

การแปลงเสียงเป็นข้อความสำคัญ

เสนอ

ผศ.ดร.ณัฐโชติ พรหมฤทธิ์

อ.ดร.สัจจาภรณ์ ไวจรรยา

จัดทำโดย

นางสาวกมลวัฒน์	ไตรรักษา	รหัสนักศึกษา 650710150
นางสาวศิริวรรณ	พอกสนิท	รหัสนักศึกษา 650710432
นางสาวรวินท์นิภา	ดำรงบุรณะกุลชัย	รหัสนักศึกษา 650710713
นางสาวศศิมา	พังยาง	รหัสนักศึกษา 650710722

รายงานนี้เป็นส่วนหนึ่งของวิชาวิทยาการข้อมูลและเครื่องมือ (Data science and tools)

ภาคเรียนที่ 1 ปีการศึกษา 2567

สาขาเทคโนโลยีสารสนเทศ คณะวิทยาศาสตร์ มหาวิทยาลัยศิลปากร

บทคัดย่อ

รายงานเรื่องการแปลงเสียงเป็นข้อความสำคัญ มีวัตถุประสงค์เพื่อแปลงเสียงเป็นข้อความช่วยแก้ปัญหาในการติดตามเนื้อหาในชั้นเรียน โดยช่วยลดเวลาในการฟังคลิปเสียงย้อนหลังและทำให้ผู้เรียนสามารถสรุปเนื้อหาได้รวดเร็วขึ้น ทั้งยังเพิ่มประสิทธิภาพในการจัดเก็บและเข้าถึงข้อมูล ทำให้เหมาะสำหรับผู้เรียนทั่วไปและผู้ที่มีความต้องการพิเศษ เช่น ผู้ที่มีปัญหาทางการได้ยิน โดยในการทำรายงานครั้งนี้ได้ดำเนินการตัดคลิปเป็นคลิปละ 5 นาที, แปลงไฟล์ .mp4 เป็น .wav, กำจัด noise ของไฟล์เสียง, แปลงไฟล์เสียงเป็นข้อความ, ซ่อมคำ และสรุปข้อความ

การแปลงเสียงเป็นข้อความสำคัญผลการดำเนินงานพบว่าจากการทำไฟล์เสียงที่ได้ไปแปลงเป็นข้อความโดยใช้ whisper มีข้อความที่หายไปหรือผิดพลาดจึงต้องทำการซ่อมแซมคำ และทำการสรุปข้อความด้วยทำจำนวนคลิปทั้งหมด 10 ไฟล์ การวัดประสิทธิภาพของการสรุปข้อความแสดงค่า rouge เฉลี่ยของไฟล์เสียงทั้งหมด

สารบัญ

เรื่อง	หน้า
บทคัดย่อ	ก
บทที่ 1 บทนำ	1
บทที่ 2 ทฤษฎีและความรู้ที่เกี่ยวข้อง	2
บทที่ 3 ขั้นตอนและวิธีการดำเนินงาน	3
บทที่ 4 ผลการดำเนินงาน	6
บทที่ 5 สรุปผลและอภิปราย	9
เอกสารอ้างอิง	10

บทที่ 1 บทนำ

เทคโนโลยีในปัจจุบันมีความก้าวหน้าในการพัฒนาอุปกรณ์อิเล็กทรอนิกส์ ที่ส่งผลให้สามารถตอบสนองความต้องการของผู้ใช้งานได้หลากหลาย ทำให้การรับ-ส่งข้อมูลข่าวสารทางอิเล็กทรอนิกส์เป็นที่นิยมอย่างแพร่หลาย ดังนั้นสารสนเทศ (Information) เป็นสิ่งที่จำเป็นสำหรับทุกองค์กร มีการใช้สารสนเทศเข้ามาช่วยในการบริหารจัดการ และข้อมูลทั้งหมดจะต้องถูกจัดเก็บเข้าสู่ระบบคอมพิวเตอร์ด้วยอุปกรณ์นำเข้าตามประเภทของข้อมูล ข้อมูลต่าง ๆ นั้นอาจจะเป็นได้ทั้ง ข้อความ (Text) ตัวเลข (Number) ภาพนิ่ง (Still Images) ภาพเคลื่อนไหว (Video) ซึ่งขึ้นอยู่กับแหล่งข้อมูลและประเภทของแหล่งข้อมูล

