลบรารี Stanford NER คิดเป็นร้อยละ 100 ไลบรารี NLTK คิดเป็นร้อยละ 100 ไลบรารี spaCy คิดเป็นร้อยละ 100 และเมื่อรวมการทำนายของแต่ละไลบรารีเข้าด้วยกันพร้อมกับสร้างเงื่อนไขจาก Regular Expressions มีความแม่นยำคิดเป็นร้อยละ 100 ในบางครั้งอาจสรุปได้ว่า บท

สนทนานี้ไม่มีการกล่าวถึงค่าเงินต่าง ๆ จึงส่งผลให้ไลบรารีทุกไลบรารีมีค่าความแม่นยำสูงสุด

- การประเมินผลความแม่นยำในการติดแท็กประเภทของ PII Number ทุกประเภท

```
----- PII NUMBER Prediction accuracies -----
Stanford Accuracy: 95.27%
NLTK Accuracy: 95.27%
spaCy Accuracy: 95.27%

** Combined Models and using Regular Expressions Accuracy: 97.63% **
```

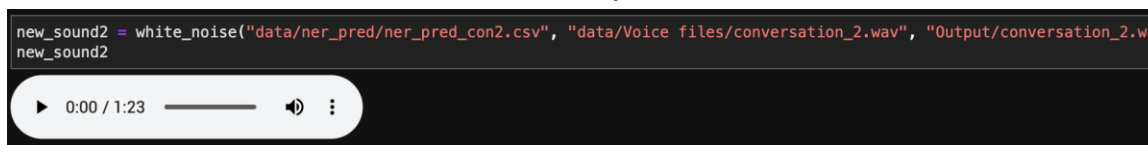
รูปที่ 4.16 การประเมินผลความแม่นยำในการติดแท็กประเภทของ PII Number ทุกประเภท

จากรูปที่ 4.16 ทางผู้จัดทำได้ดำเนินการประเมินผลความแม่นยำของเลขที่เป็นข้อมูลส่วนบุคคลทุก ๆ ประเภทเข้าด้วยกัน สามารถสรุปได้ว่า ความแม่นยำในการติดแท็กประเภทของ PII Number ทุกประเภทของไลบรารี Stanford NER คิดเป็นร้อยละ 95.27 ไลบรารี NLTK คิดเป็นร้อยละ 95.27 ไลบรารี spaCy คิดเป็นร้อยละ 95.27 และเมื่อรวมการทำนายของแต่ละไลบรารีเข้าด้วยกันพร้อมกับสร้างเงื่อนไขจาก Regular Expressions มีความแม่นยำคิดเป็นร้อยละ 97.63 สาเหตุที่ไลบรารีทั้ง 3 แบบมีค่าความแม่นยำเท่ากันเป็นเพราะทางผู้จัดทำไม่ได้มีการติดแท็กเลขในไลบรารีทั้ง 3 แบบ แต่มีการติดแท็กแค่ในการรวมไลบรารีเท่านั้น และสาเหตุที่ความแม่นยำของการทำนายไม่ถึงร้อยละ 100 นั้น อาจเป็นผลมาจากการแปลงเสียงพูดให้อยู่ในรูปแบบข้อความของ Cloud Speech to Text นั้นไม่แม่นยำมากพอ อาจจะทำนายตัวเลขหลักที่เงื่อนไขกำหนด หรือมีการแบ่งโทเค็นไว้ไม่เท่ากัน ทำให้ไม่สามารถติดแท็กได้อย่างสมบูรณ์

4.3 การแทนที่เสียงที่เป็นข้อมูลส่วนบุคคลด้วยเสียงรบกวน

เมื่อทำการแทนที่เสียงแล้วจึงดำเนินการบันทึกไฟล์เสียงที่ผ่านการแทนที่เสียงที่เป็นข้อมูลส่วนบุคคลด้วยเสียงรบกวนเป็นไฟล์เสียงประเภท .wav ดังรูปที่ 4.17

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



รูปที่ 4.17 การบันทึกเสียงที่ผ่านการแทนที่เสียงที่เป็นข้อมูลส่วนบุคคลด้วยเสียงรบกวน

