การยกระดับบริการข้อมูลของมหาวิทยาลัยด้วยปัญญาประดิษฐ์เชิงสร้างสรรค์

ENHANCING UNIVERSITY INFORMATION SERVICES WITH GENERATIVE AI

จิระศักดิ์ สายเมฆ

บัณฑิตวิทยาลัย มหาวิทยาลัยศรีนครินทรวิโรฒ

2567

การยกระดับบริการข้อมูลของมหาวิทยาลัยด้วยปัญญาประดิษฐ์เชิงสร้างสรรค์

จิระศักดิ์ สายเมฆ

สารนิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
วิทยาศาสตรมหาบัณฑิต สาขาวิชาวิทยาการข้อมูล
คณะวิทยาศาสตร์ มหาวิทยาลัยศรีนครินทรวิโรฒ
ปีการศึกษา 2567

ENHANCING UNIVERSITY INFORMATION SERVICES WITH GENERATIVE AI

JIRASAK SAIMEK

An Master's Project Submitted in Partial Fulfillment of the Requirements

for the Degree of MASTER OF SCIENCE

(Data Science)

Faculty of Science, Srinakharinwirot University

2024

สารนิพนธ์

เรื่อง

การยกระดับบริการข้อมูลของมหาวิทยาลัยด้วยปัญญาประดิษฐ์เชิงสร้างสรรค์

ของ

จิระศักดิ์ สายเมฆ

ได้รับอนุมัติจากบัณฑิตวิทยาลัยให้นับเป็นส่วนหนึ่งของการศึกษาตามหลักสูตร

ปริญญาวิทยาศาสตรมหาบัณฑิต สาขาวิชาวิทยาการข้อมูล

ของมหาวิทยาลัยศรีนครินทรวิโรฒ

(รองศาสตราจารย์ นายแพทย์ฉัตรชัย เอกปัญญาสกุล)

คณบดีบัณฑิตวิทยาลัย

คณะกรรมการสอบปากเปล่าสารนิพนธ์

..... ที่ปรึกษาหลัก

(ผู้ช่วยศาสตราจารย์ ดร.รัตน์ชัยนันท์ ธรรมสุจริต)

..... ประธาน

(ผู้ช่วยศาสตราจารย์ ดร.จันตรี ผลประเสริฐ)

..... กรรมการ

(ผู้ช่วยศาสตราจารย์ ดร.ศิริสรรพ เหล่าหะเกียรติ)

ชื่อเรื่อง	การยกระดับบริการข้อมูลของมหาวิทยาลัยด้วยปัญญาประดิษฐ์เชิงสร้างสรรค์
ผู้วิจัย	จิระศักดิ์ สายเมฆ
ปริญญา	วิทยาศาสตรมหาบัณฑิต
ปีการศึกษา	2567
อาจารย์ที่ปรึกษา	ผู้ช่วยศาสตราจารย์ ดร. รัตน์ชัยนันท์ ธรรมสุจริต

งานวิจัยนี้ได้พัฒนาระบบ AI Chatbot โดยใช้เทคโนโลยี Retrieval-Augmented Generation (RAG) เพื่อปรับปรุงบริการข้อมูลของมหาวิทยาลัย เนื่องจากมหาวิทยาลัยประสบปัญหาข้อมูลกระจัดกระจายในหลาย แพลตฟอร์ม ทำให้เกิดความล่าช้าและความไม่ถูกต้องของข้อมูล ระบบนี้ใช้ Hybrid Retrieval ซึ่งผสมผสานระหว่าง BM25 และ Semantic Search เพื่อเพิ่มประสิทธิภาพในการดึงข้อมูล ข้อมูลที่ใช้ในการพัฒนาระบบถูกรวบรวมจากเว็บไซต์ของมหาวิทยาลัย 7 ประเภท ได้แก่ ปฏิทินการศึกษา, รายละเอียดหลักสูตร, ข้อมูลติดต่อ, โครงสร้างหลักสูตร, แผนการเรียน ค่าธรรมเนียม และทุนการศึกษาของสาขา จากการทดสอบด้วยคำถาม 65 ข้อ พบว่าระบบมีประสิทธิภาพในการดึงข้อมูลถึง 95% และคุณภาพการตอบสนองอยู่ที่ 87.83% นอกจากนี้ การ ประเมินด้วย BERTScore แสดงให้เห็นถึงความสอดคล้องทางความหมายที่ดีระหว่างคำตอบที่สร้างขึ้นและข้อมูล อ้างอิง โดยมีค่า Precision 85.19%, Recall 90.94% และ F1 87.83%

คำสำคัญ : การสร้างข้อมูลแบบเสริมด้วยการค้นคืน, ประสิทธิภาพในการดึงข้อมูล, คุณภาพการตอบสนอง, ความสอดคล้องทางความหมาย

Title	ENHANCING UNIVERSITY INFORMATION SERVICES WITH GENERATIVE AI
Author	JIRASAK SAIMEK
Degree	MASTER OF SCIENCE
Academic Year	2024
Thesis Advisor	Assistant Professor Dr. Ratchainant Thammasudjarit

This research develops an AI Chatbot using Retrieval-Augmented Generation (RAG) to improve university information services. Universities often struggle with data scattered across multiple platforms, which can lead to delays and inaccuracies in providing information. The developed system utilizes Hybrid Retrieval, a combination of BM25 and Semantic Search, to enhance the efficiency of data retrieval. Data for the system was gathered from university websites, focusing on seven key categories: academic calendar, curriculum details, contact information, course structure, study plans, fees and scholarship of the branch. The system's performance was tested using 65 questions, achieving a 95% retrieval evaluation score and a 87.83% response quality score. Furthermore, BERTScore was employed to measure semantic alignment, yielding a Precision of 85.19%, Recall of 90.94%, and an F1 score of 87.83%, indicating a good level of understanding between the generated answers and the reference information.

Keyword : Hybrid Retrieval, RAG, BM25, BERTScore, retrieval evaluation score, response quality score

กิตติกรรมประกาศ

จิระศักดิ์ สายเมฆ

สารบัญ

หน้า

สารบัญตาราง

หน้า

สารบัญรูปภาพ

หน้า

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

อย่างไรก็ตาม ความท้าทายสำคัญที่มหาวิทยาลัยจำนวนมากกำลังเผชิญ คือการจัดการข้อมูลที่กระจัดกระจายอยู่ตามแพลตฟอร์มต่างๆ ทั้งเว็บไซต์หลักของมหาวิทยาลัย เว็บไซต์ย่อยของคณะหรือหน่วยงาน หน้าเพจสาขาวิชาใน Facebook และสื่อสังคมออนไลน์อื่นๆ สภาพการณ์เช่นนี้ก่อให้เกิดปัญหาด้าน ความถูกต้องและสอดคล้องของข้อมูล (Data Integrity) และ ความพร้อมใช้งานของข้อมูล (Data Availability) อย่างหลีกเลี่ยงไม่ได้ การที่ผู้ใช้ไม่สามารถเข้าถึงข้อมูลที่ต้องการได้อย่างรวดเร็วและมั่นใจได้ในความถูกต้อง กลายเป็นอุปสรรคสำคัญ โดยเฉพาะสำหรับบุคคลภายนอกที่ไม่คุ้นเคยกับโครงสร้างข้อมูลภายใน การที่ข้อมูลในแต่ละแหล่งไม่สอดคล้องหรือขาดการเชื่อมโยงกัน ย่อมนำไปสู่ความล่าช้า ความสับสน และความสิ้นเปลืองเวลาในการค้นหา ซึ่งส่งผลกระทบต่อภาพลักษณ์และความน่าเชื่อถือของมหาวิทยาลัยโดยตรง ปัญหาเหล่านี้ในมุมมองของ ผู้บริหารหลักสูตร ซึ่งเป็นบุคลากรด่านแรกที่ต้องเผชิญกับคำถามจำนวนมหาศาลจากผู้ที่สนใจสมัครเข้าศึกษาต่อ คำถามเหล่านี้มักมีลักษณะซ้ำๆ กัน เช่น เกณฑ์การรับสมัคร รายละเอียดทุนการศึกษา และกำหนดการต่างๆ การที่ต้องตอบคำถามเหล่านี้ด้วยตนเอง ไม่เพียงแต่สิ้นเปลืองเวลาและทรัพยากรบุคคล แต่ยังเสี่ยงต่อความคลาดเคลื่อนของข้อมูลหากผู้ตอบแต่ละคนเข้าถึงข้อมูลจากแหล่งที่แตกต่างกันหรือไม่ได้รับการอัปเดตอย่างทันท่วงที ซึ่งส่งผลกระทบโดยตรงต่อประสิทธิภาพการทำงานและการให้ข้อมูลเชิงรุกเพื่อดึงดูดนักศึกษาที่มีศักยภาพ ซึ่งสะท้อนให้เห็นถึงความจำเป็นเร่งด่วนในการมีช่องทางการสื่อสารที่มีประสิทธิภาพและเชื่อถือได้

เพื่อแก้ไขปัญหาเหล่านี้ เราจึงนำเสนอแนวคิดในการพัฒนาระบบ Chatbot ที่ใช้เทคโนโลยี Generative AI ซึ่งถูกออกแบบมาเพื่อตอบคำถามได้อย่างถูกต้อง รวดเร็ว และสามารถโต้ตอบผ่านการสนทนาด้วยข้อความ ระบบนี้จะทำหน้าที่เป็นช่องทางหลักให้ผู้ใช้ทั้งภายในและภายนอกมหาวิทยาลัยสามารถเข้าถึงข้อมูลที่ต้องการได้ โดยอ้างอิงข้อมูลจากฐานข้อมูลกลางของสถาบันที่ได้รับการอัพเดตข้อมูลอยู่เสมอ การนำ AI Chatbot มาใช้จะก่อให้เกิดประโยชน์หลายด้าน ไม่เพียงแต่ช่วยประหยัดเวลาในการค้นหาข้อมูลของผู้ใช้งานและลดภาระงานของเจ้าหน้าที่ในการตอบคำถามซ้ำๆ แล้วยังสามารถให้บริการได้ตลอด 24 ชั่วโมง ซึ่งเป็นการยกระดับคุณภาพการให้บริการข้อมูลอีกด้วย โดยเฉพาะสำหรับบุคคลภายนอก

โครงการนี้มีเป้าหมายหลักเพื่อยกระดับการบริการข้อมูลของมหาวิทยาลัยให้แก่ผู้ใช้บริการทั้งภายในและบุคคลภายนอก ลดขั้นตอนและความซับซ้อน ในการสืบค้นข้อมูล พร้อมทั้ง ยกระดับมาตรฐานการให้บริการข้อมูล ของสถาบันให้ก้าวทันพลวัตของยุคดิจิทัล ผ่านการใช้ AI Chatbot เราเชื่อมั่นว่าจะสามารถ ตอบสนองความต้องการด้านข้อมูล ของส่วนต่างๆ ได้อย่างมีประสิทธิภาพและครอบคลุมยิ่งขึ้น

1.2 วัตถุประสงค์ของงานวิจัย

การวิจัยครั้งนี้มีวัตถุประสงค์หลักเพื่อพัฒนาระบบ AI Chatbot สำหรับการให้บริการข้อมูลของมหาวิทยาลัย โดยมีความสามารถในการตอบสนองต่อข้อซักถามผ่านการสื่อสารในรูปแบบข้อความ ซึ่งเป็นวิธีการสื่อสารที่ผู้ใช้บริการมีความคุ้นเคยและสามารถเข้าถึงได้โดยสะดวก ระบบดังกล่าวได้รับการออกแบบให้ครอบคลุมการให้บริการข้อมูลในประเด็นสำคัญ อันได้แก่ 1). ปฏิทินวิชาการและกำหนดการสำคัญ 2). กระบวนการรับสมัครและคุณสมบัติของผู้สมัคร 3). ช่องทางการติดต่อและสถานที่ตั้ง 4). โครงสร้างหลักสูตรและรายละเอียดรายวิชา 5). แผนการเรียนและการศึกษา 6). อัตราค่าธรรมเนียมการศึกษาและชำระล่าช้า 7). ทุนการศึกษาของสาขา

นอกจากนี้ อีกหนึ่งวัตถุประสงค์ที่สำคัญคือการศึกษาพัฒนา Large Language Model LLM และเทคนิค Retrieval-Augmented Generation (RAG) ให้สามารถตอบสนองต่อคำถามของผู้ใช้งานได้อย่างถูกต้องและตรงประเด็น LLM เป็นเทคโนโลยี Artificial Intelligence AI ที่มีความสามารถในการประมวลผลและสร้างเนื้อหาที่มีความคล้ายคลึงกับภาษามนุษย์ ในขณะที่เทคนิค RAG เป็นวิธีการที่ช่วยเพิ่มประสิทธิภาพให้กับแบบจำลองโดยการดึงข้อมูลที่เกี่ยวข้องและเป็นปัจจุบันมาประกอบการสร้างคำตอบ ทั้งนี้ การพัฒนาดังกล่าวยังรวมถึงการเพิ่มขีดความสามารถในการอัปเดตข้อมูลแบบ Real-time ซึ่งเป็นคุณสมบัติสำคัญที่จะช่วยให้ระบบมีความทันสมัยและสามารถให้บริการข้อมูลที่ถูกต้องและเป็นปัจจุบันแก่ผู้ใช้งาน ด้วยเหตุนี้ จึงสามารถยกระดับการให้บริการผ่านระบบ AI Chatbot ให้มีประสิทธิภาพและตอบสนองความต้องการของผู้ใช้งานได้อย่างมีประสิทธิผล

อีกประการหนึ่ง งานวิจัยนี้ยังมุ่งเน้นการทำ Performance Evaluation ของระบบ AI Chatbot โดยพิจารณาจากปัจจัยสำคัญหลายประการ โดยเฉพาะอย่างยิ่งด้าน Accuracy ซึ่งเป็นองค์ประกอบพื้นฐานที่มีนัยสำคัญต่อคุณภาพการให้บริการ การประเมินประสิทธิภาพอย่างเป็นระบบจะช่วยให้สามารถระบุ Strengths และ Weaknesses ของระบบ ซึ่งข้อมูลเชิงลึกดังกล่าวมีความสำคัญอย่างยิ่งต่อการพัฒนาและปรับปรุงระบบให้มี Optimization สูงขึ้นในอนาคต โดยเฉพาะในมิติของ Accuracy ที่มุ่งเน้นการวัดความแม่นยำของข้อมูลที่ระบบให้บริการ ซึ่งปัจจัยนี้มีความสำคัญอย่างยิ่งต่อการสร้าง User Experience ให้แก่ผู้ใช้บริการ ด้วยเหตุนี้ จึงสามารถยกระดับการให้บริการผ่านระบบ AI Chatbot ให้มีประสิทธิภาพและตอบสนองความต้องการของผู้ใช้งานได้

1.3 ขอบเขตของงานวิจัย

การวิจัยครั้งนี้ได้กำหนดขอบเขตการดำเนินงานทั้งในส่วนของขอบเขตด้านข้อมูล งานวิจัยมุ่งเน้นการศึกษาและรวบรวมข้อมูลจากแหล่งข้อมูลทางการของมหาวิทยาลัย เช่น เว็บไซต์หลัก เว็บไซต์คณะ และเอกสารทางการต่างๆ เพื่อให้ได้ข้อมูลที่มีความถูกต้องและน่าเชื่อถือสำหรับนำไปใช้ในการพัฒนาระบบ นอกจากนี้ ยังมีการพัฒนาระบบจัดเก็บข้อมูลในรูปแบบ Vector Database ซึ่งเป็นเทคโนโลยีที่สามารถจัดเก็บและเชื่อมโยงข้อมูลองค์ความรู้ของมหาวิทยาลัย โดยลักษณะของฐานข้อมูลนี้จะช่วยให้การเข้าถึงและการค้นหาข้อมูลเป็นไปอย่างมีประสิทธิภาพและรวดเร็วยิ่งขึ้น

สำหรับขอบเขตด้านการพัฒนาระบบ งานวิจัยนี้มุ่งเน้นการพัฒนาส่วน Natural language processing ซึ่งเป็นเทคโนโลยีที่เอื้ออำนวยให้ระบบคอมพิวเตอร์สามารถประมวลผลและตีความภาษามนุษย์ได้อย่างมีประสิทธิภาพ โดยระบบจะดำเนินการวิเคราะห์ Input Query จากผู้ใช้บริการเพื่อให้สามารถตอบสนองได้อย่างถูกต้องและสอดคล้องกับความต้องการ นอกจากนี้ การวิจัยยังครอบคลุมถึงการออกแบบและพัฒนาระบบ Vector Database สำหรับการจัดเก็บข้อมูล ซึ่งเป็นโครงสร้างพื้นฐานสำคัญที่ช่วยในการจัดเก็บและ Retrieval ข้อมูลในรูปแบบ Vector Embedding ที่มีประสิทธิภาพสูง อีกทั้งยังมีการประยุกต์ใช้เทคนิค Prompt Engineering ซึ่งเป็นกระบวนการในการออกแบบและปรับแต่ง Prompt Template ที่ใช้ในการสื่อสารกับ LLM เพื่อให้ได้ผลลัพธ์ที่มีความแม่นยำและตรงตาม Requirement ที่กำหนดไว้

ในส่วนของขอบเขตด้านการตอบสนองผู้ใช้งาน งานวิจัยนี้มุ่งพัฒนา Automated Response System ที่มีความสามารถในการให้บริการแบบ 24/7 Availability เพื่อรองรับการใช้งานได้อย่างต่อเนื่องโดยไม่มีข้อจำกัดด้านเวลา สำหรับขอบเขตด้าน Performance Evaluation งานวิจัยนี้ให้ความสำคัญกับการประเมิน Accuracy ของ Response Generation ที่ระบบให้บริการ ซึ่งเป็นปัจจัยที่มีนัยสำคัญต่อ System Reliability และ User Satisfaction อย่างยิ่ง

1.4 สมมติฐานการวิจัย

ระบบ AI Chatbot ที่พัฒนาขึ้น โดยใช้เทคโนโลยี Generative AI ร่วมกับเทคนิค RAG สามารถตอบคำถามเกี่ยวกับบริการข้อมูลของมหาวิทยาลัยได้ด้วยความถูกต้องและความเกี่ยวข้อง ในระดับที่น่าพอใจสำหรับการนำไปใช้งานจริง

1.5 ประโยชน์ที่คาดว่าจะได้รับ

การพัฒนาระบบ AI chatbot เพื่อให้บริการข้อมูลของมหาวิทยาลัยมีประโยชน์ที่คาดว่าจะได้รับในหลายด้าน โดยในด้านประสิทธิภาพการให้บริการข้อมูล ระบบดังกล่าวจะสามารถเพิ่มความถูกต้องในการเข้าถึงข้อมูลสำหรับผู้ใช้บริการทั้งภายในและภายนอกของมหาวิทยาลัย อีกทั้งยังเป็นการยกระดับมาตรฐานการให้บริการข้อมูลของสถาบันการศึกษาด้วยระบบที่ให้บริการตลอด 24 ชั่วโมง ซึ่งส่งผลให้เกิดการเพิ่มประสิทธิภาพในการจัดการและเผยแพร่ข้อมูลของมหาวิทยาลัยผ่านระบบ AI chatbot ที่มีการปรับปรุงข้อมูลอย่างต่อเนื่อง

ในด้านการบริหารจัดการทรัพยากรบุคคล ระบบดังกล่าวสามารถช่วยลดภาระงานของบุคลากรในการตอบคำถามที่พบบ่อย ซึ่งเป็นงานที่มีลักษณะซ้ำซ้อนและใช้เวลามาก นอกจากนี้ยังเป็นการส่งเสริมการทำงานของบุคลากรให้มีความยืดหยุ่นและมีประสิทธิภาพมากขึ้น เนื่องจากบุคลากรสามารถทุ่มเทเวลาและทรัพยากรไปกับงานที่ต้องใช้ทักษะเฉพาะทางและความเชี่ยวชาญได้อย่างเต็มที่
ด้านประสบการณ์ผู้ใช้งาน ระบบ AI chatbot จะอำนวยความสะดวกในการเข้าถึงข้อมูลผ่านระบบการสนทนาที่เป็นธรรมชาติ เพื่อสร้างประสบการณ์การใช้งานที่ดี ทำให้ผู้ใช้สามารถสอบถามข้อมูลได้อย่างสะดวกและรวดเร็ว ไม่ว่าจะเป็นนักศึกษา บุคลากร หรือบุคคลภายนอก นอกจากนี้ยังช่วยลดความซับซ้อนในการค้นหาข้อมูลจากหลายๆ แหล่งข้อมูล ซึ่งเป็นอุปสรรคสำคัญในการเข้าถึงข้อมูลที่ต้องการอย่างรวดเร็วและมีประสิทธิภาพ

ในด้านการพัฒนาเทคโนโลยีสารสนเทศ การนำระบบ AI chatbot มาใช้จะเป็นการยกระดับระบบสารสนเทศของมหาวิทยาลัยให้ทันสมัยด้วยเทคโนโลยี Generative AI ซึ่งเป็นเทคโนโลยีที่กำลังได้รับความนิยมและมีการพัฒนาอย่างรวดเร็วในยุคปัจจุบัน อีกทั้งยังเป็นการสร้างฐานข้อมูลที่มีการปรับปรุงอย่างต่อเนื่องและมีความน่าเชื่อถือ ระบบดังกล่าวยังสามารถเรียนรู้และปรับปรุงประสิทธิภาพได้อย่างต่อเนื่องจากการมีปฏิสัมพันธ์กับผู้ใช้งาน นอกจากนี้ การนำเทคโนโลยีทันสมัยมาประยุกต์ใช้ยังเป็นการเสริมสร้างภาพลักษณ์ความทันสมัยและความเป็นผู้นำด้านเทคโนโลยีของสถาบันการศึกษา ซึ่งเป็นปัจจัยสำคัญในการดึงดูดผู้เรียนและบุคลากรที่มีคุณภาพในอนาคต

บทที่ 2
ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

การศึกษาวิจัยนี้ได้ดำเนินการศึกษาและพัฒนาระบบ AI Chatbot (1) โดยมุ่งเน้นการประยุกต์ใช้เทคนิค RAG ร่วมกับการบริหารจัดการและ Information Retrieval ด้วย Vector Database เพื่อเพิ่มประสิทธิภาพในการตอบคำถามและการให้บริการข้อมูลที่มีความแม่นยำสูง โดยองค์ประกอบของเนื้อหาในบทที่ 2 ประกอบด้วยการทบทวนวรรณกรรมและแนวคิดทฤษฎีที่เกี่ยวข้อง ได้แก่ 1). Natural language processing 2). Prompt Engineering 3). RAG 4). Vector Database และ 5). วิจัยที่เกี่ยวข้อง

1. Natural language processing (NLP)

การวิจัย NLP มีจุดเริ่มต้นในช่วงทศวรรษ 1950-1960 โดยได้รับอิทธิพลจากแนวคิดทางตรรกศาสตร์และทฤษฎีภาษาศาสตร์เชิงสัญลักษณ์ งานวิจัยในยุคบุกเบิกนี้มุ่งพัฒนาเทคโนโลยีที่ทำให้เครื่องจักรสามารถเข้าใจและประมวลผลภาษามนุษย์ได้อย่างมีประสิทธิภาพ หนึ่งในรากฐานสำคัญของ AI และ NLP คือหลักการ Turing Test ที่นำเสนอโดย Alan Turing ในบทความ "Computing Machinery and Intelligence" 1950 ซึ่งได้วางกรอบแนวคิดในการทดสอบความสามารถของเครื่องจักรในการโต้ตอบกับภาษามนุษย์ แนวคิดนี้ถือเป็นจุดเริ่มต้นสำคัญในการพัฒนาระบบ NLP (2) ที่มีความซับซ้อนในยุคต่อมา

ในช่วงเวลาเดียวกัน การพัฒนา Machine Translation MT กลายเป็นงานวิจัยที่โดดเด่นโดยเฉพาะอย่างยิ่งในบริบทของสงครามเย็น ทั้งสหรัฐอเมริกาและสหภาพโซเวียตต่างให้ความสำคัญกับการแปลภาษาอัตโนมัติ เช่น การแปลจากภาษารัสเซียเป็นภาษาอังกฤษ ระบบในยุคนี้อาศัยแนวทาง Rule-Based Systems ซึ่งนักวิจัยจำเป็นต้องสร้างกฎไวยากรณ์และความหมายที่ชัดเจนเพื่อให้เครื่องจักรสามารถประมวลผลภาษาได้อย่างถูกต้อง แม้ว่าวิธีการนี้จะมีข้อจำกัดในการรองรับความยืดหยุ่นและความซับซ้อนของภาษามนุษย์ก็ตาม

โดยทฤษฎี Generative Grammar ของ Noam Chomsky มีบทบาทสำคัญในการเปลี่ยนแปลงวิธีการวิเคราะห์ภาษา Chomsky เสนอว่าสามารถสร้างประโยคภาษามนุษย์ได้จากโครงสร้างไวยากรณ์เชิงนามธรรม แนวคิดนี้นำไปสู่การพัฒนาโมเดลการวิเคราะห์ประโยค เช่น Syntactic Parsing ซึ่งเป็นการระบุและแยกแยะโครงสร้างของประโยคตามกฎไวยากรณ์ ผลงานตัวอย่างที่โดดเด่นในช่วงปลายยุคนี้ได้แก่ SHRDLU โดย Terry Winograd 1972 ซึ่งเป็นระบบที่สามารถโต้ตอบภาษาธรรมชาติในโดเมนที่มีขอบเขตจำกัดได้อย่างน่าประทับใจ

อย่างไรก็ตาม ระบบที่อาศัยกฎเชิงสัญลักษณ์ยังคงเผชิญกับข้อจำกัดที่สำคัญหลายประการ ได้แก่ ความยากลำบากในการสร้างกฎที่ครอบคลุมทุกบริบทและสถานการณ์ที่หลากหลายในภาษามนุษย์ อีกทั้งจำนวนกฎที่เพิ่มขึ้นยังทำให้เกิดปัญหาความซับซ้อนและลดประสิทธิภาพในการทำงานของระบบ ด้วยข้อจำกัดดังกล่าว แนวทางการวิจัย NLP จึงเริ่มเปลี่ยนจากการใช้กฎเชิงไวยากรณ์ไปสู่แนวทาง Statistical and Machine Learning Approaches ซึ่งนับเป็นจุดเริ่มต้นของยุคแบบจำลองภาษาเชิงสถิติที่อาศัยโมเดล N-gram ในการประมวลผลและสร้างภาษา

แบบจำลอง N-gram เป็นกระบวนวิธีหลักใน NLP และภาษาศาสตร์เชิงคำนวณที่ประกอบด้วยลำดับต่อเนื่องของหน่วยภาษาย่อยจำนวน N หน่วย ไม่ว่าจะเป็นคำ พยางค์ ตัวอักษร หรือ Token ที่สกัดจากข้อความต้นฉบับ โดยสามารถจำแนกเป็น 1-gram unigram สำหรับหนึ่งคำหรือตัวอักษร, 2-gram bigram สำหรับคู่ของคำหรือตัวอักษร, 3-gram trigram สำหรับสามคำหรือตัวอักษร และ N-gram สำหรับลำดับที่มีจำนวนมากกว่าสามคำหรือตัวอักษร

แบบจำลองนี้อาศัยทฤษฎี Markov assumption ซึ่งตั้งสมมติฐานว่าการปรากฏของหน่วยภาษาหนึ่งๆ จะขึ้นอยู่กับหน่วยภาษาก่อนหน้าเพียง N-1 หน่วยเท่านั้น แนวคิดนี้ช่วยลดความซับซ้อนในการคำนวณและเพิ่มประสิทธิภาพการประมวลผล N-gram งเป็นองค์ประกอบพื้นฐานสำคัญในการพัฒนาระบบเทคโนโลยีภาษาหลากหลายประเภท ได้แก่ Language Modeling, Machine Translation, Text Prediction, Information Retrieval, Content Analysis, Document Classification

แบบจำลอง N-gram ได้รับการประยุกต์ใช้อย่างแพร่หลายในงาน NLP โดยเฉพาะอย่างยิ่งในการสร้าง Language Modeling ซึ่งเป็นการทำนายคำหรือวลีในลำดับถัดไปโดยอาศัยค่าความน่าจะเป็นที่ประมาณได้จากความถี่การปรากฏร่วมของคำในคลังข้อมูลตามสมการ:

$$P(W_t|W_1, W_2, \dots, W_n) \approx \prod_{i=k}^{n} P(W_i|W_{i-1}, W_{i-2}, \dots, W_{i-k+1}) \quad (1)$$

โดยที่

W_i คือ คำที่ตำแหน่ง i ในลำดับ

K ขนาดของ n-gram เช่น k=2 คือ bigram, k=3 คือ trigram

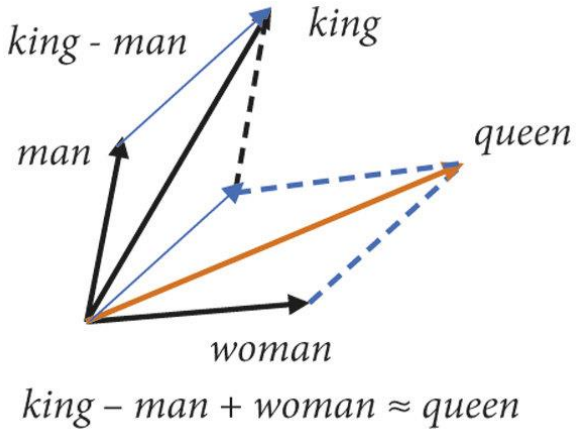
$P(W_i|W_{i-1}, W_{i-2}, \dots, W_{i-k+1})$ คือความน่าจะเป็นที่คำ W_i จะปรากฏหลังจากคำ$k-1$ คำก่อนหน้า

นอกจากนี้ ยังมีบทบาทสำคัญในการพัฒนา Text Prediction and Autocompletion ซึ่งช่วยอำนวยความสะดวกในการพิมพ์ ลดความผิดพลาด และเพิ่มประสิทธิภาพในการป้อนข้อมูล รวมถึงการ Spell Checking and Correction ที่ช่วยระบุคำที่มีแนวโน้มว่าผิดพลาดและเสนอทางเลือกที่ถูกต้อง

อย่างไรก็ตาม แบบจำลองของ N-gram ได้ประสบความท้าทายที่สำคัญหลายประการในการประยุกต์ใช้จริง โดยเฉพาะอย่างยิ่งปัญหา Data Sparsity ซึ่งเมื่อค่าของ N เพิ่มขึ้น จำนวน N-grams ที่เป็นเอกลักษณ์จะเพิ่มขึ้นในลักษณะ Exponential ซึ่งจะส่งผลให้แม้คลังข้อมูลขนาดใหญ่ก็ไม่สามารถครอบคลุม N-grams (3) ทั้งหมดที่อาจเกิดขึ้นได้ในภาษาจริง นอกจากนี้ ยังมีข้อจำกัดด้าน Memory and Computational Cost เนื่องจากแบบจำลองที่มีค่า N สูงจำเป็นต้องใช้พื้นที่จัดเก็บและกำลังประมวลผลมหาศาล รวมถึง Context Limitation เนื่องจากแบบจำลอง N-gram อาศัยหน้าต่างบริบทขนาดคงที่ N-1 ทำให้ไม่สามารถจับความสัมพันธ์ทางภาษาระยะยาวหรือโครงสร้างประโยคที่ซับซ้อนได้อย่างมีประสิทธิภาพ

เพื่อแก้ไขข้อจำกัดดังกล่าว นักวิจัยได้พัฒนาเทคนิคต่างๆ เช่น Smoothing Techniques, Laplace Smoothing, Good-Turing Estimation และ Kneser-Ney Smoothing รวมถึงการใช้เทคนิคการ Adaptive Model, Tokenization, Dimensionality Reduction แม้ว่าในปัจจุบันเทคโนโลยี AI และการเรียนรู้เชิงลึกจะได้รับการพัฒนาอย่างก้าวกระโดด แต่หลักการของ N-gram ยังคงเป็นพื้นฐานสำคัญที่มีอิทธิพลต่อการออกแบบและพัฒนาระบบประมวลผลภาษาสมัยใหม่

ยุคถัดมาเป็นการพัฒนาแบบจำลอง Neural Probabilistic Language Models NPLMs ซึ่งนับเป็นจุดเปลี่ยนที่สำคัญใน NLP เป็นการก้าวข้ามจากแนวทางเชิงสถิติแบบดั้งเดิมไปสู่การใช้โครงข่ายประสาทในงาน NLP โดยงานวิจัยของ Bengio และคณะ ในปี 2003 ได้วางรากฐานสำคัญสำหรับการประยุกต์ใช้โครงข่ายประสาทในการสร้าง Language Modeling หนึ่งในนวัตกรรมที่สำคัญที่สุดคือการพัฒนา Word Embeddings ซึ่งเป็นการแทนค่าคำด้วย Continuous Vectors ในปริภูมิ Dense Vector Space ที่มีมิติต่ำกว่าแบบจำลองแบบดั้งเดิม เวกเตอร์เหล่านี้มีความสามารถอันโดดเด่นในการบันทึกความสัมพันธ์เชิงความหมายและไวยากรณ์ระหว่างคำได้อย่างมีประสิทธิภาพ ซึ่งแตกต่างอย่างสิ้นเชิงจากการแทนค่าแบบเวกเตอร์ One-Hot ในโมเดล N-gram ที่มีลักษณะความเบาบางและไร้ซึ่งมิติความสัมพันธ์ทางความหมาย นวัตกรรมอันล้ำสมัยนี้ส่งผลให้แบบจำลองสามารถสร้างการเชื่อมโยงความหมายและการอนุมาน ได้อย่างมีประสิทธิภาพที่เหนือกว่า ที่น่าสนใจยิ่งไปกว่านั้นคือ การฝังตัวคำแสดงให้เห็นถึงความสามารถในการจับความสัมพันธ์ทางพีชคณิต เช่น:



ภาพประกอบ 1 ความสัมพันธ์เชิงความหมายในแบบจำลองเวกเตอร์คำ

จากภาพประกอบ 1 ปรากฏแสดงให้เห็นถึงปรากฏการณ์อันน่าทึ่งใน Word Embeddings ที่พบในแบบจำลองภาษาขั้นสูง ซึ่งได้รับการยอมรับอย่างกว้างขวางในวงการ NLP การแสดงผลเชิงกราฟิกนี้นำเสนอความสัมพันธ์เชิงพีชคณิตอันลึกซึ้งที่ปรากฏในปริภูมิเวกเตอร์หลายมิติ ซึ่งสะท้อนให้เห็นถึงความสัมพันธ์เชิงความหมายและโครงสร้างทางภาษาศาสตร์ที่แฝงอยู่ในภาษามนุษย์

ลักษณะที่ปรากฏในภาพคือการแสดงสมการพีชคณิต $king - man + woman \approx queen$ ซึ่งอธิบายด้วยลูกศรเวกเตอร์ที่แสดงความสัมพันธ์ระหว่างคำในปริภูมิเชิงความหมาย เมื่อนำเวกเตอร์ที่แทนคำว่า "king" มาลบด้วยเวกเตอร์ที่แทนคำว่า "man" และนำผลลัพธ์มาบวกกับเวกเตอร์ที่แทนคำว่า "woman" ผลลัพธ์จะปรากฏเป็นเวกเตอร์ที่มีความใกล้เคียงอย่างยิ่งกับเวกเตอร์ที่แทนคำว่า "queen" แสดงให้เห็นว่าแบบจำลองสามารถจับความสัมพันธ์ analogical reasoning ได้อย่างแม่นยำ การค้นพบนี้นับเป็นหลักฐานสำคัญที่แสดงถึงศักยภาพของ NLP ในการจับโครงสร้างความสัมพันธ์เชิงความหมายที่ซับซ้อนในภาษามนุษย์และเป็นรากฐานสำคัญสำหรับการพัฒนาแบบจำลองภาษาขั้นสูงที่มีความสามารถในการเข้าใจบริบทและความสัมพันธ์ทางภาษาในระดับที่ลึกซึ้งยิ่งขึ้น

Neural Probabilistic Framework เป็นแนวคิดพื้นฐานที่มีความสำคัญอย่างยิ่ง โดยมุ่งเน้นการประมาณค่าฟังก์ชันความน่าจะเป็นแบบมีเงื่อนไข $P(W_t|W_1, W_2, \ldots, W_n)$ โดยที่ W_n คือคำหรือโทเค็น tokens ในลำดับหรือประโยค เพื่อการทำนายคำถัดไปจากลำดับคำที่ปรากฏก่อนหน้า โดยอาศัยโครงข่ายประสาทเทียมแบบ Feedforward เป็นกลไกหลักในกระบวนการเรียนรู้ทั้งเวกเตอร์ฝังตัวคำและพารามิเตอร์ที่เกี่ยวข้องกับการทำนาย ซึ่งกระบวนการดังกล่าวประกอบด้วยการปรับค่าถ่วงน้ำหนักผ่านกระบวนการฝึกฝนด้วยข้อมูลขนาดใหญ่และอัลกอริทึมการเรียนรู้แบบ Backpropagation

คุณลักษณะที่ทำให้ NPLMs มีประสิทธิภาพเหนือกว่าโมเดลแบบดั้งเดิมคือ Generalization Beyond N-grams (4) เนื่องจากความสามารถในการเรียนรู้รูปแบบภาษาในปริภูมิ Continuous Space ซึ่งมีส่วนสำคัญในการลดปัญหา Data Sparsity อย่างมีนัยสำคัญ จากผลการทดลองในงานวิจัยหลายชิ้นได้แสดงให้เห็นอย่างชัดเจนว่า NPLMs สามารถลดความคลาดเคลื่อนในการทำนายได้ถึงร้อยละ 20-40 เมื่อเทียบกับโมเดล N-gram ที่มีความซับซ้อนในระดับเดียวกัน

อย่างไรก็ตาม ในระยะเริ่มต้นของการพัฒนา NPLMs นักวิจัยต้องเผชิญกับความท้าทายที่สำคัญหลายประการ โดยเฉพาะอย่างยิ่งข้อจำกัดด้าน Computational Resources ซึ่งเป็นอุปสรรคสำคัญต่อการขยายขนาดโมเดลและการฝึกฝนกับชุดข้อมูลขนาดใหญ่ โดยในช่วงเวลาก่อนที่เทคโนโลยีสถาปัตยกรรมขนานของหน่วยประมวลผลกราฟิกประมวลผลกราฟิก Graphics Processing Units: GPU จะได้รับการพัฒนาและแพร่หลาย นอกจากนี้ ยังประสบกับปัญหาด้านเสถียรภาพใน Training Stability โดยเฉพาะอย่างยิ่งปรากฏการณ์การหายไปของ Vanishing Gradients ซึ่งส่งผลให้การฝึก Deep Networks เป็นไปด้วยความยากลำบาก

Recurrent Neural Network RNN ถูกพัฒนาขึ้นเพื่อแก้ปัญหาดังกล่าว เป็นสถาปัตยกรรมโครงข่ายประสาทที่ได้รับการออกแบบมาเป็นพิเศษเพื่อการประมวลผล Sequential Data อย่างมีประสิทธิภาพ ลักษณะอันโดดเด่นของสถาปัตยกรรมนี้คือความแตกต่างอย่างมีนัยสำคัญจากโครงข่ายประสาทแบบ Feedforward โดย RNN มี Recurrent Connections อันเป็นกลไกสำคัญที่ทำให้ข้อมูลสามารถถูกบันทึกและประมวลผลในรูปแบบ Hidden State ได้อย่างมีประสิทธิภาพ โดยจะมีกลไกการประมวลผลข้อมูลเชิงลำดับที่มีลักษณะเฉพาะและมีประสิทธิภาพสูง ในกระบวนการ Sequential Processing นั้น โครงข่ายจะดำเนินการปรับปรุง Hidden State h_t ในแต่ละช่วงเวลา (t) ตามสมการ:

$$h_t = f(w_h h_{t-1} + w_x x_i + b) \quad (2)$$

โดยที่

w_hและ w_x เป็นเมทริกซ์น้ำหนัก Weight Matrices

b คือค่าคงที่ไบแอส Bias

x_i คืออินพุตในช่วงเวลา t

f คือฟังก์ชันกระตุ้น Activation Function เช่น tan หรือ ReLU

RNN จะอาศัยหลักการทางคณิตศาสตร์ขั้นสูงผ่านอัลกอริทึมที่เรียกว่า Backpropagation Through Time BPTT ซึ่งมีกลไกการทำงานที่ซับซ้อนโดยการคลี่โครงข่ายออกตามมิติของช่วงเวลาเพื่อดำเนินการคำนวณค่า Gradients และทำการปรับปรุงค่าน้ำหนักในแต่ละชั้นของเครือข่าย