ข้อมูลสารสนเทศในระบบไฟล์เสียง เป็นระบบสารสนเทศแบบหนึ่งที่มีความสำคัญและได้รับความนิยมเนื่องจากเข้าถึงได้ในวงกว้าง และได้รับความนิยม อีกทั้งประโยชน์ของการแปลงเสียงเป็นข้อความเป็นเรื่องที่ท้าทายในการจำแนกและเรียงลำดับข้อมูลในคลังที่มีไฟล์วิดีโอและเสียงจำนวนมาก สามารถใช้ที่เก็บข้อมูลนี้สำหรับการอ้างอิงและวิจัยโดยการถอดเสียงเป็นข้อความ ตัวอย่างเช่น Audioburst ใช้ซอฟต์แวร์การถอดเสียงอัตโนมัติเพื่อสร้างพื้นที่เก็บข้อมูลการบันทึกเสียงของรายการทอล์คโชว์ที่มีเนื้อหาที่ทุกคนสามารถค้นหาและแชร์ได้และสอดคล้องกับงานวิจัยของ เซวาลิต จันภิรมย์ (2565) และคณะ ได้ศึกษาวิจัยเรื่อง การพัฒนาระบบแปลงไฟล์ภาพและเสียงเป็นข้อความจากหน้าจอของจดหมาย ด้วยการประมวลผลภาพ พบว่าระบบแปลงไฟล์ภาพและเสียงเป็นข้อความจากหน้าจอของจดหมายด้วยการประมวลผลภาพให้มีประสิทธิภาพด้วยภาษาไทยและอังกฤษทำการแก้ไขและพัฒนาระบบให้สามารถทำงานได้ครบทุกฟังก์ชัน ส่วนความพึงพอใจต่อระบบแปลงไฟล์ภาพและเสียงเป็นข้อความจากหน้าจอของจดหมายด้วยการประมวลผลภาพ ในภาพรวมอยู่ในระดับมาก และสอดคล้องกับงานวิจัยของ ธาณิล ม่วงพูล (2564) และคณะ ได้ศึกษาวิจัยเรื่อง แอปพลิเคชันสำหรับการแปลงเสียงเป็นข้อความสำหรับผู้พิการทางหูโดยใช้ Google API แอปพลิเคชันนี้สามารถนำไปใช้แปลงข้อความในที่ประชุม ผู้ใช้เพียงแก้ไขข้อความบางส่วน โดยไม่จำเป็นต้องถอดเทปใหม่ และทำให้การเปลี่ยนแปลงมีความถูกต้องมากขึ้น

ดังนั้นทางกลุ่มของพวกเราจึงจัดทำการแปลงเสียงเป็นข้อความใจความสำคัญเพื่อได้ข้อความสรุปบทเรียนที่ต้องการโดยไม่ให้เสียเวลาฟังไฟล์เสียงแล้วสรุป

บทที่ 2 ทฤษฎีและความรู้ที่เกี่ยวข้อง

2.1. การแปลงไฟล์เสียงเป็นข้อความ (Speech-to-Text)

ใช้ Whisper ซึ่งเป็นโมเดล Speech-to-Text ที่พัฒนาโดย OpenAI สามารถแปลงเสียงเป็นข้อความได้อย่างมีประสิทธิภาพ รองรับหลายภาษาและมีความแม่นยำสูง แม้ในสภาพเสียงที่มีเสียงรบกวน

- การกำจัดเสียงรบกวน

ใช้ไลบรารี noisereduce สำหรับลดเสียงรบกวนในไฟล์เสียง เพื่อปรับปรุงคุณภาพเสียงก่อนประมวลผลด้วย Whisper

2.2. การสรุปข้อความ (Text Summarization)

ใช้โมเดล BERT ร่วมกับ PageRank Algorithm ในการสรุปข้อความภาษาไทย โดย:

