**การปกปิดข้อมูลเสียงพูดเพื่อปกป้องข้อมูลที่ระบุตัวบุคคล**

**ณัฏฐณิชา ชัยศิริพานิช1 และ ประวิตรานันท์ บุตรโพธิ์2**

*1คณะเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง กรุงเทพฯ*

*2คณะเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง กรุงเทพฯ*

*Emails: 60070135@it.kmitl.ac.th, 60070148@it.kmitl.ac.th*

**บทคัดย่อ**

ในปัจจุบันข้อมูลนับว่าเป็นสิ่งสำคัญต่อการนำไปประยุกต์ใช้กับปัญญาประดิษฐ์เพื่อพัฒนาประสิทธิภาพขององค์กร แต่ข้อมูลเหล่านั้นมักมีข้อจำกัดในเรื่องของข้อมูลส่วนบุคคล โดยเฉพาะข้อมูลของลูกค้า ซึ่งการบันทึกเสียงการสนทนาการทำธุรกรรมกับทางธนาคารก็ถือว่ามีข้อมูลส่วนบุคคลของลูกค้าเป็นจำนวนมาก จึงเสียงต่อการลักลอบนำข้อมูลไปใช้ในทางที่ไม่ถูต้อง

ผู้จัดทำได้เล็งเห็นถึงปัญหานี้และได้ทดลองสร้างระบบโดยใช้ปัญญาประดิษฐ์ โดยภายในระบบมีการแปลงไฟล์เสียงพูดให้อยู่ในรูปแบบข้อความ จากนั้นตรวจจับคำที่เป็นข้อมูลส่วนบุคคลพร้อมกับเก็บระยะเวลาที่พูดในไฟล์เสียงและทำการแทนที่คำพูดที่เป็นข้อมูลส่วนบุคคลด้วยเสียงรบกวน และนำผลลัพธ์ที่เป็นไฟล์เสียงที่ผ่านการปกปิดข้อมูลส่วนบุคคลแล้วไปใช้ประโยชน์ในการวิเคราะห์ทางด้านอื่น ๆ ต่อไป

***คำสำคัญ –*** *ข้อมูลส่วนบุคคล; ศูนย์บริการข้อมูลลูกค้าทางโทรศัพท์ (Call Center); การประมวลผลภาษาธรรมชาติ (Natural Language Processing); นิพจน์ระบุนาม (Named Entities); การแปลงเสียงพูดให้อยู่ในรูปแบบข้อความ (Speech-to-Text)*

**1. บทนำ**

ปัจจุบันการละเมิดข้อมูลส่วนบุคคลนั้นเกิดขึ้นได้หลายรูปแบบ ซึ่งการละเมิดข้อมูลส่วนบุคคลจากการบันทึกบทสนทนาการทำธุรกรรมกับทางธนาคารก็ถือเป็นหนึ่งในปัญหาการละเมิดสิทธิส่วนบุคคลเช่นกัน ทางผู้จัดทำได้เล็งเห็นถึงความสำคัญของการรักษาข้อมูลส่วนบุคคลของลูกค้าในการทำธุรกรรมกับทางธนาคารผ่านศูนย์บริการข้อมูลลูกค้าทางโทรศัพท์โดยจะมีการทำการตรวจจับการสนทนาบางส่วนในไฟล์บันทึกเสียง โดยเฉพาะส่วนที่เป็นข้อมูลส่วนบุคคลของลูกค้า เช่น ชื่อ - นามสกุล วันเกิด เบอร์โทรศัพท์ เลขที่บัญชี และเลขหน้าบัตรเครดิต หรือเดบิต ก่อนจะนำข้อมูลผลลัพธ์เหล่านั้นไปใช้ประโยชน์ในด้านอื่น ๆ โดยมีการทดลองสร้างระบบในการปกปิดข้อมูลที่ระบุตัวบุคคล ซึ่งภายในระบบจะดำเนินการแปลงไฟล์เสียงพูดให้อยู่ในรูปแบบข้อความ จากนั้นวิเคราะห์คำที่เป็นข้อมูลส่วนบุคคลพร้อมกับเก็บค่าของระยะเวลาที่พูดในไฟล์เสียงนั้น ระบบจะทำการแทรกเสียงรบกวนแทนที่คำพูดที่เป็นข้อมูลส่วนบุคคล และผลลัพธ์ที่ได้คือไฟล์เสียงที่ผ่านการปกปิดข้อมูลส่วนบุคคลแล้ว ซึ่งสามารถนำไปใช้ประโยชน์ในการวิเคราะห์ทางด้านอื่น ๆ ต่อไป

**2. แนวคิด ทฤษฎี เทคโนโลยีและเครื่องมือที่เกี่ยวข้อง**

ผู้จัดทำได้แบ่งรายละเอียดของแนวคิด ทฤษฎี เทคโนโลยีและเครื่องมือที่เกี่ยวข้องที่จะนำมาใช้กับการพัฒนาระบบเป็น 3 ส่วนหลัก ๆ ดังนี้

**2.1 แนวคิดที่เกี่ยวข้อง**

*2.2.1 สิทธิความเป็นอยู่ส่วนบุคคล*

มีการบัญญัติรับรองสิทธิดังกล่าวในรัฐธรรมนูญแห่งราชอาณาจักรไทย พ.ศ. 2560 มาตรา 32 บัญญัติไว้ว่า “สิทธิของบุคคลในครอบครัว เกียรติยศ ชื่อเสียง ตลอดจนความเป็นอยู่ส่วนบุคคล ย่อมได้รับความคุ้มครอง การกล่าวหรือไขข่าวแพร่หลายซึ่งข้อความหรือภาพไม่ว่าด้วยวิธีใดไปยังสาธารณชนอันเป็นการละเมิดหรือกระทบถึงสิทธิของบุคคลในครอบครัว เกียรติยศ ชื่อเสียง หรือความเป็นอยู่ส่วนบุคคล จะกระทำมิได้ เว้นแต่กรณีที่เป็น ประโยชน์ต่อสาธารณะ บุคคลย่อมมีสิทธิได้รับความคุ้มครองจากการแสวงประโยชน์โดยมิชอบจากข้อมูลส่วนบุคคลที่เกี่ยวกับตน ทั้งนี้ ตามที่กฎหมายบัญญัติ” [1]