แม้ว่าจะแสดงให้เห็นถึงประสิทธิภาพอันโดดเด่นในการประมวลผลข้อมูลเชิงลำดับ อย่างไรก็ตาม ยังคงพบข้อจำกัดที่มีนัยสำคัญหลายประการ โดยเฉพาะอย่างยิ่งปัญหา Vanishing and Exploding Gradients ในกระบวนการ BPTT ซึ่งปัญหาดังกล่าวส่งผลกระทบอย่างมีนัยสำคัญต่อความสามารถในการเรียนรู้ความสัมพันธ์ระยะยาว ข้อจำกัดเหล่านี้ได้นำไปสู่การพัฒนาสถาปัตยกรรมที่มีความซับซ้อนมากยิ่งขึ้น เช่น Long Short-Term Memory LSTM และ Gated Recurrent Unit GRU ซึ่งมีการออกแบบกลไก Gates ที่มีประสิทธิภาพในการบรรเทาปัญหาการหายไปของความลาดชัน

การเปิดตัวสถาปัตยกรรม Transformer โดย Vaswani และคณะ ในงานวิจัยชื่อ "Attention Is All You Need" 2017 นับเป็นจุดเปลี่ยนสำคัญในพัฒนาการของ NLP ด้วยการแก้ปัญหาข้อจำกัดของโครงข่าย RNN และ LSTM โมเดล Transformer จึงสามารถบรรลุผลลัพธ์ที่ล้ำหน้าที่สุดในหลากหลายงาน NLP โดยมีองค์ประกอบสำคัญได้แก่ 1). Self-Attention 2). Parallelization 3). Positional Encoding 4). Encoder-Decoder 5). Multi-Head Attention 6). Feedforward และ Residual

Self-Attention เป็นกลไกสำคัญที่อยู่ภายใต้ความสำเร็จของสถาปัตยกรรม Transformer ซึ่งได้ปฏิวัติวง NLP และการเรียนรู้เชิงลึก โดยกลไกนี้อาศัยหลักการของการคำนวณความสัมพันธ์ระหว่างทุกๆ โทเคนในลำดับข้อมูลเดียวกัน ทั้งในด้านความหมายและบริบท กระบวนการทำงานของ Self-Attention เริ่มต้นด้วยการแปลงเวกเตอร์อินพุต $\{x_1, x_2, \ldots, x_n\}$ ให้กลายเป็นสามเมทริกซ์ ได้แก่ Query Q, Key K และ Value V ผ่านการคูณด้วยเมทริกซ์น้ำหนักที่แตกต่างกัน โดยแต่ละเมทริกซ์มีบทบาทเฉพาะในการคำนวณ จากนั้นคำนวณคะแนนความสนใจ Attention Scores โดยใช้สมการ:

$$Attention(Q, K, V) = softma\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (3)$$

โดยที่

Q, K, V คือเมทริกซ์ Query, Key, และ Value ที่ได้จาก x

d_k คือมิติของเวกเตอร์ Key ที่ใช้ในการปรับขนาด

Parallelization ในสถาปัตยกรรม Transformer นับเป็นนวัตกรรมสำคัญที่เปลี่ยนโฉมหน้าวงการ NLP โดยที่ความสามารถอันโดดเด่นนี้เกิดจากกลไกพื้นฐานของโมเดลที่ออกแบบให้สามารถคำนวณความสัมพันธ์ระหว่างโทเคนได้พร้อมกันทั้งหมด เดิมแทนที่จะต้องประมวลผลไปตามลำดับเวลาเหมือนในโมเดลแบบวนซ้ำอย่าง RNN หรือ LSTM (5) การออกแบบที่ไม่มีการพึ่งพา Previous State ทำให้ Transformer สามารถใช้ประโยชน์จาก GPU และหน่วยประมวลผลเทนเซอร์ Tensor Processing Unit: TPU ได้อย่างมีประสิทธิภาพสูงสุด

Positional Encoding เป็นองค์ประกอบสำคัญในสถาปัตยกรรม Transformer ที่แก้ไขข้อจำกัดพื้นฐานของกลไก Self-Attention ซึ่งโดยธรรมชาติแล้วไม่สามารถรับรู้ลำดับหรือตำแหน่งของโทเคนในข้อมูลได้ เนื่องจากการคำนวณแบบขนานที่ไม่คำนึงถึงลำดับเวลา Vaswani และคณะ 2017 จึงได้นำเสนอวิธีการ Encoding ตำแหน่งโดยใช้ฟังก์ชันตรีโกณมิติ sin และ cos ที่มีความถี่แตกต่างกัน โดยที่ตำแหน่งคู่ $(2i)$ จะใช้ฟังก์ชัน sine และตำแหน่งคี่ $2i+1$ จะใช้ฟังก์ชัน cosine ตามสมการ:

$$PE(pos, 2i) = sin\left(\frac{pos}{10000^{2i/d_{model}}}\right) \tag{4}$$

$$PE(pos, 2i+1) = cos\left(\frac{pos}{10000^{\frac{2i}{d_{model}}}}\right) \tag{5}$$

โดยที่

pos คือตำแหน่งของโทเคน และ d_{model} คือมิติของเวกเตอร์ฝังตัว

i คือดัชนีของมิติในเวกเตอร์ $\left(i = 0, 1, 2, \ldots, \left[\frac{d}{2}\right]\right)$

d_{model} คือจำนวนมิติทั้งหมดของเวกเตอร์ที่ใช้แทนโทเคน

Encoder-Decoder เป็นแกนหลักของสถาปัตยกรรม Transformer ที่นำเสนอโดย Vaswani และคณะในปี 2017 ซึ่งได้รับการดัดแปลงให้ทำงานร่วมกับกลไก Attention ได้อย่างมีประสิทธิภาพสูงสุด ส่วน Encoder มีหน้าที่ประมวลผลข้อมูลอินพุตทั้งหมดพร้อมกันเพื่อสร้าง Contextual Representation ในขณะที่ส่วน Decoder ทำหน้าที่สร้างผลลัพธ์ทีละโทเคนโดยใช้ข้อมูลที่ประมวลผลแล้วจาก Encoder ร่วมกับโทเคนที่สร้างมาก่อนหน้า

Multi-Head Attention เป็นนวัตกรรมสำคัญที่พัฒนาต่อยอดจาก Self-Attention ในสถาปัตยกรรม Transformer โดยแนวคิดพื้นฐานของกลไกนี้คือการเพิ่มความสามารถในการเรียนรู้ความสัมพันธ์ที่หลากหลายในข้อมูลผ่านการใช้ Attention Heads หลายหัวที่ทำงานขนานกัน แทนที่จะใช้เพียงหัวเดียวเหมือนใน Vanilla Self-Attention กระบวนการทำงานของ Multi-Head Attention เริ่มจากการแบ่งมิติของเวกเตอร์ Query Q, Key K และ Value V ออกเป็นหลายส่วนเท่ากับจำนวนหัว โดยแต่ละหัวมีเมทริกซ์น้ำหนัก W_i^Q, W_i^K และ W_i^v ที่ใช้ในการแปลงข้อมูลเป็นมิติย่อยที่แตกต่างกัน ตามสมการ:

$$head_i = Attention\left(QW_i^Q, KW_i^K, VW_i^v\right) \tag{6}$$

โดยที่

Q คือ เมทริกซ์คำถาม Query matrix

K คือ เมทริกซ์กุญแจ Key matrix

V คือ เมทริกซ์ค่า Value matrix

W_i^Q คือ เมทริกซ์น้ำหนักเฉพาะสำหรับการแปลง Q ในหัวที่ i

W_i^k คือ เมทริกซ์น้ำหนักเฉพาะสำหรับการแปลง K ในหัวที่ i

W_i^v คือ เมทริกซ์น้ำหนักเฉพาะสำหรับการแปลง V ในหัวที่ i

ผลลัพธ์จากทุกหัวจะถูกนำมารวมกันด้วย Concatenation ก่อนที่จะถูกแปลงกลับไปสู่มิติเดิมด้วยเมทริกซ์ผลลัพธ์ W^o ตามสมการ:

$$MultiHead(Q,K,V) = Concat(head_1, head_2, \dots, head_n)W^o \quad (7)$$

โดยที่

n คือ จำนวนหัวความสนใจทั้งหมด

W^o คือเมทริกซ์น้ำหนักสุดท้ายที่ใช้แปลงผลลัพธ์ที่ต่อกันแล้วให้กลับมามีขนาดเท่ากับมิติของโมเดล

Feedforward และ Residual นั้นนับเป็นองค์ประกอบสำคัญที่จะช่วยเสริมประสิทธิภาพและเสถียรภาพให้กับสถาปัตยกรรม Transformer โดยในแต่ละชั้นของ Encoder และ Decoder จะประกอบด้วย Position-wise Feedforward Network ซึ่งเป็นโครงข่ายประสาทเทียมแบบเชื่อมต่อเต็มรูปแบบที่ประมวลผลแต่ละตำแหน่งของลำดับข้อมูลแยกจากกัน โดยโครงสร้างพื้นฐานของเครือข่ายนี้ประกอบด้วยสองชั้นเชื่อมต่อเต็มรูปแบบที่มีฟังก์ชัน Rectified Linear Unit ReLU คั่นกลาง ตามสมการ: $FFN(X) = \max(0, W^2 + b^2)$ ส่วนของเมทริกซ์น้ำหนัก และ b_1, b_2 คือค่าไบอัส

Pre-trained Language Models (6) เป็นก้าวกระโดดครั้งสำคัญในวงการ NLP ซึ่งเริ่มต้นอย่างเป็นรูปธรรมในปี 2018 หลังจากการนำเสนอสถาปัตยกรรม Transformer โดย Vaswani และคณะ โมเดลเหล่านี้ได้เปลี่ยนแปลงกระบวนทัศน์การพัฒนาระบบ NLP (7) เดิมที่ต้องออกแบบสถาปัตยกรรมเฉพาะสำหรับแต่ละงาน มาเป็นการใช้โมเดลพื้นฐานขนาดใหญ่ที่ผ่านการฝึกฝนบนคลังข้อมูลมหาศาลแบบ Unsupervised Learning ก่อนนำไปปรับแต่งเพิ่มเติม

Bidirectional Encoder Representations from Transformers BERT ที่พัฒนาโดยนักวิจัยจาก Google เป็นหนึ่งในโมเดลบุกเบิกที่ใช้ส่วน Encoder ของ Transformer เพื่อเรียนรู้การแทนค่าภาษาแบบ

สองทิศทาง ผ่านการฝึกฝนด้วยงาน Masked Language Modeling MLM และ Next Sentence Prediction NSP โมเดลนี้มีสถาปัตยกรรมพื้นฐานประกอบด้วยชั้น Transformer Encoder จำนวน 12 ชั้น สำหรับ BERT-base หรือ 24 ชั้น สำหรับ BERT-large โดยมีพารามิเตอร์ 110 ล้านและ 340 ล้านตัว ตามลำดับ

ในขณะที่ Generative Pre-trained Transformer GPT (8) ที่พัฒนาโดย OpenAI ใช้ส่วน Decoder ของ Transformer ในการเรียนรู้โมเดลภาษาแบบ Autoregressive ซึ่งทำนายคำถัดไปตามบริบทก่อนหน้า GPT-1 (9) ถือกำเนิดขึ้นในปี 2018 โดย Alec Radford และคณะนักวิจัยจาก OpenAI เป็นโมเดลบุกเบิกที่วางรากฐานให้กับการพัฒนา LLM ซึ่งในยุคปัจจุบัน โครงสร้างพื้นฐานที่ประกอบด้วยชั้น Transformer Decoder จำนวน 12 ชั้น และมีพารามิเตอร์ประมาณ 117 ล้านตัว

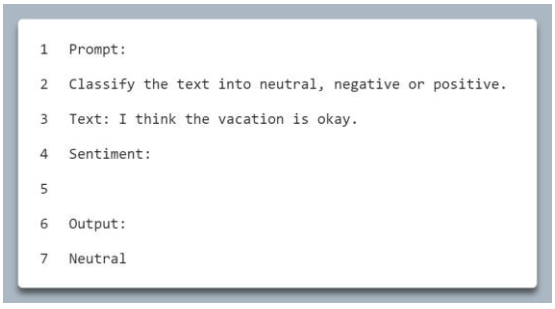
ความก้าวหน้าเหล่านี้นำไปสู่การพัฒนาโมเดลรุ่นต่อมาที่มีขนาดและความสามารถเพิ่มขึ้นอย่างต่อเนื่อง เช่น Robustly Optimized BERT Pretraining Approach RoBERTa ที่ปรับปรุงกระบวนการฝึกของ BERT, eXtreme Multi-Label Text Classification XLNet ที่ผสมผสานข้อดีของทั้ง BERT และ GPT, Text-to-Text Transfer Transformer T5 ที่มองทุกงาน NLP เป็นการแปลง Text-to-Text และ Bidirectional and Auto-Regressive Transformer BART ที่ใช้ทั้ง Encoder และ Decoder ในการสร้างโมเดลแบบ Sequence-to-Sequence แม้ LLM ได้แสดงให้เห็นถึงความสามารถที่น่าประทับใจ แต่ยังคงมีข้อบกพร่องที่ต้องได้รับการแก้ไข Prompt Engineering จึงเป็นแนวทางที่ได้รับความสนใจอย่างมากในการยกระดับประสิทธิภาพของ LLM

2. Prompt Engineering

Prompt Engineering คือกระบวนการออกแบบและปรับแต่งคำสั่งหรือ Input Prompts สำหรับโมเดล AI โดยเฉพาะในระบบที่ใช้ NLP เช่น LLM อย่าง GPT-4 เป็นต้น เป้าหมายหลักของ Prompt Engineering คือ การเพิ่มประสิทธิภาพการตอบสนองของระบบ AI ให้สามารถสร้างผลลัพธ์ที่ตรงตามความต้องการของผู้ใช้งานมากที่สุด แนวคิดนี้เริ่มเป็นที่รู้จักอย่างแพร่หลายในช่วงปี 2020-2021 เมื่อความสามารถของ LLM พัฒนาถึงจุดที่สามารถตอบสนองต่อคำสั่งที่ซับซ้อนได้ Prompt Engineering มีลักษณะเป็นทั้งศาสตร์และศิลป์ โดยอาศัยความเข้าใจในหลักการทำงานของโมเดลภาษาและควบคู่กับความคิดสร้างสรรค์ในการออกแบบคำสั่งที่มีประสิทธิภาพ โดยองค์ประกอบของ Prompt Engineering ประกอบด้วยทั้งหมดดังนี้ Context Setting ที่ช่วยให้โมเดลเข้าใจสถานการณ์และขอบเขตของงาน Role Specification ที่กำหนดว่าโมเดลควรตอบสนองในฐานะใด ซึ่งการให้ Specific Instructions ที่ระบุรายละเอียดของงานที่ต้องการ และ Output Formatting ที่ระบุโครงสร้างของคำตอบที่ต้องการ การประยุกต์ใช้ Prompt Engineering เพื่อพัฒนาการตอบสนองของ LLM

Prompt Engineering (10) มีบทบาทสำคัญยิ่งในการพัฒนาระบบ AI Chat เนื่องจากการสร้างคำสั่งที่เหมาะสมสามารถชี้นำและควบคุมพฤติกรรมของการตอบคำถามได้ ส่งผลให้เกิดการใช้งานที่หลากหลาย เช่น การสรุปข้อมูล การเขียนบทความ การแปลภาษา การวิเคราะห์ข้อมูลเชิงลึก และการสร้างเนื้อหาที่มีคุณภาพสูง นอกจากนี้ การออกแบบคำสั่งที่มีประสิทธิภาพยังช่วยลดข้อผิดพลาดในการทำงานของ AI และเพิ่มความแม่นยำในการประมวลผลข้อมูล ซึ่งเป็นประโยชน์ต่อหลากหลายภาคส่วน เช่น อุตสาหกรรม การศึกษา การแพทย์ โดยเฉพาะงานที่ต้องการข้อมูลที่น่าเชื่อถือและถูกต้องสูง การพัฒนาเทคนิค Prompt Engineering (11) จึงเป็นองค์ประกอบสำคัญในการเพิ่มประสิทธิภาพการทำงานระหว่างมนุษย์ได้ดียิ่งขึ้น โดยเฉพาะในยุคที่เทคโนโลยี AI มีการพัฒนาอย่างก้าวกระโดด ความท้าทายสำคัญของ Prompt Engineering คือการสร้างสมดุลระหว่างความชัดเจนและความยืดหยุ่น กล่าวคือ คำสั่งที่มีรายละเอียดมากเกินไปอาจจำกัดความคิดสร้างสรรค์และความสามารถในการปรับตัวของ AI ในขณะที่คำสั่งที่กว้างเกินไปอาจนำไปสู่ผลลัพธ์ที่ไม่ตรงตามความต้องการหรือขาดความแม่นยำ นักวิจัยในปัจจุบันจึงให้ความสำคัญกับการพัฒนาวิธีการและแนวทางการออกแบบคำสั่งที่สามารถปรับเปลี่ยนได้ตามบริบทและวัตถุประสงค์ที่แตกต่างกัน โดยผสมผสานองค์ความรู้จากหลากหลายศาสตร์ เช่น ภาษาศาสตร์ จิตวิทยาการรับรู้ และวิทยาการคอมพิวเตอร์ เพื่อพัฒนากรอบแนวคิดและหลักการที่สามารถนำไปประยุกต์ใช้ได้อย่างกว้างขวาง อย่างไรก็ตาม เพื่อให้เกิดความเข้าใจที่ชัดเจนเกี่ยวกับแนวคิดของ Prompt Engineering โดยจะรูปแบบตัวอย่างดังนี้

รูปแบบแรกคือ Zero-shot Learning (12) ซึ่งเป็นเทคนิคที่โมเดลภาษาสามารถตอบสนองต่อคำสั่งได้โดยไม่จำเป็นต้องมีตัวอย่างหรือบริบทเพิ่มเติม กระบวนการนี้อาศัยความสามารถในการถ่ายโอนความรู้ที่โมเดลได้เรียนรู้มาก่อนหน้าเพื่อประมวลผลคำสั่งใหม่ที่ไม่เคยพบมาก่อน ตัวอย่างเช่น การถามให้โมเดลแปลข้อความจากภาษาหนึ่งไปยังอีกภาษาหนึ่งโดยไม่ได้แสดงตัวอย่างการแปลมาก่อน หรือการให้โมเดลสรุปประเด็นสำคัญของบทความโดยไม่มีตัวอย่างการสรุปให้ดู (13) แม้ว่าเทคนิคนี้จะมีความยืดหยุ่นสูง แต่อาจมีข้อจำกัดในด้านความแม่นยำเมื่อเทียบกับเทคนิคอื่น ๆ โดยเฉพาะในกรณีที่ต้องการผลลัพธ์ที่มีรูปแบบเฉพาะหรือต้องการความเชี่ยวชาญในสาขาเฉพาะทาง

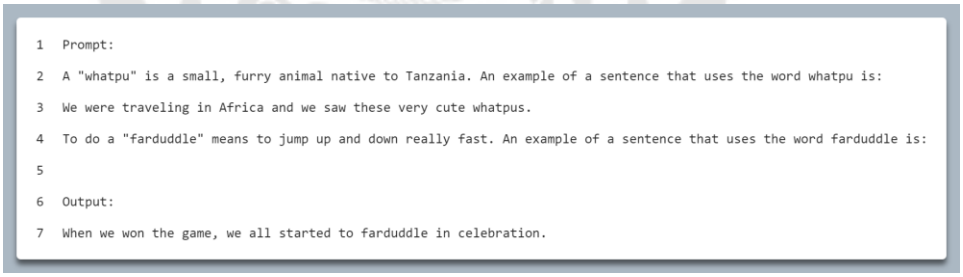


```
Prompt:
Classify the text into neutral, negative or positive.
Text: I think the vacation is okay.
Sentiment:

Output:
Neutral
```

ภาพประกอบ 2 ตัวอย่างของ Prompt Engineering รูปแบบ Zero-shot

จากภาพประกอบ 2 แสดงให้เห็น เป็นตัวอย่างของการใช้ Zero-shot Prompting (14) ในการวิเคราะห์ความรู้สึก Sentiment Analysis โดยมีโครงสร้างประกอบด้วยส่วนคำสั่ง Prompt ที่ระบุให้จำแนกข้อความเป็นกลาง Neutral, เชิงลบ Negative หรือเชิงบวก Positive ข้อความที่นำมาวิเคราะห์คือ "I think the vacation is okay." ซึ่งผลลัพธ์ที่ได้คือ "Neutral" แสดงให้เห็นว่าโมเดลสามารถประมวลผลและตีความความรู้สึกของข้อความได้อย่างถูกต้องโดยไม่จำเป็นต้องมีตัวอย่างการจำแนกความรู้สึกมาก่อน นี่เป็นการสาธิตความสามารถของ Zero-shot Learning ในการถ่ายโอนความรู้จากการฝึกฝนทั่วไปไปสู่งานเฉพาะทางได้อย่างมีประสิทธิภาพ

เทคนิคที่สองคือ Few-shot Learning ซึ่งเป็นการพัฒนาต่อยอดจากแนวคิด Zero-shot โดยการเพิ่มตัวอย่างจำนวนเล็กน้อยเข้าไปในคำสั่งเพื่อให้โมเดลเข้าใจรูปแบบหรือโครงสร้างของผลลัพธ์ที่ต้องการมากขึ้น (15) วิธีการนี้ช่วยให้โมเดลสามารถเรียนรู้จากตัวอย่างและปรับตัวให้เข้ากับบริบทเฉพาะได้อย่างรวดเร็ว โดยไม่จำเป็นต้องฝึกฝนโมเดลใหม่ทั้งหมด ตัวอย่างของการใช้ Few-shot Learning ได้แก่ การให้ตัวอย่างการแยกประเภทข้อความ 2-3 ตัวอย่างก่อนที่จะให้โมเดลทำการแยกประเภทข้อความใหม่ หรือการแสดงตัวอย่างการแปลงข้อมูลจากรูปแบบหนึ่งไปยังอีกรูปแบบหนึ่งจำนวนไม่กี่ตัวอย่างเพื่อให้โมเดลเข้าใจรูปแบบที่ต้องการ เทคนิคนี้มีประสิทธิภาพสูงในการปรับปรุงความแม่นยำของผลลัพธ์โดยใช้ทรัพยากรน้อย และเป็นที่นิยมในการประยุกต์ใช้กับงานที่ต้องการความเฉพาะเจาะจงสูง



ภาพประกอบ 3 ตัวอย่างของ Prompt Engineering รูปแบบ Few-shot

จากภาพประกอบ 3 ภาพที่แสดงเป็นตัวอย่างที่ชัดเจนของการใช้ Few-shot Prompting ในการเรียนรู้การใช้คำศัพท์สมมติ ในส่วนของ Prompt มีการแนะนำคำศัพท์ "whatpu" ซึ่งอธิบายว่าเป็นสัตว์ขนาดเล็ก มีขนฟู ที่อาศัยอยู่ในประเทศแทนซาเนีย พร้อมตัวอย่างประโยคที่ใช้คำดังกล่าว: "We were traveling in Africa and we saw these very cute whatpus." ตามด้วยการแนะนำคำว่า "farduddle" ซึ่งหมายถึงการกระโดดขึ้นลงอย่างรวดเร็ว เมื่อโมเดลได้รับตัวอย่างเหล่านี้ ก็สามารถสร้างประโยคที่ใช้คำว่า

"farduddle" ได้อย่างถูกต้องตามบริบท: "When we won the game, we all started to farduddle in celebration."

เทคนิคที่สามคือ Chain of Thought CoT ซึ่งเป็นวิธีการที่ซับซ้อนมากขึ้นโดยการออกแบบคำสั่งให้โมเดลแสดงกระบวนการคิดเป็นขั้นตอนก่อนที่จะให้คำตอบสุดท้าย วิธีการนี้มีประสิทธิภาพอย่างยิ่งในการแก้ปัญหาที่ต้องการเหตุผลและการคิดวิเคราะห์ เนื่องจากช่วยให้โมเดลสามารถแตกปัญหาออกเป็นส่วนย่อย ๆ และพิจารณาแต่ละส่วนอย่างเป็นระบบ ตัวอย่างของการใช้ CoT ได้แก่ การแก้โจทย์ปัญหาทางคณิตศาสตร์ที่ซับซ้อนโดยการแสดงวิธีทำทีละขั้นตอน หรือการวิเคราะห์ประเด็นทางกฎหมายโดยการพิจารณาองค์ประกอบต่าง ๆ อย่างเป็นระบบ การศึกษาวิจัยพบว่าการใช้เทคนิค CoT สามารถเพิ่มความแม่นยำของโมเดลได้อย่างมีนัยสำคัญ โดยเฉพาะอย่างยิ่งในโจทย์ปัญหาที่ต้องการการคิดวิเคราะห์และการใช้เหตุผลในระดับสูง

```
Prompt:
The odd numbers in this group add up to an even number: 4, 8, 9, 15, 12, 2, 1.
A: Adding all the odd numbers (9, 15, 1) gives 25. The answer is False.
The odd numbers in this group add up to an even number: 17,  10, 19, 4, 8, 12, 24.
A: Adding all the odd numbers (17, 19) gives 36. The answer is True.
The odd numbers in this group add up to an even number: 16,  11, 14, 4, 8, 13, 24.
A: Adding all the odd numbers (11, 13) gives 24. The answer is True.
The odd numbers in this group add up to an even number: 17,  9, 10, 12, 13, 4, 2.
A: Adding all the odd numbers (17, 9, 13) gives 39. The answer is False.
The odd numbers in this group add up to an even number: 15, 32, 5, 13, 82, 7, 1.
A:

Output:
Adding all the odd numbers (15, 5, 13, 7, 1) gives 41. The answer is False.
```

ภาพประกอบ 4 ตัวอย่างของ Prompt Engineering รูปแบบ CoT

จากภาพประกอบ 4 เป็นตัวอย่างที่ชัดเจนของการใช้ CoT ในการแก้ปัญหาทางคณิตศาสตร์ โดยโจทย์กำหนดให้ตรวจสอบว่าผลรวมของตัวเลขคี่ในชุดตัวเลขที่กำหนดให้จะเป็นจำนวนคู่หรือไม่ ในส่วนของ Prompt แสดงตัวอย่างการแก้ปัญหาหลายชุด โดยในแต่ละตัวอย่างมีการแสดงขั้นตอนการคิดอย่างชัดเจน ได้แก่ การระบุตัวเลขคี่ในชุดข้อมูล การคำนวณผลรวม และการสรุปผลว่าคำตอบเป็นจริงหรือเท็จ เช่น "Adding all the odd numbers 17, 19 gives 36. The answer is True." หรือ "Adding all the odd numbers 17, 9, 13 gives 39. The answer is False."

โดยในส่วนของ Output แสดงให้เห็นว่าโมเดลได้เรียนรู้รูปแบบการคิดจากตัวอย่างและสามารถประยุกต์ใช้กับโจทย์ใหม่ได้อย่างถูกต้อง โดยมีการระบุว่า "Adding all the odd numbers 15, 5, 13, 7, 1 gives 41. The answer is False." ซึ่งแสดงให้เห็นว่าโมเดลไม่เพียงแต่ให้คำตอบสุดท้าย แต่ยังแสดง

วิธีการคิดโดยการระบุตัวเลขคี่ คำนวณผลรวม และสรุปว่าเป็นเท็จเนื่องจาก 41 เป็นจำนวนคี่ ไม่ใช่จำนวนคู่ตามที่โจทย์ต้องการ

แต่ละเทคนิคของ Prompt Engineering ที่กล่าวมานั้นมีจุดแข็งและข้อจำกัดที่แตกต่างกัน โดย Zero-shot Learning มีความยืดหยุ่นสูงและใช้งานได้ง่ายแต่อาจมีข้อจำกัดด้านความแม่นยำ Few-shot Learning ซึ่งจะช่วยเพิ่มความแม่นยำโดยใช้ตัวอย่างเพียงไม่กี่ตัวอย่างซึ่งเหมาะกับงานที่มีรูปแบบเฉพาะ ในขณะที่ CoT จะมีประสิทธิภาพสูงในการแก้ปัญหาที่ซับซ้อนแต่อาจต้องใช้ทรัพยากรและเวลาในการประมวลผลมากกว่า การเลือกใช้เทคนิคใดขึ้นอยู่กับลักษณะของงาน ความซับซ้อนของปัญหา และทรัพยากรที่มีอยู่ อย่างไรก็ตาม แม้ว่าเทคนิคเหล่านี้จะช่วยปรับปรุงประสิทธิภาพของโมเดลภาษาได้อย่างมีนัยสำคัญ แต่ยังคงมีข้อจำกัดบางประการดังที่ได้กล่าวไปแล้วในตอนต้น ซึ่งนำไปสู่การพัฒนาเทคโนโลยี RAG เพื่อแก้ไขข้อจำกัดเหล่านั้นและยกระดับความสามารถของระบบ AI ให้สูงขึ้น

แม้จะเป็นกระบวนการที่มีความสำคัญอย่างยิ่งในการยกระดับประสิทธิภาพของระบบ AI แต่กระบวนการดังกล่าวประสบข้อจำกัดหลายประการที่มีนัยสำคัญต่อการพัฒนาเทคโนโลยีสารสนเทศในปัจจุบัน ประเด็นสำคัญประการแรกคือการขาดความยืดหยุ่นในการปรับปรุงข้อมูลแบบ Real-time ซึ่งส่งผลให้ระบบไม่สามารถจัดการกับข้อมูลที่มีการเปลี่ยนแปลงอย่างรวดเร็วได้อย่างมีประสิทธิภาพ อีกทั้งยังมีข้อจำกัดด้านทรัพยากรที่ต้องใช้ในกระบวนการทดลองและปรับแต่งคำสั่งซ้ำหลายครั้ง ซึ่งสร้างภาระด้านต้นทุนและเวลาอย่างมีนัยสำคัญ นอกจากนี้ ระบบ Prompt Engineering ยังประสบปัญหาความไม่แม่นยำในการตอบคำถามที่ต้องการ Domain-specific Knowledge และความไม่แน่นอนในการควบคุมผลลัพธ์ โดยเฉพาะอย่างยิ่งในสถานการณ์ที่คำสั่งมีความกำกวมหรือซับซ้อน

ข้อจำกัดที่มีความสำคัญอย่างยิ่งอีกประการหนึ่งของ Prompt Engineering คือการขาดความสามารถในการเข้าถึงและประมวลผลข้อมูลที่อยู่นอกชุด Knowledge Cutoff Limitations ซึ่งเป็นอุปสรรคสำคัญในการตอบสนองต่อคำถามที่เกี่ยวข้องกับข้อมูลปัจจุบันหรือข้อมูลเฉพาะทางที่ไม่ปรากฏในชุดข้อมูลฝึกฝน เพื่อตอบสนองต่อข้อจำกัดเหล่านี้ เทคโนโลยี RAG จึงได้รับการพัฒนาขึ้นโดยมีวัตถุประสงค์เพื่อเพิ่มประสิทธิภาพในการประมวลผลข้อมูลของ LLM แนวคิดหลักของเทคโนโลยี RAG คือการผสมผสานความสามารถของ Information Retrieval Systems เข้ากับกลไกการสร้างเนื้อหาของโมเดลภาษา ทำให้ระบบสามารถดึงข้อมูลจากแหล่งภายนอกมาประกอบการตอบคำถามได้อย่างแม่นยำและเป็นปัจจุบัน

3. Retrieval-Augmented Generation (RAG)

RAG นับเป็นพัฒนาการสำคัญที่ส่งผลต่อการเปลี่ยนแปลงขีดความสามารถของระบบปัญญาประดิษฐ์เชิงสร้างสรรค์อย่างมีนัยสำคัญ โดยสถาปัตยกรรมดังกล่าวมีวัตถุประสงค์เพื่อปรับปรุงกระบวนการทำงานของ LLM ให้สามารถประมวลผลและสร้างคำตอบที่มีความถูกต้อง แม่นยำ อีกทั้งยังสอดคล้องกับบริบทเฉพาะทางที่ผู้ใช้ต้องการ การผสานรวมระหว่าง Retrieval และ Generation (16) นี้ได้นำไปสู่การยกระดับประสิทธิภาพของระบบ NLP ซึ่งสามารถเข้าถึงแหล่งข้อมูลภายนอกที่มีความทันสมัยและเฉพาะทางมากขึ้น

โดยมีกระบวนการทำงานมีลักษณะเป็นสองขั้นตอนที่มีความสัมพันธ์กันอย่างแนบแน่น โดยขั้นตอนแรกคือ คลังความรู้ภายนอก ซึ่งอาจเป็นเอกสาร ฐานข้อมูล หรือแหล่งข้อมูลเฉพาะทางขององค์กร ข้อมูลที่ถูก Retrieval จะถูกประมวลผลและคัดกรองให้มีความเกี่ยวข้องกับคำถามหรือบริบทที่ผู้ใช้กำหนด จากนั้นในขั้นตอนที่สอง ข้อมูลดังกล่าวจะถูกนำมาผสานเข้ากับความรู้พื้นฐานที่แบบจำลองได้รับการฝึกฝนมาก่อนหน้า เพื่อสร้างเนื้อหาผลลัพธ์ที่มีความสมบูรณ์ทั้งในแง่ของความรู้ทั่วไปและข้อมูลเฉพาะทางที่ทันสมัย กลไกนี้จึงเป็นการแก้ไขข้อจำกัดพื้นฐานของแบบจำลองภาษาดั้งเดิมที่มักประสบปัญหาเกี่ยวกับความเป็นปัจจุบันของข้อมูลและความเชี่ยวชาญเฉพาะทาง

การประยุกต์ใช้ RAG (17) ในบริบททางวิชาการและอุตสาหกรรมปรากฏให้เห็นอย่างแพร่หลายและหลากหลาย โดยในภาคธุรกิจและองค์กร RAG สามารถพัฒนาเป็นระบบแชตบอทอัจฉริยะที่มีความสามารถในการเข้าถึงและประมวลผลข้อมูลภายในองค์กร ยกตัวอย่างเช่น เอกสารนโยบาย คู่มือการปฏิบัติงาน หรือฐานความรู้เฉพาะทาง ส่งผลให้การให้บริการและการสนับสนุนการทำงานของบุคลากรมีประสิทธิภาพสูงขึ้นอย่างเห็นได้ชัด ในขณะเดียวกัน ในบริบททางวิชาการโดยเฉพาะในสาขาที่ต้องการความถูกต้องและความน่าเชื่อถือของข้อมูลสูง ยกตัวอย่างเช่น การแพทย์ การศึกษา หรือการวิจัยวิทยาศาสตร์ RAG ได้แสดงศักยภาพในการนำเสนอข้อมูลที่มีความถูกต้องตามหลักวิชาการโดยอ้างอิงจากแหล่งที่มาที่เชื่อถือได้เท่านั้น เช่น วารสารวิชาการ เอกสารวิจัย หรือข้อมูลจากหน่วยงานที่มีความน่าเชื่อถือ ดังนั้น RAG จึงมิได้เป็นเพียงการพัฒนาทางด้านเทคโนโลยีเท่านั้น หากแต่ยังเป็นการยกระดับมาตรฐานการปฏิสัมพันธ์ระหว่างมนุษย์และ AI ให้มีความลึกซึ้งและมีประสิทธิภาพมากยิ่งขึ้น

ผลการศึกษาวิจัยเชิงประจักษ์ล่าสุดได้แสดงให้เห็นว่า การนำ RAG มาประยุกต์ใช้สามารถลดปัญหาของการ Hallucination ของแบบจำลองภาษาได้อย่างมีนัยสำคัญทางสถิติ ทั้งนี้เนื่องจากการมีแหล่งข้อมูลอ้างอิงที่ชัดเจนและเชื่อถือได้ช่วยให้กระบวนการสร้างคำตอบมีความโปร่งใสและสามารถตรวจสอบย้อนกลับได้ ซึ่งเป็นคุณลักษณะสำคัญในการสร้างความน่าเชื่อถือให้กับระบบ AI นอกจากนี้ ยังพบว่า RAG มีศักยภาพในการเพิ่มประสิทธิภาพการทำงานขององค์กรผ่านการบูรณาการระหว่าง

เทคโนโลยี AI กับฐานความรู้เฉพาะทาง ก่อให้เกิดระบบสนับสนุนการตัดสินใจ ที่มีความแม่นยำสูงและสามารถปรับเปลี่ยนให้เข้ากับบริบทเฉพาะของแต่ละองค์กรได้อย่างมีประสิทธิภาพ

ด้วยคุณลักษณะดังกล่าว RAG จึงมิได้เป็นเพียงนวัตกรรมทางเทคโนโลยีที่มีความน่าสนใจในเชิงวิชาการเท่านั้น หากแต่ยังเป็นเครื่องมือที่มีศักยภาพสูงในการแก้ไขปัญหาและพัฒนาการให้บริการในบริบทที่หลากหลาย การวิจัยและพัฒนาในอนาคตควรมุ่งเน้นไปที่การปรับปรุงประสิทธิภาพของกลไก Retrieval ข้อมูลให้มีความแม่นยำสูงขึ้น การลดต้นทุนในการประมวลผล และการพัฒนาระบบที่สามารถรองรับภาษาและวัฒนธรรมที่หลากหลายมากขึ้น เพื่อให้ประโยชน์ของเทคโนโลยี RAG สามารถแพร่กระจายไปสู่ผู้ใช้งานในวงกว้างและส่งผลกระทบเชิงบวกต่อสังคมในภาพรวม โดย RAG กระบวนการทำงานประกอบด้วยขั้นตอนสำคัญทั้งหมด 4 ขั้นตอนที่มีความสัมพันธ์กันอย่างเป็นระบบ อันได้แก่ อันได้แก่ 1). Indexing 2). Retrieval 3). Augmentation 4). Generation

Indexing ซึ่งเป็นกลไกพื้นฐานที่มีบทบาทสำคัญในการแปลงข้อมูลข้อความให้อยู่ในรูปแบบที่ระบบประมวลผลสามารถจัดการได้อย่างมีประสิทธิภาพนับเป็นจุดเริ่มต้นที่สำคัญของระบบ RAG กระบวนการดังกล่าวมีวัตถุประสงค์เพื่อสร้างโครงสร้างข้อมูลที่เอื้ออำนวยต่อ Retrieval อย่างรวดเร็วและแม่นยำ อันเป็นปัจจัยสำคัญที่ส่งผลต่อคุณภาพของระบบ RAG (18) โดยรวม วิธีการที่ได้รับความนิยมอย่างแพร่หลายในการจัดทำดัชนีข้อมูลคือการแปลงข้อความให้อยู่ในรูปแบบ Vector Representation ผ่านเทคนิค Embedding ซึ่งเป็นการถ่ายทอดความหมายและความสัมพันธ์ของข้อความในรูปแบบของตัวเลขหลายมิติ กระบวนการนี้อาศัยสถาปัตยกรรมโครงข่ายประสาทเทียมที่ผ่านการฝึกฝนให้มีความเข้าใจเชิงความหมายของภาษา โดยเวกเตอร์ที่ได้จะเก็บคุณลักษณะเชิงความหมาย ของข้อความไว้ในรูปแบบที่สามารถคำนวณความคล้ายคลึงและความสัมพันธ์ระหว่างข้อความได้ การแปลงข้อมูลในลักษณะนี้เป็นพื้นฐานสำคัญที่ทำให้ LLM สามารถประมวลผลและเข้าใจบริบทของข้อมูลได้อย่างลึกซึ้ง