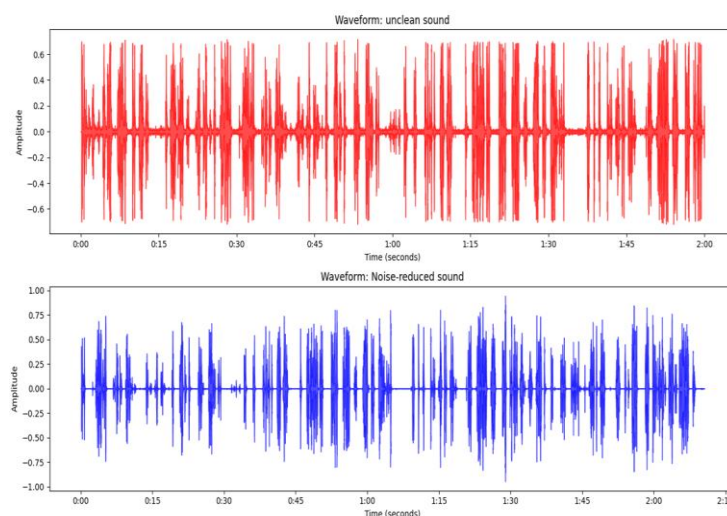
- BERT สร้างเวกเตอร์แสดงความสัมพันธ์ระหว่างประโยค
- PageRank จัดอันดับความสำคัญของประโยค เพื่อเลือกประโยคที่สำคัญที่สุดมาสร้างสรุป

บทที่ 3 ขั้นตอนและวิธีการดำเนินงาน

ขั้นตอนและวิธีการดำเนินงานเพื่อแปลงไฟล์เสียงเป็นข้อความสำคัญโดยใช้เทคนิค noisereduce, whisper, nlp , word tokenize ผู้วิจัยจะอธิบายถึง 3.1 การกำจัด noise ของเสียง 3.2 แปลงเสียงเป็นข้อความ 3.3 ซ่อมแซมข้อความ และ 3.4การนำข้อความมาสรุป โดยมีรายละเอียดดังนี้

3.1 การกำจัด noise ของเสียง

ดาวน์โหลดคลิป ใช้โปรแกรมหรือเว็บไซต์ที่สามารถดาวน์โหลดคลิปยูทูปมาเก็บไว้ในโฟลเดอร์ แปลงคลิป mp4 ให้อยู่ในรูปของไฟล์นามสกุลของ .wav โดยใช้ library moviepy, subprocess และกำจัดเสียงรบกวนในไฟล์เสียงเครื่องมือที่ใช้ในขั้นตอนนี้คือ noisereduce ซึ่งเป็นไลบรารีที่สามารถลดเสียงรบกวนจากไฟล์เสียงได้อย่างมีประสิทธิภาพโดยจะทำการกรองเสียงที่ไม่ต้องการออกไปเพื่อให้ได้เสียงที่คมชัดและเหมาะสมสำหรับการแปลงเป็นข้อความเนื่องจากเสียงรบกวนที่ไม่จำเป็นอาจส่งผลให้ผลลัพธ์จากการแปลงเสียงเป็นข้อความไม่แม่นยำ



ภาพ 1 กราฟเวลาของไฟล์เสียงที่ไม่สะอาดกับไฟล์เสียงที่ผ่านการลดเสียงรบกวน

3.2 การแปลงเสียงเป็นข้อความ

ในขั้นตอนนี้ เราจะใช้ Whisper ซึ่งเป็นโมเดลการแปลงเสียงเป็นข้อความจาก OpenAI ที่สามารถแปลงไฟล์เสียงเป็นข้อความได้อย่างแม่นยำ โดยโมเดลนี้สามารถรองรับหลายภาษาและมีความสามารถในการแปลงเสียงจากไฟล์ต่างๆ รวมถึงเสียงที่มีคุณภาพไม่สูงมาก หลังจากที่ได้ทำการกรองเสียงรบกวนแล้ว ข้อความที่ได้จากการแปลงเสียงจะถูกนำไปใช้งานต่อไปโดยมีขั้นตอนดังต่อไปนี้

3.2.1 ติดตั้งและตั้งค่าโมเดล Whisper

เริ่มต้นด้วยการติดตั้งไลบรารี Whisper

3.2.2 โหลดไฟล์เสียง

อ่านไฟล์เสียงที่ต้องการแปลงเป็นข้อความ เช่น ไฟล์ WAV, MP3, หรือ M4A โดยใช้ไลบรารี pydub หรือ librosa ตัวอย่างการโหลดไฟล์