บทที่ 5

บทสรุป

5.1 สรุปผลโครงการ

5.1.1 การแปลงเสียงพูดให้อยู่ในรูปแบบข้อความ

การแปลงเสียงพูดให้อยู่ในรูปแบบข้อความนั้น หากเป็นการประเมินผลโดยไม่คำนึงถึงความถูกต้องของเครื่องหมายวรรคตอน ถือว่าค่าของความแม่นยำอยู่ในระดับที่ดี อาจจะมีการแปลงชื่อบุคคลที่ไม่ตรงกับข้อมูลบทสนทนาจริงเล็กน้อย อาจเป็นสาเหตุมาจากเสียงที่ใช้ในการดำเนินการบันทึกเสียงที่แต่ละบุคคลมีสำเนียงการพูดที่ไม่เหมือนกัน เช่น นามสกุล Applebaum เมื่อเป็นเสียงของ Siri Male ทางแบบจำลองแปลงได้เป็น 2 โทเค็น คือ “Appel” และ “board.” แต่เมื่อเป็นเสียงของ “Siri Female” ทางแบบจำลองกลับแปลงคำได้ถูกต้อง จึงสรุปได้ว่าบางครั้งสำเนียงการพูดของแต่ละตัวบุคคลอาจส่งผลต่อความแม่นยำของการแปลงข้อมูลเสียงให้อยู่ในรูปแบบข้อความ นอกจากนี้ ยังมีการแปลงเลขที่ผิดพลาดไปบ้าง เช่น เมื่อสิริพูดว่า “oh” ในบางครั้งแบบจำลองจะแปลงเป็นเลข “0” ซึ่งส่งผลให้ความแม่นยำของแบบจำลองลดลง

5.1.2 การตรวจจับคำที่เป็นข้อมูลส่วนบุคคลจากข้อมูลรูปแบบข้อความ

ในขั้นตอนนี้ ผู้จัดทำจะอธิบายรายละเอียดของแต่ละไลบรารี ดังนี้

- Stanford NER สามารถติดแท็กบุคคล และค่าเงิน ได้ค่อนข้างแม่นยำ ส่วนนิพจน์ระบุนาม (Named Entities) ประเภทอื่น ๆ มีความแม่นยำเฉลี่ยเท่า ๆ กันกับไลบรารีอื่น ๆ แต่ในการติดแท็กวันที่ ด้วยข้อจำกัดของไลบรารีที่ไม่มีการติดแท็กตัวเลขที่เป็นประเภท Cardinal เหมือนไลบรารี 2 แบบ จึงส่งผลให้มีการติดแท็กตัวเลขธรรมดา เป็นประเภทของวันที่ (Date) ทำให้ความแม่นยำของไลบรารีลดลง
- NLTK สามารถติดแท็กองค์กร ได้แม่นยำมากที่สุด ส่วนนิพจน์ระบุนาม (Named Entities) ประเภทอื่น ๆ มีความแม่นยำเฉลี่ยเท่า ๆ กันกับไลบรารีอื่น ๆ แต่ไลบรารีนี้ มักมีการติดแท็กที่ผิดพลาดตรงส่วนของสถานที่ กล่าวคือ หากโทเค็นนั้น ๆ ขึ้นต้นด้วยตัวอักษรพิมพ์ใหญ่ เช่น คำว่า “Hello” ไลบรารีจะติดแท็กเป็นสถานที่ทันที นอกจากนี้ ไลบรารีนี้สามารถติดแท็กตัวเลขประเภท Cardinal ได้ดีที่สุด แต่เนื่องจากทางผู้จัดทำไม่ได้มุ่งเน้นติดแท็กตัวเลขจากไลบรารี จึงไม่ได้ส่งผลต่อความแม่นยำในส่วนนี้