**2.2 ทฤษฎีที่เกี่ยวข้อง**

*2.2.1 การรู้จำเสียงพูด (Speech Recognition)*

เป็นสิ่งที่ช่วยให้โปรแกรมสามารถประมวลผลคำพูดของมนุษย์ให้อยู่ในรูปแบบลายลักษณ์อักษร โดยเน้นที่การแปลงเสียงพูดจากรูปแบบคำพูดเป็นข้อความ [2]

*2.2.2 การประมวลผลภาษาธรรมชาติ (Natural Language Processing: NLP)*

เป็นศาสตร์ที่ศึกษาเกี่ยวกับการทำให้คอมพิวเตอร์สามารถสื่อสารโต้ตอบด้วยภาษาของมนุษย์ และทำให้คอมพิวเตอร์เข้าใจภาษามนุษย์มากขึ้น เช่น Siri, Google Assistant และ Alexa [3]

*2.2.3* *การรู้จำนิพจน์ระบุนาม (Named Entity Recognition)*

การรู้จำนิพจน์ระบุนาม (Named Entity Recognition) เป็นขั้นตอนที่สำคัญสำหรับการพัฒนาระบบประมวลผลเอกสาร โดยเฉพาะอย่างยิ่งสำหรับระบบที่เกี่ยวข้องกับการเข้าถึงข้อมูล เช่น ระบบสกัดสารสนเทศ (Information Extraction) หรือในระบบค้นคืนเอกสาร (Information Retrieval) [4] ผู้จัดทำได้นำทฤษฎีนี้มาใช้ในการพัฒนาระบบส่วนของการตรวจจับคำที่เป็นข้อมูลส่วนบุคคล

*2.2.4 Jaccard’s Coefficient Similarity*

เป็นสถิติประยุกต์แนวคิดในทฤษฎีเซตเพื่อใช้เปรียบเทียบความคล้ายคลึงและความหลากหลายของกลุ่มตัวอย่าง แนวคิดของค่าสัมประสิทธิ์ Jaccard’s Coefficient Similarity คือ การวัดค่าความคล้ายคลึงระหว่างกลุ่มประชากร 2 กลุ่ม โดยคำนวณจากขนาดของประชากรที่ทั้งสองกลุ่มมีตัวอย่างร่วมกัน [5] ดังสมการที่ 1

(1)

**2.2 เทคโนโลยีและเครื่องมือที่เกี่ยวข้อง**

*2.2.1 Cloud Speech to Text by Google Cloud*

กูเกิลคลาวด์แพลตฟอร์มเป็นเว็บเซิร์ฟเวอร์ที่ให้บริการคลาวด์ ซึ่งภายในกูเกิลคลาวด์แพลตฟอร์มนั้นมีบริการแยกย่อยอีก เช่น Cloud Speech to Text, Cloud Storage, Compute Engine, และ Machine Learning เป็นต้น ทั้งนี้การใช้งานกูเกิลคลาวด์แพลตฟอร์มจะคิดค่าใช้จ่ายตามจำนวนการใช้งาน

ทางผู้จัดทำเลือกบริการ Cloud Storage ในการเก็บไฟล์เสียง และใช้ Cloud Speech to Text ไลบรารี Speech ในการแปลงเสียงพูดให้อยู่ในรูปแบบข้อความ [6]

*2.2.2 Stanford Named Entity Recognizer (Stanford NER)*

เป็นการประยุกต์ใช้จากภาษาจาวาสำหรับการรู้จำนิพจน์ระบุนาม (Named Entity Recognizer) ซึ่งเป็นการจัดประเภทของคำในข้อความ เช่น ชื่อสิ่งของ ชื่อบุคคล และบริษัท เป็นการกำหนดโครงสร้างการสกัดคุณสมบัติที่เหมาะสมสำหรับการรู้จำนิพจน์ระบุนาม [7]

*2.2.3 Natural Language Toolkit (NLTK)*

เป็นแพลตฟอร์มที่นิยมในโปรแกรมภาษาไพทอน เพื่อทำงานกับข้อมูลภาษาของมนุษย์ พร้อมกับชุดของไลบรารีที่ช่วยในการประมวลข้อความ แบ่งประเภทของคำ (Classification) การแบ่งโทเค็นของคำ (Tokenization) การตัดคำ (Stemming) การติดแท็กคำ (Tagging) และการแยกวิเคราะห์คำ (Parsing) [8]

*2.2.4 spaCy*

เป็นไลบรารีสำหรับการทำการระมวลผลภาษาธรรมชาติขั้นสูงในภาษาไพทอน โดยที่ spaCy จะทำความเข้าใจข้อความจำนวนมาก สามารถใช้ในการดำเนินการสกัดข้อมูล (Information Extraction) หรือระบบการทำความเข้าใจภาษาธรรมชาติเพื่อดำเนินการประมวลผลข้อความล่วงหน้าสำหรับการเรียนรู้เชิงลึก (Deep Learning) [9]

*2.2.5 Regular Expressions*

เป็นสัญลักษณ์ที่ใช้ระบุชุดของอักขระตัวอักษร เมื่อชุดของอักขระตัวอักษรที่เฉพาะเจาะจงนั้นอยู่ในชุดอักขระตัวอักษรที่มีการกำหนดให้เป็น Regular Expressions โดยทั่วไปแล้วจะใช้สัญลักษณ์ “\*”, “+”, “?”, “()” และ “|” ในการกำหนดเงื่อนไขของชุดตัวอักษร [10]

*2.2.6**Pydub*

เป็นไลบรารีหนึ่งของภาษาไพทอนที่ใช้ในการจัดการกับข้อมูลที่เป็นไฟล์เสียง

**3. ขั้นตอนการพัฒนาระบบ**

ผู้จัดทำได้นำเทคโนโลยีและเครื่องมือมาประยุกต์ใช้โดยมีการดำเนินงานตามแผนงานการปกป้องข้อมูลที่ระบุตัวบุคคล ดังรูปที่ 1