3.2.4 โหลดโมเดล Whisper

ใช้ `whisper.load_model("large")` เพื่อโหลดโมเดลขนาดใหญ่ของ Whisper สำหรับการถอดเสียง (Transcription)

3.2.5 แปลงเสียงเป็นข้อความ

ใช้ฟังก์ชัน `model.transcribe` เพื่อถอดเสียงจากไฟล์เสียง โดยระบุภาษาเป็นภาษาไทย (`language="th"`)

3.2.6 บันทึกผลลัพธ์เป็นไฟล์ .txt

3.3. การซ่อมแซมข้อความ

การซ่อมแซมข้อความ (Text Repair) เป็นกระบวนการที่สำคัญในการปรับปรุงและแก้ไขข้อความที่มีข้อผิดพลาด หรือไม่สมบูรณ์ เพื่อให้ข้อความพร้อมใช้งานในการประมวลผลในขั้นตอนถัดไป กระบวนการซ่อมแซมข้อความนี้ใช้เทคนิคจาก การประมวลผลภาษาธรรมชาติ (Natural Language Processing - NLP) เพื่อช่วยให้การซ่อมแซมมีประสิทธิภาพมากขึ้น โดยมีขั้นตอนที่สำคัญในการดำเนินการ ดังนี้:

ขั้นตอนในการซ่อมแซมข้อความด้วย NLP

3.3.1 การซ่อมแซมคำที่ผิด โดยใช้ Natural Language Processing

- เนื่องจากเสียงที่แปลงออกมาเป็นข้อความมีคำผิดที่เยอะ จึงใช้เทคนิคการประมวลผล

การประมวลผลภาษาธรรมชาติ Natural Language Processing

- นำเข้าไลบรารี Natural Language Processing
- ใส่ข้อความที่ต้องการแก้ไข
- กดประมวลผลให้คำที่ต้องการแก้ไขประมวลผลออกมา โดยใช้

3.3.2 การซ่อมแซมคำที่ผิด โดยใช้ Google Suggest API

- เนื่องจากเสียงที่แปลงออกมาเป็นข้อความมีคำผิดที่เยอะ และเป็นคำที่ไม่คุ้นชิน จึงใช้เทคนิคการดึงคำจาก Google Suggest API
 - นำเข้าไลบรารี request, xmldict
 - ใส่ข้อความที่ผิดที่ต้องการได้รับการแก้ไข
 - กดประมวลผลคำที่ต้องการให้แก้ไข โดยจะไปดึงคำจาก google search

4.4. การสรุปข้อความ

ในกรณีที่ข้อความที่ได้รับมีความยาวมาก การสรุปข้อความสามารถช่วยให้ข้อความสั้นลงและจับสาระสำคัญได้ง่ายขึ้น ขั้นตอนนี้จะใช้การประมวลผลข้อความเพื่อลดขนาดและรักษาความหมายส่วนของการนำข้อความมาสรุป (Text Summarization) ซึ่งเป็นการใช้โมเดลประมวลผลภาษา (เช่น Wangchanberta) เพื่อสรุปใจความสำคัญจากเอกสารต้นฉบับ โดยใช้กระบวนการดังนี้:

4.4.1 โหลดโมเดลและเตรียมข้อมูล

ติดตั้งและโหลดโมเดล Wangchanberta เตรียมไฟล์ข้อความและแบ่งออกเป็นประโยค

4.4.2 ประมวลผลข้อมูลเบื้องต้น

ลบช่องว่าง (Whitespace) ที่ไม่จำเป็นและเข้ารหัสข้อความเป็นหน่วยย่อย (Subword)

4.4.3 คำนวณ Sentence Similarity และสร้างกราฟ

ใช้โมเดลคำนวณ Sentence Vector, คำนวณค่า Similarity ระหว่างประโยคสร้างกราฟเชื่อมโยงความสัมพันธ์ระหว่างประโยค