Retrieval ซึ่งเป็นองค์ประกอบสำคัญอีกประการหนึ่งของระบบ RAG มีหน้าที่หลักในการดึงข้อมูลบริบทที่เกี่ยวข้องจากฐานความรู้ภายนอกมาสนับสนุนการสร้างคำตอบที่ถูกต้องและน่าเชื่อถือ การทำงานในส่วนนี้มุ่งเน้นไปที่การดึงข้อมูลที่มีความเกี่ยวข้องสูงสุดจากแหล่งความรู้ภายนอก ซึ่งอาจเป็นได้ทั้งฐานข้อมูลองค์กร เอกสารวิชาการ หรือแหล่งข้อมูลเฉพาะทางอื่นๆ โดยมีระบบที่เรียกว่า Retriever ทำหน้าที่เป็นกลไกหลักในการสืบค้นและคัดกรองข้อมูลจากคลังข้อมูลที่ได้รับการจัดทำดัชนีไว้ล่วงหน้า กลไกการทำงานของตัวเรียกค้นข้อมูลประกอบด้วยขั้นตอนที่มีความซับซ้อนและละเอียดอ่อน โดยเริ่มจากการแปลงคำถามหรือข้อความสืบค้นให้อยู่ในรูปแบบ Dense Vector ด้วยโมเดลการ Encoding ที่มีประสิทธิภาพสูง เช่น BERT หรือ Dense Passage Retriever DPR ซึ่งมีความสามารถในการจับความหมายและบริบทของคำถามได้อย่างแม่นยำ

กระบวนการถัดมาใน Retrieval ข้อมูลคือการคำนวณ Semantic Similarity ระหว่างเวกเตอร์ของคำถามและเวกเตอร์ของเอกสารหรือข้อความที่อยู่ในฐานข้อมูล โดยอาศัยมาตรวัดทางคณิตศาสตร์ เช่น Dot Product หรือ Cosine Similarity เพื่อจัดอันดับความเกี่ยวข้องของข้อมูล จากนั้นระบบจะคัดเลือกข้อความที่มีค่าความคล้ายคลึงสูงสุดจำนวน k รายการ Top-k Retrieval โดยค่าพารามิเตอร์ k สามารถปรับแต่งได้ตามลักษณะและวัตถุประสงค์ของงาน เพื่อสร้างสมดุลระหว่างความครอบคลุมของข้อมูลและประสิทธิภาพในการประมวลผล การเลือกค่า k ที่เหมาะสมนับเป็นปัจจัยสำคัญที่ส่งผลต่อคุณภาพของผลลัพธ์ โดยค่า k ที่ต่ำเกินไปอาจทำให้ขาดข้อมูลสำคัญ ในขณะที่ค่า k ที่สูงเกินไปอาจนำไปสู่การประมวลผลที่ล่าช้าและการรวมข้อมูลที่ไม่เกี่ยวข้องเข้ามาในกระบวนการสร้างคำตอบ

พัฒนาการในเทคโนโลยี Retrieval ข้อมูลได้นำไปสู่การเกิดขึ้นของ Hybrid Retrieval (19) ซึ่งเป็นการผสมผสานจุดแข็งของเทคนิค Semantic Search และ Keyword Search เข้าด้วยกัน วิธีการดังกล่าวช่วยเพิ่มประสิทธิภาพในการดึงข้อมูลที่มีความเกี่ยวข้องอย่างครอบคลุมและแม่นยำมากขึ้น โดยการค้นหาเชิงความหมายมีความสามารถในการเข้าใจบริบทและความสัมพันธ์เชิงความหมายที่ซับซ้อน ในขณะที่การค้นหาเชิงคำสำคัญมีประสิทธิภาพในการระบุและดึงข้อมูลที่มีคำสำคัญตรงกับคำถามอย่างชัดเจน การผสมผสานวิธีการทั้งสองเข้าด้วยกันจึงช่วยลดข้อจำกัดและเพิ่มความครอบคลุมใน Retrieval ข้อมูลได้อย่างมีนัยสำคัญ

Indexing และ Retrieval ที่มีประสิทธิภาพนับเป็นรากฐานสำคัญของระบบ RAG ที่มีคุณภาพ เนื่องจากส่งผลโดยตรงต่อความสามารถในการสร้างข้อความที่มีความถูกต้องตามข้อเท็จจริงและลดปัญหา Hallucination ซึ่งเป็นประเด็นท้าทายสำคัญในการพัฒนาระบบปัญญาประดิษฐ์เชิงสร้างสรรค์ วัตถุประสงค์สำคัญของกระบวน Retrieval ข้อมูลคือการเพิ่มประสิทธิภาพในการสร้างผลลัพธ์ที่มีความสอดคล้องกับข้อเท็จจริงและลดปัญหาการสร้างข้อมูลที่ขาดมูลความจริงรองรับ ซึ่งเป็นความท้าทายสำคัญของระบบปัญญาประดิษฐ์เชิงสร้างสรรค์ในปัจจุบัน การลงทุนทรัพยากรและความพยายามในการพัฒนาและปรับปรุงกระบวนการเหล่านี้จึงมีความคุ้มค่าอย่างยิ่ง เนื่องจากส่งผลต่อคุณภาพของผลลัพธ์โดยรวมและความน่าเชื่อถือของระบบ ซึ่งเป็นปัจจัยสำคัญต่อการยอมรับและนำไปประยุกต์ใช้ในบริบทที่หลากหลายทั้งในเชิงวิชาการและอุตสาหกรรม

Augmentation ในระบบ RAG ได้รับการออกแบบขึ้นเพื่อแก้ไขข้อจำกัดที่สำคัญของ LLM โดยมุ่งเน้นการบูรณาการองค์ความรู้จากแหล่งข้อมูลภายนอกเข้าสู่บริบทของอินพุต การดำเนินการดังกล่าวส่งผลให้ความสามารถในการสร้างคำตอบที่ถูกต้องและสอดคล้องกับบริบทมีประสิทธิภาพสูงขึ้นอย่างมี

นัยสำคัญ ทั้งนี้ กระบวนการเสริมข้อมูลประกอบด้วยขั้นตอนที่มีความสัมพันธ์เชื่อมโยงและเกื้อหนุนซึ่งกันและกัน ซึ่งแต่ละองค์ประกอบล้วนมีความสำคัญอย่างยิ่งต่อประสิทธิภาพโดยรวมของระบบ RAG

Integration of Retrieved Context เป็นกระบวนการพื้นฐานที่มีความสำคัญในระบบ LLM โดยกลไกหลักของกระบวนการนี้เริ่มจากการทำงานของโมดูล Retriever ซึ่งทำหน้าที่สืบค้นและดึงข้อมูลที่มีความเกี่ยวข้องจากแหล่งความรู้ภายนอกที่หลากหลาย ครอบคลุมทั้งข้อความจากเอกสารและเว็บไซต์ ข้อมูลเชิงโครงสร้างจากฐานข้อมูลหรือกราฟความรู้ ตลอดจนข้อมูลในรูปแบบมัลติโหมด เช่น รูปภาพและตาราง กระบวน Retrieval ดังกล่าวอาศัยอัลกอริธึมการค้นหาเชิงความหมายที่มีประสิทธิภาพสูง เช่น Dense Vector Retrieval หรือ Best Matching 25 BM25 ซึ่งสามารถประเมินและคัดกรองข้อมูลที่มีความเกี่ยวข้องกับคำถามหรือหัวข้อที่กำลังพิจารณาได้อย่างแม่นยำ

เมื่อได้ข้อมูลที่ผ่าน Retrieval แล้ว ขั้นตอนต่อไปคือการผสมผสานข้อมูลเหล่านั้นเข้ากับคำถามของผู้ใช้เพื่อสร้างบริบทอินพุตที่เสริมข้อมูล โดยวิธีการบูรณาการข้อมูลมีหลายรูปแบบที่แตกต่างกันไปตามลักษณะของงานและโครงสร้างของข้อมูล วิธีการที่พบได้บ่อย ได้แก่ Concatenation ซึ่งเป็นการเพิ่มข้อมูลที่ดึงมาไว้ต่อท้ายคำถามของผู้ใช้ ในขณะที่ Hierarchical Integration มีความซับซ้อนมากกว่าโดยจัดโครงสร้างอินพุตในลักษณะที่รักษาความแตกต่างระหว่างคำถามต้นฉบับและข้อมูลที่เสริมเข้ามา การบูรณาการข้อมูลอย่างมีประสิทธิภาพไม่เพียงแต่ช่วยให้บริบทอินพุตมีความสอดคล้องกันเท่านั้น แต่ยังส่งเสริมให้แบบจำลองสามารถประมวลผลและใช้ประโยชน์จากข้อมูลเพิ่มเติมได้อย่างเต็มประสิทธิภาพ อันนำไปสู่การตอบสนองที่มีความแม่นยำและสอดคล้องกับข้อเท็จจริงมากยิ่งขึ้น อย่างไรก็ตาม โดยประสิทธิภาพของการบูรณาการข้อมูลยังขึ้นอยู่กับการออกแบบอย่างละเอียดรอบคอบในการกำหนดวิธีการจัดเรียงและการให้น้ำหนักของข้อมูลที่ดึงมานั้น ทั้งนี้เพื่อป้องกันการบิดเบือนหรือการให้ความสำคัญกับข้อมูลที่ไม่เกี่ยวข้องมากเกินไป ซึ่งอาจส่งผลเสียต่อคุณภาพของคำตอบที่สร้างขึ้น การพิจารณาปัจจัยเหล่านี้อย่างรอบคอบจึงเป็นสิ่งสำคัญในการพัฒนาระบบ RAG ที่มีประสิทธิภาพและน่าเชื่อถือ

Preprocessing for Compatibility เป็นขั้นตอนสำคัญอีกประการหนึ่งในการประมวลผลข้อมูลก่อนที่จะป้อนเข้าสู่แบบจำลองภาษา กระบวนการนี้ประกอบด้วยการแปลงข้อมูล ซึ่งเป็นการปรับเปลี่ยนรูปแบบของข้อมูลที่ได้รับจาก Retrieval ให้มีความเข้ากันได้กับโครงสร้างอินพุตของแบบจำลอง โดยผ่านกระบวนการต่างๆ เช่น Tokenization ที่แปลงข้อความให้อยู่ในรูปของลำดับโทเคนที่แบบจำลองสามารถประมวลผลได้ การสรุปความที่ช่วยย่อข้อมูลที่มีขนาดใหญ่โดยยังคงรักษาสาระสำคัญไว้ และการปรับรูปแบบที่จัดโครงสร้างข้อมูลให้สอดคล้องกับข้อกำหนดเฉพาะของแบบจำลอง นอกจากนี้ การควบคุม

คุณภาพ ยังเป็นองค์ประกอบสำคัญของกระบวนการเตรียมข้อมูล โดยใช้กลไกการกรองและการจัดลำดับความสำคัญเพื่อคัดกรองข้อมูลที่ไม่เกี่ยวข้องออกและเลือกเฉพาะข้อมูลที่มีความสำคัญสูงสุด วิธีการที่ใช้ในการควบคุมคุณภาพประกอบด้วยเทคนิค Relevance Scoring ซึ่งกำหนดค่าคะแนนให้กับข้อมูลแต่ละชิ้นตามระดับความสอดคล้องกับคำถาม และการกำจัดความซ้ำซ้อน ที่ช่วยลบข้อมูลที่มีเนื้อหาซ้ำกันหรือมีความหมายคล้ายคลึงกันเพื่อปรับแต่งอินพุตให้มีประสิทธิภาพสูงสุด ประสิทธิภาพของการเตรียมข้อมูลยังได้รับการเสริมด้วยการประยุกต์ใช้การเรียนรู้แบบวนซ้ำ ที่ปรับปรุงกระบวนการตามผลลัพธ์ที่ได้รับ และการบูรณา Advanced NLP Technologies เพื่อเพิ่มความแม่นยำในการวิเคราะห์ความสัมพันธ์เชิงความหมายระหว่างข้อมูลและคำถาม

ขั้นตอนต่อมาในกระบวนการเสริมข้อมูลคือ Dynamic Context Creation ซึ่งเป็นการรวมข้อมูลที่ผ่าน Retrieval และเตรียมพร้อมแล้วเข้ากับคำถามของผู้ใช้เพื่อสร้างบริบทเฉพาะงานที่มีความยืดหยุ่นและสามารถปรับเปลี่ยนตามความต้องการของอินพุตแต่ละรายการ กระบวนการนี้มีบทบาทสำคัญในการลดความคลุมเครือของคำถามที่อาจมีหลายความหมายหรือสามารถตีความได้หลากหลาย โดยข้อมูลเพิ่มเติมที่เฉพาะเจาะจงช่วยให้แบบจำลองสามารถเข้าใจบริบทและความต้องการที่แท้จริงของผู้ใช้ได้อย่างชัดเจนมากขึ้น การสร้างบริบทแบบไดนามิกยังช่วยเพิ่มความเฉพาะเจาะจงในการตอบคำถาม ทำให้คำตอบที่สร้างขึ้นมีความสอดคล้องและเหมาะสมกับความต้องการเฉพาะของผู้ใช้มากยิ่งขึ้น นอกจากนี้ ยังมีส่วนสำคัญในการปรับปรุงความสามารถทั่วไปของแบบจำลอง โดยการอ้างอิงข้อมูลที่ได้จาก Retrieval ช่วยเติมเต็มข้อจำกัดของแบบจำลองที่ผ่านการฝึกล่วงหน้า และส่งเสริมให้แบบจำลองสามารถตอบคำถามที่อยู่นอกเหนือขอบเขตของชุดข้อมูลการฝึกอบรมดั้งเดิมได้อย่างมีประสิทธิภาพ องค์ประกอบสำคัญอีกประการหนึ่งของการสร้างบริบทแบบไดนามิกคือ Context Window Scaling ซึ่งเป็นการปรับความยาวของบริบทให้เหมาะสมตามความซับซ้อนของคำถามและปริมาณข้อมูลที่จำเป็นต้องใช้ในการสร้างคำตอบ รวมถึง Information Prioritization ที่วางข้อมูลที่มีความเกี่ยวข้องสูงสุดไว้ในตำแหน่งที่มีความสำคัญภายในโครงสร้างของบริบท กระบวนการเหล่านี้ล้วนมีส่วนช่วยเพิ่มประสิทธิภาพในการประมวลผลและความแม่นยำของการตอบสนองโดยแบบจำลองภาษา

ประโยชน์ที่สำคัญอย่างยิ่งของการเสริมบริบทด้วยข้อมูลจากภายนอกคือการแก้ไขข้อจำกัดของแบบจำลองภาษาแบบฝึกล่วงหน้า โดยเฉพาะอย่างยิ่งปัญหาเกี่ยวกับการแก้ไขปัญหาความล้าสมัยของความรู้ เนื่องจากแบบจำลอง LLM มีข้อจำกัดด้านชุดข้อมูลการฝึกที่ถูกจำกัดอยู่ในช่วงเวลาหนึ่ง การเสริมข้อมูลจึงเป็นวิธีการที่มีประสิทธิภาพในการช่วยให้แบบจำลองสามารถบูรณาการข้อมูลที่ทันสมัยและมาจากแหล่งที่เชื่อถือได้จากภายนอก นอกจากนี้ Grounding Responses ยังช่วยให้

แบบจำลองสามารถสร้างคำตอบที่มีความน่าเชื่อถือและเกี่ยวข้องกับบริบท ลดความเสี่ยงในการสร้าง Hallucination ซึ่งเป็นปัญหาที่พบบ่อยใน LLM

ตัวอย่างที่ชัดเจนของการประยุกต์ใช้กระบวนการเสริมข้อมูลสามารถเห็นได้จากกรณีที่ผู้ใช้สอบถามเกี่ยวกับ "ความก้าวหน้าล่าสุดในด้านการประมวลผลควอนตัม" โดยระบบจะเริ่มจากการดึงข้อมูลโดยโมดูล Retriever ซึ่งเข้าถึงแหล่งข้อมูลที่น่าเชื่อถือ เช่น บทความวิชาการ รายงานเชิงเทคนิค หรือแหล่งข่าวที่เกี่ยวข้องกับความก้าวหน้าในด้านการประมวลผลควอนตัม จากนั้นข้อมูลที่ได้รับจะผ่านการประมวลผลและผสานเข้ากับคำถามของผู้ใช้เพื่อสร้างบริบทอินพุตแบบไดนามิกที่มีความครอบคลุมและเฉพาะเจาะจง ก่อนที่แบบจำลองภาษาจะนำบริบทที่ได้รับการเสริมนี้ไปใช้ในการสร้างคำตอบที่ทันสมัยและมีความถูกต้องตามข้อเท็จจริง นอกจากนี้ กระบวนการเสริมข้อมูลยังรวมถึงการปรับแต่งความเฉพาะเจาะจงตามโดเมน ซึ่งช่วยให้แบบจำลองสามารถปรับตัวเข้ากับศัพท์เฉพาะทางและแนวคิดในสาขาวิชาที่หลากหลาย ทำให้การตอบสนองมีความเหมาะสมกับบริบทเฉพาะทางมากยิ่งขึ้น อีกทั้งยังมีการเพิ่มความโปร่งใสในแหล่งที่มา ซึ่งช่วยสร้างความน่าเชื่อถือให้กับผู้ใช้โดยการระบุแหล่งที่มาของข้อมูลที่นำมาใช้ในการสร้างคำตอบ ทำให้ผู้ใช้สามารถตรวจสอบและประเมินความน่าเชื่อถือของข้อมูลได้ด้วยตนเอง

ขั้นตอนสุดท้ายในกรอบการทำงานของ RAG คือ Generation ซึ่งถือเป็นกลไกหลักที่ส่งมอบคุณค่าให้แก่ผู้ใช้งานระบบ ในขั้นตอนนี้ Augmented Input Context ซึ่งเกิดจากการผสานคำถามของผู้ใช้เข้ากับข้อมูลภายนอกที่ผ่าน Retrieval และประมวลผลแล้ว จะถูกนำไปประมวลผลโดย LLM เพื่อสร้างคำตอบที่ถูกต้อง สอดคล้องกับบริบท และตรงตามความต้องการของผู้ใช้ ความสำเร็จของระบบ RAG โดยรวมจึงขึ้นอยู่กับประสิทธิภาพของขั้นตอนนี้เป็นอย่างมาก เนื่องจากเป็นจุดที่แปลงข้อมูลและความรู้ทั้งหมดให้กลายเป็นผลลัพธ์ที่มีคุณค่าและนำไปใช้ประโยชน์ได้จริง

การประมวลผลบริบทอินพุตที่ได้รับการเสริมเป็นจุดเริ่มต้นสำคัญของกระบวนการสร้างเนื้อหา โดยบริบทดังกล่าวซึ่งเกิดจากขั้นตอน Augmentation ก่อนหน้านี้ จะถูกป้อนเข้าสู่แบบจำลอง LLM เพื่อการประมวลผลในเชิงลึก บริบทที่ได้รับการเสริมนี้ประกอบด้วยองค์ประกอบสำคัญสองส่วน ได้แก่ คำถามหรือคำอธิบายงานที่เป็นต้นฉบับจากผู้ใช้ และข้อมูลภายนอกที่เกี่ยวข้องซึ่งถูกดึงมาและผสานเข้าไปในบริบท โดยแบบจำลองจะดำเนินการตีความบริบทนี้อย่างละเอียดเพื่อสร้างความเข้าใจที่ครอบคลุมเกี่ยวกับคำถามและบริบทที่เกี่ยวข้อง กระบวนการนี้มีความสำคัญอย่างยิ่งในการลดความคลุมเครือที่อาจเกิดขึ้นในคำถามดั้งเดิม และเพิ่มความเฉพาะเจาะจงให้กับคำตอบที่จะถูกสร้างขึ้น ความสามารถในการเข้าใจบริบทอย่างลึกซึ้งนี้เป็นปัจจัยสำคัญที่ส่งผลต่อคุณภาพของคำตอบที่จะได้รับในขั้นตอนถัดไป

เมื่อแบบจำลองเข้าใจบริบทอย่างถ่องแท้แล้ว จะเข้าสู่ขั้นตอน Response Generation ซึ่งเป็นขั้นตอนสำคัญที่อาศัยการผสานระหว่าง Pre-trained Knowledge กับ Contextual Information ที่ปรากฏในอินพุต กลไกพื้นฐานสำคัญที่ทำให้แบบจำลองสามารถสร้างคำตอบที่มีคุณภาพสูงได้คือ Self-Attention ในสถาปัตยกรรม Transformer ซึ่งช่วยให้แบบจำลองสามารถวิเคราะห์และบูรณาการข้อมูลจากคำถามและแหล่งข้อมูลที่หลากหลายได้อย่างมีประสิทธิภาพ แบบจำลองจะคำนวณความสัมพันธ์และความสำคัญของคำและวลีต่างๆ ภายในบริบท และใช้ความสัมพันธ์เหล่านี้ในการสร้างคำตอบที่มีความสอดคล้องกับบริบทอย่างเป็นธรรมชาติ

ผลลัพธ์ที่ได้จากกระบวนการสร้างคำตอบมีความหลากหลายตามวัตถุประสงค์การใช้งานและความต้องการเฉพาะของแต่ละสถานการณ์ ในบางกรณี คำตอบอาจอยู่ในรูปแบบของบทสรุปกระชับ ที่มุ่งเน้นการย่อความสาระสำคัญจากข้อมูลปริมาณมากให้อยู่ในรูปแบบที่เข้าใจง่ายและกระชับ ในขณะที่บางสถานการณ์อาจต้องการคำอธิบายเชิงลึก ซึ่งจะให้รายละเอียดและเหตุผลประกอบอย่างครบถ้วน นอกจากนี้ คำตอบยังอาจอยู่ในรูปแบบของคำแนะนำเชิงปฏิบัติ ที่ได้รับการออกแบบมาโดยเฉพาะเพื่อสนับสนุนการตัดสินใจและการดำเนินการของผู้ใช้ หรืออาจเป็นคำตอบแบบหลายรูปแบบ ที่สามารถตีความและประมวลผลข้อมูลที่หลากหลายรูปแบบ เช่น ภาพ ตาราง และกราฟิก เพื่อนำเสนอข้อมูลในรูปแบบที่เหมาะสมที่สุดสำหรับเนื้อหานั้นๆ คุณภาพของคำตอบที่ได้จากกระบวนการสร้างนั้นขึ้นอยู่กับปัจจัยสำคัญหลายประการ ปัจจัยแรกคือความถูกต้องของข้อมูลที่ใช้ในการสร้างคำตอบ ซึ่งส่งผลโดยตรงต่อความน่าเชื่อถือของผลลัพธ์ โดยปัจจัยที่สองคือความสามารถในการเชื่อมโยงความสัมพันธ์ของข้อมูลต่างๆ ที่ปรากฏในบริบท ซึ่งช่วยให้คำตอบมีความสอดคล้องและเป็นเหตุเป็นผล และปัจจัยที่สามคือประสิทธิภาพของกลไกการประมวลผลความหมายและบริบททางภาษาของแบบจำลอง ซึ่งจะส่งผลต่อความลื่นไหลและความเป็นธรรมชาติของภาษาในคำตอบ ปัจจัยเหล่านี้ล้วนมีความสัมพันธ์กันและส่งผลร่วมกันต่อคุณภาพของคำตอบที่ผู้ใช้จะได้รับ

การรับรองความเกี่ยวข้องและความถูกต้องเป็นองค์ประกอบสำคัญอีกประการหนึ่งในกระบวนการ Generation ที่ช่วยยกระดับคุณภาพของคำตอบให้มีความน่าเชื่อถือและตรงตามความต้องการของผู้ใช้มากยิ่งขึ้น ในด้านความเกี่ยวข้องตามบริบท แบบจำลองจะทำการประเมินความสัมพันธ์ระหว่างบริบทที่ได้รับการเสริมกับคำตอบที่สร้างขึ้น เพื่อให้แน่ใจว่าผลลัพธ์มีความสอดคล้องกับคำถามของผู้ใช้และข้อมูลที่ได้รับการนำเสนอ การประเมินนี้ช่วยให้แบบจำลองสามารถปรับแต่งและขัดเกลาคำตอบให้มีความเฉพาะเจาะจงและตรงประเด็นมากยิ่งขึ้น ในแง่ของความถูกต้องตามข้อเท็จจริง การรวมข้อมูลที่ได้จาก Retrieval เข้ามาในกระบวนการสร้างคำตอบมีส่วนสำคัญอย่างยิ่งในการลดข้อผิดพลาดที่อาจเกิดขึ้นจากข้อจำกัดของฐานความรู้ที่แบบจำลองได้รับการฝึกฝนมาล่วงหน้า คำตอบที่

สร้างขึ้นจะได้รับการสนับสนุนด้วยข้อมูลที่มีความถูกต้องและสอดคล้องกับข้อเท็จจริงที่เป็นปัจจุบัน ซึ่งช่วยเพิ่มความน่าเชื่อถือและลดความเสี่ยงของการเกิด Hallucination ที่เป็นปัญหาสำคัญในแบบจำลองภาษาทั่วไป การรับรองทั้งในด้านความเกี่ยวข้องและความถูกต้องนี้จึงเป็นกลไกสำคัญที่ช่วยยกระดับคุณภาพและความน่าเชื่อถือของระบบ RAG โดยรวม

Evaluation and Refinement เป็นขั้นตอนสุดท้ายที่มีความสำคัญไม่น้อยไปกว่าขั้นตอนอื่นๆ ในกระบวนการ Generation โดยเป็นการวิเคราะห์คุณภาพและประสิทธิภาพของคำตอบที่สร้างโดย LLM อย่างเป็นระบบ กระบวนการประเมินผลนี้อาศัยทั้งตัวชี้วัดการประเมินผลแบบอัตโนมัติ เช่น BLEU, ROUGE, และ METEOR ซึ่งเป็นเครื่องมือมาตรฐานในการวัดคุณภาพของข้อความที่สร้างขึ้นโดยเปรียบเทียบกับข้อความอ้างอิง รวมถึงการประเมินโดยมนุษย์ซึ่งให้มุมมองเชิงคุณภาพที่ละเอียดและลึกซึ้งกว่า การประเมินผลมุ่งเน้นไปที่การวิเคราะห์คุณลักษณะสำคัญ 4 ประการที่ส่งผลต่อคุณภาพของคำตอบ ประการแรกคือ Fluency ซึ่งแสดงถึงคุณภาพและความต่อเนื่องของการเรียบเรียงข้อความให้มีความเป็นธรรมชาติและอ่านเข้าใจง่าย ประการที่สองคือ Relevance ซึ่งสะท้อนระดับความสอดคล้องของคำตอบกับคำถามและความต้องการของผู้ใช้ ประการที่สามคือ Accuracy ซึ่งบ่งชี้ความน่าเชื่อถือและความสอดคล้องกับข้อเท็จจริงของข้อมูลที่นำเสนอในคำตอบ และประการสุดท้ายคือ Specificity ซึ่งแสดงถึงระดับความละเอียดและความแม่นยำของคำตอบในการตอบสนองความต้องการเฉพาะของผู้ใช้

ภายหลังจากการประเมินผลเบื้องต้น ระบบจะดำเนินการประมวลผล เพื่อปรับปรุงคุณภาพของคำตอบให้มีความสมบูรณ์ยิ่งขึ้น กระบวนการนี้ประกอบด้วยขั้นตอนสำคัญหลายประการ ขั้นตอนแรกคือ Filtering ซึ่งมุ่งเน้นไปที่การกำจัดข้อมูลที่ไม่เกี่ยวข้องหรือมีความซ้ำซ้อนออกจากคำตอบ ขั้นตอนที่สองคือ Re-ranking ซึ่งเป็นการจัดเรียงเนื้อหาภายในคำตอบให้มีลำดับตามระดับความสำคัญหรือความเกี่ยวข้องกับคำถาม และขั้นตอนสุดท้ายคือ Formatting ซึ่งเป็นการนำเสนอข้อมูลในโครงสร้างที่เหมาะสมและเข้าใจง่าย เช่น การจัดเรียงเป็นรายการหัวข้อย่อยหรือตาราง เพื่อให้ผู้ใช้สามารถเข้าถึงและเข้าใจข้อมูลได้อย่างมีประสิทธิภาพ ตัวอย่างที่ชัดเจนของกระบวนการทั้งหมดสามารถเห็นได้จากการตอบคำถามเกี่ยวกับความก้าวหน้าในเทคโนโลยีพลังงานหมุนเวียน ซึ่งเริ่มต้นจากการรวบรวมข้อมูลจากแหล่งที่เชื่อถือได้เกี่ยวกับนวัตกรรมและพัฒนาการล่าสุดในด้านพลังงานแสงอาทิตย์ พลังงานลม และเทคโนโลยีการกักเก็บพลังงาน ข้อมูลเหล่านี้จะผ่านการสังเคราะห์และประเมินคุณภาพอย่างละเอียดเพื่อให้แน่ใจว่ามีความถูกต้องและทันสมัย จากนั้นจึงนำมาสร้างเป็นคำตอบที่มีการจัดโครงสร้างอย่างเป็นระบบ ประกอบด้วยภาพรวมของแนวโน้มปัจจุบัน ตามด้วยรายละเอียดเฉพาะของแต่ละเทคโนโลยี และปิดท้ายด้วยการคาดการณ์เกี่ยวกับทิศทางในอนาคต ผลลัพธ์สุดท้ายที่ได้จากกระบวนการทั้งหมดคือคำตอบที่มีความกระชับ ถูกต้อง และตรงประเด็นกับคำถามของผู้ใช้

กระบวนการประเมินและปรับปรุงนี้จึงเป็นองค์ประกอบสำคัญที่ช่วยยกระดับประสิทธิภาพของระบบสร้างคำตอบอัตโนมัติให้สามารถตอบสนองความต้องการของผู้ใช้ได้อย่างมีประสิทธิภาพและน่าเชื่อถือ การประเมินผลอย่างต่อเนื่องและการปรับปรุงกระบวนการทำงานตามผลการประเมินเป็นวงจรที่มีความสำคัญในการพัฒนาระบบ RAG ให้มีประสิทธิภาพสูงขึ้นอย่างต่อเนื่อง ซึ่งในที่สุดแล้วจะส่งผลให้ผู้ใช้ได้รับประสบการณ์และคุณค่าที่ดียิ่งขึ้นจากระบบดังกล่าว โดยสรุป กระบวนการ Generation ในระบบ RAG เป็นขั้นตอนสุดท้ายที่มีความซับซ้อนและละเอียดอ่อน ประกอบด้วยการประมวลผลบริบทอินพุตที่ได้รับการเสริม การสร้างคำตอบโดยอาศัยความรู้และข้อมูลบริบท การรับรองความเกี่ยวข้องและความถูกต้อง รวมถึงการประเมินผลและการปรับปรุงคุณภาพ กระบวนการเหล่านี้ทำงานร่วมกันอย่างประสานสอดคล้องเพื่อสร้างคำตอบที่มีคุณภาพสูง ตอบสนองความต้องการของผู้ใช้ได้อย่างตรงประเด็นและมีความน่าเชื่อถือ อันเป็นเป้าหมายสูงสุดของระบบ RAG โดยรวม

จากการทบทวนกระบวนการทั้ง 4 ขั้นตอนของ RAG (20) จะเห็นได้ว่าแต่ละส่วนมีความเชื่อมโยงและเกื้อหนุนซึ่งกันและกันอย่างเป็นระบบ โดยเริ่มจากการจัดทำดัชนีข้อมูลที่เป็นรากฐานสำคัญต่อด้วย Retrieval ข้อมูลที่เกี่ยวข้องอย่างแม่นยำ สู่การเสริมข้อมูลเข้ากับบริบทการสอบถามของผู้ใช้ และท้ายที่สุดคือการสร้างคำตอบที่มีคุณภาพสูงจากข้อมูลทั้งหมดที่มี ความก้าวหน้าในด้านเทคโนโลยี RAG จึงไม่เพียงแต่เป็นการพัฒนานวัตกรรมทาง AI เท่านั้น แต่ยังเป็นการยกระดับการปฏิสัมพันธ์ระหว่างมนุษย์และคอมพิวเตอร์ไปสู่มิติใหม่ที่มีความเข้าใจและตอบสนองต่อความต้องการได้อย่างลึกซึ้งมากขึ้น การพัฒนาและวิจัยเพิ่มเติมในอนาคตจะยิ่งเสริมความแข็งแกร่งให้กับเทคโนโลยีนี้ และเปิดโอกาสให้เกิดการประยุกต์ใช้ในวงกว้างที่ส่งผลกระทบเชิงบวกต่อสังคมในภาพรวม

4. Vector Database

Vector Database เป็นระบบฐานข้อมูลที่ได้รับการออกแบบขึ้นโดยเฉพาะสำหรับการจัดเก็บ บริหารจัดการ และ Retrieval ข้อมูลในรูปแบบเวกเตอร์ ซึ่งเป็นตัวแทนเชิงตัวเลขหลายมิติที่เกิดจากการแปลงข้อมูลประเภทต่างๆ เช่น ข้อความ รูปภาพ หรือเสียง ผ่าน Deep Learning Models หรือโมเดล AI อื่นๆ คุณลักษณะอันโดดเด่นที่สุดของ Vector Database ประการหนึ่งคือ Approximate Nearest Neighbor Search ANN ซึ่งทำให้ Retrieval เวกเตอร์ที่มีความคล้ายคลึงกันในชุดข้อมูลขนาดมหึมา เช่น ข้อมูลระดับล้านหรือพันล้านรายการ โดยจะสามารถดำเนินการได้อย่างมีประสิทธิภาพสูงและใช้เวลาประมวลผลน้อย ทั้งนี้ ยังช่วยบรรเทาข้อจำกัดด้านทรัพยากรและระยะเวลาในการประมวลผลข้อมูลที่มีความซับซ้อนและมีปริมาณมหาศาลได้อย่างมีนัยสำคัญ

ในบริบทของ RAG ซึ่งเป็นแนวทางในการสร้างข้อความที่ผสาน Retrieval และ Generation เข้าด้วยกัน โดยอาศัยแหล่งข้อมูลภายนอกเป็นฐานอ้างอิงเพื่อยกระดับความถูกต้องและความสอดคล้องของผลลัพธ์ที่ได้จากโมเดลการสร้างข้อความ Vector Database (21) ได้ทำหน้าที่เป็นกลไกหลักใน Retrieval Engine อย่างมีประสิทธิภาพ กล่าวคือ ข้อมูลปฐมภูมิประเภทต่างๆ จะได้รับการแปลงสภาพให้อยู่ในรูปแบบเวกเตอร์ผ่านกระบวนการ Encoding ของโมเดลการเรียนรู้ ก่อนที่จะถูกบันทึกลงใน Vector Database ส่งผลให้ข้อมูลสามารถถูกสืบค้นและเรียกใช้ได้อย่างรวดเร็วและแม่นยำ เมื่อระบบ RAG ได้รับคำถามหรือคำสั่ง โมเดลจะดำเนินการสร้างเวกเตอร์ของคำถามดังกล่าว แล้วนำไปเปรียบเทียบกับเวกเตอร์ที่จัดเก็บไว้ในฐานข้อมูล โดยอาศัยหลักการของการค้นหาแบบใกล้เคียงที่สุด ทำให้ได้ผลลัพธ์เป็นข้อมูลที่มีความเกี่ยวข้องและคล้ายคลึงสูงสุด ข้อมูลที่ได้รับ Retrieval จะถูกนำไปใช้เป็นฐานความรู้สำหรับโมเดลสร้างข้อความ เช่น GPT (22) เพื่อสนับสนุนให้การสร้างคำตอบมีความถูกต้อง มีบริบทที่เหมาะสม และสอดคล้องกับคำถามหรือคำสั่งที่ได้รับอย่างมีประสิทธิภาพ

โดย Vector Database จำเป็นจะต้องประกอบด้วยคุณสมบัติที่สำคัญหลายประการเพื่อเพิ่มประสิทธิภาพการทำงานของระบบ RAG อย่างสมบูรณ์ ได้แก่ Search Efficiency ที่สามารถประมวลผลข้อมูลเชิงความหมายที่ซับซ้อนได้อย่างรวดเร็วและแม่นยำแม้ในสภาวะที่มีทรัพยากรจำกัด ความสามารถในการปรับขยาย รองรับการจัดเก็บข้อมูลปริมาณมหาศาลในระดับพันล้านรายการและสามารถเพิ่มเติมข้อมูลใหม่ได้อย่างต่อเนื่องโดยไม่ส่งผลกระทบต่อประสิทธิภาพโดยรวมของระบบ การรองรับข้อมูลหลากหลายประเภท ที่สามารถจัดการกับข้อมูลหลากหลายรูปแบบโดยการแปลงให้อยู่ในรูปแบบเวกเตอร์ที่เหมาะสม และการปรับให้เหมาะสมกับงานเฉพาะด้าน ที่สามารถปรับแต่งพารามิเตอร์และโครงสร้างให้สอดคล้องกับบริบทการใช้งานเฉพาะทาง นอกจากนี้ ระบบยังควรมีความสามารถในการบูรณาการที่ไร้รอยต่อ เพื่อเชื่อมโยงกับองค์ประกอบอื่นในระบบ RAG ได้อย่างราบรื่น มี Retrieval Precision ที่สามารถจัดลำดับและคัดกรองผลลัพธ์ตามระดับความเกี่ยวข้องกับคำถาม มีการจัดการด้าน Data Security ที่รองรับการ Encoding และการควบคุมการเข้าถึงข้อมูลอย่างละเอียด ตลอดจนความสามารถในการติดตามแหล่งที่มาของข้อมูล ที่ช่วยให้ระบบสามารถอ้างอิงแหล่งที่มาของข้อมูลที่ใช้ในการสร้างคำตอบได้อย่างโปร่งใสและตรวจสอบได้ คุณสมบัติเหล่านี้มีความสำคัญอย่างยิ่งต่อการพัฒนาระบบ RAG ที่มีประสิทธิภาพสูง มีความน่าเชื่อถือ และสามารถตอบสนองความต้องการของผู้ใช้ได้อย่างแม่นยำในยุคที่ข้อมูลมีความซับซ้อนและมีปริมาณเพิ่มขึ้นอย่างรวดเร็วและต่อเนื่อง

ตอนที่ 5 วิจัยที่เกี่ยวข้อง

Least-to-most prompting (23) เป็นกลยุทธ์ใหม่ที่ได้รับการพัฒนาขึ้นเพื่อเพิ่มความสามารถในการใช้เหตุผลที่ซับซ้อนใน LLM โดยมีเป้าหมายเพื่อแก้ไขข้อจำกัดของ chain-of-thought prompting ซึ่งพบว่ามีความยากลำบากในการขยายผลไปสู่ปัญหาที่ซับซ้อนหรือยากกว่าตัวอย่างที่ให้ไว้ใน prompts วิธีการนี้ใช้แนวคิดในการแยกปัญหาที่ซับซ้อนออกเป็นปัญหาย่อยที่ง่ายกว่า แล้วดำเนินการแก้ไขทีละส่วนตามลำดับ ส่งผลให้โมเดลสามารถจัดการกับปัญหาที่ซับซ้อนได้อย่างมีประสิทธิภาพมากขึ้น การทดสอบประสิทธิภาพในงาน last-letter concatenation พบว่า least-to-most prompting สามารถแสดงผลลัพธ์ที่เหนือกว่า chain-of-thought และ standard prompting อย่างมีนัยสำคัญ โดยเฉพาะอย่างยิ่งเมื่อความยาวของข้อมูลนำเข้าเพิ่มขึ้น ตัวอย่างเช่น ในการใช้ 2-shot prompts กับข้อมูลนำเข้า 12 คำ least-to-most prompting ให้ความแม่นยำถึง 74% เทียบกับ 31.8% ของ chain-of-thought ทั้งนี้ แม้การเพิ่มจำนวนตัวอย่างจะช่วยปรับปรุงประสิทธิภาพของทั้งสองวิธี แต่ least-to-most prompting ยังคงรักษาความได้เปรียบในการแก้ปัญหาไว้ได้อย่างชัดเจน นอกจากนี้ การวิเคราะห์ข้อผิดพลาดชี้ให้เห็นว่าปัญหาส่วนใหญ่เกิดจากการเชื่อมต่อตัวอักษรที่ผิดพลาด มากกว่าการระบุตัวอักษรสุดท้ายผิดพลาด สำหรับการทดสอบใน SCAN compositional generalization benchmark ซึ่งเน้นการประเมินความสามารถในการสร้างความเข้าใจแบบผสมผสาน พบว่า least-to-most prompting มีความแม่นยำสูงถึง 99.7% ใน length split ที่ท้าทาย โดยใช้ตัวอย่างเพียง 14 ตัวอย่าง เมื่อเปรียบเทียบกับ chain-of-thought ที่มีความแม่นยำเพียง 16.2% และยังมีประสิทธิภาพดีกว่าโมเดล neural-symbolic แบบเดิมที่ต้องใช้ตัวอย่างฝึกฝนมากกว่า 15,000 ตัวอย่างอย่างมีนัยสำคัญ อย่างไรก็ดี ปัญหาบางส่วนที่เหลืออยู่ ได้แก่ การตีความคำว่า twice/thrice และ after ซึ่งยังคงท้าทายการทำงานของโมเดลในบางกรณี การประเมินความสามารถในด้านการให้เหตุผลทางคณิตศาสตร์ผ่านชุดข้อมูล GSM8K และ DROP ชี้ให้เห็นว่า least-to-most prompting มีประสิทธิภาพสูงกว่า chain-of-thought ในหลายด้าน แม้ว่าผลการดำเนินการโดยรวมใน GSM8K จะสูงกว่าเพียงเล็กน้อย 62.39% เทียบกับ 60.87% แต่ในปัญหาที่ซับซ้อนที่ต้องใช้ขั้นตอนการให้เหตุผลตั้งแต่ 5 ขั้นขึ้นไป least-to-most prompting ให้ผลลัพธ์ที่ดีกว่าอย่างมีนัยสำคัญ 45.23% เทียบกับ 39.07% นอกจากนี้ บนชุดข้อมูล DROP least-to-most prompting ยังสามารถแสดงประสิทธิภาพที่เหนือกว่า chain-of-thought ได้อย่างชัดเจนในทั้งโจทย์ที่เกี่ยวกับ football และ non-football ซึ่งเป็นการแสดงศักยภาพในการปรับใช้กับปัญหาหลากหลายประเภท อย่างไรก็ตาม การประเมินวิธีการนี้ยังพบข้อจำกัดที่สำคัญ ได้แก่ การที่ decomposition prompts ไม่สามารถขยายผลข้ามโดเมนได้อย่างมีประสิทธิภาพ และความยากลำบากในการแยกส่วนปัญหาที่ซับซ้อนมากขึ้นภายในโดเมนเดียวกัน เช่น คณิตศาสตร์ ทั้งนี้ ข้อจำกัดเหล่านี้สะท้อนให้เห็นว่าวิธีการดังกล่าวยังต้องการการ