Diagram

Description automatically generated

**รูปที่ 1.** แผนงานการปกป้องข้อมูลที่ระบุตัวบุคคล

การดำเนินงานประกอบด้วย 3 ส่วนหลัก ดังรูปที่ 3.1 มีรายละเอียดการดำเนินงาน ดังนี้

**3.1 การแปลงเสียงพูดให้อยู่ในรูปแบบข้อความ**

*3.1.1 ประยุกต์ใช้ Cloud Speech to Text*

สร้างโปรเจกต์บนกูเกิลคลาวด์ และเปิดใช้งาน API 2 ประเภท คือ Cloud Storage และ Cloud Speech to Text จากนั้นอัปโหลดไฟล์เสียงขึ้นบน Cloud Storage ดังรูปที่ 2

Graphical user interface, application, table

Description automatically generated

**รูปที่ 2.** อัปโหลดไฟล์เสียงขึ้นบน Cloud Storage

ต่อมาทำการแปลงเสียงพูดให้อยู่ในรูปแบบข้อความโดยการสร้างฟังก์ชันต่าง ๆ ได้แก่ ฟังก์ชัน print\_word\_offsets ระบุเวลากับคำในบทสนทนา (Timestamp) โดยระบุเวลาที่เริ่มต้นในแต่ละคำ และเวลาที่สิ้นสุดของคำ ๆ นั้น โดยหน่วยเวลาเป็นวินาทีเพื่อให้ง่ายต่อการเข้าใจ ฟังก์ชัน print\_sentences แสดงผลประโยคที่ผ่านการแปลงเสียงพูดให้อยู่ในรูปแบบข้อความและแสดงผลค่าความเชื่อมั่น (Confidence) ส่วนสุดท้ายในขั้นตอนการแปลงเสียงพูดให้อยู่ในรูปแบบของข้อความในฟังก์ชัน speech\_to\_text ใช้โมดูลของไลบรารี Speech ในการแปลงข้อมูลเสียงพูดให้อยู่ในรูปแบบข้อความ และระบุไฟล์

Text

Description automatically generated

**รูปที่ 3** ฟังก์ชันการแปลงเสียงพูดให้อยู่ในรูปแบบข้อความ

*3.1.2 บันทึกไฟล์ให้อยู่ในรูปแบบ Dictionary*

บันทึกเป็นประเภทไฟล์ JSON เพื่อทำการตรวจจับคำที่เป็นข้อมูลส่วนบุคคลในขั้นตอนต่อไป

**3.2 การตรวจจับคำที่เป็นข้อมูลส่วนบุคคลจากข้อมูลรูปแบบข้อความ**

*3.2.1 กระบวนการตรวจจับนิพจน์ระบุนาม*

ขั้นตอนนี้มีการใช้ไลบรารีทั้งหมด 3 ไลบรารี เพื่อเพิ่มความแม่นยำในการตรวจจับนิพจน์ระบุนาม ได้แก่ Stanford NER, NLTK และ spaCy มีกระบวนการดำเนินงาน ดังนี้

- พัฒนาไลบรารี Stanford NER โดยเลือกประเภทของนิพจน์ระบุนามในการติดแท็กบทสนทนาทั้งหมดเป็นจำนวน 5 ประเภท ได้แก่ PERSON, ORGANIZATION, LOCATION, DATE และ MONEY ซึ่งในฟังก์ชันมีการทำ Word Tokenization เพื่อแยกโทเค็นของคำในข้อความ ต่อมามีการติดแท็กนิพจน์ระบุนามจากอัลกอริทึมของ Stanford NER จากนั้นสร้างเงื่อนไขเก็บเฉพาะโทเค็นที่เป็นนิพจน์ระบุนามเท่านั้น จากนั้นจึงแก้ไขประเภทของนิพจน์ระบุนามที่ถูกติดแท็ก เพื่อให้ประเภทของนิพจน์ระบุนามตรงกับไลบรารีอื่น ๆ เช่น คำว่า “ORG” ที่ติดแท็กไว้ จะดำเนินการเปลี่ยนเป็นคำว่า “ORGANIZATION” จากนั้นทำการจับคู่โทเค็นที่ไลบรารีแบ่งออกมาเทียบกับโทเค็นที่ Cloud Speech to Text แบ่งไว้ เพื่อให้แน่ใจว่าโทเค็นที่ถูกติดแท็กนั้นตรงกับระยะเวลาที่ Cloud Speech to Text ทำนายออกมา และเก็บค่าของคำที่ติดแท็กได้ พร้อมกับประเภทของนิพจน์ระบุนาม

- พัฒนาไลบรารี NLTK ผู้จัดทำได้เลือกการติดแท็กในบทสนทนาเป็นจำนวนทั้งหมด 6 ประเภท ได้แก่ ORGANIZATION, PERSON, LOCATION, GPE, DATE และ MONEY โดยเริ่มจากการทำ Word Tokenization จากนั้นทำการติดแท็กนิพจน์ระบุนามจากอัลกอริทึม NLTK ซึ่งต้องมีการติดแท็กส่วนของประโยค (Part-of-Speech) ก่อนจึงจะติดแท็กได้ และกระบวนการหลังจากนั้นมีวิธีการทำเช่นเดียวกันกับ Stanford NER คือ เลือกโทเค็นที่เป็นนิพจน์ระบุนาม และทำการเปลี่ยนประเภทนิพจน์ระบุนามให้เหมือนกันทุกไลบรารี จากนั้นจับคู่โทเค็นที่ไลบรารีแบ่งเทียบกับโทเค็นของ Cloud Speech to Text และเก็บค่าของโทเค็น