4.4.4 คำนวณค่า PageRank และจัดลำดับประโยค

คำนวณค่า PageRank ของแต่ละประโยค จัดเรียงประโยคตามความสำคัญ

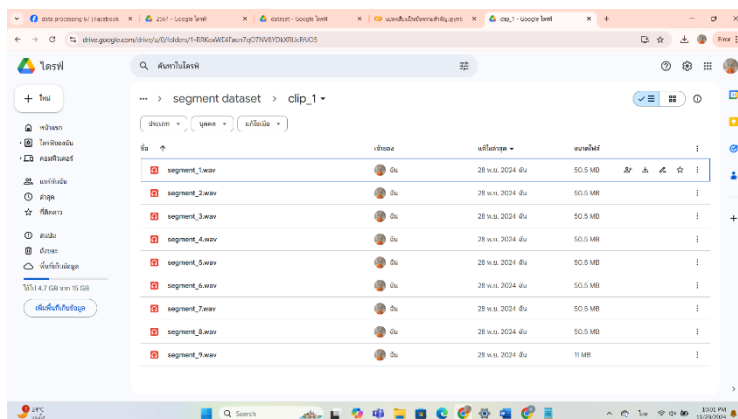
4.4.5 เลือกประโยคสำคัญและไฮไลต์

เลือก 20% ของประโยคที่มีค่า PR สูงที่สุดต่อมาบันทึกข้อความพร้อมไฮไลต์

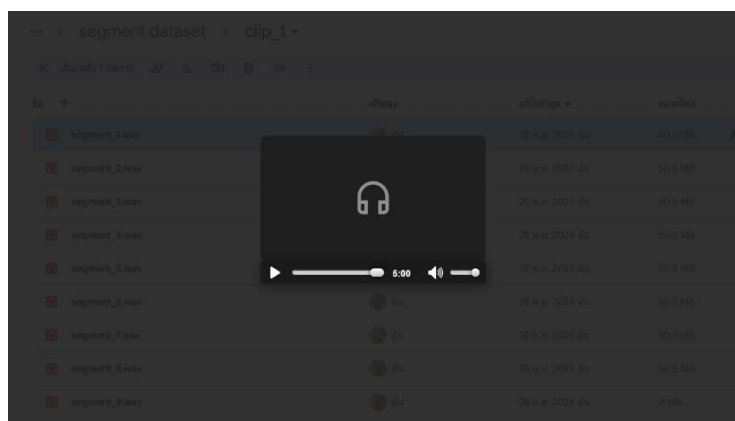
บทที่ 4 ผลการดำเนินงาน

4.1 การตัดเสียงให้เหลือ 5 นาที

การตัดคลิปเสียงเป็นการตัดคลิปเสียงให้ย่อย ๆ เป็นคลิปละ 5 นาที เพื่อเพิ่มประสิทธิภาพในการทำงานของงานในขั้นตอนอื่นๆ เช่น คลิปต้นแบบ 1 ชั่วโมง แบ่งเป็นคลิปละ 5 นาที ได้ทั้งหมด 12 คลิป



ภาพ 2 ไฟล์เสียงทั้งหมดที่ตัดเป็น 5 นาทีของคลิปที่ 1



ภาพ 3 ไฟล์เสียงที่ถูกตัดเป็น 5 นาที

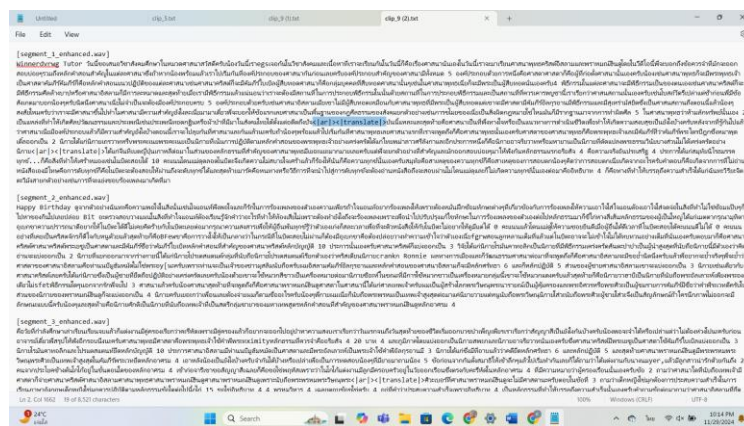
4.2 การกำจัด noise ของคลิปเสียง

การกำจัด noise ของคลิปเสียงเป็นการจัดการเพื่อเพิ่มประสิทธิภาพของเสียง โดยในกระบวนการเพิ่มความดังของเสียง มีการใช้ซอฟต์แวร์ดังนี้