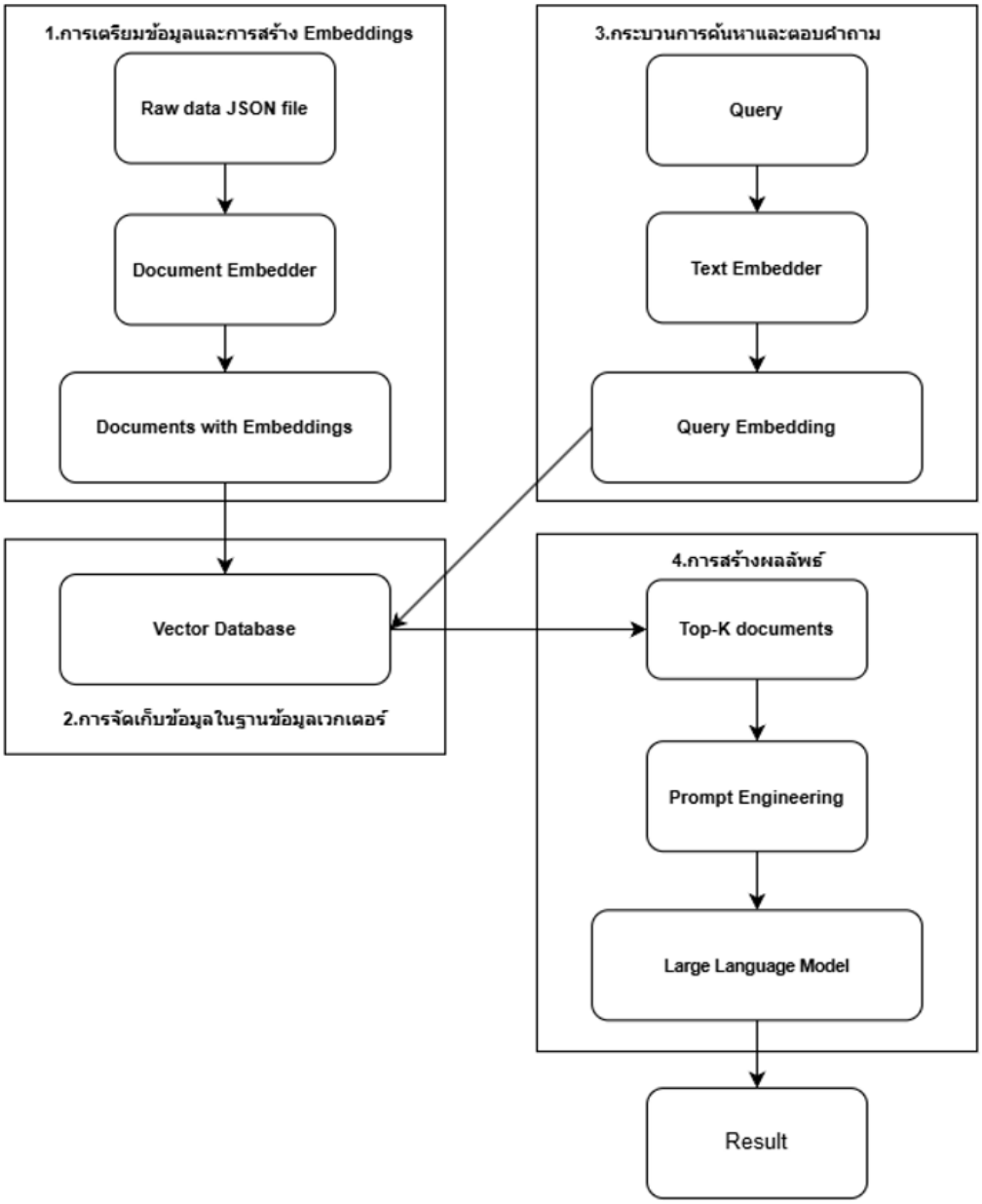
พัฒนาต่อไป แม้ least-to-most prompting จะมีประสิทธิภาพที่น่าพึงพอใจและสามารถเพิ่มศักยภาพของโมเดลภาษาได้อย่างมีนัยสำคัญในงาน

นำเสนอการศึกษาที่ครอบคลุมเกี่ยวกับเทคนิคการพัฒนา prompt engineering ซึ่งถือเป็นเครื่องมือสำคัญในการเพิ่มประสิทธิภาพของโมเดลภาษาแบบขนาดใหญ่ LLM และโมเดลที่ผสานการประมวลผลภาษากับภาพ Vision-Language Models - VLMs (24) โดยเน้นการใช้คำสั่งเฉพาะเพื่อปรับแต่งผลลัพธ์โดยไม่ต้องเปลี่ยนแปลงพารามิเตอร์ภายในของโมเดล ความสำคัญของศาสตร์นี้อยู่ที่ความสามารถในการประยุกต์ใช้ในหลากหลายงานและโดเมนผ่านเทคนิคที่ครอบคลุมตั้งแต่แนวทางพื้นฐานจนถึงวิธีการที่ซับซ้อน บทความนี้มุ่งเน้นการจัดระเบียบความรู้เกี่ยวกับ prompt engineering อย่างเป็นระบบ พร้อมทั้งสรุปความก้าวหน้าล่าสุดในสาขานี้ โดยมีการจำแนกหมวดหมู่ตามลักษณะการใช้งาน การสำรวจเชิงลึกของเทคนิคทั้งหมด 29 รูปแบบครอบคลุมตั้งแต่การสร้างงานใหม่โดยไม่ต้องฝึกฝนข้อมูล Zero-Shot Prompting, Few-Shot Prompting การเสริมศักยภาพด้านการให้เหตุผลและตรรกะ เช่น Chain-of-Thought CoT และ Tree-of-Thoughts ToT ไปจนถึงการลดปัญหาการสร้างข้อมูลที่ไม่ถูกต้อง hallucination ผ่าน Retrieval Augmented Generation RAG และ Chain-of-Verification CoVe รวมถึงเทคนิคที่ช่วยเพิ่มประสิทธิภาพการปรับแต่งและการสร้างผลลัพธ์ที่เหมาะสม เช่น Automatic Prompt Engineer APE และ Optimization by Prompting OPRO นอกจากนี้ ยังมีการประยุกต์ใช้เทคนิคในด้านต่าง ๆ เช่น การสร้างเหตุผลบนฐานข้อมูล Knowledge-Based Reasoning, การเพิ่มความสอดคล้องและความสมเหตุสมผลของคำตอบ Consistency and Coherence, การจัดการด้านอารมณ์และโทนคำตอบ Emotion Prompting, และการสร้างโค้ดที่มีประสิทธิภาพ Code Generation and Execution เช่น Scratchpad Prompting และ Program of Thoughts PoT การวิจัยยังครอบคลุมถึงแนวทางที่เน้นการเข้าใจเจตนาของผู้ใช้ Understanding User Intent และการพัฒนากลไกเพื่อการสะท้อนและการคิดเชิงเมตา Metacognition and Self-Reflection เช่น Take a Step Back Prompting ซึ่งช่วยเพิ่มศักยภาพของโมเดลในการสรุปและวิเคราะห์แนวคิดเชิงนามธรรม สรุปได้ว่าการสำรวจเชิงระบบของ prompt engineering เป็นพื้นฐานสำคัญที่ช่วยผลักดันการพัฒนา AI ในปัจจุบัน แม้จะมีความก้าวหน้า แต่ยังคงมีอุปสรรคที่ต้องแก้ไข เช่น อคติของข้อมูล ความไม่ถูกต้อง และข้อจำกัดในการตีความ อย่างไรก็ตาม อนาคตของศาสตร์นี้มีศักยภาพที่น่าสนใจ โดยเฉพาะการพัฒนาแนวคิด meta-learning และสถาปัตยกรรมแบบผสมผสาน ซึ่งอาจช่วยยกระดับความสามารถของ AI อย่างมีนัยสำคัญ อีกทั้งบทความยังเน้นย้ำถึงความจำเป็นในการพิจารณาด้านจริยธรรม เพื่อให้การพัฒนาและการใช้งานเทคโนโลยีดังกล่าวเกิดประโยชน์และความรับผิดชอบต่อสังคมโดย

บทที่ 3
วิธีดำเนินการวิจัย

การศึกษาวิจัยนี้มุ่งพัฒนาระบบการจัดการองค์ความรู้สำหรับมหาวิทยาลัยที่มีศักยภาพในการรองรับกลุ่มผู้ใช้งานทั้งภายในและภายนอกมหาวิทยาลัย โดยประยุกต์ใช้ RAG ซึ่งจะประกอบด้วยองค์ประกอบสำคัญสี่ส่วนตามที่แสดงในภาพประกอบ 5 ได้แก่ ขั้นตอนที่ 1 กระบวนการเตรียมข้อมูลและการสร้าง Embeddings ที่เริ่มจากแหล่งข้อมูลดิบในรูปแบบ JSON ซึ่งจะถูกแปลงเป็นเวกเตอร์เชิงความหมายด้วยตัวฝังตัวเอกสารแบบแปลงประโยค SentenceTransformersDocumentEmbedder บนโมเดล "paraphrase-multilingual-mpnet-base-v2" ที่รองรับภาษาไทย ต่อมาขั้นตอนที่ 2 การจัดเก็บข้อมูลใน Vector Database ผ่าน HybridDocumentStore ที่ผสานเทคนิค Semantic Search และ Lexical Search ด้วย Hybrid Search และขั้นตอนที่ 3 Query Processing ที่วิเคราะห์เจตนาและจำแนกประเภทของคำถามผ่าน AdvancedQueryProcessor ก่อนแปลงเป็นเวกเตอร์คำค้นด้วย Text Embedding เพื่อค้นหาข้อมูลที่เกี่ยวข้อง และขั้นตอนที่ 4 การสร้างผลลัพธ์โดยการคัดเลือกเอกสารที่มีความคล้ายคลึงสูงสุด Top-K Documents ด้วยวิธี Cosine Similarity ตามด้วยการประยุกต์ใช้เทคนิค Prompt Engineering (25) ผ่าน ResponseGenerator ที่ส่งข้อมูลไปยัง LLM เพื่อสังเคราะห์ข้อมูลและสร้างคำตอบที่มีความถูกต้อง สมบูรณ์ และสอดคล้องกับบริบทของคำถาม ซึ่งกระบวนการทั้งหมดนี้ช่วยเพิ่มประสิทธิภาพการเข้าถึงและจัดการองค์ความรู้อย่างเป็นระบบ ลดปัญหา Hallucination และส่งเสริมคุณภาพการให้บริการข้อมูลที่ตอบสนองต่อความต้องการของผู้ใช้งาน โดยการวิจัยครั้งนี้ประกอบด้วยขั้นตอนการดำเนินการ 5 ระยะ ได้แก่ 1 กรอบแนวคิดและภาพรวมของการทดลองพัฒนาระบบโปรแกรมสนทนาอัตโนมัติแบบปัญญาประดิษฐ์ AI Chatbot 2 การรวบรวมและจัดการข้อมูล จากแหล่งข้อมูลทางการของมหาวิทยาลัย 3 การดึงข้อมูลและการเตรียมข้อมูล Data Preparation 4 การพัฒนาระบบโต้ตอบอัตโนมัติ ที่รองรับการสื่อสารข้อความ โดยใช้ LLM 5 การประเมินผล Evaluation

1. **กรอบแนวคิดและภาพรวมของการทดลองพัฒนาระบบ AI Chatbot**



ภาพประกอบ 5 กรอบแนวคิดและภาพรวมของการทดลองพัฒนาระบบ AI Chatbot

การพัฒนาระบบตอบคำถามอัตโนมัติด้วยเทคนิค RAG มีความสำคัญอย่างยิ่งในการยกระดับการให้บริการข้อมูลสถาบันการศึกษา โดยระบบที่นำเสนอนี้ประกอบด้วยกระบวนการหลัก 4 ส่วนตามที่แสดงในรูปภาพ ได้แก่ 1). การเตรียมข้อมูลและการสร้าง Embeddings 2). การจัดเก็บข้อมูลใน Vector Database 3). กระบวนการค้นหาและตอบคำถาม และ 4). การสร้างผลลัพธ์ ซึ่งแต่ละส่วนมีบทบาทสำคัญดังต่อไปนี้

1.1 การเตรียมข้อมูลและการสร้าง Embeddings

ขั้นตอนแรกคือ Raw data JSON file จากภาพประกอบ 5 สังเกตได้ว่าขั้นตอน Raw data JSON file โดยจะเป็นจุดเริ่มต้นในกระบวนการที่ 1 คือการเตรียมข้อมูลและการสร้าง Embeddings กระบวนการพัฒนาระบบ RAG เริ่มต้นจากการประมวลผลข้อมูลดิบในรูปแบบ JSON ซึ่งจะรวบรวมสารสนเทศจากแหล่งข้อมูลต่างๆ ของสถาบันการศึกษา โดยผ่านการประมวลผลด้วย CalendarDataProcessor ที่ทำหน้าที่แยกประเภทข้อมูลและจัดโครงสร้าง เช่น academic_calendar, program_details, contact_details, course_structure, program_study_plan และ fees ข้อมูลเหล่านี้จะถูกแปลงเป็นวัตถุที่มีโครงสร้าง Structured Objects ผ่านกลไก @dataclass เช่น CalendarEvent, ContactDetail, CourseStructure และ TuitionFee ซึ่งการจัดรูปแบบข้อมูลอย่างเป็นระบบนี้เป็นพื้นฐานสำคัญที่ทำให้ขั้นตอนต่อไปในกระบวนการมีประสิทธิภาพและความแม่นยำสูงและขั้นตอนถัดมาคือ **Document Embedde** จากภาพประกอบ 5 จะเห็นว่าขั้นตอน Document Embedder เป็นกระบวนการถัดจาก Raw data JSON file ในส่วนของการเตรียมข้อมูลและการสร้าง Embeddings กระบวนการสร้างเวกเตอร์ความหมายของเอกสาร เป็นองค์ประกอบสำคัญในระบบ RAG ที่ทำหน้าที่แปลงข้อมูลดิบจากไฟล์ JSON โดยจะให้อยู่ในรูปแบบของเวกเตอร์เชิงความหมาย ซึ่งจะใช้โมเดล SentenceTransformersDocumentEmbedder ที่มีพื้นฐานจากแบบจำลอง "paraphrase-multilingual-mpnet-base-v2" ซึ่งเป็นโมเดลที่รองรับภาษาไทยและภาษาอื่นๆ ทำให้สามารถเข้าใจความหมายเชิงบริบทที่ซับซ้อนของข้อความได้ดี โดยระบบมีการนำเทคนิค Hybrid Retrieval มาประยุกต์ใช้ร่วมกับ Cache Management เพื่อเพิ่มประสิทธิภาพการประมวลผลและลดเวลาในการคำนวณซ้ำ นอกจากนี้ยังมี Advanced Query Processing ที่สามารถปรับแต่ง weight ระหว่าง Semantic Search กับ Lexical Search ได้อย่างเหมาะสมตามลักษณะของข้อมูล และขั้นตอนถัดมาคือ **Documents with Embeddings** จากภาพประกอบ 5 จะเห็นได้ว่า Documents with Embeddings เป็นผลลัพธ์สุดท้ายของกระบวนการที่ 1 คือการเตรียมข้อมูลและการสร้าง Embeddings กระบวนการสร้าง Embeddings ในระบบ RAG ดำเนินการผ่านการประมวลผลข้อความให้เป็นเวกเตอร์เชิงความหมายด้วย SentenceTransformersDocumentEmbedder โดยจะมีการใช้โมเดลภาษาแบบ paraphrase-multilingual-mpnet-base-v2 ซึ่งรองรับการประมวลผลภาษาไทย ผลลัพธ์คือชุดเวกเตอร์ความ

หนาแน่นสูงที่มีมิติประมาณ 384-1536 ขึ้นอยู่กับสถาปัตยกรรมของโมเดลที่เลือกใช้ ระบบได้ออกแบบให้มีการจัดเก็บ Embeddings ไว้ในรูปแบบ cache เพื่อลดการประมวลผลซ้ำและเพิ่มประสิทธิภาพการทำงาน เมื่อได้เอกสารพร้อม Embeddings แล้ว จะเป็นข้อมูลที่พร้อมสำหรับการจัดเก็บใน Vector Database โดยในขั้นตอนต่อไป ซึ่งเป็นองค์ประกอบสำคัญที่ทำให้ระบบสามารถค้นหาข้อมูลเชิงความหมายได้อย่างมีประสิทธิภาพ

1.2 **การจัดเก็บข้อมูลใน** Vector Database

ขั้นตอนแรกคือ Vector Database จากภาพประกอบ 5 จะเห็นว่า Vector Database เป็นองค์ประกอบหลักในกระบวนการที่ 2 คือการจัดเก็บข้อมูลใน Vector เป็นโครงสร้างพื้นฐานที่ออกแบบเฉพาะสำหรับการจัดเก็บและสืบค้นข้อมูลในรูปแบบเวกเตอร์ ซึ่งเป็นองค์ประกอบสำคัญในระบบ RAG ที่มีประสิทธิภาพ โดยในสถาปัตยกรรมที่พัฒนาขึ้น จะมีการประยุกต์ใช้ HybridDocumentStore ที่ผสาน semantic search ผ่าน InMemoryEmbeddingRetriever และ lexical search ผ่าน InMemoryBM25Retriever โดยมี weight semantic ระหว่างทั้งสองวิธี ระบบนี้ใช้ SentenceTransformersDocumentEmbedder ในการแปลงข้อความเป็นเวกเตอร์ที่เก็บความหมายเชิงอรรถศาสตร์ ซึ่งสามารถประมวลผลการค้นหาด้วยหลักการความคล้ายคลึงแบบ cosine similarity

Vector Database นี้ยังจัดเก็บข้อมูลของ ข้อมูลปฏิทินการศึกษา รายละเอียดหลักสูตร ข้อมูลการติดต่อ โครงสร้างรายวิชา แผนการเรียนและการศึกษา ค่าธรรมเนียมการศึกษา และทุนการศึกษาของสาขา นอกจากนี้ ยังมีกลไก re-ranking ที่ช่วยปรับปรุงความแม่นยำใน Retrieval ข้อมูล ทำให้ระบบสามารถรองรับการสืบค้นข้อมูลที่ซับซ้อนได้อย่างมีประสิทธิภาพ

1.3 **กระบวนการค้นหาและตอบคำถาม**

ขั้นตอนแรกคือ Query จากภาพประกอบ 5 ด้านบน จะเห็นว่า Query เป็นจุดเริ่มต้นในกระบวนการที่ 3 คือกระบวนการค้นหาและตอบคำถาม กระบวนการค้นหาและตอบคำถามของระบบ RAG โดยจะเริ่มต้นจากการรับ Query จากผู้ใช้งาน ซึ่งจะถูกประมวลผลโดยคลาส AdvancedQueryProcessor ที่ทำหน้าที่วิเคราะห์เจตนาและจำแนกประเภทของคำถามผ่านการตรวจจับคำสำคัญพร้อมทั้งปรับแต่งคำค้นด้วยกระบวนการ Normalization เพื่อให้ตรงกับรูปแบบมาตรฐาน เช่น การแปลงรูปแบบปีการศึกษาและภาคเรียนให้เป็นรูปแบบเดียวกัน เช่น "ปี 1" เป็น "ปีที่ 1"

นอกจากนี้ ระบบยังมีการพิจารณาประวัติการสนทนา เพื่อสร้างคำค้นที่มีบริบท ซึ่งช่วยให้ระบบเข้าใจคำถามที่อาจมีการอ้างอิงถึงบทสนทนาก่อนหน้า และสามารถจัดการกับคำถามที่มีความกำกวมหรือไม่ชัดเจนได้อย่างมีประสิทธิภาพ ทั้งนี้ ระบบยังมีการปรับเปลี่ยน Weight Semantic และพยายามค้นหาซ้ำหากไม่พบผลลัพธ์ที่เกี่ยวข้อง เพื่อเพิ่มโอกาสในการค้นพบข้อมูลที่ตรงกับความต้องการของผู้ใช้และขันตอนถัดมาคือ Text Embedder จากภาพประกอบ 5 จะเห็นว่า Text Embedder เป็นขั้นตอนถัดจาก Query ในกระบวนการค้นหาและตอบคำถาม กระบวนการแปลงคำถาม Query เป็นเวกเตอร์ในขั้นตอน Text Embedder เป็นกระบวนการสำคัญที่ใช้โมเดล SentenceTransformersDocumentEmbedder ด้วยโมเดล "paraphrase-multilingual-mpnet-base-v2" ซึ่งเป็นโมเดลที่รองรับภาษาไทยและภาษาอื่นๆ โดยการแปลงนี้จำเป็นต้องใช้โมเดลเดียวกันกับที่ใช้ในการแปลงเอกสารเพื่อรักษาความสอดคล้องในปริภูมิเวกเตอร์ ซึ่งทำให้การคำนวณความคล้ายคลึงระหว่างคำถามและเอกสารมีความแม่นยำ

ระบบมีการจัดเก็บ embedding ในรูปแบบ cache เพื่อเพิ่มประสิทธิภาพผ่านคลาส CacheManager ทำให้ไม่ต้องคำนวณใหม่สำหรับข้อความเดิม ค่า embedding ที่ได้จะถูกนำไปใช้ในกระบวนการ Hybrid Search ซึ่งจะรวมเทคนิค Semantic Search ด้วยเทคนิค embedding_retriever และ Lexical Search ด้วย BM25_retriever เพื่อให้ได้ผลลัพธ์ที่สมดุลระหว่างความเข้าใจเชิงความหมายและการจับคู่คำสำคัญและขันตอนถัดมาคือการใช้งาน Query Embedding จากภาพประกอบ 5 จะเห็นว่า Query Embedding เป็นผลลัพธ์จากขั้นตอน Text Embedder ในกระบวนการค้นหาและตอบคำถาม Query Embedding เป็นขั้นตอนสำคัญในระบบ RAG ที่แปลงคำค้นของผู้ใช้ให้เป็นเวกเตอร์เชิงความหมายในพื้นที่หลายมิติ โดยในระบบที่พัฒนาขึ้น ได้ใช้โมเดล SentenceTransformers ร่วมกับ AdvancedQueryProcessor ที่ทำหน้าที่วิเคราะห์และ normalization คำค้น ก่อนส่งไปประมวลผลด้วย OpenAI API (26) เพื่อจำแนกประเภทคำถามและดึงคำสำคัญ

เมื่อได้เวกเตอร์คำค้นแล้ว ระบบจะใช้ค่านี้ในการค้นหาเอกสารที่เกี่ยวข้องด้วยวิธี hybrid search ที่ผสมผสาน semantic search กับ Lexical Search แบบ BM25 ซึ่งวิธีการนี้ช่วยเพิ่มประสิทธิภาพใน document retrieval ที่เกี่ยวข้อง แม้จะใช้คำที่แตกต่างแต่มีความหมายใกล้เคียงกัน โดยเวกเตอร์ที่ได้จะถูกใช้ในการคำนวณ cosine similarity กับเอกสารใน Vector Database เพื่อค้นหาข้อมูลที่เกี่ยวข้องที่สุด

1.4 การสร้างผลลัพธ์

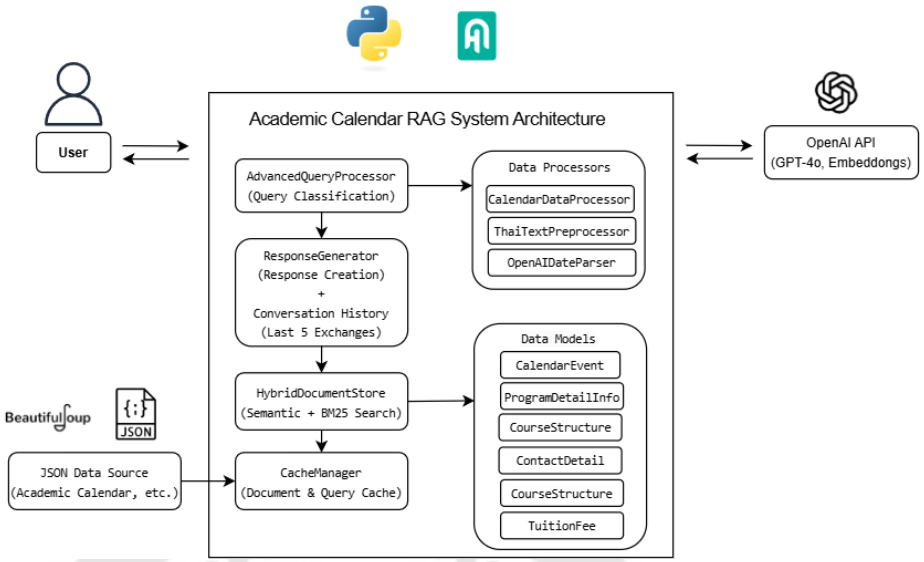
ขั้นตอนแรกคือ Top K documents จากภาพประกอบ 5 จะเห็นว่า Top-K documents เป็นขั้นตอนแรกในกระบวนการที่ 4 คือการสร้างผลลัพธ์ กระบวนการ Top-K documents ในระบบ RAG ที่พัฒนาขึ้นดำเนินการผ่านการผสมผสานระหว่าง Semantic Search และ Lexical Search โดยใช้ Hybrid Search ที่กำหนดค่าน้ำหนัก weight semantic เพื่อควบคุมสัดส่วนระหว่างสองวิธีการ ระบบจะคำนวณค่าความคล้ายคลึงระหว่างเวกเตอร์ของคำถามกับเวกเตอร์ของเอกสารในฐานข้อมูลด้วยวิธี Cosine Similarity และระยะห่างแบบ Euclidean หลังจากนั้นคัดเลือกเอกสารที่มีค่าความคล้ายคลึงสูงสุดตามจำนวน K โดยค่าเริ่มต้นคือ 5

ในกรณีที่ต้องการความแม่นยำสูง ระบบยังมีกระบวนการ Re-ranking ที่ใช้ Cross-encoder เพื่อวิเคราะห์ความสัมพันธ์ระหว่างคำถามและเอกสารอย่างละเอียดอีกครั้ง และในกรณีที่ไม่พบข้อมูลเพียงพอในครั้งแรก ระบบจะทดลองปรับเปลี่ยนค่า weight semantic ไปมาระหว่าง 0.3 และ 0.7 และอาจทดลองค้นหาซ้ำสูงสุดถึง 4 ครั้ง เพื่อเพิ่มโอกาสในการค้นพบเอกสารที่เกี่ยวข้อง เอกสารที่ได้รับการคัดเลือกจากขั้นตอนนี้จะถูกนำไปใช้ในการสร้าง prompt สำหรับการสร้างคำตอบในขั้นตอนต่อไปและขันตอนถัดมาคือ Prompt Engineering จากภาพประกอบ 5 จะเห็นว่า Prompt Engineering เป็นขั้นตอนถัดจาก Top-K documents ในกระบวนการสร้างผลลัพธ์ Prompt Engineering ในระบบ RAG เป็นกระบวนการสำคัญที่ส่งผลต่อคุณภาพของคำตอบและประสิทธิภาพของการให้บริการข้อมูล โดยเป็นขั้นตอนหลังจาก document retrieval ซึ่งในระบบที่พัฒนาขึ้นนี้ ได้ใช้โครงสร้าง Prompt ที่ซับซ้อนและมีประสิทธิภาพผ่านคลาส ResponseGenerator ที่รวมองค์ประกอบสำคัญหลายประการเข้าด้วยกัน ได้แก่: 1). บริบทจากประวัติการสนทนา Conversation History ที่ช่วยให้ระบบเข้าใจคำถามที่อ้างอิงถึงข้อมูลในบทสนทนาก่อนหน้า 2). คำถามปัจจุบันที่ได้รับการวิเคราะห์และจำแนกประเภทของข้อมูลที่ต้องการ 3). เอกสารที่เกี่ยวข้องซึ่งมีการระบุประเภทและรายละเอียดของข้อมูล 4). ชุดคำแนะนำในการตอบที่ครอบคลุมประเด็นสำคัญ ระบบมีการปรับเปลี่ยนค่าถ่วงน้ำหนัก Weight ในการค้นหาและทำซ้ำหลายครั้งเพื่อให้ได้คำตอบที่มีคุณภาพสูงสุด ซึ่งช่วยลดปัญหา Knowledge Neglect และ Hallucination อันเป็นปัจจัยสำคัญต่อความน่าเชื่อถือของระบบให้บริการข้อมูลมหาวิทยาลัยและขันตอนถัดมาคือ LLM จากภาพประกอบ 5 จะเห็นว่า LLM เป็นขั้นตอนถัดจาก Prompt Engineering ในกระบวนการสร้างผลลัพธ์ หลังจากที่ระบบได้สร้าง prompt ที่มีโครงสร้างเหมาะสมแล้ว ข้อมูลจะถูกส่งไปยัง LLM ซึ่งในระบบที่พัฒนาขึ้นนี้ใช้ OpenAI API ในการประมวลผลและสร้างคำตอบ โมเดลภาษานี้จะวิเคราะห์ข้อมูลที่ได้รับจาก prompt ซึ่ง

ประกอบด้วยคำถามของผู้ใช้ เอกสารอ้างอิงที่เกี่ยวข้อง และคำแนะนำในการตอบ เพื่อที่จะสร้างคำตอบที่มีความถูกต้อง ครบถ้วน และเป็นธรรมชาติ

LLM นี้มีความสามารถในการเข้าใจบริบทและการสร้างเนื้อหา ที่สอดคล้องกับข้อมูลที่ได้รับ ซึ่งทำให้สามารถสร้างคำตอบที่มีความเชื่อมโยงกับข้อมูลที่ Retrieval มาได้อย่างแม่นยำ และยังสามารถระบุกรณีที่ไม่พบข้อมูลเพียงพอ หรือข้อมูลมีความไม่ชัดเจนได้อย่างเหมาะสม ซึ่งเป็นคุณสมบัติที่สำคัญในการให้บริการข้อมูลที่น่าเชื่อถือและขันตอนถัดมาคือ Result จากภาพประกอบ 5 จะเห็นว่า Result เป็นผลลัพธ์สุดท้ายของระบบ RAG ที่ได้จากกระบวนการทั้งหมด กระบวนการสร้างผลลัพธ์ในระบบ RAG ดำเนินการโดยการส่งข้อมูลที่ผ่านกระบวนการ Prompt Engineering ไปยัง LLM ผ่านการทำงานของคลาส ResponseGenerator ซึ่งจะประกอบด้วย OpenAIGenerator ที่เชื่อมต่อกับ API ภายนอก และ PromptBuilder ที่กำหนดโครงสร้างคำถาม

ระบบจะทำการประมวลผลทางความหมายของข้อมูลที่ได้รับ Retrieval จาก Vector Database ร่วมกับการวิเคราะห์ประวัติการสนทนา เพื่อสร้างคำตอบที่สอดคล้องกับบริบทของคำถาม ทั้งนี้ ระบบได้ถูกออกแบบให้พิจารณาคำสำคัญเฉพาะสำหรับคำถามที่อ้างอิงถึงบทสนทนาก่อนหน้า มีการจัดการกับคำถามที่ไม่ชัดเจนด้วยการใช้ข้อมูลจากประวัติการสนทนา และสามารถระบุกรณีที่ไม่พบข้อมูลที่เกี่ยวข้องได้อย่างถูกต้อง ก่อนที่จะทำการส่งผลลัพธ์กลับไปยังผู้ใช้งานในรูปแบบที่มีโครงสร้างเป็นภาษาไทยที่สมบูรณ์และเข้าใจง่าย ตอบสนองความต้องการของผู้ใช้ได้อย่างมีประสิทธิภาพ



ภาพประกอบ 6 Academic Calendar RAG System

จากภาพประกอบ 6 สถาปัตยกรรมระบบ Academic Calendar RAG System ที่แสดงในภาพประกอบ 4 นำเสนอโครงสร้างการทำงานแบบบูรณาการของระบบตอบคำถามอัตโนมัติด้วยเทคนิค RAG โดยมีการผสมผสานเทคโนโลยีขั้นสูงหลายประการเพื่อให้บรรลุประสิทธิภาพสูงสุดในการให้บริการข้อมูลแก่ผู้ใช้งาน สถาปัตยกรรมนี้เริ่มต้นจากการปฏิสัมพันธ์ระหว่างผู้ใช้งาน User กับระบบผ่านส่วนติดต่อที่ออกแบบให้ใช้งานได้อย่างสะดวก ซึ่งเมื่อผู้ใช้ส่งคำถามเข้ามา ระบบจะประมวลผลด้วยองค์ประกอบหลักหลายส่วนที่ทำงานประสานกัน

ส่วนแรกของการประมวลผลคือ AdvancedQueryProcessor ที่ทำหน้าที่จำแนกประเภทคำถาม Query Classification โดยวิเคราะห์โครงสร้างและเจตนาของคำถาม เพื่อระบุประเภทข้อมูลที่ผู้ใช้ต้องการ ซึ่งมีความสำคัญอย่างยิ่งในการกำหนดทิศทางการค้นหาข้อมูลในขั้นตอนต่อไป จากนั้นข้อมูลจะถูกส่งต่อไปยัง ResponseGenerator ที่รับผิดชอบในการสร้างคำตอบที่สมบูรณ์ Response Creation โดยพิจารณาประกอบกับประวัติการสนทนาย้อนหลัง Conversation History จำนวน 5 การแลกเปลี่ยนล่าสุด ซึ่งช่วยให้ระบบสามารถเข้าใจบริบทของคำถามปัจจุบันที่อาจมีการอ้างอิงถึงการสนทนาก่อนหน้าได้อย่างแม่นยำ โดยองค์ประกอบสำคัญอีกประการหนึ่งคือ HybridDocumentStore ที่จะใช้เทคนิคการค้นหาแบบผสมผสานระหว่าง Semantic Search และ BM25 Search โดยการค้นหาเชิงความหมายอาศัย vector embeddings ในการวิเคราะห์ความคล้ายคลึงเชิงความหมายระหว่างคำถามและเอกสาร ในขณะที่การค้นหาแบบ BM25 จะช่วยในการจับคู่คำสำคัญโดยตรง การผสมผสานทั้งสองเทคนิคนี้ช่วยเพิ่มประสิทธิภาพใน Retrieval ซึ่งข้อมูลที่เกี่ยวข้องแม้ในกรณีที่ผู้ใช้ใช้คำศัพท์ที่

แตกต่างจากที่ปรากฏในฐานข้อมูล สำหรับการเพิ่มประสิทธิภาพในการประมวลผล ระบบได้ติดตั้ง CacheManager ที่ทำหน้าที่จัดการ cache โดยเก็บผลลัพธ์การคำนวณเวกเตอร์ของเอกสารและคำถามไว้ชั่วคราว ซึ่งจะช่วยลดเวลาในการประมวลผลซ้ำและเพิ่มความเร็วในการตอบสนองต่อคำถามที่มีลักษณะคล้ายคลึงกัน ในส่วนของการประมวลผลข้อมูล ระบบมีกลุ่มของ Data Processors ที่ประกอบด้วย CalendarDataProcessor สำหรับจัดการข้อมูลปฏิทินการศึกษา, ThaiTextPreprocessor สำหรับการประมวลผลข้อความภาษาไทยฉฉพาะ และ OpenAIDateParser สำหรับการวิเคราะห์และแปลงรูปแบบวันที่ให้เป็นมาตรฐานเดียวกัน ระบบยังได้ออกแบบโครงสร้างข้อมูล ที่จะครอบคลุมประเภทข้อมูลทั้งหมดที่สำคัญในบริบทของสถาบันการศึกษา ซึ่งจะประกอบไปด้วย CalendarEvent สำหรับเหตุการณ์ในปฏิทินการศึกษา, ProgramDetailInfo สำหรับรายละเอียดหลักสูตร, CourseStructure สำหรับโครงสร้างรายวิชา, ContactDetail สำหรับข้อมูลการติดต่อ และ TuitionFee สำหรับค่าธรรมเนียมการศึกษา โดยแต่ละโมเดลถูกออกแบบให้สามารถรองรับลักษณะเฉพาะของข้อมูลแต่ละประเภทได้อย่างเหมาะสม ข้อมูลที่ใช้ในระบบนี้มาจากแหล่งข้อมูล JSON ที่ประกอบด้วยข้อมูลปฏิทินการศึกษาและข้อมูลอื่นๆ ที่เกี่ยวข้อง ซึ่งผ่านการประมวลผลด้วย BeautifulSoup เพื่อดึงข้อมูลจากเว็บไซต์และจัดโครงสร้างให้อยู่ในรูปแบบที่เหมาะสมสำหรับการนำเข้าสู่ระบบ นอกจากนี้ ระบบยังเชื่อมต่อกับ OpenAI API GPT-4o, Embeddings เพื่อใช้ในการสร้าง embeddings สำหรับเอกสารและคำถาม รวมถึงการสร้างคำตอบที่เป็นธรรมชาติและตรงตามความต้องการของผู้ใช้ การทำงานของระบบเริ่มต้นเมื่อผู้ใช้ส่งคำถามเข้ามาใน AI Chat Bot คำถามจะถูกประมวลผลโดย AdvancedQueryProcessor เพื่อวิเคราะห์และจำแนกประเภท จากนั้นระบบจะค้นหาข้อมูลที่เกี่ยวข้องใน HybridDocumentStore โดยใช้เทคนิคการค้นหาแบบผสมผสาน เมื่อได้ข้อมูลที่เกี่ยวข้องแล้ว ResponseGenerator จะนำข้อมูลเหล่านั้นไปสร้างคำตอบที่สมบูรณ์โดยพิจารณาประกอบกับประวัติการสนทนา และจะส่งคำตอบกลับไปยังผู้ใช้ในรูปแบบที่เข้าใจง่ายและตรงประเด็น ทั้งนี้ ข้อมูลการสนทนาจะถูกเก็บไว้เพื่อใช้ในการวิเคราะห์คำถามในอนาคต ซึ่งช่วยเพิ่มประสิทธิภาพในการให้บริการอย่างต่อเนื่อง