- พัฒนาไลบรารี spaCy โดยผู้จัดทำได้เลือกการติดแท็กในบทสนทนาเป็นจำนวนทั้งหมด 6 ประเภท ได้แก่ ORGANIZATION, PERSON, LOCATION, GPE, DATE และ MONEY ในฟังก์ชันมีการใช้อัลกอริทึมของ spaCy ซึ่งในอัลกอริทึมจะดำเนินการวิเคราะห์ข้อความต่าง ๆ อัตโนมัติ ซึ่งสามารถเรียกดูค่าได้จากอัลกอริทึมได้ทันที และกระบวนการหลังจากนั้นมีวิธีการทำเช่นเดียวกันกับ Stanford NER และ NLTK คือ เลือกโทเค็นที่เป็นนิพจน์ระบุนาม และทำการเปลี่ยนประเภทนิพจน์ระบุนามให้เหมือนกันทุกไลบรารี จากนั้นจับคู่โทเค็นที่แบบจำลองแบ่งเทียบกับโทเค็นของ Cloud Speech to Text และเก็บค่าของโทเค็น

*3.2.2 เลือกการทำนายประเภทของนิพจน์ระบุนามที่เหมือนกันตั้งแต่ 2 ใน 3 ของไลบรารี*

เลือกการทำนายประเภทของนิพจน์ระบุนามที่เหมือนกันตั้งแต่ 2 จาก 3 ไลบรารีขึ้นไปจากการสร้างฟังก์ชันจับคู่โทเค็นที่มีการทำนายนิพจน์ระบุนามค่าเดียวกัน และเก็บค่าของโทเค็นนั้นใหม่ เพื่อนำไปใช้วิเคราะห์กระบวนการถัดไป ในที่นี้ ผู้จัดทำขอแทนผลลัพธ์ของกระบวนการนี้ว่าค่าทำนายจริง

*3.2.3 สร้างประเภทของนิพจน์ระบุนามเพิ่มเพื่อติดแท็กเลขที่เป็นข้อมูลส่วนบุคคลโดยใช้ Regular Expressions*

ขั้นตอนนี้มีการดึงโทเค็นคำของ Cloud Speech to Text เฉพาะที่เป็นเลขมาตรวจสอบเงื่อนไขเพื่อติดแท็กเลขที่เป็นข้อมูลส่วนบุคคลเท่านั้น โดยแบ่งประเภทของเลขที่เป็นข้อมูลส่วนบุคคลไว้ 5 ประเภท คือ IDCARD (เลขบัตรประชาชน 13 หลัก) PHONENUM (เบอร์โทรศัพท์ 10 หลัก) ACCNUM (เลขบัญชี 9 หลัก) CARDNUM (เลขบัตรเดบิต หรือบัตรเครดิต 16 หลัก) และ PIINUM (เลขอื่น ๆ ที่ไม่เข้าเงื่อนไขประเภทก่อนหน้านี้ แต่มีตั้งแต่ 9 หลักขึ้นไป มีไว้ในกรณีที่ Cloud Speech to Text แปลงเป็นข้อความออกมาได้ไม่แม่นยำ) จากนั้นนำค่าที่ได้ไปรวมกับค่าทำนายจริง และเก็บค่านั้นไว้ในรูปแบบไฟล์ CSV เพื่อนำไปดำเนินการต่อในขั้นถัดไป

*3.2.4 เก็บค่าต่าง ๆ ให้อยู่ในรูปของไฟล์ CSV*

หลังจากทำนายนิพจน์ระบุนามทั้งหมดแล้ว จึงจัดเก็บค่าเหล่านั้นให้อยู่ในรูปแบบตารางและบันทึกเป็นไฟล์ CSV โดยมีจำนวนทั้งหมด 5 คอลัมน์ ได้แก่ ลำดับโทเค็น โทเค็นคำ เวลาที่เริ่มพูดโทเค็นนั้นในไฟล์เสียง เวลาที่พูดโทเค็นนั้นจบ และประเภทของนิพจน์ระบุนาม

**3.3 การแทนที่เสียงที่เป็นข้อมูลส่วนบุคคลด้วยเสียงรบกวน**

นำข้อมูลไฟล์ CSV มาใช้ในการดำเนินงาน คือ นำคอลัมน์ของเวลาเริ่มต้นของคำพูดและเวลาสิ้นสุดมาใช้ โดยแปลงให้ค่าของเวลาอยู่ในหน่วยของมิลลิวินาที จากนั้นแทนที่เสียงรบกวนในช่วงเวลาที่คำนวณไว้ จากนั้นดำเนินการบันทึกไฟล์เสียงที่มีการแทนที่ข้อมูลส่วนบุคคลแล้วเป็นไฟล์ประเภท .wav เพื่อที่จะสามารถนำข้อมูลไปวิเคราะห์ประโยชน์ในด้านอื่น ๆ ต่อได้

**3.5 การประเมินผล (Evaluation)**

มีการประเมินผล 2 กระบวนการหลัก ๆ คือ ประเมินผลความแม่นยำของการแปลงเสียงพูดให้อยู่ในรูปแบบข้อความโดยนำแนวคิดของ Jaccard’s Coefficient Similarity มาประยุกต์ใช้ในการประเมินผล และกระบวนการประเมินผลความแม่นยำของการตรวจจับคำที่เป็นข้อมูลส่วนบุคคลจากข้อมูลรูปแบบข้อความ จากการสร้างผลเฉลยของการทำนายข้อความและโทเค็นต่าง ๆ เพื่อใช้ตรวจสอบความแม่นยำในการทำนายของไลบรารีทั้งหมดโดยวัดจากค่า Accuracy

**4. การทดลองและผลลัพธ์ที่ได้จากระบบ**

ผู้จัดทำได้แบ่งการแสดงผลการทดลองและผลลัพธ์ที่ได้เป็น 4 หัวข้อหลัก มีรายละเอียด ดังนี้

**4.1 การวิเคราะห์ข้อมูลที่สร้างขึ้น**

ในการเตรียมข้อมูลเกิดปัญหาที่ไม่สามารถหาชุดข้อมูลจากแหล่งข้อมูลสาธารณะมาพัฒนาระบบได้เนื่องจากข้อมูลเหล่านี้มีข้อจำกัดในเรื่องของข้อมูลส่วนบุคคล จึงต้องสร้างชุดข้อมูลขึ้นมาเอง โดยมีรายละเอียดการสร้างข้อมูล ดังนี้