- Noise Gate
- Compressor
- Low Shelf Filter

4.3 การแปลงเสียงให้เป็นข้อความ

การแปลงเสียงเป็นข้อความ เป็นการทดสอบเพื่อได้ข้อความที่จะนำไปสรุป ซึ่งเสียงนำมาแปลงเป็นข้อความจะผ่านการกำจัด noise เพื่อให้เสียงประสิทธิภาพแล้ว ในการแปลงเสียงเป็นข้อความแสดงให้เห็นถึงศักยภาพของโมเดล Whisper ในการถอดเสียงพูดภาษาไทย ซึ่งอาจจะมียางคำที่ไม่สามารถทราบความหมาย หรือถอดออกมาได้ไม่ครบครัน มีคำที่แปลงหรือถอดออกมาจากไฟล์เสียงผิด มีการสะกดไม่ถูกต้อง คำถอดมาไม่ตรงกับคำพูดในเสียง จึงต้องนำข้อความที่แปลงออกมาไปสู่ขั้นตอนการซ่อมแซมข้อความ



ภาพ 4 คำที่ผิดจากการแปลงเสียงเป็นข้อความ

4.4 การซ่อมแซมคำ

การซ่อมคำนั้น เป็นการแก้ไขคำที่ผิดพลาดจากการแปลงเสียงเป็นข้อความ เช่น สะกดผิด คำภาษาอื่นๆ เป็นต้น มาซ่อมแซมโดย ใช้ไลบรารี PyThaiNLP ร่วมกัน function spell (pythainlp.spell) และสุดท้ายเป็นวิธีการใช้ไลบรารี requests และ xmltodict ใช้ประโยชน์จาก Google Suggest API ในการตรวจเช็คคำผิดได้ โดยเข้าถึงได้ผ่านลิงค์ <https://www.google.com/comple/search?output=toolbar&q=คำผิด> จะได้ข้อมูล xml กลับมา

4.5 การสรุปข้อความ

การสรุปข้อความ เป็นการนำข้อความที่ได้จากการแปลงจากเสียงและซ่อมแซมคำและมาทำ text highlight และทดสอบความแม่นยำเมื่อเทียบกับที่ทางผู้จัดทำได้สรุปข้อความในชุดเดียวกัน และแสดงเป็นข้อมูล rouge ได้ผลดังนี้

```

SumVS.ipynb
[Cell 1]
print("File not found: (human_summary_file)")
except Exception as e:
    print("Error processing (human_summary_file): (e)")

Comparing with clip_0_1.txt...
Test (human summary): human tutor ในนี้ จะสอน ทักษะการเขียน ใน ภาษา อังกฤษ ใน 10 นาที เรา จะ ทำ การ สอนด้วย แบบ สั้นๆ และ ท้าทาย โดย จะใช้ประโยค สั้นๆ ของ ภาษา 5 ข้อ
Predicted (highlighted summary): เนื้อหา ส่วน นี้ สั้นๆ ของ ภาษาแล้ว เนื้อหา 10 ประการ เป็น เนื้อ ส่วน ของ ภาษาไทย 0 ๖ มี หอพระเจ้า เป็น ภาษา สั้นๆ และ เนื้อหาของ ของ ภาษาไทย 0
Rouge1: 0.75
Rouge2: 0.61
RougeL: 0.42
Avg Rouge: 0.59

Comparing with clip_0_2.txt...
Test (human summary): ใน 10 นาที เรา จะ ทำ การ สอนด้วย แบบ สั้นๆ และ ท้าทาย โดย จะใช้ประโยค สั้นๆ ของ ภาษา 5 ข้อ และ จะใช้ประโยค 1 ข้อ ส่วน ภาษา 0 6 มี หอพระเจ้า เป็น ภาษา สั้นๆ
Predicted (highlighted summary): เนื้อหา ส่วน นี้ สั้นๆ ของ ภาษาแล้ว เนื้อหา 10 ประการ เป็น เนื้อ ส่วน ของ ภาษาไทย 0 ๖ มี หอพระเจ้า เป็น ภาษา สั้นๆ และ เนื้อหาของ ของ ภาษาไทย 0
Rouge1: 0.59
Rouge2: 0.5
RougeL: 0.44
Avg Rouge: 0.51

Comparing with clip_0_3.txt...
Test (human summary): จะใช้ประโยค สั้นๆ ของ ภาษา 5 ข้อ และจะใช้ประโยค 1 ข้อ ส่วน ภาษา 0 6 มี หอพระเจ้า เป็น ภาษา สั้นๆ และ เนื้อหาของ ของ ภาษาไทย 0 6 มี หอพระเจ้า เป็น ภาษา สั้นๆ
Predicted (highlighted summary): เนื้อหา ส่วน นี้ สั้นๆ ของ ภาษาแล้ว เนื้อหา 10 ประการ เป็น เนื้อ ส่วน ของ ภาษาไทย 0 ๖ มี หอพระเจ้า เป็น ภาษา สั้นๆ และ เนื้อหาของ ของ ภาษาไทย 0
Rouge1: 0.43
Rouge2: 0.34
RougeL: 0.25
Avg Rouge: 0.32

Comparing with clip_0_4.txt...
Test (human summary): ภาษา 0 จะใช้ประโยค สั้นๆ 5 ประการ ใน ภาษา สั้นๆ , สั้นๆ , สั้นๆ , สั้นๆ , และ ภาษาของ ภาษา ใน ภาษาไทย 0 6 มี หอพระเจ้า เป็น ภาษา สั้นๆ และ เนื้อหาของ ของ ภาษาไทย 0
Predicted (highlighted summary): เนื้อหา ส่วน นี้ สั้นๆ ของ ภาษาแล้ว เนื้อหา 10 ประการ เป็น เนื้อ ส่วน ของ ภาษาไทย 0 ๖ มี หอพระเจ้า เป็น ภาษา สั้นๆ และ เนื้อหาของ ของ ภาษาไทย 0
Rouge1: 0.4
Rouge2: 0.37
RougeL: 0.3
Avg Rouge: 0.36