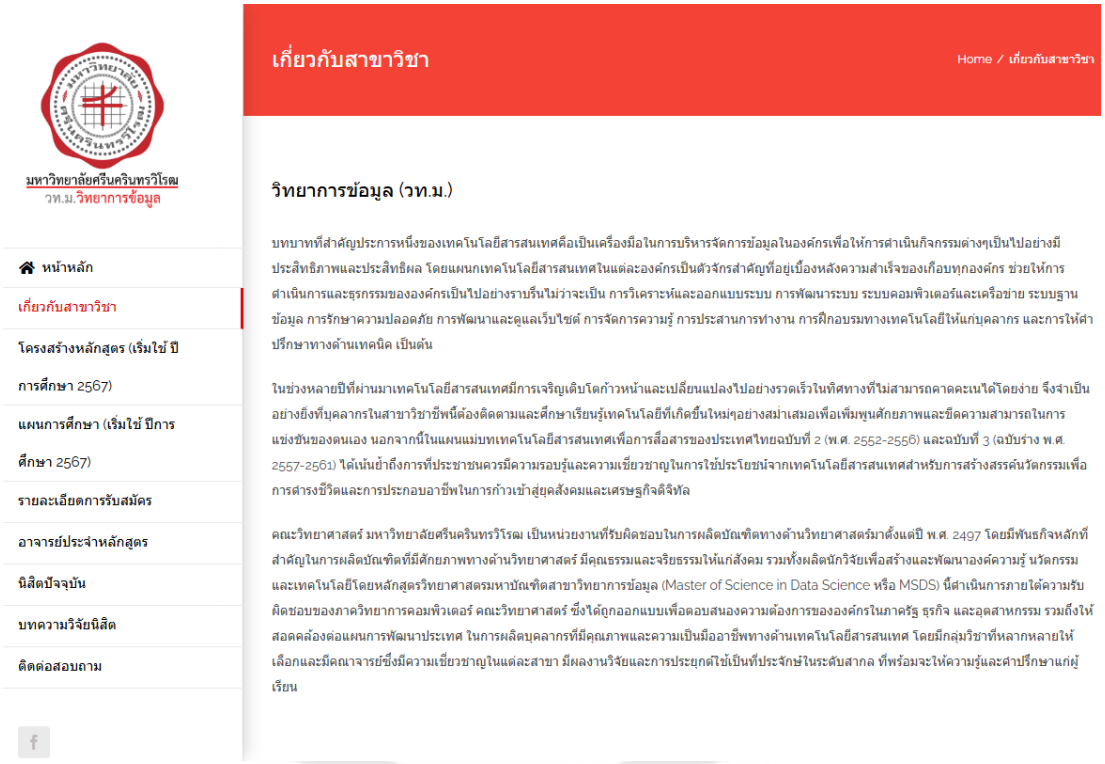
สถาปัตยกรรมนี้ได้รับการออกแบบให้มีความยืดหยุ่นและขยายขอบเขตได้ง่าย โดยสามารถเพิ่มประเภทข้อมูลหรือปรับปรุงองค์ประกอบต่างๆ ได้ตามความต้องการที่เปลี่ยนแปลงไป นอกจากนี้ ยังมีกลไกในการติดตามและประเมินประสิทธิภาพของระบบ เพื่อนำไปสู่การพัฒนาและปรับปรุงอย่างต่อเนื่อง ซึ่งจะเป็นปัจจัยสำคัญในการสร้างระบบให้บริการข้อมูลที่มีประสิทธิภาพและน่าเชื่อถือสำหรับสถาบันการศึกษา

2. การรวบรวมและจัดการข้อมูลจากแหล่งข้อมูลทางการของมหาวิทยาลัย

ในระยะแรกของโครงการนี้ เราจะทำการรวบรวมข้อมูลจากเว็บไซต์ทางการของมหาวิทยาลัย โดยครอบคลุมเนื้อหาต่าง ๆ ที่เกี่ยวข้องกับระดับบัณฑิตศึกษา ซึ่งประกอบด้วยข้อมูลดังต่อไปนี้:

2.1 ข้อมูลเกี่ยวกับสาขาวิชา

รายละเอียดเกี่ยวกับโปรแกรมการศึกษา สาขาวิชาต่าง ๆ ที่เปิดสอน และข้อมูลเกี่ยวกับการเรียนการสอน ต่อไปนี้เป็นตัวอย่างของข้อมูลฝ่ายเกี่ยวกับสาขาวิชา

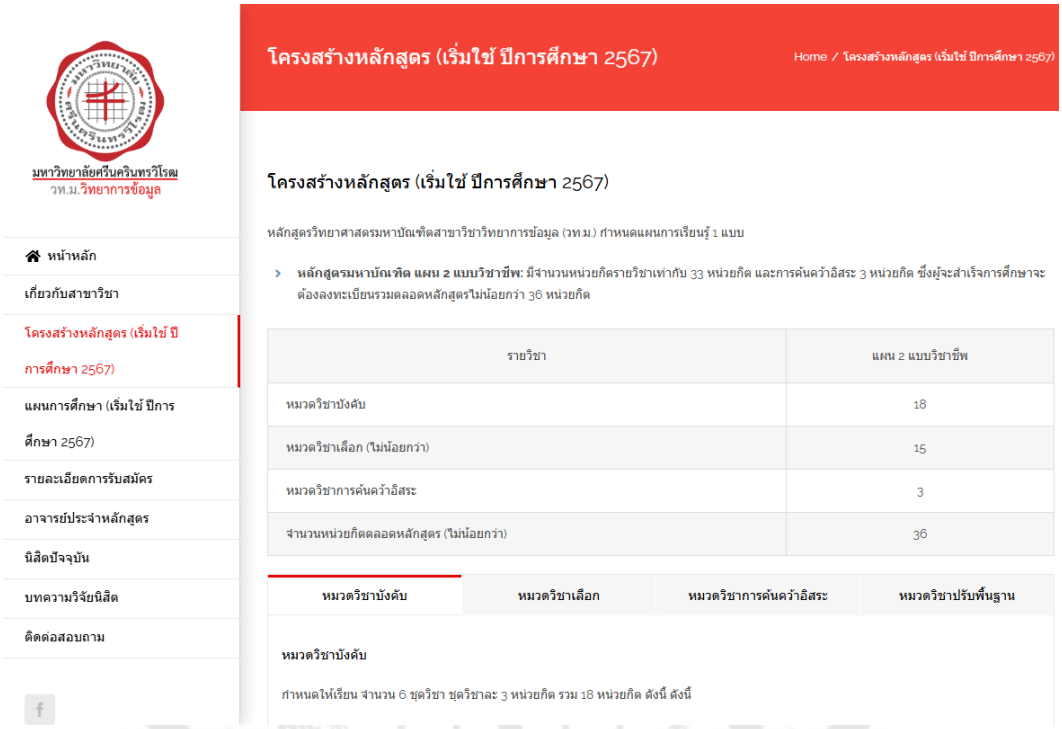


ภาพประกอบ 7 ข้อมูลเกี่ยวกับสาขาวิชา

ภาพประกอบ 7 แสดงรายละเอียดเกี่ยวกับหลักสูตรวิทยาศาสตร์มหาบัณฑิต สาขาวิทยาการข้อมูล Master of Science in Data Science: MSDS ซึ่งจัดขึ้นโดยคณะวิทยาศาสตร์ มหาวิทยาลัยศรีนครินทรวิโรฒ โดยหลักสูตรนี้มุ่งเน้นการพัฒนาบุคลากรที่มีศักยภาพสูงทางด้านวิทยาการข้อมูล พร้อมทั้งส่งเสริมการสร้างสรรค์นวัตกรรม และการพัฒนาขีดความสามารถในการแข่งขันในระดับสากล เพื่อตอบสนองความต้องการขององค์กรในยุคดิจิทัลอย่างมีประสิทธิภาพ

2.2 ข้อมูลเกี่ยวกับโครงสร้างหลักสูตร

การจัดวางวิชา การวางแผนการเรียน และจำนวนหน่วยกิตที่เกี่ยวข้องต่อไปนี้เป็นตัวอย่างของข้อมูลฝ่ายเกี่ยวกับโครงสร้างหลักสูตร

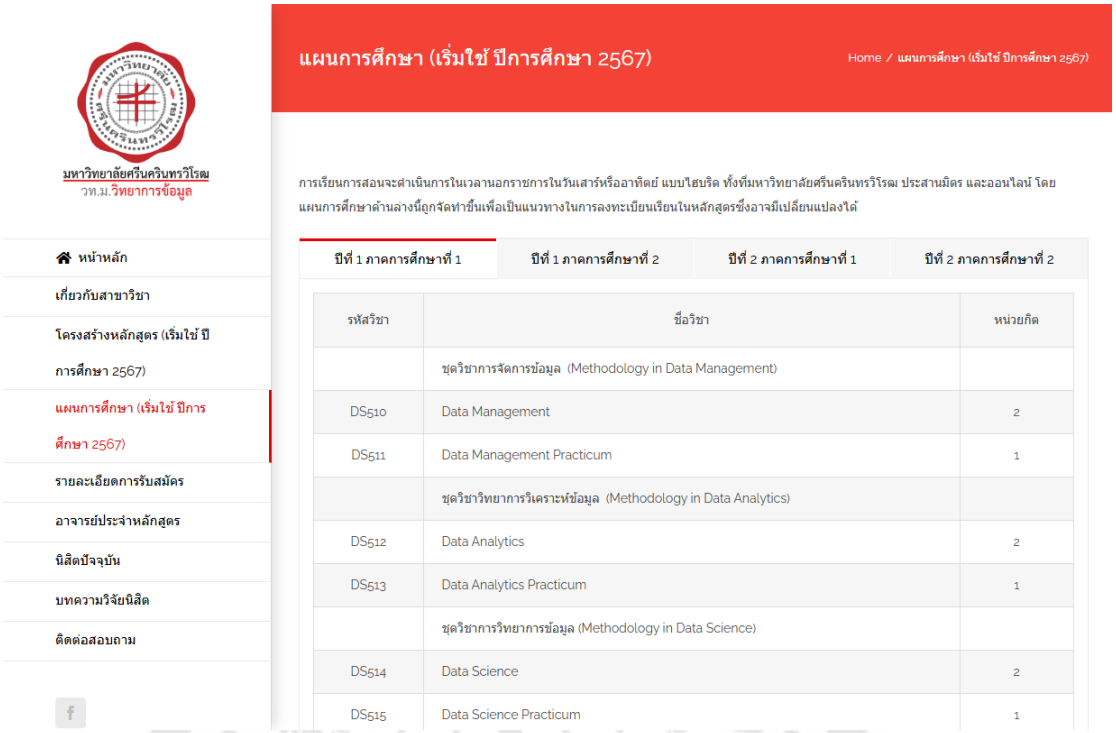


ภาพประกอบ 8 โครงสร้างหลักสูตร

จากภาพประกอบ 8 หลักสูตรวิทยาศาสตร์มหาบัณฑิต สาขาวิทยาการข้อมูล Master of Science in Data Science: MSDS ได้ระบุโครงสร้างรายวิชาและจำนวนหน่วยกิตที่จำเป็นสำหรับการสำเร็จการศึกษา โดยประกอบด้วยรายวิชาบังคับจำนวน 18 หน่วยกิต รายวิชาเลือกจำนวน 15 หน่วยกิต และการค้นคว้าอิสระจำนวน 3 หน่วยกิต รวมทั้งหมดต้องมีหน่วยกิตสะสมไม่น้อยกว่า 36 หน่วยกิต ทั้งนี้ โครงสร้างหลักสูตรได้รับการออกแบบเพื่อสร้างความสมดุลระหว่างความรู้พื้นฐาน ทักษะวิชาชีพ และการประยุกต์ใช้งานจริง เพื่อให้บัณฑิตมีความพร้อมในด้านวิชาการและการประกอบอาชีพในยุคดิจิทัลอย่างมีประสิทธิภาพ

2.3 ข้อมูลเกี่ยวกับแผนการเรียนและการศึกษา

ตารางเรียน ตารางสอบ และกำหนดการที่เกี่ยวข้องกับการศึกษา ต่อไปนี้เป็นตัวอย่างของข้อมูลฝ่ายเกี่ยวกับการศึกษา

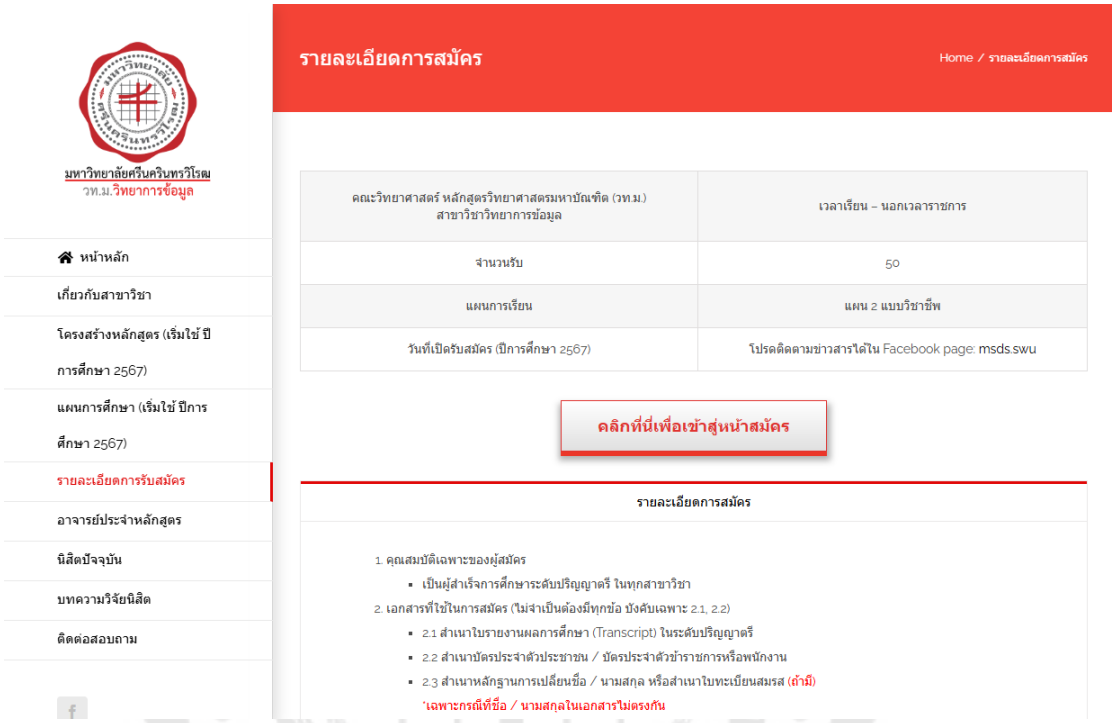


ภาพประกอบ 9 แผนการเรียนและการศึกษา

จากภาพประกอบ 9 ได้แสดงรายละเอียดรายวิชาตามแผนการเรียนและการศึกษา ในแต่ละภาคการศึกษา โดยมีการระบุรหัสวิชา ชื่อวิชา และจำนวนหน่วยกิตที่เกี่ยวข้อง ตัวอย่างเช่น ในปีการศึกษาที่ 1 ภาคการศึกษาที่ 1 มีรายวิชาในกลุ่ม ชุดวิชาการจัดการข้อมูล เช่น Data Management รหัส DS510 จำนวน 2 หน่วยกิต และ Data Management Practicum รหัส DS511 จำนวน 1 หน่วยกิต ในขณะที่กลุ่ม ชุดวิชาการวิเคราะห์ข้อมูล และ ชุดวิชาการวิทยาการข้อมูล ประกอบด้วยวิชาที่เน้นการวิเคราะห์และปฏิบัติการ เพื่อเตรียมความพร้อมสำหรับการประยุกต์ใช้ในสาขาวิชาชีพที่เกี่ยวข้อง หลักสูตรดังกล่าวถูกออกแบบอย่างเป็นระบบเพื่อส่งเสริมความรู้และทักษะเชิงปฏิบัติในระดับที่สามารถนำไปใช้จริงได้อย่างมีประสิทธิภาพ

2.4 ข้อมูลเกี่ยวกับกระบวนการรับสมัครและคุณสมบัติของผู้สมัคร

ขั้นตอนและเงื่อนไขการรับสมัคร รวมถึงข้อมูลที่ผู้สมัครควรทราบ ต่อไปนี้เป็นตัวอย่างของข้อมูลฝ่ายเกี่ยวกับการรับสมัคร

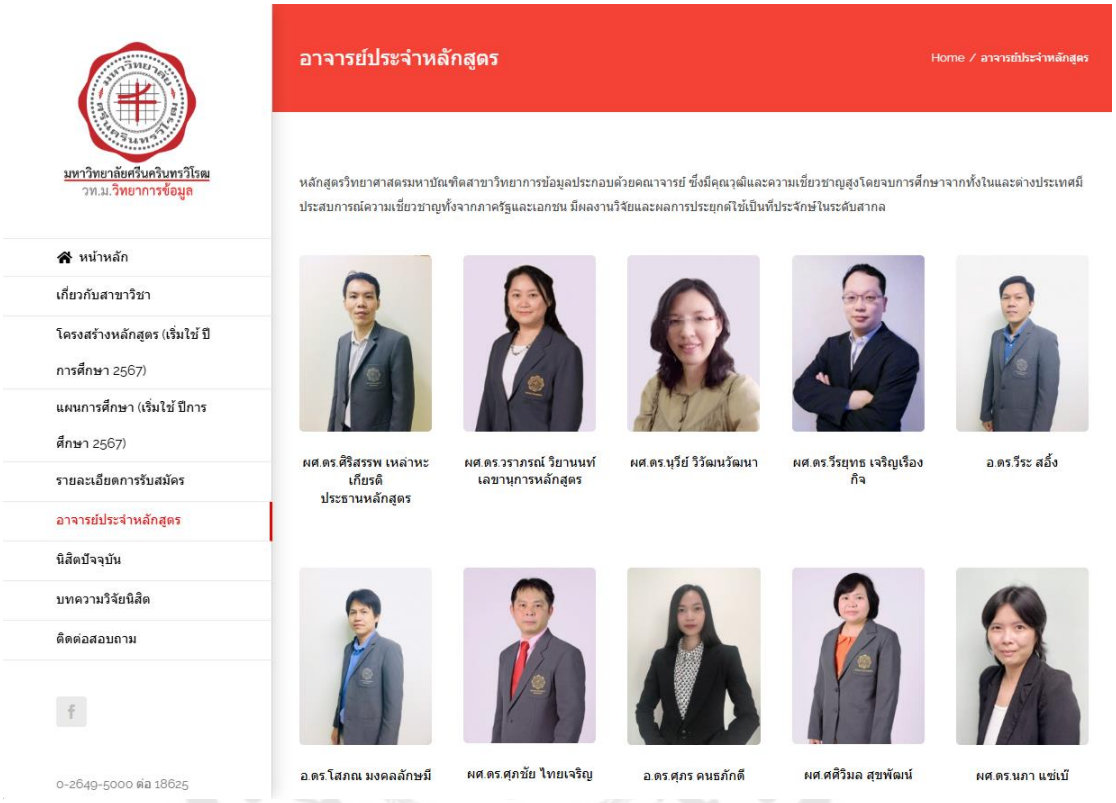


ภาพประกอบ 10 กระบวนการรับสมัครและคุณสมบัติของผู้สมัคร

จากภาพประกอบ 10 แสดงรายละเอียดเกี่ยวกับการรับสมัครหลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิทยาการข้อมูล วท.ม. โดยหลักสูตรนี้เปิดรับสมัครในแผนการเรียนแบบแผน 2 แผนวิชาชีพ และรองรับผู้เรียนจำนวน 50 คน ต่อปีการศึกษา การเรียนการสอนจัดในรูปแบบนอกเวลาราชการ ผู้สมัครต้องมีคุณสมบัติสำคัญ เช่น สำเร็จการศึกษาระดับปริญญาตรีในสาขาที่เกี่ยวข้อง และต้องยื่นเอกสารสำคัญ เช่น ใบแสดงผลการศึกษา Transcript ฉบับสมบูรณ์ พร้อมทั้งหนังสือรับรองจากอาจารย์หรือผู้บังคับบัญชา ทั้งนี้ การเปิดรับสมัครจะมีการกำหนดช่วงเวลาและรายละเอียดเพิ่มเติมซึ่งสามารถติดตามได้ทางเว็บไซต์หรือช่องทางสื่อสารของหลักสูตรอย่างเป็นทางการ

2.5 ข้อมูลเกี่ยวกับอาจารย์ประจำหลักสูตร

ประวัติการศึกษา ผลงานวิจัย และข้อมูลการติดต่อของอาจารย์ในหลักสูตร ต่อไปนี้เป็นตัวอย่างของข้อมูลฝ่ายเกี่ยวกับข้อมูลการติดต่อของอาจารย์ในหลักสูตร

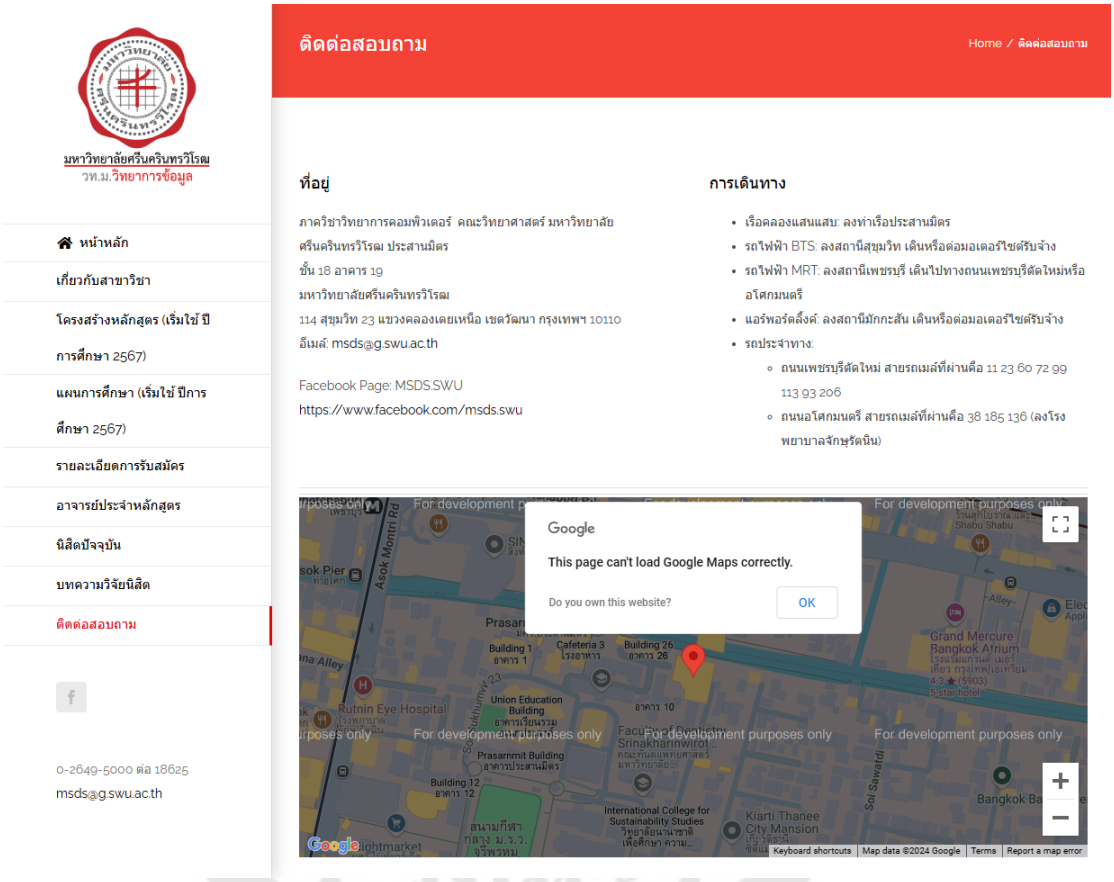


ภาพประกอบ 11 อาจารย์ประจำหลักสูตร

จากภาพประกอบ 11 ได้แสดงรายชื่อคณาจารย์ในหลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิทยาการข้อมูล พร้อมทั้งข้อมูลเกี่ยวกับความเชี่ยวชาญและความถนัดเฉพาะด้านของแต่ละท่าน โดยคณาจารย์ในหลักสูตรล้วนมีคุณวุฒิในระดับสูงและผ่านการศึกษาทั้งในประเทศและต่างประเทศ มีประสบการณ์ด้านการวิจัยและการประยุกต์ใช้ความรู้ในระดับสากล ซึ่งสามารถสนับสนุนการเรียนการสอนให้มีคุณภาพสูงสุด ตลอดจนช่วยเสริมสร้างศักยภาพของผู้เรียนให้สอดคล้องกับความต้องการของยุคสังคมดิจิทัลและการแข่งขันในระดับนานาชาติ

2.6 ข้อมูลเกี่ยวกับช่องทางการติดต่อและสถานที่ตั้ง

ช่องทางการติดต่อ เช่น อีเมล เบอร์โทรศัพท์ หรือที่อยู่สำหรับการติดต่อกับส่วนงานต่าง ๆ ต่อไปนี้เป็นตัวอย่างของข้อมูลฝ่ายช่องทางการติดต่อ



ภาพประกอบ 12 ช่องทางการติดต่อและสถานที่ตั้ง

จากภาพประกอบ 12 ได้แสดงข้อมูลเกี่ยวกับที่ตั้งและรายละเอียดการติดต่อของภาควิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยศรีนครินทรวิโรฒ โดยตั้งอยู่ที่ชั้น 18 อาคาร 19 มหาวิทยาลัยศรีนครินทรวิโรฒ เลขที่ 114 สุขุมวิท 23 เขตวัฒนา กรุงเทพฯ 10110 พร้อมทั้งมีช่องทางการติดต่อผ่านอีเมล (msds@g.swu.ac.th) และ Facebook Page MSDS.SWU นอกจากนี้ ยังมีรายละเอียดเกี่ยวกับการเดินทางมายังมหาวิทยาลัย ทั้งโดยรถไฟฟ้า BTS, รถไฟฟ้า MRT, รถประจำทาง และแอร์พอร์ตลิงก์ เพื่อความสะดวกในการเข้าถึงสถานที่เรียนและทำกิจกรรมทางวิชาการ

2.7 ข้อมูลเกี่ยวกับปฏิทินกำหนดการ

กำหนดการสำคัญ เช่น วันเปิดภาคเรียน วันสอบ และกำหนดส่งงานวิจัย ต่อไปนี้เป็นตัวอย่างของข้อมูลกำหนดการสำคัญของบัณฑิต

ประกาศมหาวิทยาลัยศรีนครินทรวิโรฒ
เรื่อง ปฏิทินการศึกษาระดับบัณฑิตศึกษา ประจำปีการศึกษา 2567

เพื่อให้การจัดการเรียนการสอนระดับบัณฑิตศึกษา ประจำปีการศึกษา 2567 เป็นไปด้วยความเรียบร้อย และเป็นไปตามข้อบังคับมหาวิทยาลัยศรีนครินทรวิโรฒ ว่าด้วยการจัดการศึกษาระดับบัณฑิตศึกษา พ.ศ. 2566 รวมถึงข้อบังคับมหาวิทยาลัยศรีนครินทรวิโรฒ ว่าด้วยเกณฑ์มาตรฐานระดับบัณฑิตศึกษา พ.ศ. 2566 อาศัยอำนาจตามความในมาตรา 29 มาตรา 34 และมาตรา 44 แห่งพระราชบัญญัติมหาวิทยาลัยศรีนครินทรวิโรฒ พ.ศ. 2559 และคำสั่งมหาวิทยาลัยศรีนครินทรวิโรฒ ที่ 10189/2563 ลงวันที่ 29 ธันวาคม 2563 เรื่อง การมอบอำนาจให้ผู้ปฏิบัติการแทนอธิการบดี จึงขอกำหนดปฏิทินการศึกษา ระดับบัณฑิตศึกษา ประจำปีการศึกษา 2567 ดังต่อไปนี้

ภาคเรียนที่ 1/2567 (ภาคต้น)
จ 19 สิงหาคม 2567 – อา 19 มกราคม 2568

วัน/เดือน/ปี	เวลา	กิจกรรม	หมายเหตุ
กรกฎาคม 2567			
จ 8 ก.ค. – จ 19 ส.ค. 67		ผู้ประสานคณะ/สถาบัน เปิด/แก้ไขรายวิชาในระบบ ภาค 1/2567	คณะ/สถาบัน/สำนัก http://supreme.swu.ac.th
จ 22 ก.ค. 67	*วันหยุดชดเชยวันอาสาฬหบูชา*		
จ 29 ก.ค. 67	*วันหยุดชดเชยเนื่องในวันเฉลิมพระชนมพรรษาพระบาทสมเด็จพระปรเมนทรรามาธิบดีศรีสินทรมหาวชิราลงกรณ พระวชิรเกล้าเจ้าอยู่หัว*		

ภาพประกอบ 13 ปฏิทินกำหนดการ

จากภาพประกอบ 13 แสดงให้เห็นถึง Academic Calendar สำหรับภาคเรียนที่ 1/2567 ภาคต้น โดยครอบคลุมระยะเวลาตั้งแต่วันที่ 19 สิงหาคม 2567 ถึงวันที่ 19 มกราคม 2568 ปฏิทินดังกล่าวได้รับการจัดทำในรูปแบบของ Tabular Format ซึ่งจะประกอบด้วยคอลัมน์แสดง Date/Month/Year, Time Period, Academic Activities และ Additional Notes โดยมีการระบุรายละเอียดกิจกรรมสำคัญทางวิชาการ เช่น วันผู้ประสานงานและสถาบันเปิด/แก้ไขข้อมูลในระบบสำหรับภาคการศึกษา 1/2567 ในวันที่ 19 กรกฎาคม 2567 รวมถึงวันหยุดพิเศษอื่นๆ ที่มีความเกี่ยวข้องกับการดำเนินงานทางวิชาการ

2.8 ข้อมูลเกี่ยวกับทุนการศึกษาของสาขา

ทุนสนับสนุนค่าครองชีพนิสิต ระดับบัณฑิตศึกษา และกำหนดการรับสมัคร ต่อไปนี้เป็นตัวอย่างของข้อมูลทุนการศึกษาสำคัญของภาควิชาวิทยาการคอมพิวเตอร์

ประกาศคณะวิทยาศาสตร์ มหาวิทยาลัยศรีนครินทรวิโรฒ
เรื่อง เงินทุนสนับสนุนค่าครองชีพนิสิต ระดับบัณฑิตศึกษา ภาควิชาวิทยาการคอมพิวเตอร์
ภาคเรียนที่ 2/2567

ภาควิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยศรีนครินทรวิโรฒ ได้ตระหนักถึงความสำคัญของการศึกษา จึงมีนโยบายสนับสนุนช่วยเหลือนิสิตระดับปริญญาโท จำนวน 28 ทุน ในภาคเรียนที่ 2/2567 ดังนี้

ระดับปริญญาโท

1.หลักสูตร วท.ม.วิทยาการข้อมูล

จำนวน 28 ทุน ๆ ละ 4,000 บาท/เดือน/ทุน จำนวน 1 เดือน เป็นเงิน 112,000 บาท

รวมเป็นเงินทั้งสิ้น 112,000 บาท
(หนึ่งแสนหนึ่งหมื่นสองพันบาทถ้วน)

ภาควิชาวิทยาการคอมพิวเตอร์ ได้จัดสรรเงินทุนสนับสนุนค่าครองชีพนิสิต ระดับบัณฑิตศึกษา ภาคเรียนที่ 2/2567 โดยมีรายละเอียดคุณสมบัติ ดังนี้

ภาพประกอบ 14 ทุนการศึกษาของสาขา

แสดงให้เห็นถึงประกาศเงินทุนสนับสนุนค่าครองชีพนิสิตระดับบัณฑิตศึกษา ภาควิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยศรีนครินทรวิโรฒ สำหรับภาคเรียนที่ 2/2567 โดยมีรายละเอียดการให้ทุนสำหรับหลักสูตร วท.ม.วิทยาการข้อมูล จำนวน 28 ทุน ๆ ละ 4,000 บาท/เดือน/ทุน จำนวน 1 เดือน รวมเป็นเงินทั้งสิ้น 112,000 บาท (หนึ่งแสนหนึ่งหมื่นสองพันบาทถ้วน) โดยคุณสมบัติของผู้รับทุนประกอบด้วย การเป็นนิสิตระดับปริญญาโท สังกัดภาควิชาวิทยาการคอมพิวเตอร์ ซึ่งจะมีหลักฐานการยื่นสอบปากเปล่าสารนิพนธ์/ปริญญานิพนธ์ มีร่าง Manuscript ที่เตรียมยื่นนำเสนอทางวิชาการ มีคะแนนเฉลี่ยสะสม (GPAX) ไม่ต่ำกว่า 2.50 และมีความประพฤติดี

จากข้อมูลที่รวบรวมมาทั้งหมดจะประกอบด้วย ข้อมูลที่เกี่ยวกับสาขาวิชาซึ่งครอบคลุมรายละเอียดโปรแกรมการศึกษาและสาขาวิชาต่างๆ รวมถึงโครงสร้างหลักสูตรที่แสดงการจัดวางวิชาและจำนวนหน่วยกิต แผนการเรียนและการศึกษาที่ประกอบด้วยตารางเรียน และกำหนดการที่เกี่ยวข้อง นอกจากนี้ยังมีกระบวนการรับสมัครและคุณสมบัติของผู้สมัครที่รวมถึงขั้นตอนและเงื่อนไขในการรับสมัคร รวมทั้งข้อมูลเกี่ยวกับกิจกรรมของนิสิตในระดับบัณฑิตศึกษาเช่น การจัดกิจกรรมและข้อมูลสนับสนุนต่างๆ ช่องทางการติดต่อและสถานที่ตั้งผ่านช่องทางต่างๆ เช่น อีเมลหรือเบอร์โทรศัพท์ ปฏิทินกำหนดการที่รวมวันสำคัญต่างๆ เช่น วันเปิดภาคเรียนและกำหนดส่งงานวิจัย ค่าธรรมเนียมการศึกษาที่บอกทั้งค่าเทอมและค่าปรับล่าช้า และรวมถึงทุนการศึกษาของสาขาที่มีคุณสมบัติและขั้นตอนต่างๆ ข้อมูลทั้งหมดนี้จะถูกเก็บไว้ในรูปแบบ JSON เพื่อความสะดวกในการนำไปใช้งานต่อไป

3. **การดึงข้อมูลและการเตรียมข้อมูล** Data Preparation

ประกอบด้วยกระบวนการตรวจสอบและจัดการข้อมูลอย่างเป็นระบบ โดยจะเริ่มจากการเก็บรวบรวมข้อมูลจากเว็บไซต์มหาวิทยาลัยผ่านเทคนิค Web Scraping ด้วยภาษา Python โดยใช้ไลบรารี BeautifulSoup และ Requests เพื่อดึงข้อมูลจากหน้าเว็บที่มีโครงสร้าง HTML จากนั้นนำข้อมูลที่ได้มาผ่านกระบวนการทำความสะอาด Data Cleaning ด้วยคลาส ThaiTextPreprocessor ที่ทำหน้าที่ปรับแก้ความไม่สอดคล้องของรูปแบบตัวอักษรภาษาไทย แปลงตัวเลขไทยเป็น Western Digits และแก้ไขปัญหาช่องว่างที่ไม่สม่ำเสมอ โดยจะมีการแปลงรูปแบบข้อมูลวันที่ให้เป็นมาตรฐานเดียวกันผ่านคลาส OpenAIDateParser ที่ใช้เทคนิค LLM-based parsing สำหรับจัดการกับรูปแบบวันที่ภาษาไทยที่หลากหลาย เช่น "จ 8 ก.ค. – จ 19 ส.ค. 67" เป็น "start_date": "2024-07-08", "end_date": "2024-08-19", "is_range": true และมีการตรวจสอบคุณภาพข้อมูลผ่านคลาส CalendarEventValidator ที่ช่วยระบุข้อผิดพลาดและคำเตือนเกี่ยวกับความสมบูรณ์ของข้อมูล สุดท้ายจัดเก็บข้อมูลที่ผ่านการเตรียมเรียบร้อยแล้วในรูปแบบ JSON ที่มีโครงสร้างชัดเจนตามคลาส CalendarEvent, ContactDetail, CourseStructure และคลาสอื่นๆ ที่กำหนดไว้ เพื่อให้พร้อมสำหรับการนำไปใช้ในระบบ RAG ต่อไป

โดยตัวอย่างของการที่ได้ทำการเก็บได้จากเว็บไซต์ของมหาลัยจะมีตัวอย่างข้อมูลดังนี้ ประกอบไปด้วย 1). ปฏิทินวิชาการและกำหนดการสำคัญ 2). กระบวนการรับสมัครและคุณสมบัติของผู้สมัคร 3). ช่องทางการติดต่อและสถานที่ตั้ง 4). โครงสร้างหลักสูตรและรายละเอียดรายวิชา 5). แผนการเรียนและการศึกษา 6). อัตราค่าธรรมเนียมการศึกษาและชำระล่าช้า 7). ทุนการศึกษาของสาขา

โครงสร้างของข้อมูลปฏิทินวิชาการและกำหนดการสำคัญมหาวิทยาลัยถูกจัดโครงสร้างในรูปแบบ JavaScript Object Notation JSON ซึ่งเป็นมาตรฐานที่มีประสิทธิภาพในการแลกเปลี่ยนข้อมูลผ่านเครือข่าย โดยประกอบด้วยโครงสร้างหลักคือ "academic_calendar" ซึ่งเป็นอาร์เรย์หลักที่บรรจุชุดข้อมูลปฏิทินวิชาการและกำหนดการสำคัญทั้งหมด ภายในประกอบด้วยแอตทริบิวต์ "education" ที่ระบุภาคการศึกษา เช่น ภาคต้นปี 1/2567 และแอตทริบิวต์ "schedule" ซึ่งเป็นอาร์เรย์ของกิจกรรมทางวิชาการที่จะเกิดขึ้นในภาคการศึกษานั้นๆ โดยแต่ละกิจกรรมประกอบด้วยข้อมูลสำคัญ ได้แก่ "date" วันที่จัดกิจกรรม "time" ช่วงเวลาที่จัดกิจกรรม ซึ่งอาจเป็นค่าว่างได้ "activity" รายละเอียดของกิจกรรม และ "note" ข้อมูลเพิ่มเติมหรือลิงก์อ้างอิง โครงสร้างข้อมูลในลักษณะนี้เอื้อประโยชน์ต่อการพัฒนาระบบสารสนเทศที่ต้องการความยืดหยุ่นในการจัดเก็บและเรียกใช้ข้อมูล

```
{
    "academic_calendar": [
        {
            "education": "ภาคเรียนที่ 1/2567 (ภาคต้น)",
            "schedule": [
                {
                    "date": "จ 8 ก.ค. - จ 19 ส.ค. 67",
                    "time": "",
                    "activity": "ผู้ประสานคณะ/สถาบัน เปิด/แก้ไขรายวิชาในระบบ ภาค 1/2567",
                    "note": "คณะ/สถาบัน/สำนัก http://supreme.swu.ac.th"
                },
```