*4.1.1 สร้างบทสนทนาระหว่างลูกค้าและศูนย์บริการข้อมูลลูกค้าทางโทรศัพท์*

สร้างชุดข้อมูลในรูปแบบข้อความเป็นจำนวนทั้งหมด 23 บทสนทนาเพื่อใช้ในการพัฒนาและประเมินผลระบบ จากการวิเคราะห์ประโยคในบทสนทนาคิดเป็น 566 ประโยค มีจำนวนเฉลี่ยทั้งหมด 24.61 ประโยค แบ่งย่อยลงไปเป็นการวิเคราะห์คำที่ยังไม่ผ่านการทำความสะอาดข้อมูลมีทั้งหมด 4,095 คำ โดยที่ใน 1 บทสนทนาจะมีคำเป็นจำนวนเฉลี่ย 178.04 คำ และหากวิเคราะห์คำผ่านการทำความสะอาดข้อมูลแล้ว กล่าวคือ ดำเนินการตัดเครื่องหมายวรรคตอนและ Stop words บางส่วนออก มีทั้งหมด 1732 คำ ค่าเฉลี่ยใน 1 บทสนทนาจะมีคำโดยเฉลี่ยจำนวน 75.3 คำ

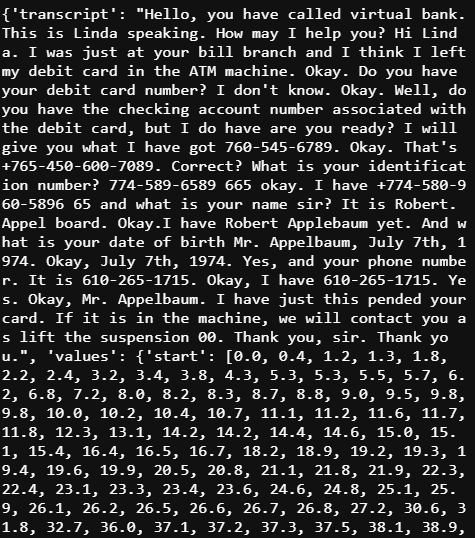
*4.1.2 นำข้อมูลในรูปแบบข้อความมาดำเนินการบันทึกเสียง*

เนื่องจากบทสนทนาเป็นบทสนทนาภาษาอังกฤษ จึงต้องนำบทสนทนาไปบันทึกเสียงโดยใช้ระบบสั่งการด้วยเสียงของระบบปฏิบัติการ iOS หรือ “สิริ” (Siri) ในการช่วยอ่านบทสนทนาเหล่านั้น ใน 1 บทสนทนาจะประกอบไปด้วยเสียงของพนักงานและลูกค้า โดยที่เสียงของพนักงานจะมีเพียงเพศเดียว คือ เพศหญิง โดยใช้เสียงของ “Siri Female” และในส่วนของเสียงลูกค้าจะแบ่งออกเป็น 2 เพศ ได้แก่ เพศหญิง ใช้เสียงของ “Samantha” และเพศชาย ใช้เสียงของ “Siri Male”

*4.1.3 แปลงประเภทของไฟล์เสียงบทสนทนา*

แปลงประเภทไฟล์เสียงจาก “.m4a” ให้อยู่ในประเภทไฟล์ “.wav” เพื่อสะดวกต่อการนำไปประมวลผลโดยได้ดำเนินการแปลงบนเว็บไซต์ที่ชื่อว่า “Convert MP4 to WAV” [11]

**4.1 การแปลงเสียงพูดให้อยู่ในรูปแบบข้อความ**

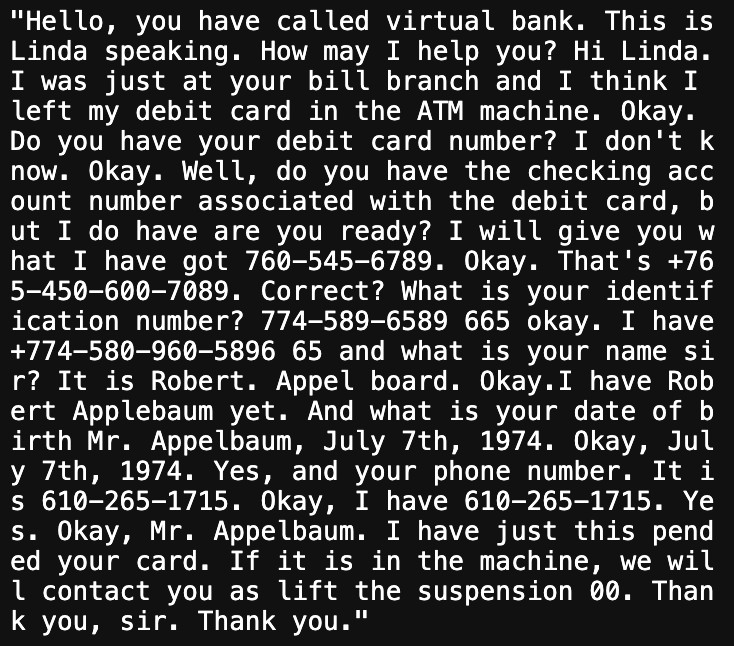


**รูปที่ 4.** ตัวอย่างการแปลงข้อมูลเสียงให้อยู่ในรูปแบบข้อความโดยใช้ Cloud Speech to Text

จากรูปที่ 4 แปลงให้อยู่ในรูปแบบของ Dictionary และสร้างคีย์ที่ชื่อว่า transcript เพื่อเก็บข้อความในบทสนทนาทั้งหมด ในส่วนของโทเค็นคำ ได้มีการสร้างคีย์ที่ชื่อว่า values เพื่อเก็บค่าของเวลาที่เริ่มพูดโทเค็นนั้น ๆ (start) เวลาที่พูดจบ (end) และโทเค็นนั้น ๆ (word) และประเมินผลความแม่นยำในการทำนายของแบบจำลอง โดยการนำข้อมูลบทสนทนาจริงเทียบกับข้อมูลที่แบบจำลองทำนายโดยใช้ Jaccard’s Coefficient Similarity ดังนี้