- spaCy จากผลลัพธ์การประเมินผลความแม่นยำ จะสังเกตได้ว่าส่วนใหญ่แล้ว spaCy มักมีค่าความแม่นยำสูงในการติดแท็กโทเค็น แต่หากให้สรุปเป็นรายประเภท จะสามารถสรุปได้ว่า ไลบรารีนี้สามารถติดแท็กบุคคล สถานที่ วันที่ และค่าเงินได้ดีที่สุด ส่วนนิพจน์ระบุนาม (Named Entities) ประเภทอื่น ๆ มีความแม่นยำเฉลี่ยเท่า ๆ กันกับไลบรารีอื่น ๆ แต่เนื่องจากการติดแท็กของไลบรารีนี้ยังมีความไม่แม่นยำอยู่บ้าง ทางผู้จัดทำจึงมีความเห็นว่าควรรวมไลบรารีเข้าด้วยกันเพื่อเพิ่มประสิทธิภาพในการติดแท็ก

ในส่วนของการรวมไลบรารีเข้าด้วยกัน มีความแม่นยำค่อนข้างสูง ซึ่งเฉลี่ยแล้วคิดเป็นร้อยละ 90 ถือเป็นค่าความแม่นยำที่น่าพึงพอใจ

และในส่วนสุดท้าย คือ การตรวจจับเลขที่เป็นข้อมูลส่วนบุคคล โดยใช้ Regular Expressions ก็มีความแม่นยำค่อนข้างสูงเช่นกัน แต่ในบางครั้งอาจไม่แม่นยำอย่างสมบูรณ์เนื่องจากรูปแบบการแปลงตัวเลขของ Cloud Speech to Text อาจแบ่งโทเค็นได้ไม่ตรงกับตัวเลขที่ควรจะเป็น เช่น เลขบัตรเดบิตหรือบัตรเครดิต 16 หลัก ทางแบบจำลองอาจมีรูปแบบการแปลงตัวเลขได้เพียงแค่ 13 หลัก แล้วจึงแบ่งเลขอีก 3 หลักหลังเป็นอีกโทเค็น ซึ่งในเงื่อนไขมักจะติดแท็กเลขที่มากกว่า 9 หลักขึ้นไปโดยไม่สนใจเครื่องหมายต่าง ๆ เช่น +111-111-111-1111 หรือ 111-111-1111 เป็นต้น แต่หากพิจารณาถึงภาพรวมของค่าความแม่นยำแล้ว ถือเป็นที่น่าพึงพอใจเช่นกัน

5.1.3 การแทนที่เสียงที่เป็นข้อมูลส่วนบุคคลด้วยเสียงรบกวน

การแทนที่เสียงที่เป็นข้อมูลส่วนบุคคลด้วยเสียงรบกวน ในบางช่วงที่มีการแทนที่เสียงรบกวนนั้นอาจเกินหรือขาดจากระยะเวลาของคำพูดที่เป็นข้อมูลส่วนบุคคล แต่โดยภาพรวมแล้วถือว่าปิดบังคำพูดที่เป็นข้อมูลส่วนบุคคลได้ดี

5.2 ปัญหาในการทำโครงงานและสรุปผล

โดยส่วนใหญ่แล้ว ปัญหาในการทำโครงงานนี้ คือ ความแม่นยำของการแปลงข้อมูลเสียงให้อยู่ในรูปแบบข้อความนั้น มีความแม่นยำในระดับปานกลางจนถึงค่อนข้างสูง แต่เมื่อดำเนินการเข้าสู่กระบวนการตรวจจับคำที่เป็นข้อมูลส่วนบุคคลจากข้อมูลรูปแบบข้อความ ส่งผลให้ไลบรารีไม่สามารถติดแท็กประเภทของโทเค็นที่ควรจะมีนิพจน์ระบุนาม (Named Entities) ได้ เช่น ชื่อบุคคล หรือส่วนเล็ก ๆ ของเลขที่เป็นข้อมูลสำคัญ จึงอาจส่งผลให้เป็นปัญหาต่อการปิดบังคำที่เป็นข้อมูลส่วนบุคคลใน

ขั้นตอนสุดท้ายได้ และในการเป็นการแทนที่เสียงที่เป็นข้อมูลส่วนบุคคลด้วยเสียงรบกวน ในบางช่วงของการแทนที่เสียงรบกวนอาจเกินหรือขาดจากระยะเวลาของคำพูดที่เป็นข้อมูลส่วนบุคคล