ภาพประกอบ 15 ตัวอย่างข้อมูลปฏิทินวิชาการและกำหนดการสำคัญ

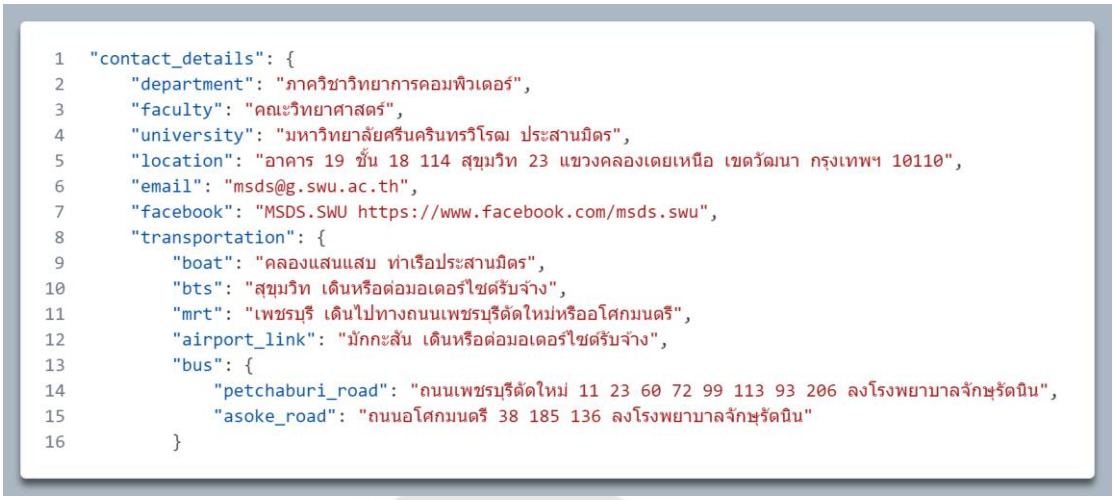
โครงสร้างของข้อมูลกระบวนการรับสมัครและคุณสมบัติของผู้สมัครเข้าศึกษาต่อในระดับอุดมศึกษาถูกจัดเก็บในรูปแบบโครงสร้างข้อมูล JavaScript Object Notation JSON ซึ่งประกอบด้วยหมวดหมู่หลัก "program_details" ที่รวบรวมองค์ประกอบสำคัญของกระบวนการสมัคร ได้แก่ "application_info" ที่ระบุช่องทางการเข้าถึงระบบรับสมัคร application_portal และช่องทางการติดต่อสื่อสาร program_email "required_documents" ซึ่งแบ่งเป็นเอกสารที่จำเป็นต้องมี mandatory เช่น สำเนาใบรับรองการศึกษา ใบแสดงผลการศึกษา และเอกสารแสดงตัวตน เอกสารเพิ่มเติม optional ในกรณีพิเศษ เช่น หลักฐานการเปลี่ยนชื่อ-นามสกุล และเอกสารรับรองความสามารถทางภาษาอังกฤษ english_proficiency ที่ระบุเกณฑ์คะแนนขั้นต่ำของการทดสอบมาตรฐานต่างๆ เช่น TOEFL ไม่ต่ำกว่า 450 คะแนน pBT/iTP IELTS ไม่ต่ำกว่า 5.0 พร้อมทั้งข้อกำหนดเรื่องอายุของเอกสารและข้อยกเว้นสำหรับผู้สำเร็จการศึกษาจากหลักสูตรนานาชาติ นอกจากนี้ ยังประกอบด้วย "submission_process" ที่แสดงรายละเอียดวิธีการส่งเอกสารในรูปแบบไฟล์ PDF ผ่านระบบออนไลน์หรือทางอีเมล และ

"selection_process" ที่อธิบายลำดับขั้นตอนการคัดเลือกผู้สมัคร ซึ่งประกอบด้วยการตรวจสอบคุณสมบัติเบื้องต้น การสัมภาษณ์เชิงวิชาการ และการตัดสินขั้นสุดท้ายโดยคณะกรรมการผู้ทรงคุณวุฒิ โครงสร้างข้อมูลในลักษณะนี้เอื้อประโยชน์ต่อการพัฒนาระบบสารสนเทศที่สามารถประมวลผลและนำเสนอข้อมูลแก่ผู้สมัครได้อย่างมีประสิทธิภาพ


```
"program_details": {
    "application_info": {
        "application_portal": "http://admission.swu.ac.th",
        "program_email": "msds@g.swu.ac.th"
    },
    "required_documents": {
        "mandatory": {
            "education": "ผู้สำเร็จการศึกษาระดับปริญญาตรี ในทุกสาขาวิชา",
            "transcript": "สำเนาใบรายงานผลการศึกษา (Transcript) ในระดับปริญญาตรี",
            "id_card": "สำเนาบัตรประจำตัวประชาชน / บัตรประจำตัวข้าราชการหรือพนักงาน"
        },
        "optional": {
            "name_change": {
                "name": "สำเนาหลักฐานการเปลี่ยนชื่อ / นามสกุล หรือสำเนาใบทะเบียนสมรส",
                "condition": "เฉพาะกรณีที่ชื่อ / นามสกุลในเอกสารไม่ตรงกัน"
            },
```


ภาพประกอบ 16 ตัวอย่างข้อมูลกระบวนการรับสมัครและคุณสมบัติของผู้สมัคร

โครงสร้างของข้อมูลช่องทางการติดต่อและสถานที่ตั้งถูกจัดโครงสร้างในรูปแบบ JavaScript Object Notation JSON ภายใต้องค์ประกอบหลัก "contact_details" ซึ่งประกอบด้วยข้อมูลหน่วยงานที่รับผิดชอบ ได้แก่ ภาควิชาวิทยาการคอมพิวเตอร์ สังกัดคณะวิทยาศาสตร์ มหาวิทยาลัยศรีนครินทรวิโรฒ ประสานมิตร โดยระบุสถานที่ตั้งอย่างละเอียดคือ อาคาร 19 ชั้น 18 ห้อง 114 เลขที่ สุขุมวิท 23 แขวง คลองเตยเหนือ เขตวัฒนา กรุงเทพมหานคร 10110 พร้อมทั้งช่องทางการติดต่อสื่อสารทางอิเล็กทรอนิกส์ ได้แก่ อีเมล msds@g.swu.ac.th และเครือข่ายสังคมออนไลน์เฟซบุ๊ก MSDS.SWU ซึ่งสามารถเข้าถึงได้ผ่านลิงก์ https://www.facebook.com/msds.swu นอกจากนี้ ยังมีข้อมูลทางเลือกในการเดินทางมายังสถานที่ตั้งที่ครอบคลุมหลากหลายรูปแบบ ประกอบด้วย การเดินทางทางน้ำผ่านคลองแสนแสนและท่าเรือประสานมิตร ระบบขนส่งมวลชนทางราง ได้แก่ รถไฟฟ้า BTS สถานีสุขุมวิท รถไฟฟ้า MRT สถานีเพชรบุรี และรถไฟฟ้าเชื่อมท่าอากาศยาน Airport Rail Link สถานีมักกะสัน โดยจากสถานีรถไฟฟ้าสามารถเดินทางต่อด้วยการเดินเท้าหรือใช้บริการมอเตอร์ไซค์รับจ้าง รวมถึงการเดินทางด้วยรถโดยสารประจำทางหลากหลายสาย ที่แบ่งตามเส้นทางถนนหลัก ได้แก่ ถนนเพชรบุรีตัดใหม่ สาย 11, 23, 60, 72, 99, 113, 93, 206 และถนนอโศกมนตรี สาย 38, 185, 136 โดยแนะนำให้ลงป้ายโรงพยาบาลจักษุรัตนินเพื่อความสะดวกในการเดินทางไปยังสถานที่ตั้ง



```
"contact_details": {
    "department": "ภาควิชาวิทยาการคอมพิวเตอร์",
    "faculty": "คณะวิทยาศาสตร์",
    "university": "มหาวิทยาลัยศรีนครินทรวิโรฒ ประสานมิตร",
    "location": "อาคาร 19 ชั้น 18 114 สุขุมวิท 23 แขวงคลองเตยเหนือ เขตวัฒนา กรุงเทพฯ 10110",
    "email": "msds@g.swu.ac.th",
    "facebook": "MSDS.SWU https://www.facebook.com/msds.swu",
    "transportation": {
        "boat": "คลองแสนแสบ ท่าเรือประสานมิตร",
        "bts": "สุขุมวิท เดินหรือต่อมอเตอร์ไซต์รับจ้าง",
        "mrt": "เพชรบุรี เดินไปทางถนนเพชรบุรีตัดใหม่หรือออโศกมนตรี",
        "airport_link": "มักกะสัน เดินหรือต่อมอเตอร์ไซต์รับจ้าง",
        "bus": {
            "petchaburi_road": "ถนนเพชรบุรีตัดใหม่ 11 23 60 72 99 113 93 206 ลงโรงพยาบาลจักษุรัตนิน",
            "asoke_road": "ถนนอโศกมนตรี 38 185 136 ลงโรงพยาบาลจักษุรัตนิน"
        }
```

ภาพประกอบ 17 ตัวอย่างข้อมูลช่องทางการติดต่อและสถานที่ตั้ง

โครงสร้างของข้อมูลหลักสูตร (เริ่มใช้ ปีการศึกษา 2567) ถูกนำเสนอในรูปแบบ JavaScript Object Notation JSON ภายใต้องค์ประกอบหลัก "course_structure" ซึ่งประกอบด้วยสองส่วนสำคัญ ได้แก่ "program_metadata" ที่บรรจุข้อมูลพื้นฐานของหลักสูตร เช่น ชื่อหลักสูตร หลักสูตรนวัตกรรมทางความคิด แบบ 2 แขนงวิชา สังกัดภาควิชา Data Science จำนวนหน่วยกิตรวมตลอดหลักสูตร 36 หน่วยกิต และระดับการศึกษา ปริญญาโท และ "curriculum_structure" ที่อธิบายโครงสร้างรายวิชาในหลักสูตร แบ่งเป็น "foundation_courses" หรือรายวิชาพื้นฐานที่ไม่นับหน่วยกิต ซึ่งออกแบบสำหรับนิสิตที่ขาดพื้นฐานความรู้ในหัวข้อที่เกี่ยวข้องหรือผู้ที่ต้องการทบทวนก่อนเริ่มเรียนตามหลักสูตร ประกอบด้วยวิชาเช่น วข501 DS501 การเขียนโปรแกรมและอัลกอริทึมสำหรับวิทยาการข้อมูล และ วข502 DS502 คณิตศาสตร์และสถิติสำหรับวิทยาการข้อมูล และ "core_courses" หรือรายวิชาบังคับที่นิสิตต้องลงทะเบียนเรียนอย่างน้อย 18 หน่วยกิต ประกอบด้วยวิชาต่างๆ ที่จัดเป็นหมวดหมู่หรือโมดูล เช่น วข510 DS510 การจัดการข้อมูล 2 หน่วยกิต และ วข511 DS511 ปฏิบัติการจัดการข้อมูล 1 หน่วยกิต โครงสร้างข้อมูลในลักษณะนี้เอื้อต่อการประมวลผลเชิงอัตโนมัติและการนำเสนอข้อมูลในรูปแบบที่เข้าใจง่ายผ่านระบบสารสนเทศของมหาวิทยาลัย เพื่อประโยชน์ในการให้คำแนะนำทางวิชาการและการวางแผนการเรียนและการศึกษา แก่นิสิต

```
"course_structure": {
    "program_metadata": {
        "name": "หลักสูตรบัณฑิต แผน 2 แบบวิชาชีพ",
        "department": "Data Science",
        "total_credits": 36,
        "degree_level": "master"
    },
    "curriculum_structure": {
        "foundation_courses": {
            "metadata": {
                "description": "สำหรับนิสิตที่ไม่เคยศึกษาในหัวข้อดังกล่าว หรือ นิสิตที่ต้องการทบทวนเนื้อหาพื้นฐาน",
                "credits": "non-credit"
            },
            "courses": [
                {
                    "code": "วข501 (DS501)",
                    "title": {
                      "th": "การเขียนโปรแกรมและอัลกอริทึมสำหรับวิทยาการข้อมูล",
                      "en": "Programming and Algorithms for Data Science"
                    },
                    "credits": 0
                },
```

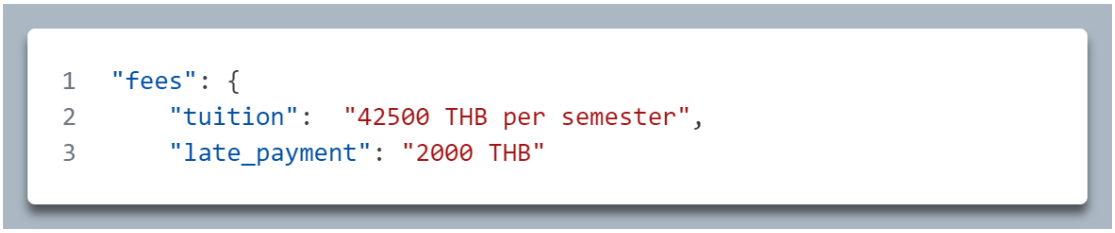
ภาพประกอบ 18 ตัวอย่างข้อมูลโครงสร้างหลักสูตรและรายละเอียดรายวิชา

โครงสร้างของข้อมูลแผนการเรียนและการศึกษา หลักสูตรถูกนำเสนอในรูปแบบโครงสร้าง JavaScript Object Notation JSON ภายใต้องค์ประกอบหลัก "program_study_plan" ซึ่งจัดลำดับตามปีการศึกษาและภาคการศึกษา โดยเริ่มจาก "year1" และแบ่งเป็น "semester1" ที่แสดงรายละเอียดภาคเรียนแรก ประกอบด้วยส่วน "metadata" ที่ระบุข้อมูลรวมของภาคการศึกษา เช่น จำนวนหน่วยกิตรวม 9 หน่วยกิต และประเภทรายวิชาหลัก core และส่วน "modules" ที่แสดงรายวิชาที่นิสิตควรลงทะเบียนในภาคการศึกษานั้น ซึ่งประกอบด้วยรายละเอียดของแต่ละรายวิชา ได้แก่ รหัสวิชา code เช่น วข510 DS510 ชื่อวิชาทั้งภาษาไทยและภาษาอังกฤษ title เช่น การจัดการข้อมูล Data Management และจำนวนหน่วยกิต credits เช่น 2 หน่วยกิต โดยในภาคการศึกษานี้ประกอบด้วยรายวิชาหลัก 3 วิชา ได้แก่ การจัดการข้อมูล 2 หน่วยกิต ปฏิบัติการจัดการข้อมูล 1 หน่วยกิต และวิทยาการวิเคราะห์ข้อมูล 2 หน่วยกิต โดยการจัดโครงสร้างข้อมูลในลักษณะนี้มีประโยชน์อย่างยิ่งต่อการพัฒนาระบบสารสนเทศเพื่อการบริหารจัดการหลักสูตร การให้คำปรึกษาทางวิชาการ และการวางแผนการเรียนและการศึกษา นิสิต โดยเอื้อต่อการประมวลผลข้อมูลแบบอัตโนมัติ การตรวจสอบความสอดคล้องของหลักสูตร และการนำเสนอข้อมูลในรูปแบบที่เข้าถึงได้ง่ายผ่านแพลตฟอร์มดิจิทัลต่างๆ ของสถาบันการศึกษา

```
"program_study_plan": {
    "year1": {
        "semester1": {
            "metadata": {
                "total_credits": 9,
                "course_type": "core"
            },
            "modules": [
                {
                    "code": "วข510 (DS510)",
                    "title": {
                        "th": "การจัดการข้อมูล",
                        "en": "Data Management"
                    },
                    "credits": 2
                },
                {
                    "code": "วข511 (DS511)",
                    "title": {
                        "th": "ปฏิบัติการจัดการข้อมูล",
                        "en": "Data Management Practicum"
                    },
                    "credits": 1
                },
```

ภาพประกอบ 19 ตัวอย่างข้อมูลแผนการเรียนและการศึกษา

โครงสร้างของข้อมูลค่าธรรมเนียมการศึกษาถูกจัดโครงสร้างในรูปแบบ JavaScript Object Notation JSON ภายใต้หมวดหมู่หลัก "fees" ซึ่งประกอบด้วยองค์ประกอบสำคัญสองส่วน ได้แก่ "tuition" ที่ระบุอัตราค่าเล่าเรียนพื้นฐาน 42,500 บาทต่อภาคการศึกษา และ "late_payment" ที่กำหนดค่าปรับกรณีชำระเงินเกินกำหนดเวลา 2,000 บาท โครงสร้างข้อมูลดังกล่าวถือเป็นส่วนสำคัญของระบบสารสนเทศเพื่อการบริหารจัดการทางการเงินของสถาบันการศึกษา การจัดเก็บข้อมูลในรูปแบบ JSON นี้จะช่วยเอื้อประโยชน์ต่อการพัฒนาระบบสารสนเทศที่สามารถตอบคำถามเกี่ยวกับค่าธรรมเนียมการศึกษาและค่าปรับล่าช้าได้อย่างถูกต้องและรวดเร็ว โดยเฉพาะอย่างยิ่งในการพัฒนาระบบของ AI chatbot (27) ที่จะสามารถช่วยให้ข้อมูลทางการเงินที่สำคัญแก่นักศึกษาและผู้ปกครองเพื่อประกอบการตัดสินใจและการวางแผนทางการเงิน



```
"fees": {
    "tuition":  "42500 THB per semester",
    "late_payment": "2000 THB"
```

ภาพประกอบ 20 ตัวอย่างข้อมูลอัตราค่าธรรมเนียมการศึกษาและชำระล่าช้า

โครงสร้างของข้อมูลทุนการศึกษาของสาขาถูกจัดโครงสร้างในรูปแบบ JavaScript Object Notation JSON ภายใต้หมวดหมู่หลัก "scholarship_details" ซึ่งประกอบด้วยองค์ประกอบสำคัญหกส่วน ได้แก่ "program_info" ที่ระบุรายละเอียดโครงการทุน "funding_details" ที่กำหนดจำนวนทุน 28 ทุน มูลค่าทุนละ 4,000 บาทต่อเดือน รวมมูลค่าทั้งสิ้น 112,000 บาท "eligibility_requirements" ที่ครอบคลุมเกณฑ์คุณสมบัติทางวิชาการและทั่วไป เช่น การมี GPAX ไม่ต่ำกว่า 2.50 และการมีร่าง Manuscript "application_process" ที่ระบุกำหนดเวลาสมัครภายในวันที่ 10 เมษายน 2568 "required_documents" ที่แจงรายการเอกสารที่จำเป็นทั้ง 4 ประเภท และ "selection_process" ที่กำหนดกระบวนการคัดเลือกผ่านการสัมภาษณ์ในวันที่ 21 เมษายน 2568

```
"scholarship_details": {
    "program_info": {
        "title": "เงินทุนสนับสนุนค่าครองชีพนิสิต ระดับบัณฑิตศึกษา ภาควิชาวิทยาการคอมพิวเตอร์",
        "institution": "ภาควิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยศรีนครินทรวิโรฒ",
        "semester": "ภาคเรียนที่ 2/2567",
        "program_type": "หลักสูตร วท.ม.วิทยาการข้อมูล (Master of Science in Data Science)"
    },
```

ภาพประกอบ 21 ตัวอย่างข้อมูลทุนการศึกษาของสาขา

การพัฒนาระบบ RAG ที่มีประสิทธิภาพสูงจำเป็นต้องอาศัยการจัดโครงสร้างข้อมูลที่เป็นระเบียบและเป็นระบบ โดยเฉพาะอย่างยิ่งในกระบวนการแบ่งเอกสาร Document Splitting ซึ่งถือเป็นขั้นตอนที่มีความสำคัญอย่างยิ่งต่อประสิทธิภาพ Retrieval ข้อมูล กล่าวคือ การแปรรูปข้อมูลจากโครงสร้างดั้งเดิมที่มีลักษณะเป็นอาร์เรย์รวมของกิจกรรมหลากหลายประเภทให้กลายเป็นเอกสารย่อย Document ที่แต่ละเอกสารเป็นตัวแทนของกิจกรรมเดี่ยว ช่วยให้ระบบสามารถระบุและดึงข้อมูลที่มีความเฉพาะเจาะจงกับความต้องการของผู้ใช้ได้อย่างมีประสิทธิภาพ

กระบวนการดังกล่าวดำเนินการผ่านการจัดโครงสร้างข้อมูลจาก JSON ต้นฉบับที่ได้รับการรวบรวมจากแหล่งข้อมูลหลากหลาย นำมาประมวลผลและแบ่งแยกเป็นเอกสารย่อยที่แต่ละเอกสารมีความสมบูรณ์ในตัวเอง และเมื่อนำมาประยุกต์ใช้กับข้อมูลปฏิทินวิชาการและกำหนดการสำคัญของมหาวิทยาลัย การแบ่งเอกสารในลักษณะนี้จะทำให้ระบบสามารถ Retrieval เฉพาะกิจกรรมที่เกี่ยวข้องกับคำถามของผู้ใช้ได้อย่างแม่นยำและรวดเร็ว เดิมแทนที่จะต้องประมวลผลข้อมูลทั้งปฏิทินวิชาการและกำหนดการสำคัญ ซึ่งไม่เพียงแต่ช่วยเพิ่มความเร็วในการตอบสนองของระบบเท่านั้น แต่ยังช่วยยกระดับความถูกต้องและความเกี่ยวข้องของข้อมูลที่นำเสนอให้แก่ผู้ใช้บริการทั้งนิสิต คณาจารย์ และบุคลากร

ของมหาวิทยาลัยได้อย่างมีประสิทธิภาพสูงสุด โดยการจัดโครงสร้างข้อมูลสำหรับระบบ RAG ได้รับการพัฒนาอย่างเป็นระบบเพื่อสนับสนุนประสิทธิภาพ Retrieval สารสนเทศ โดยกระบวนการดังกล่าวประกอบด้วยการออกแบบเชิงโครงสร้างที่มุ่งเน้นการผสานระหว่างองค์ประกอบหลักสองส่วน กล่าวคือ Content ซึ่งบรรจุข้อมูลเชิงปฏิบัติการ และ Metadata เป็นกระบวนการสำคัญที่ช่วยเสริมประสิทธิภาพการดำเนินงานของระบบ RAG โดยข้อมูลกำกับดังกล่าวทำหน้าที่เป็นโครงสร้างเชิงการจัดหมวดหมู่ซึ่งประกอบด้วยตัวแปรที่สกัดจากเนื้อหาหลัก เช่น event_type สำหรับจำแนกประเภทกิจกรรม semester สำหรับระบุภาคการศึกษา date สำหรับระบุวันที่จัดกิจกรรมในรูปแบบมาตรฐาน และ event_idx สำหรับกำหนดรหัสอ้างอิงเฉพาะของแต่ละกิจกรรม การกำหนดข้อมูลกำกับในลักษณะนี้มิเพียงเอื้อประโยชน์ต่อการสืบค้นและกรองข้อมูลตามเงื่อนไขที่ผู้ใช้ระบุเท่านั้น โดยการผนวกรวมองค์ประกอบทั้งสองส่วนนี้ก่อให้เกิดโครงสร้างข้อมูลที่เอื้อต่อการประมวลผลโดยระบบ AI ได้อย่างมีประสิทธิภาพ

การจัดโครงสร้างข้อมูลปฏิทินวิชาการและกำหนดการสำคัญสำหรับระบบ RAG ส่วนแรกจะเป็นในส่วนของ Content Structuring ซึ่งได้มีการจัดระเบียบให้ข้อมูลแต่ละรายการประกอบด้วยองค์ประกอบสำคัญที่ครอบคลุมสารสนเทศเชิงปฏิบัติการอย่างครบถ้วน เช่น การระบุภาคการศึกษาที่เกี่ยวข้อง ประเภทของกิจกรรมทางวิชาการหรือวันหยุด กรอบเวลาในการดำเนินกิจกรรม และรายละเอียดเพิ่มเติมซึ่งรวมถึงช่องทางการศึกษาข้อมูลเพิ่มเติม ทั้งนี้ การจัดรูปแบบดังกล่าวมิได้เป็นเพียงการนำเสนอข้อมูลในลักษณะที่เข้าใจง่ายสำหรับผู้ใช้งานเท่านั้น ในส่วนของ Metadata กรณีของข้อมูลปฏิทินวิชาการและกำหนดการสำคัญ การกำหนดค่า event_type เป็น "academic", "holiday", "deadline", และ "registration"

```
Document 1:
Content:
ภาคการศึกษา: ภาคต้น
ประเภท: academic
วันที่: จ 8 ก.ค. - จ 19 ส.ค. 67
เวลา: -
กิจกรรม: ผู้ประสานคณะ/สถาบัน เปิด/แก้ไขรายวิชาในระบบ ภาค 1/2567
หมวดหมู่: -
หมายเหตุ: คณะ/สถาบัน/สำนัก http://supreme.swu.ac.th

Metadata:
event_type: academic
semester: ภาคต้น
date: จ 8 ก.ค. - จ 19 ส.ค. 67
event_idx: 0
```

ภาพประกอบ 22 ตัวอย่างข้อมูลที่แปลงแล้วของปฏิทินวิชาการและกำหนดการสำคัญ

การจัดการข้อมูลช่องทางการติดต่อและสถานที่ตั้งสำหรับระบบ RAG ส่วนแรกจะเป็นในส่วนของ Content Structuring สำหรับข้อมูลการติดต่อ ได้มีการออกแบบให้ข้อมูลมีความเป็นระบบระเบียบและเป็นลำดับขั้นตอนโดยคำนึงถึงความสำคัญและความถี่ในการใช้งานเป็นสำคัญ กล่าวคือ เริ่มต้นจากการนำเสนอข้อมูลทั่วไปเกี่ยวกับหน่วยงาน อันได้แก่ ภาควิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยศรีนครินทรวิโรฒ ตามด้วยข้อมูลสถานที่ตั้งซึ่งระบุรายละเอียดอย่างชัดเจน ช่องทางการติดต่อออนไลน์ทั้งอีเมลและเฟซบุ๊ก ไปจนถึงข้อมูลเส้นทางการเดินทางในรูปแบบต่างๆ เช่น ทางเรือ รถไฟฟ้า BTS MRT Airport Link และรถประจำทาง พร้อมทั้งระบุจุดลงที่เหมาะสมเพื่ออำนวยความสะดวกแก่ผู้ที่ต้องการเดินทางมาติดต่อโดยตรง การจัดลำดับเนื้อหาในลักษณะนี้ไม่เพียงแต่เอื้อต่อการรับรู้และการเข้าถึงข้อมูลของผู้ใช้งานเท่านั้น หากยังส่งเสริมประสิทธิภาพใน Retrieval ข้อมูลของระบบ RAG ที่มักจะให้น้ำหนักกับข้อมูลที่ปรากฏในลำดับต้นๆ มากกว่าข้อมูลที่ปรากฏในส่วนท้ายของเอกสาร ในส่วนของ Metadata กรณีของข้อมูลช่องทางการติดต่อและสถานที่ตั้ง การกำหนดค่า event_type เป็น "contact"

```
Document 88:
Content:

ข้อมูลการติดต่อ:
คณะ: คณะวิทยาศาสตร์
ภาควิชา: ภาควิชาวิทยาการคอมพิวเตอร์
มหาวิทยาลัย: มหาวิทยาลัยศรีนครินทรวิโรฒ ประสานมิตร
สถานที่: อาคาร 19 ชั้น 18 114 สุขุมวิท 23 แขวงคลองเตยเหนือ เขตวัฒนา กรุงเทพฯ 10110

การติดต่อ:
อีเมล: msds@g.swu.ac.th
Facebook: "MSDS.SWU https://www.facebook.com/msds.swu"

การเดินทาง:
เรือ: คลองแสนแสบ ท่าเรือประสานมิตร
BTS: สุขุมวิท เดินหรือต่อมอเตอร์ไซต์รับจ้าง
MRT: เพชรบุรี เดินไปทางถนนเพชรบุรีตัดใหม่หรืออโศกมนตรี
Airport Link: มักกะสัน เดินหรือต่อมอเตอร์ไซต์รับจ้าง
รถประจำทาง: {"petchaburi_road":
             "ถนนเพชรบุรีตัดใหม่ 11 23 60 72 99 113 93 206 ลงโรงพยาบาลจักษุรัตนิน",
             "asoke_road": "ถนนอโศกมนตรี 38 185 136 ลงโรงพยาบาลจักษุรัตนิน"}


Metadata:
event_type: contact
```

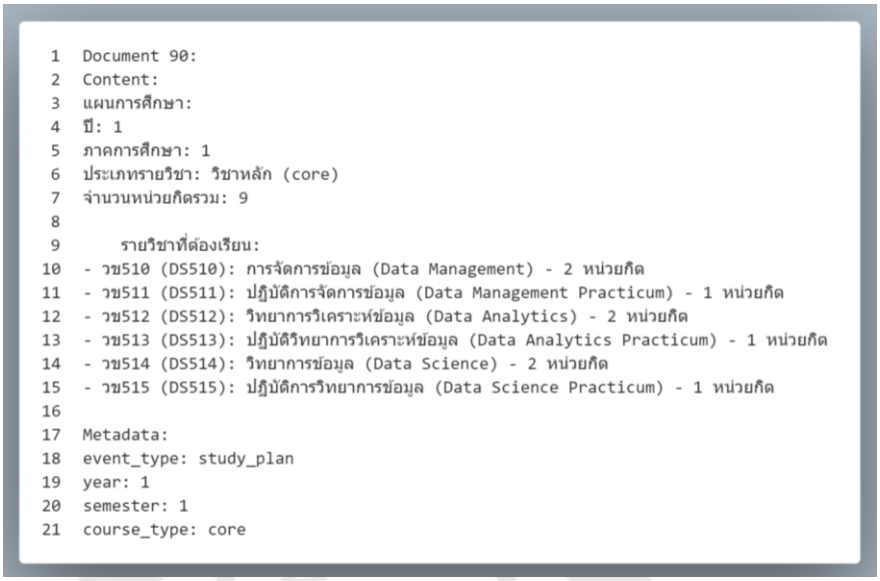
ภาพประกอบ 23 ตัวอย่างข้อมูลที่แปลงแล้วของติดต่อสอบถาม

การจัดการข้อมูลโครงสร้างหลักสูตรและรายละเอียดรายวิชาที่เริ่มใช้ในปีการศึกษา 2567 สำหรับระบบ RAG ส่วนแรกจะเป็นในส่วนของ Content Structuring สำหรับข้อมูลหลักสูตร ได้มีการดำเนินการปรับเปลี่ยนข้อมูลจากรูปแบบ JSON ดั้งเดิมให้มีความเป็นระเบียบและเอื้อต่อการประมวลผลมากยิ่งขึ้น โดยมีการจัดกลุ่มข้อมูลที่มีความสัมพันธ์กันเชิงโครงสร้างอย่างเป็นระบบ ประกอบด้วยข้อมูลทั่วไปของหลักสูตร เช่น ชื่อหลักสูตร ภาควิชาที่รับผิดชอบ จำนวนหน่วยกิตรวมที่ต้องศึกษาตลอดหลักสูตร และระดับการศึกษา ตามด้วยรายละเอียดโครงสร้างหลักสูตรและรายละเอียดรายวิชาซึ่งแบ่งเป็นหมวดหมู่ที่ชัดเจน ได้แก่ วิชาปรับพื้นฐาน วิชาบังคับ วิชาเลือก และวิชาการค้นคว้าอิสระ โดยแต่ละหมวดหมู่ประกอบด้วยคำอธิบายขอบเขตของรายวิชา จำนวนหน่วยกิตที่กำหนด และรายละเอียดของรายวิชาพร้อมรหัสวิชาที่เกี่ยวข้อง โดยการจัดโครงสร้างในลักษณะนี้ไม่เพียงแต่ช่วยให้ผู้ใช้งานสามารถทำความเข้าใจภาพรวมของหลักสูตรได้อย่างรวดเร็ว แต่ยังเอื้อต่อการประมวลผลของระบบ AI ในการวิเคราะห์ความสัมพันธ์ระหว่างองค์ประกอบต่างๆ ของหลักสูตรได้อย่างมีประสิทธิภาพ ในส่วนของ Metadata สำหรับข้อมูลหลักสูตร ได้มีการกำหนดค่า event_type เป็น "curriculum"

```
Document 89:
Content:
โครงสร้างหลักสูตร:
ชื่อหลักสูตร: หลักสูตรบัณฑิต แผน 2 แบบวิชาชีพ
ภาควิชา: Data Science
หน่วยกิตรวม: 36
ระดับการศึกษา: master

รายละเอียดโครงสร้าง:

หมวดวิชาปรับพื้นฐาน/วิชาพื้นฐาน:
คำอธิบาย: วิชาพื้นฐานที่จำเป็นต้องเรียน foundation courses รายวิชาปรับพื้นฐาน
หน่วยกิต: non-credit
รายวิชา:
- วข501 (DS501): การเขียนโปรแกรมและอัลกอริทึมสำหรับวิทยาการข้อมูล (Programming and Algorithms for Data Science) - 0 หน่วยกิต
- วข502 (DS502): คณิตศาสตร์และสถิติสำหรับวิทยาการข้อมูล (Mathematics and Statistics for Data Science) - 0 หน่วยกิต

```

ภาพประกอบ 24 ตัวอย่างข้อมูลที่แปลงแล้วของโครงสร้างหลักสูตรและรายละเอียดรายวิชา

การจัดการข้อมูลแผนการเรียนและการศึกษา ที่เริ่มใช้ในปีการศึกษา 2567 สำหรับระบบ RAG ส่วนแรกจะเป็นในส่วนของ Content Structuring สำหรับแผนการเรียนและการศึกษา ได้มีการดำเนินการแปลงข้อมูลจากรูปแบบ JSON ต้นฉบับที่มีลักษณะเป็นลำดับชั้น hierarchical structure ให้อยู่ในรูปแบบเอกสารที่มีโครงสร้างชัดเจนและเอื้อต่อการประมวลผลโดยระบบ RAG โดยแต่ละเอกสารจะประกอบด้วยสองส่วนหลัก ได้แก่ ส่วนเนื้อหา Content ที่บรรจุข้อมูลสำคัญของแผนการเรียนและการศึกษา ประกอบด้วยปีการศึกษา ภาคการศึกษา ประเภทรายวิชา จำนวนหน่วยกิตรวม และรายละเอียดของรายวิชาที่ต้องเรียนในแต่ละภาคการศึกษา ซึ่งรวมถึงรหัสวิชา ชื่อวิชา และจำนวนหน่วยกิตของแต่ละรายวิชา และส่วนข้อมูลกำกับ Metadata ที่ระบุคุณลักษณะสำคัญเพื่อ Retrieval ประกอบด้วย event_type, year, semester และ course_type การจัดโครงสร้างในลักษณะนี้สอดคล้องกับหลักการออกแบบฐานข้อมูลที่มุ่งเน้นการแยกส่วนและการจัดหมวดหมู่ข้อมูลอย่างเป็นระบบ อันเป็นพื้นฐานสำคัญของการจัดการความรู้ในระบบสารสนเทศสมัยใหม่ ในส่วนของ Metadata สำหรับแผนการเรียนและได้มีการกำหนดค่า event_type เป็น "study_plan"



```
Document 90:
Content:
แผนการศึกษา:
ปี: 1
ภาคการศึกษา: 1
ประเภทรายวิชา: วิชาหลัก (core)
จำนวนหน่วยกิตรวม: 9

    รายวิชาที่ต้องเรียน:
- วข510 (DS510): การจัดการข้อมูล (Data Management) - 2 หน่วยกิต
- วข511 (DS511): ปฏิบัติการจัดการข้อมูล (Data Management Practicum) - 1 หน่วยกิต
- วข512 (DS512): วิทยาการวิเคราะห์ข้อมูล (Data Analytics) - 2 หน่วยกิต
- วข513 (DS513): ปฏิบัติวิทยาการวิเคราะห์ข้อมูล (Data Analytics Practicum) - 1 หน่วยกิต
- วข514 (DS514): วิทยาการข้อมูล (Data Science) - 2 หน่วยกิต
- วข515 (DS515): ปฏิบัติการวิทยาการข้อมูล (Data Science Practicum) - 1 หน่วยกิต

Metadata:
event_type: study_plan
year: 1
semester: 1
course_type: core
```

ภาพประกอบ 25 ตัวอย่างข้อมูลที่แปลงแล้วของแผนการเรียนและการศึกษา

การจัดการข้อมูลกระบวนการรับสมัครและคุณสมบัติของผู้สมัครสำหรับระบบ RAG ส่วนแรกจะเป็นในส่วนของ Content Structuring สำหรับข้อมูลกระบวนการรับสมัครและคุณสมบัติของผู้สมัคร ได้มีการพัฒนาขึ้นตามหลักการออกแบบฐานข้อมูลที่มุ่งเน้น Retrieval อย่างมีประสิทธิภาพ โดยแบ่งเป็นสองส่วนหลักตามมาตรฐานของระบบ RAG ส่วนแรกคือ Content หรือเนื้อหาที่บรรจุข้อมูลเชิงรายละเอียดเกี่ยวกับเกณฑ์และเงื่อนไขการสมัคร โดยเฉพาะอย่างยิ่งข้อกำหนดด้านภาษาอังกฤษที่ได้รับการนำเสนออย่างเป็นระบบและมีโครงสร้างชัดเจน ประกอบด้วยคะแนนสอบมาตรฐานประเภทต่างๆ เช่น TOEFL ที่กำหนดให้ไม่ต่ำกว่า 450 คะแนน iBT ที่กำหนดให้ไม่ต่ำกว่า 45 คะแนน cBT ที่กำหนดให้ไม่ต่ำกว่า 133 คะแนน และ IELTS ที่กำหนดให้ไม่ต่ำกว่า 5.0 พร้อมทั้งข้อมูลเพิ่มเติมเกี่ยวกับระยะเวลาความถูกต้อง Validity และประโยชน์ Benefits ของคะแนนดังกล่าว การจัดโครงสร้างข้อมูลในลักษณะนี้ช่วยให้ผู้สนใจสมัครเข้าศึกษาสามารถเข้าถึงและเข้าใจข้อกำหนดด้านภาษาอังกฤษได้อย่างชัดเจน อันเป็นประเด็นสำคัญที่มักจะมีการสอบถามบ่อยครั้ง ในส่วนของ Metadata สำหรับแผนการเรียนและได้มีการกำหนดค่า event_type เป็น "program_details"

```
Document 94:
Content:
ข้อมูลการสมัคร:
เว็บไซต์รับสมัคร: http://admission.swu.ac.th
อีเมล: msds@g.swu.ac.th

เอกสารที่ต้องใช้:
- education: ผู้สำเร็จการศึกษาระดับปริญญาตรี ในทุกสาขาวิชา
- transcript: สำเนาใบรายงานผลการศึกษา (Transcript) ในระดับปริญญาตรี
- id_card: สำเนาบัตรประจำตัวประชาชน / บัตรประจำตัวข้าราชการหรือพนักงาน

เอกสารเพิ่มเติม:
- สำเนาหลักฐานการเปลี่ยนชื่อ / นามสกุล หรือสำเนาใบทะเบียนสมรส (เฉพาะกรณีที่ชื่อ / นามสกุลในเอกสารไม่ตรงกัน)
- ใบรับรองความรู้ภาษาอังกฤษ

เกณฑ์คะแนนที่ยอมรับ:
* TOEFL: pBT/ITP ไม่ต่ำกว่า 450 คะแนน, iBT ไม่ต่ำกว่า 45 คะแนน, cBT ไม่ต่ำกว่า 133 คะแนน
* IELTS: ไม่ต่ำกว่า 5.0
* SWU-SET: B2
Validity: ผลสอบมีอายุไม่เกิน 2 ปี นับถึงวันเปิดภาคเรียน
Benefits: ได้รับการยกเว้นไม่ต้องสอบวิชาภาษาอังกฤษทั่วไป
Exemptions: ผู้สมัครชาวไทยที่จบการศึกษาปริญญาตรี หรือโท จากต่างประเทศที่ใช้ภาษาอังกฤษในการเรียนการสอน

```

ภาพประกอบ 26 ตัวอย่างข้อมูลที่แปลงแล้วของกระบวนการรับสมัครและคุณสมบัติของผู้สมัคร

การจัดการข้อมูลค่าธรรมเนียมการศึกษาสำหรับระบบ RAG โดยส่วนแรกจะเป็นในส่วนของ Content Structuring สำหรับข้อมูลค่าธรรมเนียมการศึกษา ได้มีการดำเนินการอย่างเป็นระบบโดยการแปลงข้อมูลจากรูปแบบ JSON ต้นฉบับให้อยู่ในโครงสร้างที่เหมาะสมสำหรับ Retrieval และการนำเสนอข้อมูลแก่ผู้ใช้ ส่วนของเนื้อหาได้รับการจัดวางให้มีความชัดเจนและเข้าใจง่าย โดยแสดงประเภทของค่าธรรมเนียมการศึกษาและค่าปรับล่าช้า ประกอบด้วยค่าเล่าเรียน Tuition fee ที่กำหนดไว้ที่ 42,500.00 บาทต่อภาคการศึกษา และค่าปรับกรณีชำระเงินล่าช้า Late payment penalty จำนวน 2,000.00 บาท การนำเสนอข้อมูลในรูปแบบรายการที่ขึ้นต้นด้วยเครื่องหมายขีด - ช่วยให้ข้อมูลมีความเป็นระเบียบและสะดุดตา ทำให้ผู้ใช้สามารถเข้าถึงข้อมูลที่ต้องการได้อย่างรวดเร็วและไม่สับสน ในส่วนของ Metadata Enrichment สำหรับข้อมูลค่าธรรมเนียมการศึกษา ได้มีการกำหนดค่า event_type เป็น "fees"

```
Document 96:
Content:
ค่าธรรมเนียมการศึกษา:
- ค่าเล่าเรียน: 42,500.00 THB per semester
- ค่าปรับชำระล่าช้า: 2,000.00 THB

Metadata:
event_type: fees
```