```

ภาพ 5 ผลการสรุปของไฟล์ที่ 9

```

avg_rouge1 = sum([result["Rouge1"] for result in results]) / len(results)
avg_rouge2 = sum([result["Rouge2"] for result in results]) / len(results)
avg_rougeL = sum([result["RougeL"] for result in results]) / len(results)
avg_rouge = sum([result["Avg Rouge"] for result in results]) / len(results)

print(f"ค่าเฉลี่ยของ Rouge1 จากการทดสอบ 4 ชุดคือ: {avg_rouge1}")
print(f"ค่าเฉลี่ยของ Rouge2 จากการทดสอบ 4 ชุดคือ: {avg_rouge2}")
print(f"ค่าเฉลี่ยของ RougeL จากการทดสอบ 4 ชุดคือ: {avg_rougeL}")
print(f"ค่าเฉลี่ยของ Avg Rouge จากการทดสอบ 4 ชุดคือ: {avg_rouge}")

ค่าเฉลี่ยของ Rouge1 จากการทดสอบ 4 ชุดคือ: 0.5924999999999999
ค่าเฉลี่ยของ Rouge2 จากการทดสอบ 4 ชุดคือ: 0.45499999999999996
ค่าเฉลี่ยของ RougeL จากการทดสอบ 4 ชุดคือ: 0.36
ค่าเฉลี่ยของ Avg Rouge จากการทดสอบ 4 ชุดคือ: 0.47000000000000003

```

ภาพ 6 ผลการสรุปเฉลี่ยของไฟล์ที่ 9

```

avg_rouge1 = sum([result["Rouge1"] for result in results]) / len(results)
avg_rouge2 = sum([result["Rouge2"] for result in results]) / len(results)
avg_rougeL = sum([result["RougeL"] for result in results]) / len(results)
avg_rouge = sum([result["Avg Rouge"] for result in results]) / len(results)

print(f"ค่าเฉลี่ยของ Rouge1 จากการทดสอบ 10 ชุดคือ: {avg_rouge1}")
print(f"ค่าเฉลี่ยของ Rouge2 จากการทดสอบ 10 ชุดคือ: {avg_rouge2}")
print(f"ค่าเฉลี่ยของ RougeL จากการทดสอบ 10 ชุดคือ: {avg_rougeL}")
print(f"ค่าเฉลี่ยของ Avg Rouge จากการทดสอบ 10 ชุดคือ: {avg_rouge}")

ค่าเฉลี่ยของ Rouge1 จากการทดสอบ 10 ชุดคือ: 0.5482500000000001
ค่าเฉลี่ยของ Rouge2 จากการทดสอบ 10 ชุดคือ: 0.3577
ค่าเฉลี่ยของ RougeL จากการทดสอบ 10 ชุดคือ: 0.34875
ค่าเฉลี่ยของ Avg Rouge จากการทดสอบ 10 ชุดคือ: 0.42074999999999996