จากการทำโครงการนี้ได้ส่งผลให้ผู้จัดทำเข้าใจเรื่องการประมวลผลภาษาธรรมชาติ (Natural Language Processing) และ เข้าใจการทำงานของระบบการรู้จำเสียงพูด (Speech Recognition) มากยิ่งขึ้น สามารถปกปิดข้อมูลสำคัญในรูปแบบเสียงโดยการแทรกเสียงรบกวนแทนที่เสียงข้อมูลส่วนบุคคลเพื่อเพิ่มความปลอดภัยในการนำข้อมูลเหล่านั้นไปดำเนินการวิเคราะห์ต่อไปในอนาคต

5.3 แนวทางในการพัฒนาต่อ

ทางผู้จัดทำจะดำเนินการหาวิธีการเพิ่มความแม่นยำของการแปลงข้อมูลเสียงให้อยู่ในรูปแบบข้อความให้มีความแม่นยำมากขึ้น เพื่อให้การติดแท็กโทเค็นตรงเจี้ยนไขมากที่สุด และอาจมีการดำเนินการพัฒนาต่อเพิ่มในด้านของการตรวจจับข้อมูลส่วนบุคคล เช่น หลังจากที่ได้ติดแท็กโทเค็นนั้นแล้ว อาจมีการฝึกฝนแบบจำลองอื่น ๆ เพิ่มเติม เพื่อตรวจจับว่าโทเค็นนั้น ๆ เป็นข้อมูลส่วนบุคคลที่จำเป็นต้องปกปิดจริงหรือไม่ แต่ด้วยวิธีการนั้นอาจจะต้องดำเนินการสร้างชุดข้อมูลพร้อมกับการเฉลยผลการตรวจจับว่าเป็นข้อมูลส่วนบุคคลหรือไม่ เป็นจำนวนมาก เพื่อให้แบบจำลองสามารถทำนายได้อย่างแม่นยำ

รายการอ้างอิง

- [1] ศ. ศวัสดิ์พงษ์ศาธา, “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] “การเปิดใช้งาน Cloud Speech API. Cloud Speech API... | by Flame Sillawat | Medium”. [ออนไลน์]. Available at: <https://medium.com/@flame.zaxaou/การเปิดใช้งาน-cloud-speech-api-6b7a05c72752>.
- [8] K. Surapong, “Natural Language Processing (NLP) คืออะไร รวมคำศัพท์เกี่ยวกับ Natural Language Processing (NLP) – NLP ep.1”, 2018.
- [9] R. Alfred, L. C. Leong, C. K. On, และ P. Anthony, “Malay named entity recognition based on rule-based approach”, 2014.
- [10] “Natural Language Processing is Fun! | by Adam Geitgey | Medium”. [ออนไลน์]. Available at: <https://medium.com/@ageitgey/natural-language-processing-is-fun-9a0bff37854e>.
- [11] “Named Entity Recognition with Stanford NER Tagger”. [ออนไลน์]. Available at: <https://pythonprogramming.net/named-entity-recognition-stanford-ner-tagger/>.

- [12] “Natural Language Toolkit — NLTK 3.5 documentation”. [ออนไลน์]. Available at: <https://www.nltk.org/>.
- [13] “spaCy 101: Everything you need to know · spaCy Usage Documentation”. [ออนไลน์]. Available at: <https://spacy.io/usage/spacy-101>.
- [14] “7. Extracting Information from Text”. [ออนไลน์]. Available at: <https://www.nltk.org/book/ch07.html>.
- [15] “Regular Expression Matching Can Be Simple And Fast”. [ออนไลน์]. Available at: <https://swtch.com/~rsc/regexp/regex1.html>.
- [16] สุกวัญ, “การตรวจเทียบภายนอกหากลักลอกในงานวิชาการโดยใช้แบบจำลองซัพพอร์ตเวกเตอร์แมชชีนและ การวัดค่าความละเอียดของข้อความ”, 2560.
- [17] “How do MP3 and WAV Files Differ?” [ออนไลน์]. Available at: <https://www.dawsons.co.uk/blog/how-do-mp3-and-wav-files-differ>.
- [18] “MP4 to WAV online file converter”. [ออนไลน์]. Available at: <https://audio.online-convert.com/convert/mp4-to-wav>.