**รูปที่ 5.** ข้อมูลบทสนทนาจริง

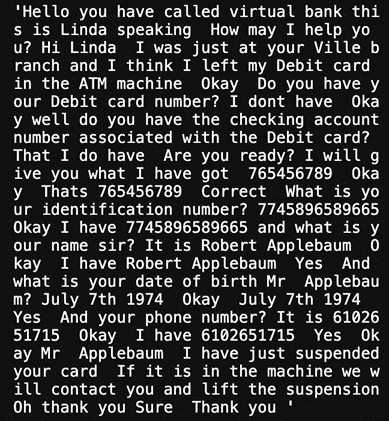


**รูปที่ 6.** บทสนทนาที่แบบจำลองทำนาย

**ตารางที่ 1.** ตารางแสดงค่าความแม่นยำของการแปลงเสียงพูดให้อยู่ในรูปของข้อความ

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

จากตารางที่ 1 ความแม่นยำในการทำนายคิดเป็นร้อยละ 57.02 ซึ่งเมื่อเทียบบทสนทนารูปที่ 5 และ 6 พบว่าสิ่งที่ส่งผลให้ค่าความแม่นยำของแบบจำลองไม่สูงนั้นส่วนใหญ่ขึ้นอยู่กับเครื่องหมายวรรคตอนของข้อมูลบทสนทนาทั้งสอง ดังนั้น จึงสร้างฟังก์ชันตัดเครื่องหมายวรรคตอนของบทสนทนาทั้งสองออก เพื่อประเมินผลค่าความแม่นยำใหม่ ดังรูปที่ 7 และ 8 และ ตารางที่ 2



**รูปที่ 7.** ข้อมูลบทสนทนาจริงที่ผ่านการทำความสะอาด



**รูปที่ 8.** บทสนทนาที่แบบจำลองทำนาย

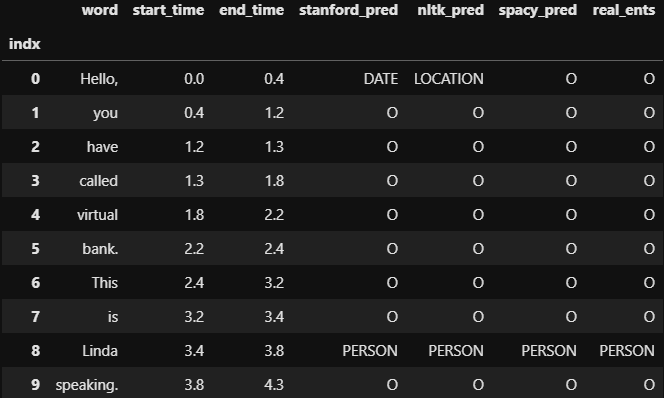
ที่ผ่านการทำความสะอาด

**ตารางที่ 2.** ตารางแสดงค่าความแม่นยำของการแปลงเสียงพูดให้อยู่ในรูปของข้อความ (ใหม่)

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

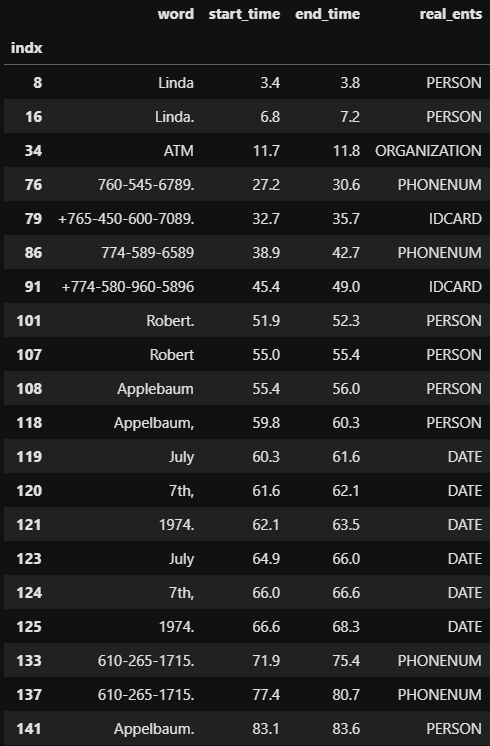
จากตารางที่ 2 ความแม่นยำในการทำนายคำพูดของแบบจำลองคิดเป็นร้อยละ 71.43 สามารถเห็นได้ชัดว่าค่าความแม่นยำสูงขึ้นอย่างชัดเจน เมื่อตัดเครื่องหมายวรรคตอนออกเบื้องต้น

**4.3 การตรวจจับคำที่เป็นข้อมูลส่วนบุคคลจากข้อมูลรูปแบบข้อความ**

เมื่อนำข้อมูลในรูปแบบข้อความที่ได้จาก Cloud Speech to Text เข้าสู่ฟังก์ชันต่าง ๆ ของไลบรารี Stanford NER, NLTK และ spaCy จากนั้นนำเข้าฟังก์ชันของการเลือกค่าทำนายจริง และสร้างนิพจน์ระบุนามเพิ่มเติมสำหรับเลขที่เป็นข้อมูลส่วนบุคคลโดยใช้ Regular Expressions ดังที่ได้กล่าวไว้ในบทขั้นตอนการพัฒนาระบบ จึงเก็บค่าของการทำนายของทุก ๆ ไลบรารีไว้ในรูปแบบตาราง ดังรูปที่ 9