```

ภาพ 7 ผลการสรุปเฉลี่ยของไฟล์ทั้งหมด

บทที่ 5 สรุปผลและอภิปราย

บทความนี้นำเสนอการสรุปข้อความจากคลิปเสียง โดยใช้วิธีการแปลงคลิปเสียงเป็นข้อความ และนำข้อความไปสรุป ในการรวบรวมคลิปเสียงการเรียนจากในยูทูบและคลิปที่ทางผู้จัดทำอัดไว้ จากนั้นแปลงไฟล์นามสกุล .mp4 เป็น .wav โดยใช้ไลบรารี moviepy, subprocess แล้วจากนั้นกำจัด noise ของเสียง โดยใช้ noisereduce ซึ่งเป็นไลบรารีที่สามารถลดเสียงรบกวน แล้วจึงทำไฟล์เสียงที่ได้ไปแปลงเป็นข้อความโดยใช้ whisper ซึ่งอาจมีข้อความที่หายไปหรือผิดพลาดจึงต้องทำการซ่อมแซมคำ และทำการสรุปข้อความด้วยทำจำนวนคลิปทั้งหมด 10 ไฟล์ การวัดประสิทธิภาพของการสรุปข้อความแสดงค่า rouge เฉลี่ยของไฟล์เสียงทั้งหมด ซึ่งจากประสิทธิภาพที่ได้ หากปรับปรุงการทำงานในขั้นตอนต่างๆให้ดีขึ้นอาจนำไปพัฒนาเป็นระบบช่วยสรุปข้อความได้ต่อไป

เอกสารอ้างอิง

- [1] อ.พรหมฤทธิ์ และ อ.สัจจาภรณ์, “การไฮไลต์ใจความสำคัญภาษาไทยแบบอัตโนมัติด้วย Bert Model และ PageRank Algorithm,” 24 กันยายน 2566 [Online]. Available: <https://blog.pijop.org/thai-text-summarization-with-bert-and-pagerank/> [Access 17 พฤศจิกายน 2567].
- [2] [Sattaya Sing](#), “สู่ความเจียบสัด EP.2: เริ่มต้นเขียน Noise Cancellation อย่างง่ายด้วยภาษา Python,” 19 มกราคม 2564 [Online]. Available: <https://medium.com/super-ai-engineer/> .[Access 18 พฤศจิกายน 2567].
- [3] เซาวลิต จันภิรมย์และคณะ, “การพัฒนาระบบแปลงไฟล์ภาพและเสียงเป็นข้อความจากหน้าของจดหมาย” 2023 [Online].Available: <https://so05.tci-thaijo.org/index.php/jmctrmutp/article/view/260003>. [Access 18 พฤศจิกายน 2567]
- [4] ธานิล ม่วงพูลและคณะ, “แอปพลิเคชันสำหรับการแปลงเสียงเป็นข้อความสำหรับผู้พิการทางหูโดยใช้ Google API” 2021 [Online].Available: <https://pws.npru.ac.th/signal/data/files.pdf> [Access 14 พฤศจิกายน 2567]
- [5] ชุติพร ยงเกียรติพานิช, “การสรุปใจความสำคัญแบบสกัดจากบทความโดยใช้ออนโทโลยีและวิธีการทางกราฟ” 2018 [Online].Available: <https://digital.car.chula.ac.th/chulaetd/3268/> [Access 15 พฤศจิกายน 2567]
- [6] ศรัณญา นาทองห่อ, “การสรุปใจความสำคัญของข้อความแบบสกัดสำหรับข่าวท่องเที่ยวภาษาไทย” 2020 [Online].Available: <https://digital.car.chula.ac.th/chulaetd/3797/> [Access 16 พฤศจิกายน 2567]