ภาพประกอบ 27 ตัวอย่างข้อมูลที่แปลงแล้วของค่าธรรมเนียมการศึกษา

การจัดการข้อมูลทุนการศึกษาสำหรับระบบ RAG โดยในส่วนแรกจะเป็นในส่วนของ Content Structuring สำหรับข้อมูลทุนการศึกษา ได้มีการดำเนินการอย่างเป็นระบบโดยการแปลงข้อมูลจากรูปแบบ JSON ต้นฉบับให้อยู่ในโครงสร้างที่เหมาะสมสำหรับ Retrieval และการนำเสนอข้อมูลแก่ผู้ใช้ ส่วนของเนื้อหาได้รับการจัดวางให้มีความชัดเจนและเข้าใจง่าย ซึ่งจะแสดงข้อมูลรายละเอียดของเงินทุนสนับสนุนค่าครองชีพนิสิตระดับบัณฑิตศึกษา ภาควิชาวิทยาการคอมพิวเตอร์ ประกอบด้วยข้อมูลสถาบัน ภาคเรียนที่ 2/2567 หลักสูตร วท.ม.วิทยาการข้อมูล จำนวนทุน 28 ทุน มูลค่าทุนละ 4,000 บาทต่อเดือน รวมมูลค่าทั้งสิ้น 112,000 บาท คุณสมบัติผู้สมัครทั้งทางวิชาการและทั่วไป กระบวนการสมัครและเอกสารที่ต้องใช้ รวมถึงกระบวนการคัดเลือกและกำหนดการสำคัญ การนำเสนอข้อมูลในรูปแบบลำดับชั้นและรายการที่ขึ้นต้นด้วยเครื่องหมายขีด - ช่วยให้ข้อมูลมีความเป็นระเบียบและสะดุดตา ทำให้ผู้ใช้สามารถเข้าถึงข้อมูลที่ต้องการได้อย่างรวดเร็วและไม่สับสน ในส่วนของ Metadata Enrichment สำหรับข้อมูลทุนการศึกษา ได้มีการกำหนดค่า event_type เป็น "scholarship"

```
ทุนการศึกษา: เงินทุนสนับสนุนค่าครองชีพนิสิต ระดับบัณฑิตศึกษา ภาควิชาวิทยาการคอมพิวเตอร์

สถาบัน: ภาควิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยศรีนครินทรวิโรฒ
ภาคเรียน: ภาคเรียนที่ 2/2567
หลักสูตร: หลักสูตร วท.ม.วิทยาการข้อมูล (Master of Science in Data Science)

รายละเอียดทุน:
จำนวนทุน: 28
จำนวนเงิน: 4,000 บาท/เดือน/ทุน
ระยะเวลา: 1 เดือน
จำนวนเงินรวม: 112,000 บาท (หนึ่งแสนหนึ่งหมื่นสองพันบาทถ้วน)

คุณสมบัติผู้สมัคร:
คุณสมบัติทางวิชาการ:
- ระดับการศึกษา: นิสิตระดับปริญญาโท สังกัดภาควิชาวิทยาการคอมพิวเตอร์
- สถานะการลงทะเบียน: ต้องเป็นนิสิตคณะวิทยาศาสตร์ ระดับปริญญาโท สังกัดภาควิชาวิทยาการคอมพิวเตอร์
- ข้อกำหนดวิทยานิพนธ์: มีหลักฐานการยื่นสอบปากเปล่าสารนิพนธ์/ปริญญานิพนธ์
- ข้อกำหนด Manuscript: มีร่าง Manuscript ที่เตรียมยื่นนำเสนอทางวิชาการ (ผ่านการรับรองจากอาจารย์ที่ปรึกษา)
- เกรดเฉลี่ยขั้นต่ำ: คะแนนเฉลี่ยสะสม (GPAX) ไม่ต่ำกว่า 2.50

คุณสมบัติทั่วไป:
- สถานะการลงทะเบียน: ไม่อยู่ในระหว่างลาพักการศึกษาในช่วงที่ยื่นสมัครขอรับทุน
- ความประพฤติ: มีความประพฤติดี และไม่เคยถูกลงโทษตามระเบียบข้อบังคับด้านวินัยนิสิต มหาวิทยาลัยศรีนครินทรวิโรฒ พ.ศ. 2559
```

ภาพประกอบ 28 ตัวอย่างข้อมูลที่แปลงแล้วของทุนการศึกษา

สรุปแล้วการแปลงโครงสร้างข้อมูลจากรูปแบบ JSON ดั้งเดิมสู่รูปแบบเอกสาร RAG เป็นขั้นตอนสำคัญในการพัฒนาระบบตอบคำถามอัตโนมัติที่มีประสิทธิภาพ การออกแบบโครงสร้างข้อมูลที่เหมาะสมจะช่วยให้ระบบสามารถ Retrieval ข้อมูลได้อย่างแม่นยำและสร้างคำตอบที่มีคุณภาพ ซึ่งจะช่วยอำนวยความสะดวกให้แก่ผู้ใช้ในการค้นหาข้อมูลเกี่ยวกับโอกาสการรับทุนการศึกษาและเงื่อนไขการสมัครได้อย่างมีประสิทธิภาพ โดยการแปลงข้อมูลตามแนวทางที่มีการนำเสนอนี้ไม่เพียงแต่จะช่วยเพิ่มประสิทธิภาพของระบบ RAG เท่านั้น แต่ยังช่วยให้ข้อมูลมีความเป็นระเบียบและสามารถนำไปใช้ในงานอื่นๆ ได้ง่ายขึ้น เช่น การวิเคราะห์ข้อมูลทุนการศึกษา การติดตามผลการสมัครทุน หรือการนำเสนอข้อมูลทุนการศึกษาในรูปแบบอื่นๆ

4. การพัฒนาระบบโต้ตอบอัตโนมัติ ที่รองรับการสื่อสารข้อความ โดยใช้ LLM

การพัฒนาระบบโต้ตอบอัตโนมัติสำหรับข้อมูลทางวิชาการได้ถูกออกแบบบนพื้นฐานของเทคโนโลยี RAG (28) ซึ่งผสานความสามารถใน Retrieval ข้อมูลที่มีความเฉพาะเจาะจงเข้ากับความสามารถในการสร้างข้อความของ LLM ระบบนี้ได้รับการพัฒนาให้สามารถให้ข้อมูลที่ถูกต้องและตรงประเด็นเกี่ยวกับปฏิทินการศึกษา โครงสร้างหลักสูตรและรายละเอียดรายวิชา และข้อมูลสำคัญอื่นๆ ทางการศึกษา โดยมีองค์ประกอบหลักดังต่อไปนี้

4.1 Hybrid Document Store Architecture

การพัฒนาสถาปัตยกรรมคลังเอกสารแบบผสมผสานในงานวิจัยนี้ถือเป็นองค์ประกอบพื้นฐานที่มีความสำคัญอย่างยิ่งต่อประสิทธิภาพโดยรวมของระบบโต้ตอบอัตโนมัติ โดยสถาปัตยกรรมดังกล่าวได้รับการออกแบบตาม Modern Information Retrieval Systems ซึ่งผสานแนวคิดระหว่าง Semantic Retrieval และ Lexical Retrieval เข้าด้วยกันอย่างมีประสิทธิภาพ เพื่อให้ได้ผลลัพธ์ Retrieval ข้อมูลที่มีความแม่นยำและครอบคลุมมากที่สุด

องค์ประกอบหลักของคลังเอกสารแบบผสมผสานประกอบด้วยส่วนสำคัญที่ทำงานประสานกันอย่างเป็นระบบ ได้แก่ In-Memory Document Store ซึ่งเป็นพื้นฐานสำคัญที่ใช้สำหรับจัดเก็บเอกสารและดัชนีข้อมูลทั้งหมด โดยใช้การจัดเก็บข้อมูลใน RAM แทนการจัดเก็บบนอุปกรณ์จัดเก็บข้อมูลแบบถาวร ซึ่งสามารถเพิ่มประสิทธิภาพในการเข้าถึงข้อมูลและลดเวลาตอบสนองได้อย่างมีนัยสำคัญ โดยคลังเอกสารในหน่วยความจำนี้ทำหน้าที่เป็นศูนย์กลางในการจัดการเอกสารทั้งหมดภายในระบบและประสานงานกับองค์ประกอบอื่นๆ อย่างมีประสิทธิภาพ

```
1  self.store = InMemoryDocumentStore()
```

ภาพประกอบ 29 การสร้างตัวแปรสำหรับจัดเก็บเอกสารในหน่วยความจำ

นอกจากนี้ ระบบยังประกอบด้วย Document Embedder ซึ่งเป็นองค์ประกอบที่ใช้เทคโนโลยี SentenceTransformers อันเป็นโมเดลภาษาที่ได้รับการฝึกฝนให้สร้าง Vector Representations จากข้อความ โดยมีวัตถุประสงค์หลักเพื่อแปลงข้อความให้อยู่ในรูปแบบเวกเตอร์หลายมิติที่สามารถนำไปใช้ในการคำนวณ Semantic Similarity ระหว่างข้อความต่างๆ ได้อย่างมีประสิทธิภาพ การฝังตัวของข้อความในรูปแบบเวกเตอร์นี้ถือเป็นพื้นฐานสำคัญสำหรับกระบวน Retrieval เชิงความหมายภายในระบบ

```
1  self.embedder = SentenceTransformersDocumentEmbedder(
2      model=config.model.embedder_model
3  )
```

ภาพประกอบ 30 การสร้างตัวแปร Document Embedder

ในส่วนของ Retrieval ข้อมูล ระบบได้นำอัลกอริทึม BM25 มาประยุกต์ใช้ผ่านองค์ประกอบ BM25 Retriever ซึ่งเป็น Ranking Function ที่ได้รับการยอมรับอย่างกว้างขวางในวง Retrieval สารสนเทศ BM25 เป็นส่วนขยายของโมเดลความน่าจะเป็น ที่คำนวณคะแนนความเกี่ยวข้องระหว่างเอกสารและคำค้นหา โดยพิจารณาจากความถี่ของคำในเอกสาร ความถี่ผกผันของเอกสาร รวมถึงปัจจัยด้านความยาวของเอกสาร ซึ่งส่วนประกอบนี้มีบทบาทสำคัญในการค้นหาเอกสารที่มีคำสำคัญตรงกับคำค้นหาของผู้ใช้งานอย่างแม่นยำ

```
self.bm25_retriever = InMemoryBM25Retriever(
    document_store=self.store,
    top_k=config.retriever.top_k
)
```

ภาพประกอบ 31 การสร้าง BM25 Retriever สำหรับ Retrieval เอกสาร

ควบคู่กับ Retrieval เชิงคำสำคัญ ระบบยังใช้ Embedding Retriever ซึ่งเป็นองค์ประกอบที่ใช้ Vector Embeddings สำหรับการค้นหาเอกสารโดยคำนวณความคล้ายคลึงเชิงความหมายระหว่างคำค้นหาและเอกสาร ซึ่งตัว Retrieval นี้ทำงานโดยการแปลงคำค้นหาให้เป็นเวกเตอร์โดยใช้โมเดลเดียวกันกับที่ใช้กับเอกสาร จากนั้นจึงคำนวณระยะห่าง Semantic Distance ระหว่างเวกเตอร์เหล่านั้น โดยทั่วไปจะใช้วิธีการวัดความคล้ายคลึงด้วย Cosine Similarity หรือ Euclidean Distance ซึ่งช่วยให้ระบบสามารถค้นหาเอกสารที่มีความหมายใกล้เคียงกับคำค้นหาได้ แม้ว่าจะใช้คำศัพท์ที่แตกต่างกันก็ตาม

```
self.embedding_retriever = InMemoryEmbeddingRetriever(
    document_store=self.store,
    top_k=config.retriever.top_k
)
```

ภาพประกอบ 32 การสร้าง InMemoryEmbeddingRetriever สำหรับ Retrieval เอกสาร

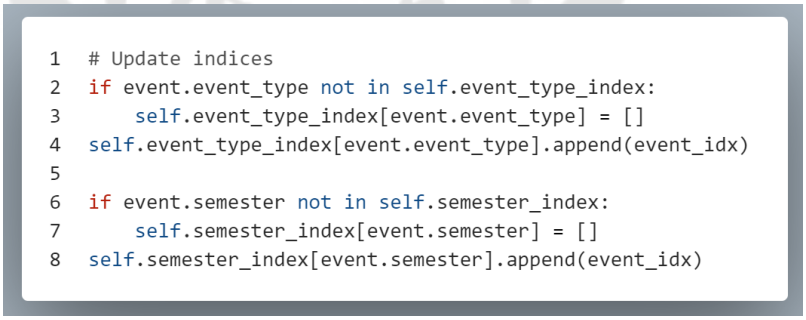
สำหรับกระบวนการจัดเก็บและการสร้างดัชนี โดยคลังเอกสารแบบผสมผสานได้ใช้กระบวนการที่มีประสิทธิภาพซึ่งเริ่มต้นด้วยการสร้างเอกสารที่มีโครงสร้าง โดยระบบจะทำการแปลงข้อมูลดิบจากแหล่งต่างๆ ให้เป็นเอกสารที่มีโครงสร้างซึ่งประกอบด้วย Content และ Metadata ที่เกี่ยวข้อง เช่น ประเภทเหตุการณ์ ภาคการศึกษา และข้อมูลอื่นๆ ที่จำเป็นสำหรับ

Retrieval ที่มีประสิทธิภาพ จากนั้นจึงดำเนินการสร้าง Embedding Vector Generation สำหรับเอกสารแต่ละฉบับโดยใช้โมเดล SentenceTransformers ที่กำหนดไว้ ซึ่งกระบวนการนี้จะแปลงเนื้อหาข้อความให้เป็นเวกเตอร์หลายมิติที่สามารถจับความหมายเชิงความหมายของข้อความได้อย่างมีประสิทธิภาพ

```python
def _compute_embedding(self, text: str) -> Any:
    """Compute embedding with caching"""
    cached_embedding = self.cache_manager.get_embedding_cache(text)
    if cached_embedding is not None:
        return cached_embedding

    doc = Document(content=text)
    embedding = self.embedder.run(documents=[doc])["documents"][0].embedding
    self.cache_manager.set_embedding_cache(text, embedding)
    return embedding
```

ภาพประกอบ 33 ฟังก์ชันการคำนวณ Embedding พร้อมระบบแคช

การสร้าง Hybrid Indexing เป็นขั้นตอนสำคัญที่ระบบจะสร้างดัชนีสองประเภทพร้อมกัน ได้แก่ Semantic Index ซึ่งเก็บเวกเตอร์ฝังตัวของเอกสารสำหรับ Retrieval เชิงความหมาย โดยการดำเนินการนี้มักใช้โครงสร้างข้อมูลที่เหมาะสม เช่น Approximate Nearest Neighbor Indexes เพื่อให้การค้นหามีประสิทธิภาพแม้ในพื้นที่เวกเตอร์ที่มีมิติสูง และ Lexical Index ซึ่งสร้าง Inverted Index สำหรับ Retrieval เชิง BM25 โดยเก็บความถี่ของคำและข้อมูลอื่นๆ ที่จำเป็นสำหรับการคำนวณคะแนน BM25 ได้อย่างมีประสิทธิภาพ



```python
# Update indices
if event.event_type not in self.event_type_index:
    self.event_type_index[event.event_type] = []
self.event_type_index[event.event_type].append(event_idx)

if event.semester not in self.semester_index:
    self.semester_index[event.semester] = []
self.semester_index[event.semester].append(event_idx)
```

ภาพประกอบ 34 การปรับปรุงดัชนีเหตุการณ์ตามประเภทและภาคการศึกษา

นอกจากดัชนีหลักแล้ว ระบบยังได้มี Specialized Indices Management โดยสร้างดัชนีเพิ่มเติมเพื่อเพิ่มประสิทธิภาพการค้นหาในกรณีเฉพาะ เช่น Event Type ที่ช่วยให้การค้นหาเฉพาะประเภทเหตุการณ์ทำได้อย่างรวดเร็ว และ Semester Index ที่อำนวยความสะดวกในการค้นหาข้อมูลที่เกี่ยวข้องกับภาคการศึกษาเฉพาะได้อย่างมีประสิทธิภาพ ท้ายที่สุด เพื่อจัดการกับ

ข้อมูลขนาดใหญ่อย่างมีประสิทธิภาพ ระบบได้ใช้ Batching Strategy ในการเขียนเอกสารเข้าคลังข้อมูล ซึ่งจะช่วยให้หลีกเลี่ยงการใช้หน่วยความจำมากเกินไปและปรับปรุงประสิทธิภาพโดยรวมของระบบได้อย่างมีนัยสำคัญ

```python
batch_size = 10
for i in range(0, len(documents), batch_size):
    batch = documents[i:i + batch_size]
    try:
        self.store.write_documents(batch)
    except Exception as e:
        logger.error(f"Error writing document batch {i//batch_size + 1}: {str(e)}")
        for doc in batch:
            try:
                self.store.write_documents([doc])
            except Exception as e2:
                logger.error(f"Failed to write document {doc.id}: {str(e2)}")
```

ภาพประกอบ 35 กระบวนการเขียนเอกสารแบบแบตช์พร้อมกลไกการจัดการข้อผิดพลาด

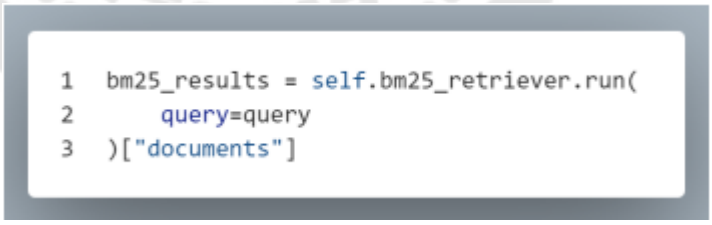
4.2 Hybrid Retrieval Mechanism

Retrieval ข้อมูลแบบผสมผสาน Hybrid Retrieval เป็นนวัตกรรมสำคัญในการพัฒนาระบบโต้ตอบอัตโนมัติที่ได้รับการออกแบบเพื่อเพิ่มประสิทธิภาพในการค้นหาข้อมูลที่เกี่ยวข้องอย่างแม่นยำ โดยผสานจุดแข็งของเทคนิค Retrieval ที่หลากหลายเข้าด้วยกัน ระบบนี้ได้รับการพัฒนาบนพื้นฐานทางทฤษฎีของ Retrieval สารสนเทศสมัยใหม่ที่มุ่งเน้นการผสมผสาน Retrieval เชิงความหมาย Semantic Retrieval และ Retrieval เชิงคำสำคัญ Lexical Retrieval เพื่อให้ได้ประสิทธิภาพสูงสุด

พื้นฐานทางทฤษฎีและหลักการ Hybrid Retrieval นั้นจะวางอยู่บนหลักการทางทฤษฎีสารสนเทศที่ว่าระบบ Retrieval ประเภทต่างๆ มีจุดแข็งและข้อจำกัดที่แตกต่างกันออกไป การผสมผสานจึงเป็นแนวทางที่มีประสิทธิภาพในการลดข้อจำกัดของแต่ละวิธี Retrieval ให้เหลือน้อยที่สุด โดยปัญหาสำคัญที่ระบบนี้มุ่งแก้ไขประกอบด้วยปัญหา Vocabulary Gap ซึ่งเกิดขึ้นเมื่อผู้ใช้และเอกสารใช้คำที่แตกต่างกันในการอธิบายแนวคิดเดียวกัน ปัญหา Semantic Ambiguity ที่เกิดขึ้นเมื่อคำเดียวกันอาจมีความหมายที่แตกต่างกันในบริบทที่ต่างกัน และปัญหา Exact Matching Problem ซึ่งระบบ Retrieval แบบดั้งเดิมที่ต้องการการจับคู่คำที่ตรงกันแบบแม่นยำอาจไม่สามารถค้นหาเอกสารที่เกี่ยวข้องแต่ใช้คำที่แตกต่างกันได้อย่างมีประสิทธิภาพ

ระบบ Retrieval แบบผสมผสานในงานวิจัยนี้ประกอบด้วยองค์ประกอบหลักสองส่วนด้วยกัน ส่วนแรกคือ Semantic Retrieval System ซึ่งใช้เทคนิค Text Embedding เพื่อแปลงข้อความเป็นเวกเตอร์ในปริภูมิหลายมิติ ทำให้สามารถวัดความคล้ายคลึงเชิงความหมายได้ด้วยการคำนวณระยะห่างระหว่างเวกเตอร์ได้อย่างมีประสิทธิภาพ การคำนวณการฝังตัวข้อความนี้ถูกดำเนินการด้วยโมเดล SentenceTransformers ซึ่งเป็นโมเดลที่ได้รับการฝึกฝนให้สร้างเวกเตอร์ของข้อความที่มีความคล้ายคลึงเชิงความหมายให้มีตำแหน่งที่ใกล้กันในปริภูมิเวกเตอร์ โดยกระบวนการนี้ประกอบด้วยขั้นตอนสำคัญหลายประการได้แก่ Preprocessing ซึ่งข้อความจะถูก Cleaning และแบ่งเป็น Token การแปลงเป็นเวกเตอร์ที่ Token จะถูกแปลงเป็นเวกเตอร์ด้วยโมเดลภาษา Normalization ซึ่งเวกเตอร์จะได้รับการปรับขนาดเพื่อให้เปรียบเทียบได้ง่ายขึ้น และ Similarity Computation ที่ใช้ Cosine Similarity เพื่อวัดความคล้ายระหว่างเวกเตอร์อย่างมีประสิทธิภาพ

องค์ประกอบหลักส่วนที่สองคือระบบ Retrieval เชิงคำสำคัญ ซึ่งใช้อัลกอริทึม BM25 อันเป็นรูปแบบที่พัฒนาต่อยอดจากโมเดล Term Frequency-Inverse Document Frequency โดยมีการเพิ่มการคำนวณความยาวของเอกสารและ Term Frequency Saturation เพื่อให้ได้ผลลัพธ์ Retrieval ที่แม่นยำยิ่งขึ้น ซึ่งระบบนี้ทำงานโดยการวิเคราะห์ความถี่ของคำในเอกสารและความถี่ของคำนั้นในคลังเอกสารทั้งหมด เพื่อคำนวณค่าความสำคัญของคำในเอกสารนั้นๆ และนำมาใช้ในการจัดอันดับความเกี่ยวข้องของเอกสารกับคำค้นหาได้อย่างมีประสิทธิภาพ



ภาพประกอบ 36 document retrieval ด้วยอัลกอริทึม BM25

กระบวนการผสมผลลัพธ์ถือเป็นขั้นตอนสำคัญที่ช่วยให้ระบบสามารถรวมผลลัพธ์จากทั้งสองวิธีเข้าด้วยกันอย่างมีประสิทธิภาพ โดยประกอบด้วยขั้นตอนสำคัญหลายประการ ขั้นตอนแรกคือ Score Normalization ซึ่งคะแนนจากทั้งสองระบบจะถูกปรับให้อยู่ในช่วง 0 ถึง 1 เพื่อให้สามารถเปรียบเทียบกันได้ โดยใช้วิธี Linear Normalization ด้วยการหารด้วยค่าสูงสุด ทำให้คะแนนจากทั้งสองระบบอยู่ในมาตรฐานเดียวกันและสามารถนำมาเปรียบเทียบกันได้อย่างเหมาะสม

Weighted Score Fusion โดยจะเป็นกระบวนการสำคัญในการผสมผลลัพธ์จากระบบ Retrieval ที่แตกต่างกัน โดยใช้ weight parameter เพื่อกำหนดอิทธิพลของแต่ละระบบต่อคะแนนรวม ในกรณีนี้ พารามิเตอร์ $\boldsymbol{w_{sem}}$ เป็นตัวกำหนดน้ำหนักให้กับคะแนนจากระบบ Retrieval เชิงความหมาย โดยคะแนนรวมคำนวณจากสมการ:

$$\boldsymbol{score_{combined} = w_{sem} \cdot score_{semantic} + (1 - w_{sem}) \cdot score_{lexical}} \quad (8)$$

โดยที่

$\boldsymbol{score_{combined}}$ คือ คะแนนรวมหลังการถ่วงน้ำหนัก

$\boldsymbol{score_{semantic}}$ คือคะแนนระบบ Retrieval เชิงความหมาย หลังการปรับนอร์มอลไซ์

$\boldsymbol{score_{lexical}}$ คือ คะแนนจากระบบ Retrieval เชิงคำสำคัญ หลังการปรับนอร์มอลไซ์

$\boldsymbol{w_{sem}}$ คือ ค่าน้ำหนักสำหรับคะแนนเชิงความหมาย ซึ่งมีค่าระหว่าง 0 ถึง 1

$\boldsymbol{1 - w_{sem}}$ คือ ค่าน้ำหนักสำหรับคะแนนเชิงคำสำคัญ

หลังจากการรวมคะแนนแล้ว จะเข้าสู่ขั้นตอน Ranking and Selection ซึ่งเอกสารทั้งหมดจะถูกจัดอันดับตามคะแนนรวมที่คำนวณได้ และมีการเลือกเฉพาะ Top_k เอกสารที่มีคะแนนสูงสุดเพื่อนำเสนอต่อผู้ใช้งานต่อไป การจัดอันดับนี้ช่วยให้ผลลัพธ์ Retrieval มีความแม่นยำและตรงกับความต้องการของผู้ใช้มากยิ่งขึ้น

ระบบได้รับการออกแบบให้สามารถปรับเปลี่ยนน้ำหนักในการผสมผลลัพธ์ตามประเภทของข้อมูลและคำถาม ซึ่งเป็นการเพิ่มความยืดหยุ่นให้กับ Retrieval อย่างมาก ตัวอย่างเช่น ในกรณีของข้อมูลปฏิทินวิชาการและกำหนดการสำคัญ ซึ่งมักต้องการความเข้าใจเชิงความหมายมากกว่า เนื่องจากผู้ใช้อาจใช้คำที่หลากหลายในการถามเกี่ยวกับวันที่หรือกิจกรรม จึงให้น้ำหนักกับ Retrieval เชิงความหมายมากกว่า Retrieval เชิงคำสำคัญ ทั้งนี้เพื่อให้ได้ผลลัพธ์ที่ตรงกับความต้องการของผู้ใช้มากที่สุด

นอกจากนี้ Intelligent Retry Mechanism ที่สามารถสลับค่าน้ำหนักระหว่าง 0.3 และ 0.7 ได้ หากไม่พบข้อมูลที่เกี่ยวข้องในการพยายามครั้งแรก ซึ่งเป็นกลไกที่ช่วยเพิ่มโอกาสในการค้นพบข้อมูลที่เกี่ยวข้องมากขึ้น โดยเฉพาะในกรณีที่ผู้ใช้อาจใช้คำศัพท์ที่แตกต่างจากที่ปรากฏในเอกสาร

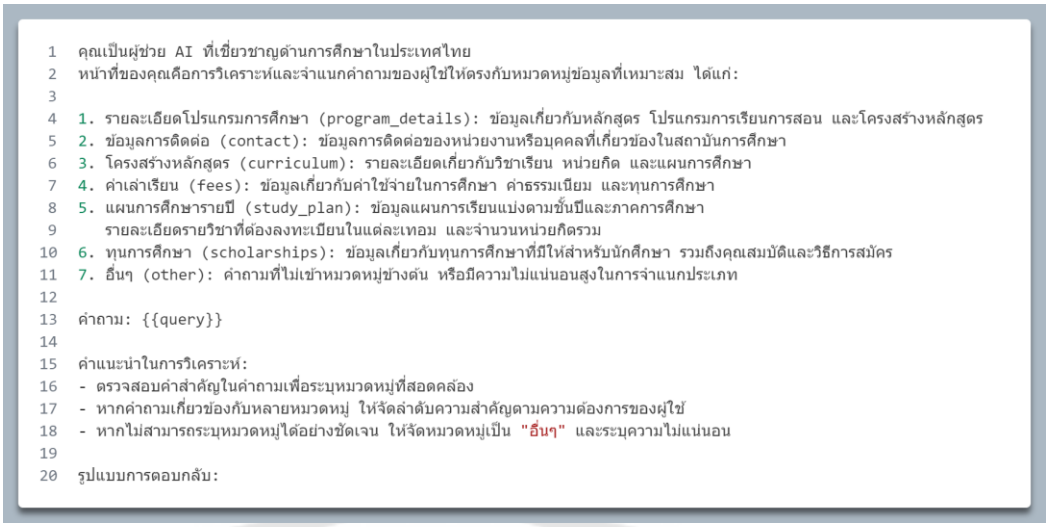
อย่างมาก หรือในกรณีที่ข้อมูลมีความกำกวมสูง กลไกนี้จะช่วยให้ระบบสามารถปรับตัวและตอบสนองต่อความต้องการของผู้ใช้ได้อย่างมีประสิทธิภาพมากขึ้น

ในส่วนของการ Re-ranking ด้วย Cross-Encoder นั้น เป็นอีกหนึ่งกระบวนการที่ช่วยเพิ่มความแม่นยำใน Retrieval โดย Cross-Encoder ทำงานแตกต่างจาก Bi-Encoder ที่ใช้ใน Retrieval เชิงความหมายทั่วไป มีลักษณะสำคัญคือการประมวลผลคู่ข้อความพร้อมกัน ซึ่ง Cross-Encoder จะรับคู่ข้อความ คำถามและเอกสาร เป็นอินพุตพร้อมกัน ทำให้สามารถเรียนรู้ความสัมพันธ์ระหว่างข้อความได้โดยตรง มีการใช้กลไก Attention ข้ามข้อความที่สามารถให้ความสนใจกับส่วนที่เกี่ยวข้องระหว่างคำถามและเอกสารได้อย่างมีประสิทธิภาพ และมีความแม่นยำสูงแต่ประสิทธิภาพต่ำกว่า Bi-Encoder ในแง่ของการคำนวณ จึงเหมาะสำหรับการ Re-ranking กับเอกสารจำนวนไม่มากที่ได้รับการคัดเลือกจากระบบ Retrieval เบื้องต้นแล้ว

กระบวนการ Re-ranking นี้ใช้วิธีการค้นหาแบบ Two-Stage Retrieval โดยขั้นตอนแรกจะใช้ Hybrid Retrieval เพื่อดึงเอกสารที่เกี่ยวข้องจำนวนมากขึ้น จากนั้นในขั้นตอนที่สองจะใช้ Cross-Encoder เพื่อ Re-ranking และเลือกเฉพาะเอกสารที่มีความเกี่ยวข้องสูงสุด เพื่อนำเสนอต่อผู้ใช้ ซึ่งวิธีการนี้ช่วยให้ระบบสามารถรวมจุดแข็งของทั้งสองวิธีเข้าด้วยกัน โดยใช้ประโยชน์จากประสิทธิภาพในการคำนวณของ Bi-Encoder ในการคัดกรองเอกสารเบื้องต้น และใช้ความแม่นยำของ Cross-Encoder ในการ Re-ranking เพื่อให้ได้ผลลัพธ์สุดท้ายที่มีความแม่นยำสูงสุด ทำให้ระบบสามารถตอบสนองต่อความต้องการของผู้ใช้ได้อย่างมีประสิทธิภาพ ทั้งในแง่ของความรวดเร็วและความแม่นยำใน Retrieval ข้อมูล

4.3 Advanced Query Processing

การประมวลผลคำถามขั้นสูงถือเป็นองค์ประกอบที่มีความสำคัญอย่างยิ่งต่อการพัฒนาระบบโต้ตอบอัตโนมัติที่มีประสิทธิภาพ โดยมีวัตถุประสงค์หลักในการตีความความตั้งใจของผู้ใช้และการแปลงคำถามให้อยู่ในรูปแบบที่เหมาะสมสำหรับกระบวน Retrieval สารสนเทศต่อไป ระบบได้นำเสนอกลไกการประมวลผลคำถามที่ซับซ้อนซึ่งสามารถวิเคราะห์โครงสร้างทางภาษาศาสตร์และบริบททางความหมายของคำถามได้อย่างมีประสิทธิภาพ โดยโครงสร้างของระบบประมวลผลคำถามได้รับการพัฒนาผ่านคลาส AdvancedQueryProcessor ซึ่งประกอบด้วยองค์ประกอบหลักที่ทำงานประสานกันอย่างเป็นระบบ ซึ่งประกอบด้วยองค์ประกอบหลักดังนี้:



```
คุณเป็นผู้ช่วย AI ที่เชี่ยวชาญด้านการศึกษาในประเทศไทย
หน้าที่ของคุณคือการวิเคราะห์และจำแนกคำถามของผู้ใช้ให้ตรงกับหมวดหมู่ข้อมูลที่เหมาะสม ได้แก่:

1. รายละเอียดโปรแกรมการศึกษา (program_details): ข้อมูลเกี่ยวกับหลักสูตร โปรแกรมการเรียนการสอน และโครงสร้างหลักสูตร
2. ข้อมูลการติดต่อ (contact): ข้อมูลการติดต่อของหน่วยงานหรือบุคคลที่เกี่ยวข้องในสถาบันการศึกษา
3. โครงสร้างหลักสูตร (curriculum): รายละเอียดเกี่ยวกับวิชาเรียน หน่วยกิต และแผนการศึกษา
4. ค่าเล่าเรียน (fees): ข้อมูลเกี่ยวกับค่าใช้จ่ายในการศึกษา ค่าธรรมเนียม และทุนการศึกษา
5. แผนการศึกษารายปี (study_plan): ข้อมูลแผนการเรียนแบ่งตามชั้นปีและภาคการศึกษา
   รายละเอียดรายวิชาที่ต้องลงทะเบียนในแต่ละเทอม และจำนวนหน่วยกิตรวม
6. ทุนการศึกษา (scholarships): ข้อมูลเกี่ยวกับทุนการศึกษาที่มีให้สำหรับนักศึกษา รวมถึงคุณสมบัติและวิธีการสมัคร
7. อื่นๆ (other): คำถามที่ไม่เข้าหมวดหมู่ข้างต้น หรือมีความไม่แน่นอนสูงในการจำแนกประเภท

คำถาม: {{query}}

คำแนะนำในการวิเคราะห์:
- ตรวจสอบคำสำคัญในคำถามเพื่อระบุหมวดหมู่ที่สอดคล้อง
- หากคำถามเกี่ยวข้องกับหลายหมวดหมู่ ให้จัดลำดับความสำคัญตามความต้องการของผู้ใช้
- หากไม่สามารถระบุหมวดหมู่ได้อย่างชัดเจน ให้จัดหมวดหมู่เป็น "อื่นๆ" และระบุความไม่แน่นอน

รูปแบบการตอบกลับ:
```

ภาพประกอบ 37 โครงสร้างหมวดหมู่การค้นหาข้อมูลโปรแกรมการศึกษา

คลาสนี้ได้ผสานเทคโนโลยี LLM เข้ากับกระบวนการศึกษาและบริบทด้านการศึกษา โดยมีการกำหนดโครงสร้างพื้นฐานที่ชัดเจนสำหรับการชี้นำโมเดลให้วิเคราะห์และจำแนกคำถามตามหมวดหมู่ที่ได้กำหนดไว้อย่างเป็นระบบ ซึ่งการวิเคราะห์คำถามเชิงภาษาศาสตร์ในระบบนี้ โดยที่ดำเนินการผ่านกระบวนการทางภาษาศาสตร์เชิงคำนวณที่มีประสิทธิภาพ ซึ่งประกอบด้วยกลไกสำคัญสองประการ ได้แก่ Query Normalization ซึ่งใช้นิพจน์ปกติในการระบุและแปลงรูปแบบที่หลากหลายของการอ้างอิง โดยเฉพาะอย่างยิ่งเกี่ยวกับปีการศึกษาและภาคเรียนให้เป็นมาตรฐานเดียวกัน และ Synonym Management ที่แก้ไขปัญหาความกำกวมทางภาษาด้วยการแทนที่คำที่มีความหมายเหมือนกันด้วยคำมาตรฐาน ทั้งสองกระบวนการนี้เป็นวิธีการแก้ไขปัญหาความไม่สอดคล้องกันของศัพท์ซึ่งเป็นอุปสรรคสำคัญใน Information Retrieval Systems

นอกจากนี้ ระบบยังได้รับการออกแบบให้รองรับ Natural Language Understanding NLU เพื่อวิเคราะห์โครงสร้างและความหมายของคำถามอย่างมีประสิทธิภาพ อันนำไปสู่ความแม่นยำของผลลัพธ์ที่สูงขึ้น องค์ประกอบสำคัญของระบบประมวลผลคำถามคือการประยุกต์ใช้ LLM ร่วมกับเทมเพลตที่กำหนดไว้อย่างเป็นระบบเพื่อการวิเคราะห์และจำแนกคำถาม โดยมีการกำหนดหมวดหมู่คำถาม อย่างชัดเจนเป็น 6 ประเภท ได้แก่ 1). ปฏิทินวิชาการและกำหนดการสำคัญ Deadline, Academic 2). กระบวนการรับสมัครและคุณสมบัติของผู้สมัคร program_details 3). ช่องทางการติดต่อและสถานที่ตั้ง contact 4). โครงสร้างหลักสูตรและรายละเอียดรายวิชา curriculum 5). แผนการเรียนและการศึกษา study_plan 6). อัตราค่าธรรมเนียมการศึกษาและชำระล่าช้า fees 7). ทุนการศึกษาของสาขา scholarship ซึ่งการ

จำแนกประเภทนี้เป็นกลไกสำคัญที่ช่วยให้ระบบสามารถเลือกใช้กลยุทธ์ Retrieval ที่เหมาะสมกับประเภทของข้อมูลที่ผู้ใช้ต้องการ

ระบบยังมี Keyword Extraction and Parameter Identification ที่ช่วยระบุองค์ประกอบสำคัญในคำถาม เช่น ปีการศึกษา ภาคเรียน หรือประเภทย่อยของข้อมูล ทำให้สามารถนำเสนอผลลัพธ์ที่เฉพาะเจาะจงตรงตามความต้องการของผู้ใช้ ภายใต้บริบทการศึกษาที่มีความซับซ้อนและหลากหลายได้อย่างมีประสิทธิภาพ กระบวนการทั้งหมดนี้ดำเนินการผ่าน Semantic Analysis ที่ช่วยให้ระบบเข้าใจเจตนาและบริบทของคำถามได้อย่างลึกซึ้ง ในการจัดการกับความไม่แน่นอนในการวิเคราะห์คำถาม ระบบได้พัฒนากลไกสำหรับการจัดการกับความไม่แน่นอนในการวิเคราะห์คำถามภาษาธรรมชาติอย่างเป็นระบบ โดยมี Uncertainty Assessment ซึ่งเป็นตัวแปรสำคัญในการตัดสินใจเกี่ยวกับกลยุทธ์ Retrieval ข้อมูล ระหว่าง Semantic Search และ Keyword Search

ระบบยังได้ออกแบบฟังก์ชัน Default Analysis Mechanism ดังแสดงในฟังก์ชัน get_default_analysis โดยที่จะทำงานในกรณีที่การวิเคราะห์ไม่สามารถดำเนินการได้หรือเกิดข้อผิดพลาด โดยใช้โครงสร้างข้อมูลพื้นฐานที่ประกอบด้วยคำถามต้นฉบับ ประเภทเหตุการณ์ ภาคการศึกษา คำสำคัญ และรูปแบบการตอบสนอง ซึ่งเป็นการเพิ่มความทนทานของระบบ อีกทั้งยังมีการวิเคราะห์เชิงบริบทและการปรับปรุงด้วยกฎเฉพาะทางที่พัฒนาจากความเข้าใจในบริบทการศึกษา แสดงให้เห็นถึงการผสมผสานอย่างลงตัวระหว่างความสามารถของ LLM และความรู้เฉพาะทางที่ได้รับการออกแบบโดยผู้เชี่ยวชาญ ส่งผลให้ระบบมีความแม่นยำสูงในการวิเคราะห์คำถามที่มีความเฉพาะเจาะจงในบริบททางการศึกษา

```python
def _get_default_analysis(self, query: str) -> Dict[str, Any]:
    logger.info("Returning default analysis")
    return {
        "original_query": query,
        "event_type": None,
        "semester": None,
        "key_terms": [],
        "response_format": "detailed"
    }
```