**รูปที่ 9.** ตารางการทำนายประเภทของนิพจน์ระบุนาม

จากรูปที่ 9 เก็บค่าการทำนายของโทเค็นไว้ในตารางเดียวกันตามประเภทของนิพจน์ระบุนาม แถวใดมีการทำนายเป็นคำว่า “O” หมายความว่าโทเค็นนั้นไม่ได้เป็นนิพจน์ระบุนาม และมีการเก็บค่าการทำนายทั้งหมด 4 คอลัมน์ ได้แก่ stanford\_pred (ค่าที่แบบจำลอง Stanford NER ทำนาย) nltk\_pred (ค่าที่ NLTK ทำนาย) spacy\_pred (ค่าที่ spaCy ทำนาย) และ real\_ents (ค่าทำนายจริง) นอกจากนี้ ยังได้เก็บบันทึกค่าการทำนายจริง เฉพาะโทเค็นที่มีการติดแท็กนิพจน์ระบุนามขึ้นอีก 1 ตาราง เพื่อบันทึกให้อยู่ในรูปแบบไฟล์ CSV และนำไปปกปิดเสียงในขั้นตอนถัดไป ดังรูปที่ 10



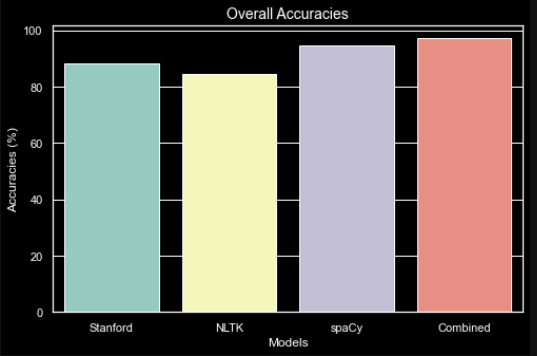
**รูปที่ 10.** ตารางค่าทำนายจริงเฉพาะที่มี

การติดแท็กนิพจน์ระบุนาม

มีการประเมินผลความแม่นยำในการทำนายนิพจน์ระบุนามของแต่ละไลบรารี ซึ่งวัดจากค่า Accuracies โดยนำข้อมูลโทเค็นที่ Cloud Speech to Text แบ่งมาทำการเฉลยนิพจน์ระบุนามจริง เพื่อที่จะนำไปประเมินผลความแม่นยำจากค่า Accuracies ของการทำนายนิพจน์ระบุนามในทุก ๆ ไลบรารี

**ตารางที่ 3.** ตารางการแสดงค่าความแม่นยำของการทำนายนิพจน์ระบุนามทุกประเภท

|  |  |
| --- | --- |
| **Libraries** | **Accuracies** |
| Stanford NER | 88.17 |
| NLTK | 84.62 |
| spaCy | 94.67 |
| Combined and Regex | 97.04 |



**รูปที่ 11.** กราฟการประเมินผลความแม่นยำ

ของแต่ละไลบรารี

จากตารางที่ 3 สามารถสรุปได้ ดังนี้

- ความแม่นยำของการทำนายนิพจน์ระบุนามของ Stanford NER คิดเป็นร้อยละ 88.17

- ความแม่นยำของการทำนายนิพจน์ระบุนามของ NLTK คิดเป็นร้อยละ 84.62

- ความแม่นยำของการทำนายนิพจน์ระบุนามของ spaCy คิดเป็นร้อยละ 94.67

- ความแม่นยำของการทำนายนิพจน์ระบุนามของการรวมไลบรารีและการทำ Regular Expressions คิดเป็นร้อยละ 97.04

สังเกตได้ว่า เมื่อดำเนินการรวมการทำนายของแต่ละแบบจำลองเข้าด้วยกัน และสร้างเงื่อนไขจาก Regular Expressions นั้น ส่งผลให้ค่าความแม่นยำในการทำนายนิพจน์ระบุนามสูงที่สุด

**4.3 การแทนที่เสียงที่เป็นข้อมูลส่วนบุคคลด้วยเสียงรบกวน**

เมื่อทำการแทนที่เสียงแล้วจึงดำเนินการบันทึกไฟล์เสียงที่ผ่านการแทนที่เสียงที่เป็นข้อมูลส่วนบุคคลด้วยเสียงรบกวนเป็นไฟล์เสียงประเภท .wav

**5. บทสรุป**

**5.1 สรุปผลโครงงาน**

การสร้างข้อมูลบทสนทนาในรูปแบบข้อความ เนื้อหาข้อมูลส่วนบุคคลของบทสนทนาประกอบด้วย ชื่อ - นามสกุล เลขที่บัญชี เลขบัตรเดบิต หรือ เครดิต เลขบัตรประชาชน วันเกิด ที่อยู่ และเบอร์โทรศัพท์ จากนั้นนำข้อมูลบทสนทนาในรูปแบบข้อความมาดำเนินการบันทึกเสียงและบันทึกเป็นไฟล์เสียง เพื่อนำไปใช้ในการพัฒนาระบบ

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

ขั้นตอนต่อมาเป็นการตรวจจับคำที่เป็นข้อมูลส่วนบุคคลจากข้อมูลรูปแบบข้อความ ได้ทำการทดลอง ทั้งหมด 3 ไลบรารี

- Stanford NER สามารถติดแท็กบุคคล และค่าเงินได้ค่อนข้างแม่นยำ แต่ในการติดแท็กวันที่ ด้วยข้อจำกัดที่ไม่มีการติดแท็กตัวเลขที่เป็นประเภท Cardinal จึงส่งผลให้มีการติดแท็กตัวเลขธรรมดา เป็นประเภทของวันที่ (Date) ทำให้ความแม่นยำลดลง

- NLTK สามารถติดแท็กองค์กรได้แม่นยำมากที่สุด แต่แบบจำลองนี้มักมีการติดแท็กที่ผิดพลาดตรงส่วนของสถานที่ กล่าวคือ หากโทเค็นนั้นขึ้นต้นด้วยตัวอักษรพิมพ์ใหญ่ เช่น คำว่า “Hello” แบบจำลองจะติดแท็กเป็นสถานที่ทันที

- spaCy จากผลลัพธ์การประเมินผลความแม่นยำ สังเกตได้ว่าส่วนใหญ่แล้ว spaCy มีค่าความแม่นยำสูงในการติดแท็กโทเค็น แต่หากให้สรุปเป็นรายประเภท สรุปได้ว่าสามารถติดแท็กบุคคล สถานที่ วันที่ และค่าเงินได้ดีที่สุด แต่เนื่องจากการติดแท็กของไลบรารีนี้ยังมีความไม่แม่นยำอยู่บ้าง ทางผู้จัดทำจึงมีความเห็นว่าควรรวมไลบรารีเข้าด้วยกันเพื่อเพิ่มประสิทธิภาพในการติดแท็ก

ในส่วนของการรวมไลบรารีเข้าด้วยกันมีความแม่นยำค่อนข้างสูง ซึ่งเฉลี่ยแล้วคิดเป็นร้อยละ 90 ถือเป็นค่าความแม่นยำที่น่าพึงพอใจ

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

ขั้นตอนสุดท้ายเป็นการแทนที่เสียงที่เป็นข้อมูลส่วนบุคคลด้วยเสียงรบกวน ในบางช่วงที่มีการแทนที่เสียงรบกวนนั้นอาจเกินหรือขาดจากระยะเวลาของคำพูดที่เป็นข้อมูลส่วนบุคคล แต่โดยภาพรวมแล้วถือว่าปิดบังคำพูดที่เป็นข้อมูลส่วนบุคคลได้ดี

**5.2 ปัญหาในการพัฒนาโครงงาน**

1) ปัญหาในการเตรียมข้อมูล โครงงานนี้ไม่สามารถหาชุดข้อมูลสาธารณะในการนำไปพัฒนาระบบได้เนื่องจากข้อมูลที่ใช้ดำเนินงานนั้นเป็นข้อมูลส่วนบุคคล ผู้จัดทำจึงจำเป็นต้องสร้างชุดข้อมูลขึ้นเองเพื่อพัฒนาระบบนี้

2) ปัญหาในการพัฒนาระบบหลัก ๆ คือ ความแม่นยำของการแปลงข้อมูลเสียงให้อยู่ในรูปแบบข้อความ หากบางบทสนทนามีค่าความแม่นยำที่ไม่สูงมากพอ เมื่อนำข้อมูลเหล่านั้นเข้าสู่กระบวนการตรวจจับคำที่เป็นข้อมูลส่วนบุคคลจากข้อมูลรูปแบบข้อความ ส่งผลให้ไลบรารีไม่สามารถติดแท็กประเภทของโทเค็นที่ควรจะมีนิพจน์ระบุนามได้ เช่น ชื่อบุคคล หรือ ส่วนเล็ก ๆ ของเลขที่เป็นข้อมูลสำคัญ ส่งผลให้เป็นปัญหาต่อการปิดบังคำที่เป็นข้อมูลส่วนบุคคลในขั้นตอนสุดท้ายได้

3) การแทนที่เสียงที่เป็นข้อมูลส่วนบุคคลด้วยเสียงรบกวน ในบางช่วงของการแทนที่เสียงรบกวนอาจเกินหรือขาดจากระยะเวลาของคำพูดที่เป็นข้อมูลส่วนบุคคล

**5.3 แนวทางในการพัฒนาต่อ**

1) โครงงานนี้สามารถต่อยอดโดยการนำระบบที่พัฒนาไปใช้ในด้านของการปกปิดข้อมูลที่เป็นส่วนบุคคลในหน่วยงานและองค์กรที่ต้องการประยุกต์ใช้ระบบได้ทั้งในรูปแบบไฟล์เสียง และข้อมูลที่เป็นข้อความ

2) สามารถนำผลลัพธ์ที่ได้จากการพัฒนาระบบไปวิเคราะห์และพัฒนาต่อในด้านอื่น ๆ ได้ โดยไม่ต้องคำนึงถึงสิทธิส่วนบุคคลเนื่องจากมีการปกปิดข้อมูลส่วนบุคคลแล้ว

**เอกสารอ้างอิง**

[1] ศ. สวัสดิ์พงศ์ธาดา. “ความเป็นส่วนตัว (Privacy)”. [Online]. Available:

<https://angsila.cs.buu.ac.th/>~58160640

/887420/hw/hw8.pdf. 2015.

[2] IBM. “What is Speech Recognition?”. [Online].

Available: <https://www.ibm.com/cloud/learn/speech-recognition>. 2020.

[3] Flame Sillawat. “การเปิดใช้งาน Cloud Speech API”.

[Online]. Available: <https://bit.ly/3orAjIe>.

2018.

[4] รัฐภูมิ ตันสุตะพานิช. “**การสกัดความสัมพันธ์ระหว่างนิพจน์ระบุนามในภาษาไทย**”. (วิทยานิพนธ์ปริญญาวิทยาศาสตร์มหาบัณฑิต, 2552).

[5] ศุภวัจน์ แต่รุ่งเรือง. “**การตรวจเทียบภายนอกหาการลักลอกในงานวิชาการโดยใช้แบบจำลองซัพพอร์ตเวกเตอร์แมชชีนและการวัดค่าความละม้ายของข้อความ**”. (วิทยานิพนธ์ปริญญาอักษรศาสตรดุษฎีบัณฑิตจุฬาลงกรณ์มหาวิทยาลัย, 2560).

[6] A. Geitgey Natural Language Processing is Fun!”. [Online]. Available: <https://bit.ly/36Vpsk8>.

2018.

[7] C. Dishmon. “Named Entity Recognition with Stanford NER Tagger”. [Online]. Available: <https://bit.ly/3lXHvKB>.

2020.

[8] NLTK. “Natural Language Toolkit”. [Online]. Available:

<https://www.nltk.org/>. 2020.

[9] spaCy. “spaCy 101: Everything you need to know”. [Online]. Available: <https://spacy.io/usage/spacy-101>. 2020.

[10] R. Cox. “Regular Expression Matching Can Be Simple And Fast (but is slow in Java, Perl, PHP, Python, Ruby, ...)”. [Online]. Available: <https://swtch.com/~rsc/regexp/regexp1.html>.

2007.

[11] “MP4 to WAV online file converter”. [Online].

Available: https://audio.online-convert.com/convert/mp4-to-wav.