ภาพประกอบ 38 ฟังก์ชันการสร้างโครงสร้างการวิเคราะห์เริ่มต้นสำหรับคำค้น

Advanced Query Processing เป็นกลไกสำคัญที่ทำหน้าที่เป็นตัวกลางในการแปลงความตั้งใจของผู้ใช้ให้อยู่ในรูปแบบที่ระบบสามารถประมวลผลได้อย่างมีประสิทธิภาพ โดยผลลัพธ์ของกระบวนการนี้จะถูกนำไปใช้ในการกำหนดพารามิเตอร์หลัก 5 ประการสำหรับ

Retrieval ข้อมูล ได้แก่ event_type ที่ใช้ในการกรองเอกสารตามประเภท, detail_type ที่ระบุประเภทย่อยของข้อมูลที่ต้องการ, semester ที่ใช้ในการกรองข้อมูลตามช่วงเวลา, key_terms ที่ใช้ในการเน้นประเด็นสำคัญในการค้นหา และ response_format ที่กำหนดลักษณะของการนำเสนอข้อมูลที่ Retrieval ได้ ความแม่นยำในการจัดหาพารามิเตอร์เหล่านี้มีความสำคัญอย่างยิ่งต่อประสิทธิภาพของระบบการสร้างคำตอบโดยการเสริม Retrieval RAG โดยส่งผลโดยตรงต่อความถูกต้องและความเกี่ยวข้องของสารสนเทศที่ระบบนำเสนอให้แก่ผู้ใช้ นอกจากนี้ ยังมีส่วนช่วยในการปรับแต่งประสบการณ์การใช้งานให้สอดคล้องกับบริบทเฉพาะทางการศึกษา ซึ่งมีความซับซ้อนและต้องการความเฉพาะเจาะจงสูง เพื่อให้การโต้ตอบระหว่างผู้ใช้และระบบเป็นไปอย่างมีประสิทธิภาพและตอบสนองความต้องการได้อย่างตรงประเด็น

4.4 Response Generation using LLM

การพัฒนากลไกการสร้างการตอบสนองในระบบ AI เพื่อการโต้ตอบเชิงวิชาการนี้ อาศัยการประยุกต์ใช้ LLM (29) ซึ่งจะได้รับการปรับแต่งอย่างเฉพาะเจาะจงเพื่อตอบสนองต่อบริบทการศึกษาในประเทศไทย โดยการออกแบบโครงสร้างของกลไกการสร้างการตอบสนองได้รับการพัฒนาบนพื้นฐานของคลาส ResponseGenerator ซึ่งมีความสามารถในการใช้ประโยชน์จากโมเดลภาษา OpenAI ผ่านทางอินเตอร์เฟซ API ที่มีการกำหนดค่าพารามิเตอร์จาก PipelineConfig ซึ่งรวมถึงการเลือกรุ่นของโมเดลที่เหมาะสมกับภาระงาน โดยมีเทมเพลตการแสดงประวัติการสนทนาระหว่างผู้ใช้และที่ปรึกษาเป็นส่วนสำคัญที่ช่วยให้ระบบสามารถรักษาความต่อเนื่องของการสนทนาได้อย่างมีประสิทธิภาพ

```
คุณเป็นที่ปรึกษาทางวิชาการ กรุณาตอบคำถามต่อไปนี้โดยใช้ข้อมูลจากเอกสารที่ให้มาและพิจารณาบริบทจากประวัติการสนทนา

{% if conversation_history %}
ประวัติการสนทนา:
{% for message in conversation_history %}
{% if message.role == 'user' %}
ผู้ใช้: {{ message.content }}
{% else %}
ที่ปรึกษา: {{ message.content }}
{% endif %}
{% endfor %}
{% endif %}

คำถามปัจจุบัน: {{query}}

ข้อมูลที่เกี่ยวข้อง:
{% for doc in context %}
---
ประเภท: {{doc.meta.event_type}}{% if doc.meta.detail_type %}, รายละเอียด: {{doc.meta.detail_type}}{% endif %}
เนื้อหา:
{{doc.content}}
{% endfor %}
```

ภาพประกอบ 39 เทมเพลตการแสดงประวัติการสนทนาระหว่างผู้ใช้และที่ปรึกษา

Prompt Engineering เป็นองค์ประกอบที่มีความสำคัญอย่างยิ่งในการกำหนดทิศทางและคุณภาพของการตอบสนองที่ได้จากโมเดลภาษา โดยระบบนี้ได้พัฒนาเทคนิค Prompt Engineering ที่มีโครงสร้างเฉพาะประกอบด้วยองค์ประกอบสำคัญสี่ประการ ได้แก่ Role Definition ที่ระบุให้โมเดลทำหน้าที่เป็น "ที่ปรึกษาทางวิชาการ" เพื่อกำหนดขอบเขตและลักษณะของการตอบสนองให้เหมาะสมกับบริบทการศึกษา, Conversation History Presentation ที่จัดโครงสร้างการสนทนาก่อนหน้าอย่างเป็นระบบโดยแยกแยะข้อความของผู้ใช้และระบบอย่างชัดเจนเพื่อสร้างความต่อเนื่องในการสนทนา, Relevant Information Presentation ที่จัดระเบียบข้อมูลที่ Retrieval ได้พร้อมด้วยเมทาดาต้าที่ช่วยให้โมเดลเข้าใจประเภทและบริบทของข้อมูลได้อย่างถูกต้อง และ Response Guidelines ที่กำหนดกรอบและวิธีการตอบสนองอย่างมีประสิทธิภาพ รวมถึงวิธีการจัดการกับข้อมูลที่ไม่ครบถ้วนหรือไม่ชัดเจน การออกแบบ Prompt Engineering อย่างพิถีพิถันนี้ส่งผลให้ระบบสามารถให้คำตอบที่มีความถูกต้อง เที่ยงตรง และสอดคล้องกับความต้องการของผู้ใช้ในบริบทการศึกษาได้อย่างมีประสิทธิภาพ

ฟังก์ชัน generate_response ในคลาส ResponseGenerator รับผิดชอบในการสร้างการตอบสนองโดยอาศัยการประมวลผลและบูรณาการข้อมูลจากหลายแหล่ง โดยกระบวนการนี้ประกอบด้วยขั้นตอนที่สำคัญหลายประการ เริ่มจากการวิเคราะห์ลักษณะของ ที่ระบบวิเคราะห์คำถามเพื่อระบุว่าเป็นคำถามที่มีการอ้างอิงถึงบทสนทนาก่อนหน้าหรือไม่ โดยใช้การตรวจจับคำสำคัญที่บ่งชี้การอ้างอิง เช่น "ก่อนหน้านี้" หรือ "คำถามที่แล้ว" ต่อมาคือ Context Preparation ที่ระบบรวบรวมบริบทการสนทนาที่จำเป็นสำหรับการตอบคำถาม โดยเฉพาะอย่างยิ่งเมื่อคำถามมีการอ้างอิงถึงบทสนทนาก่อนหน้า จากนั้นเป็นการสร้าง Dynamic Prompt Construction ที่ระบบสร้าง Prompt Engineering ที่มีการปรับเปลี่ยนตามลักษณะของคำถามและบริบทการสนทนา ผ่านการเรียกใช้ prompt_builder.runLaun & Wolff, 2025 ซึ่งรวมข้อมูลที่เกี่ยวข้องทั้งหมด ขั้นตอนต่อไปคือการเรียกใช้โมเดลภาษา LLM Invocation ที่ระบบส่ง Prompt Engineering ที่สร้างขึ้นไปยังโมเดลภาษาผ่าน generator.runLaun & Wolff, 2025 ซึ่งจะประมวลผลและสร้างการตอบสนองที่เหมาะสม และสุดท้ายคือการจัดการข้อผิดพลาด ที่ระบบมีกลไกในการตรวจจับและจัดการกับข้อผิดพลาดที่อาจเกิดขึ้นในกระบวนการสร้างการตอบสนอง โดยให้ข้อความที่เหมาะสมแก่ผู้ใช้

ความสามารถในการจัดการกับคำถามที่มีการอ้างอิงถึงบทสนทนาก่อนหน้า เป็นคุณลักษณะพิเศษของระบบนี้ ซึ่งช่วยให้การสนทนามีความต่อเนื่องและเป็นธรรมชาติ โดยเริ่มจากการตรวจจับคำถามที่มีการอ้างอิงด้วยการค้นหาคำสำคัญที่บ่งชี้การอ้างอิง ทั้งในภาษาไทยและภาษาอังกฤษ เช่น "ก่อนหน้านี้", "ที่ผ่านมา", "previous", "earlier" เป็นต้น เมื่อพบว่าเป็นคำถามที่มีการอ้างอิง ระบบจะเพิ่มการระบุและให้ความสำคัญกับบริบทการสนทนาก่อนหน้าใน Prompt Engineering (30) นอกจากนี้ ระบบยังได้รับการออกแบบให้สามารถตีความคำถามและแก้ไขการอ้างอิงที่คลุมเครือโดยพิจารณาจากบริบทการสนทนาที่ผ่านมา อันนำไปสู่การตอบสนองที่มีความต่อเนื่องกับบทสนทนาก่อนหน้า โดยจะไม่จำเป็นต้องทวนซ้ำคำถามหรือคำตอบที่ผ่านมา แต่ยังคงให้ข้อมูลที่สมบูรณ์และเข้าใจได้อย่างชัดเจน

ระบบได้พัฒนากลไก Response Formatting ที่มีความยืดหยุ่นและเหมาะสมกับบริบทการใช้งาน โดยประกอบด้วยองค์ประกอบสำคัญห้าประการ ได้แก่ Verbosity Control ที่สามารถปรับเปลี่ยนตามค่า response_format ระหว่าง "detailed" สำหรับคำตอบที่มีรายละเอียดครบถ้วนกับ "concise" สำหรับคำตอบที่กระชับตรงประเด็น การจัดรูปแบบตามประเภทข้อมูล ที่ปรับการนำเสนอให้เหมาะสมกับประเภทของข้อมูล เช่น ปฏิทินวิชาการและกำหนดการสำคัญ, กระบวนการรับสมัครและคุณสมบัติของผู้สมัคร, ช่องทางการติดต่อและสถานที่ตั้ง, โครงสร้างหลักสูตรและรายละเอียดรายวิชา, แผนการเรียนและการศึกษา, อัตราค่าธรรมเนียมการศึกษาและชำระล่าช้า, ทุนการศึกษาของสาขา การจัดลำดับความสำคัญของข้อมูล ที่นำเสนอข้อมูลที่ตรงกับคำถามมากที่สุดก่อน การเสริมข้อมูลเพิ่มเติม สำหรับหัวข้อเฉพาะทางเพื่อความสมบูรณ์ของคำตอบ และการจัดรูปแบบที่อ่านง่าย ที่ใช้หัวข้อ ย่อหน้า และการจัดวางที่เหมาะสมเพื่อสร้างความเข้าใจที่ชัดเจน นอกจากนี้ ระบบยังรองรับการปรับแต่งตามอุปกรณ์ที่ใช้งาน และมีกลไกการแก้ไขความคลุมเครือ ในกรณีที่คำถามมีความไม่ชัดเจน ทำให้การตอบสนองมีประสิทธิภาพสูงแม้ในสถานการณ์ที่มีความซับซ้อน

การรักษาคุณภาพของการตอบสนองเป็นสิ่งสำคัญในระบบโต้ตอบอัตโนมัติ ระบบนี้มีแนวทางในการวิเคราะห์และรักษาคุณภาพการตอบสนองหลายประการ โดยจะเริ่มจากการตรวจสอบความสอดคล้องกับข้อมูลที่ให้มา ที่ระบบออกแบบให้โมเดลสร้างการตอบสนองที่สอดคล้องกับข้อมูลที่ให้มาเท่านั้น โดยไม่สร้างข้อมูลที่ไม่มีในเอกสารอ้างอิง การรับมือกับความไม่ครบถ้วนของข้อมูล ที่หากข้อมูลที่ให้มาไม่ครบถ้วนหรือไม่ชัดเจน ระบบจะระบุข้อจำกัดนี้อย่างชัดเจนในการตอบสนอง การจัดการกับคำถามที่ไม่มีคำตอบ ที่สำหรับคำถามที่ไม่พบ

ข้อมูลที่เกี่ยวข้อง ระบบจะให้การตอบสนองที่สื่อถึงข้อจำกัดนี้อย่างชัดเจนและสุภาพ การรักษาความต่อเนื่องของบทสนทนา ที่ระบบตรวจสอบว่าการตอบสนองมีความต่อเนื่องกับบทสนทนาที่ผ่านมา โดยเฉพาะอย่างยิ่งเมื่อคำถามมีการอ้างอิงถึงบทสนทนาก่อนหน้า และการประเมินคุณภาพการตอบสนอง ที่แม้ว่าจะไม่ได้ระบุไว้อย่างชัดเจนในโค้ด แต่ระบบอาจมีการประเมินคุณภาพการตอบสนองโดยใช้เกณฑ์เช่นความครบถ้วน ความถูกต้อง และความเกี่ยวข้องกับคำถาม ซึ่งกระบวนการทั้งหมดนี้มีส่วนสำคัญในการยกระดับคุณภาพของการปฏิสัมพันธ์ระหว่างผู้ใช้และระบบ AI ให้มีความเป็นธรรมชาติและมีประสิทธิภาพสูงในบริบทการศึกษา

5. System Performance Evaluation

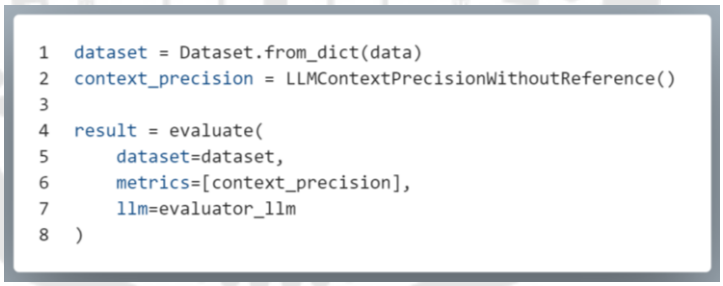
การศึกษาวิจัยนี้มุ่งพัฒนาระบบการจัดการองค์ความรู้สำหรับมหาวิทยาลัยที่มีศักยภาพในการรองรับกลุ่มผู้ใช้งานทั้งภายในและภายนอกมหาวิทยาลัย โดยประยุกต์ใช้สถาปัตยกรรมเทคโนโลยี RAG ดังแสดงในภาพประกอบ 5 แสดงองค์ประกอบของระบบและกระบวนการทำงาน ซึ่งกระบวนการทั้งหมดช่วยเพิ่มประสิทธิภาพการเข้าถึงและจัดการองค์ความรู้อย่างเป็นระบบ ลดปัญหา Hallucination และส่งเสริมคุณภาพการให้บริการข้อมูลที่ตอบสนองต่อความต้องการของผู้ใช้งานทั้งนี้ โดยระบบได้รับการประเมินประสิทธิภาพใน 2 มิติหลัก ได้แก่ 1). Retrieval Effectiveness ซึ่งเป็นการวัดความสามารถของระบบ RAG ในการค้นหาเอกสารที่เกี่ยวข้องและตรงประเด็นกับคำถามของผู้ใช้ 2). Response Quality ที่ประเมินคุณภาพของคำตอบที่ระบบสร้างขึ้นในหลายมิติ ได้แก่ ความถูกต้องของข้อมูล ความสมบูรณ์ของเนื้อหา ความสอดคล้องกับคำถาม และระดับความพึงพอใจของผู้ใช้งาน

5.1 วิเคราะห์ประเภทคำถามสำหรับการทดสอบระบบสนทนาอัตโนมัติ

ในการประเมินประสิทธิภาพของระบบโต้ตอบอัตโนมัติ AI Chatbot สำหรับสถาบันการศึกษา ผู้วิจัยได้ดำเนินการวิเคราะห์และจำแนกประเภทคำถามในชุดข้อมูลทดสอบซึ่งประกอบด้วยคำถามจำนวน 65 รายการ คำถามในชุดทดสอบสามารถจำแนกได้เป็น 7 ประเภทหลัก ได้แก่: 1). คำถามเกี่ยวกับปฏิทินวิชาการและกำหนดการจำนวน 20 คำถาม ซึ่งเกี่ยวข้องกับวันที่สำคัญต่างๆ เช่น วันเปิด-ปิดภาคเรียน กำหนดการสอบ และกำหนดส่งเอกสาร 2). คำถามเกี่ยวกับโครงสร้างหลักสูตรและรายละเอียดรายวิชาจำนวน 12 คำถาม 3). คำถามเกี่ยวกับกระบวนการรับสมัครและคุณสมบัติของผู้สมัครจำนวน 10 คำถาม 4). คำถามเกี่ยวกับอัตราค่าธรรมเนียมการศึกษาและชำระล่าช้าจำนวน 3 คำถาม 5). คำถามเกี่ยวกับช่องทางการติดต่อและสถานที่ตั้งจำนวน 5 คำถาม 6). คำถามเกี่ยวกับแผนการเรียนและการศึกษา จำนวน 10 คำถาม และ 7).คำถามเกี่ยวกับทุนการศึกษาของสาขา จำนวน 5 คำถาม

5.2 Retrieval Evaluation

เป็นกระบวนการวิเคราะห์ความสามารถของระบบในการระบุและสกัดข้อมูลที่มีความเกี่ยวข้องและเป็นประโยชน์จากฐานความรู้ เพื่อนำมาใช้ในการสร้างคำตอบ โดยในงานวิจัยนี้ได้ประยุกต์ใช้เกณฑ์การประเมินแบบ LLMContextPrecisionWithoutReference ซึ่งเป็นวิธีการประเมินที่ไม่จำเป็นต้องอาศัยข้อมูลอ้างอิงที่จัดเตรียมโดยผู้เชี่ยวชาญ Reference-free Evaluation แต่ใช้ LLM เช่น ChatOpenAI GPT-4o ในการวิเคราะห์ว่าบริบทที่ถูก Retrieval มามีความสอดคล้องและเกี่ยวข้องกับคำถามมากน้อยเพียงใด โดยเมื่อระบบรับคำถามจากผู้ใช้ กระบวนการประเมินจะวิเคราะห์ความสัมพันธ์ระหว่างคำถามและเนื้อหาที่ Retrieval มาจากฐานข้อมูล Retrieved Context แล้วให้คะแนนที่สะท้อนถึงความแม่นยำของ Retrieval โดยคะแนนที่สูงจะแสดงถึงความสามารถของระบบในการค้นหาข้อมูลที่เกี่ยวข้องได้อย่างมีประสิทธิภาพ ซึ่งเป็นปัจจัยสำคัญที่จะส่งผลต่อคุณภาพของคำตอบที่ระบบสร้างขึ้น การประเมินในลักษณะนี้มีความสำคัญอย่างยิ่งในระบบ RAG เนื่องจากความถูกต้องและความเกี่ยวข้องของข้อมูลที่ Retrieval มาเป็นรากฐานสำคัญที่จะช่วยให้ระบบสามารถสร้างคำตอบที่มีคุณภาพถูกต้อง และตรงประเด็นกับคำถามของผู้ใช้



```
1  dataset = Dataset.from_dict(data)
2  context_precision = LLMContextPrecisionWithoutReference()
3
4  result = evaluate(
5      dataset=dataset,
6      metrics=[context_precision],
7      llm=evaluator_llm
8  )
```

ภาพประกอบ 40 การประเมินประสิทธิภาพของโมเดลภาษาด้วยเมทริก Context Precision

5.3 Response Quality Evaluation

เป็นกระบวนการวิเคราะห์ความสามารถของระบบในการสร้างคำตอบที่มีคุณภาพจากข้อมูลที่ Retrieval มา โดยพิจารณาองค์ประกอบสำคัญหลายประการ ได้แก่ ความเกี่ยวข้อง ความถูกต้อง และความสมบูรณ์ของคำตอบ ในงานวิจัยนี้ ผู้วิจัยได้ประยุกต์ใช้เกณฑ์การประเมินแบบ ResponseRelevancy ซึ่งเป็นเมตริกที่วัดระดับความเกี่ยวข้องของคำตอบกับคำถาม โดยวิเคราะห์ว่าเนื้อหาคำตอบที่ระบบสร้างขึ้นนั้นตอบสนองต่อความต้องการของคำถามได้ตรงประเด็นและครบถ้วนมากน้อยเพียงใด รวมถึงพิจารณาว่ามีเนื้อหาที่ไม่เกี่ยวข้องหรือเกินความจำเป็นหรือไม่ กระบวนการประเมินดำเนินการโดยใช้ LLM เช่นเดียวกับการประเมินประสิทธิภาพ Retrieval โดยวิเคราะห์ความสัมพันธ์ระหว่างคำถาม question คำตอบที่สร้างขึ้น answer และบริบทที่ Retrieval มา contexts จากนั้นให้คะแนนที่สะท้อนถึงคุณภาพของการตอบสนอง โดยค่า

คะแนนที่สูงบ่งชี้ว่าระบบสามารถสร้างคำตอบที่มีความเกี่ยวข้อง ตรงประเด็น และมีคุณภาพสูง การประเมินในมิตินี้มีความสำคัญอย่างยิ่งเนื่องจากแม้ระบบจะสามารถ Retrieval ข้อมูลที่เกี่ยวข้องได้อย่างมีประสิทธิภาพ แต่หากไม่สามารถนำข้อมูลเหล่านั้นมาสังเคราะห์เป็นคำตอบที่มีคุณภาพได้ ก็จะไม่สามารถตอบสนองความต้องการของผู้ใช้ได้อย่างมีประสิทธิภาพ

```python
dataset = Dataset.from_dict(data)
context_precision = ResponseRelevancy()

result = evaluate(
    dataset=dataset,
    metrics=[context_precision],
    llm=evaluator_llm
)
```

ภาพประกอบ 41 การประเมินความเกี่ยวข้องของคำตอบจากโมเดลภาษา

5.4 BERTScore Evaluation

เป็นกระบวนการประเมินคุณภาพการตอบสนองของระบบด้วยวิธีการวัดเชิงความหมาย (semantic measurement) ที่มีความละเอียดและแม่นยำสูง โดยใช้โมเดลภาษาขนาดใหญ่อย่าง BERT ในการวิเคราะห์ความคล้ายคลึงระหว่างคำตอบที่ระบบสร้างขึ้นกับคำตอบอ้างอิงที่ถูกต้อง ในงานวิจัยนี้ ผู้วิจัยได้ประยุกต์ใช้ BERTScore ซึ่งประกอบด้วยเมตริกสำคัญสามส่วน ได้แก่ **BERTScore Precision** ที่วัดระดับความแม่นยำของเนื้อหาในคำตอบที่ระบบสร้างขึ้นเมื่อเทียบกับคำตอบอ้างอิง โดยพิจารณาว่าแต่ละส่วนของคำตอบนั้นตรงกับข้อมูลในคำตอบอ้างอิงมากน้อยเพียงใด **BERTScore Recall** ที่วัดความครอบคลุมของเนื้อหา โดยวิเคราะห์ว่าคำตอบที่ระบบสร้างขึ้นสามารถดึงข้อมูลสำคัญจากคำตอบอ้างอิงได้ครบถ้วนหรือไม่ และ **BERTScore F1** ซึ่งเป็นค่าเฉลี่ยแบบถ่วงน้ำหนักระหว่าง Precision และ Recall เพื่อให้ได้ตัวชี้วัดประสิทธิภาพโดยรวมของระบบ การประเมินด้วย BERTScore มีข้อได้เปรียบกว่าวิธีการวัดแบบดั้งเดิมเนื่องจากสามารถจับความคล้ายคลึงในระดับความหมาย (semantic similarity) ไม่ใช่เพียงแค่การตรงกันของคำหรือวลี ทำให้การประเมินมีความลึกซึ้งและสะท้อนถึงคุณภาพของคำตอบได้อย่างสมจริงมากขึ้น ซึ่งมีความสำคัญอย่างยิ่งในระบบ RAG ที่มุ่งเน้นการตอบคำถามที่ซับซ้อนและต้องการความเข้าใจในระดับความหมายของบริบทที่หลากหลาย

```python
from bert_score import score

hypotheses = ["ภาคเรียนที่ 1/2567 เปิดเรียนในวันจันทร์ที่ 19 สิงหาคม 2567"]
references = ["ภาคเรียนที่ 1/2567 เปิดเรียนวันจันทร์ที่ 19 สิงหาคม 2567"]

P, R, F1 = score(hypotheses, references, lang="th", model_type="bert-base-multilingual-cased", verbose=True)

print(f"BERTScore Precision: {P.mean():.4f}")
print(f"BERTScore Recall: {R.mean():.4f}")
print(f"BERTScore F1: {F1.mean():.4f}")
```

ภาพประกอบ 42 การประเมินค่า **BERTScore** ของคำตอบจากโมเดลภาษา

บทที่ 4
ผลลัพธ์จากการวิจัย

บทนี้นำเสนอผลการวิจัยจากการพัฒนาระบบโต้ตอบอัตโนมัติสำหรับข้อมูลทางวิชาการโดยใช้เทคนิค RAG ซึ่งมุ่งเน้นการประเมินประสิทธิภาพของระบบที่พัฒนาขึ้นในสองมิติหลัก ได้แก่ Retrieval Effectiveness และ Response Quality การประเมินดังกล่าวมีวัตถุประสงค์เพื่อตรวจสอบความสามารถของระบบในการระบุข้อมูลที่เกี่ยวข้องจากฐานความรู้ และความสามารถในการสร้างคำตอบที่มีความถูกต้อง ครบถ้วน และตรงประเด็นตามความต้องการของผู้ใช้

กรอบการประเมินที่นำมาใช้ในงานวิจัยนี้คือ Ragas Retrieval Augmented Generation Assessment ซึ่งเป็นเครื่องมือมาตรฐานที่ได้รับการยอมรับในวงการวิจัยด้านระบบ RAG สำหรับการประเมินแบบหลายมิติที่ครอบคลุมทั้งความแม่นยำของข้อมูลที่ Retrieval และคุณภาพของคำตอบที่สร้างขึ้น บรรลุวัตถุประสงค์ของงานวิจัยได้กำหนดดังนี้ 1). ชุดข้อมูลทดสอบ Test Dataset 2). Retrieval Evaluation 3). Response Quality Evaluation

1. ชุดข้อมูลทดสอบ Test Dataset

การประเมินประสิทธิภาพของระบบได้ดำเนินการบนชุดข้อมูลทดสอบที่ครอบคลุมประเภทคำถามที่หลากหลายเกี่ยวกับข้อมูลทางวิชาการ โดยเฉพาะข้อมูลปฏิทินการศึกษา โครงสร้างหลักสูตรและรายละเอียดรายวิชา และค่าธรรมเนียมการศึกษา ชุดข้อมูลทดสอบประกอบด้วยคำถาม คำตอบมาตรฐาน และบริบทที่เกี่ยวข้องที่ Retrieval จากฐานความรู้ โดยตัวอย่างของคำถามขอแต่ละประเภทจะมีดังนี้ 1). ปฏิทินวิชาการและกำหนดการสำคัญ 2). กระบวนการรับสมัครและคุณสมบัติของผู้สมัคร 3). ช่องทางการติดต่อและสถานที่ตั้ง 4). โครงสร้างหลักสูตรและรายละเอียดรายวิชา 5). แผนการเรียนและการศึกษา 6). อัตราค่าธรรมเนียมการศึกษาและชำระล่าช้า 7).ทุนการศึกษาของสาขา โดยคำถามเกี่ยวกับปฏิทินการศึกษาและกำหนดการ จะมีตัวอย่างดังนี้

```
ภาคเรียนที่ 1/2567 เปิดเรียนวันไหน?,
วันสุดท้ายของการลงทะเบียนเรียนและชำระเงินค่าธรรมเนียม ภาค 1/2567 คือวันไหน?,
กำหนดการสอบปากเปล่าปริญญานิพนธ์ ภาค 1/2567 คือช่วงไหน?,
วันสุดท้ายของการขอถอนรายวิชาเรียน ภาค 1/2567 คือวันไหน?,
วันหยุดชดเชยวันอาสาฬหบูชาคือวันไหน?,
วันสุดท้ายที่อาจารย์ต้องส่งผลการสอบ ภาค 1/2567 คือวันไหน?,
กำหนดการยื่นขอแต่งตั้งอาจารย์ที่ปรึกษา (บว.410) ภาคปลายคือวันไหน?,
วันสุดท้ายของการสอบเค้าโครงปริญญานิพนธ์ ภาค 2/2567 คือวันไหน?,
วันหยุดเนื่องในวันปิยมหาราชคือวันไหน?,
```

ภาพประกอบ 43 ตัวอย่างชุดคำถามของปฏิทินการศึกษาและกำหนดการ

คำถามในกลุ่มนี้มุ่งเน้นการสอบถามข้อมูลเกี่ยวกับวันเวลาสำคัญในปฏิทินการศึกษา เช่น วันเปิดเรียนภาคเรียนที่ 1 ปีการศึกษา 2567 วันสุดท้ายของการลงทะเบียนเรียนภาคเรียนที่ 1/2567 และกำหนดการปฐมนิเทศนิสิตใหม่ รวมถึงรูปแบบการจัดกิจกรรม ซึ่งข้อมูลเหล่านี้มีความสำคัญอย่างยิ่งต่อการวางแผนการเรียนและการศึกษา และการเตรียมความพร้อมของนิสิต โดยเฉพาะนิสิตใหม่ที่ต้องการทราบกำหนดการต่างๆ เพื่อเข้าร่วมกิจกรรมได้อย่างครบถ้วน โดยคำถามเกี่ยวกับโครงสร้างหลักสูตรและรายละเอียดรายวิชาและรายวิชา จะมีตัวอย่างดังนี้

```
1   หลักสูตร MSDS 2567 แผน 2 แบบวิชาชีพ มีระยะเวลาเรียนกี่ปี?,
2   วิชา DS514 และ DS515 เรียนในปีที่เท่าไหร่ ภาคเรียนที่เท่าไหร่?,
3   วิชา DS518 มีจำนวนกี่หน่วยกิต?,
4   วิชา DS610 และ DS611 เรียนเกี่ยวกับอะไร?,
5   ในปีที่ 2 ภาคเรียนที่ 2 นิสิตต้องเรียนวิชาอะไรบ้าง?,
6   วิชา DS516 และ DS517 เรียนเกี่ยวกับอะไร?,
7   วิชา GRI682 เรียนในปีที่เท่าไหร่ ภาคเรียนที่เท่าไหร่?,
8   ในแต่ละภาคเรียนของปีที่ 1 นิสิตต้องเรียนวิชาหลักทั้งหมดกี่หน่วยกิต?,
9   วิชา DS510 มีจำนวนกี่หน่วยกิต?,
10  วิชา DS519 มีรูปแบบการเรียนเป็นแบบใด?,
```

ภาพประกอบ 44 ตัวอย่างชุดคำถามของโครงสร้างหลักสูตรและรายละเอียดรายวิชาและรายวิชา

คำถามประเภทนี้เกี่ยวข้องกับรายละเอียดโครงสร้างของหลักสูตร รายวิชาที่นิสิตต้องลงทะเบียนเรียนในภาคเรียนที่ 1 ปีที่ 1 รายวิชาเลือกที่นิสิตสามารถเลือกลงทะเบียนได้ตามความสนใจ และจำนวนหน่วยกิตของวิชาต่างๆ โดยเฉพาะวิชาการค้นคว้าอิสระซึ่งเป็นรายวิชาสำคัญสำหรับการสำเร็จการศึกษา ข้อมูลเหล่านี้ช่วยให้นิสิตสามารถวางแผนการเรียนตลอดหลักสูตรได้อย่างมีประสิทธิภาพ โดยคำถามเกี่ยวกับการสมัครและคุณสมบัติผู้สมัคร จะมีตัวอย่างดังนี้

```
1   หลักสูตร MSDS ปี 2567 เปิดรับผู้สำเร็จการศึกษาระดับปริญญาตรีสาขาใด?,
2   ช่องทางการสมัครหลักสูตร MSDS คือช่องทางไหน?,
3   อีเมลสำหรับติดต่อสอบถามข้อมูลเกี่ยวกับหลักสูตร MSDS คืออีเมลอะไร?,
4   ผู้สมัครหลักสูตร MSDS ต้องมีผลคะแนน TOEFL iBT เท่าไหร่จึงจะได้รับการยกเว้นไม่ต้องสอบวิชาภาษาอังกฤษ?,
5   เอกสารใดบ้างที่ผู้สมัครหลักสูตร MSDS ต้องยื่น?,
6   ผู้สมัครหลักสูตร MSDS ที่มีผลสอบ SWU-SET ระดับใด จึงจะได้รับการยกเว้นไม่ต้องสอบวิชาภาษาอังกฤษทั่วไป?,
7   ขั้นตอนการคัดเลือกผู้เข้าศึกษาหลักสูตร MSDS มีอะไรบ้าง?,
8   ผู้สมัครหลักสูตร MSDS ต้องยื่นเอกสารในรูปแบบไฟล์ใด?,
```

ภาพประกอบ 45 ตัวอย่างชุดคำถามของการสมัครและคุณสมบัติผู้สมัคร

คำถามในกลุ่มนี้มุ่งเน้นที่กระบวนการรับสมัครนิสิตใหม่ ประกอบด้วยคำถามเกี่ยวกับคุณสมบัติที่จำเป็นสำหรับผู้สมัครเข้าศึกษา เอกสารสำคัญที่ต้องใช้ประกอบการสมัคร และขั้นตอนวิธีการสมัครเข้าเรียนในหลักสูตร ซึ่งการให้ข้อมูลที่ถูกต้องและครบถ้วนในประเด็นเหล่านี้มีความสำคัญอย่างยิ่งในการช่วยให้ผู้สนใจสมัครเข้าศึกษาสามารถเตรียมตัวได้อย่างเหมาะสมและดำเนินการสมัครได้อย่างราบรื่น โดยคำถามเกี่ยวกับค่าธรรมเนียมการศึกษาและค่าปรับล่าช้า จะมีตัวอย่างดังนี้

```
1  ค่าเล่าเรียนหลักสูตร MSDS 2567 ต่อภาคการศึกษาคือเท่าไหร่?,
2  หากชำระค่าเล่าเรียนล่าช้า จะมีค่าปรับเท่าไหร่?,
3  ค่าปรับชำระล่าช้า มีเงื่อนไขอย่างไร?,
```

ภาพประกอบ 46 ตัวอย่างชุดคำถามของค่าธรรมเนียมการศึกษาและค่าปรับล่าช้า

คำถามประเภทนี้เกี่ยวข้องกับข้อมูลด้านการเงินที่นิสิตจำเป็นต้องทราบ โดยเฉพาะอย่างยิ่งอัตราค่าธรรมเนียมการศึกษาต่อภาคเรียนของหลักสูตร และบทลงโทษหรือค่าปรับกรณีที่มีการชำระค่าธรรมเนียมการศึกษาล่าช้า ข้อมูลเหล่านี้มีความสำคัญต่อการวางแผนการเงินของนิสิตและผู้ปกครอง ช่วยให้สามารถเตรียมค่าธรรมเนียมการศึกษาได้อย่างเหมาะสมและหลีกเลี่ยงค่าปรับที่อาจเกิดขึ้นจากการชำระเงินล่าช้า โดยคำถามเกี่ยวกับการติดต่อและสถานที่ จะมีตัวอย่างดังนี้

```
1  ภาควิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยศรีนครินทรวิโรฒ ประสานมิตร ตั้งอยู่ที่อาคารใด?,
2  หมายเลขโทรศัพท์สำหรับติดต่อภาควิชาวิทยาการคอมพิวเตอร์คือหมายเลขอะไร?,
3  หากต้องการเดินทางไปภาควิชาวิทยาการคอมพิวเตอร์ โดยใช้บริการเรือโดยสารคลองแสนแสบ ต้องลงที่ท่าเรือใด?,
4  Facebook Page ของหลักสูตร MSDS คือ Page ไหน?,
5  หากเดินทางโดย BTS ไปภาควิชาวิทยาการคอมพิวเตอร์ ควรลงสถานีใด?,
```

ภาพประกอบ 47 ตัวอย่างชุดคำถามของการติดต่อและสถานที่

คำถามในกลุ่มนี้เกี่ยวข้องกับช่องทางการติดต่อสื่อสารกับหลักสูตร โดยอย่างยิ่งเฉพาะอีเมลสำหรับติดต่อสอบถามข้อมูล รวมถึงสถานที่ตั้งของมหาวิทยาลัยและวิธีการเดินทางมายังมหาวิทยาลัย ข้อมูลเหล่านี้มีความจำเป็นสำหรับนิสิตในการติดต่อประสานงานกับหลักสูตรและการวางแผนการเดินทางมาเรียนหรือเข้าร่วมกิจกรรมต่างๆ ที่มหาวิทยาลัยจัดขึ้น โดยคำถามเกี่ยวกับทุนการศึกษาของสาขาจะมีตัวอย่างดังนี้

```
1  สมัครทุนการศึกษาต้องเตรียมเอกสารอะไรบ้าง และต้องส่งในรูปแบบไหน?,
2  กระบวนการคัดเลือกทุนการศึกษามีขั้นตอนอะไรบ้าง และสัมภาษณ์เมื่อไหร่?,
3  จำนวนเงินทุนรวมทั้งหมดเท่าไหร่?,
4  มี Manuscript แล้วแต่ยังไม่ได้รับการรับรองจากอาจารย์ที่ปรึกษา สมัครได้หรือไม่?,
5  ถ้าเกรดเฉลี่ยต่ำกว่า 2.50 จะสมัครได้หรือไม่?
```

ภาพประกอบ 48 ตัวอย่างชุดคำถามของทีนการศึกษาของสาขา

คำถามในกลุ่มนี้เกี่ยวข้องกับรายละเอียดของโครงการทุนการศึกษาต่างๆ ที่มีให้กับนิสิตระดับบัณฑิตศึกษา โดยเฉพาะเงินทุนสนับสนุนค่าครองชีพนิสิตภาควิชาวิทยาการคอมพิวเตอร์ ครอบคลุมข้อมูลเกี่ยวกับเอกสารที่ต้องเตรียมสำหรับการสมัครและรูปแบบการส่งเอกสาร กระบวนการคัดเลือกและกำหนดเวลาสัมภาษณ์ จำนวนเงินทุนและระยะเวลาการให้ทุน คุณสมบัติที่จำเป็นเช่นการมี Manuscript ที่ได้รับการรับรองจากอาจารย์ที่ปรึกษา และเกณฑ์เกรดเฉลี่ยสะสมขั้นต่ำ ข้อมูลเหล่านี้มีความจำเป็นสำหรับนิสิตในการประเมินโอกาสการได้รับทุนการศึกษา การเตรียมความพร้อมสำหรับการสมัคร และการวางแผนทางการเงินเพื่อสนับสนุนการศึกษาในระดับบัณฑิตศึกษา โดยแต่ละหัวข้อจะมีจำนวนคำถามดังนี้ ดังแสดงในตารางที่ 1

ประเภทข้อมูล	จำนวนคำถาม
1. คำถามเกี่ยวกับปฏิทินวิชาการและกำหนดการสำคัญ	20
2. คำถามเกี่ยวกับโครงสร้างหลักสูตรและรายละเอียดรายวิชา	12
3. คำถามเกี่ยวกับกระบวนการรับสมัครและคุณสมบัติของผู้สมัคร	10
4. คำถามเกี่ยวกับอัตราค่าธรรมเนียมการศึกษาและชำระล่าช้า	3
5. คำถามเกี่ยวกับช่องทางการติดต่อและสถานที่ตั้ง	5
6. คำถามเกี่ยวกับแผนการเรียนและการศึกษา	10
7. คำถามเกี่ยวกับทุนการศึกษาของสาขา	5

ตารางที่ 1 ตัวอย่างชุดข้อมูลทดสอบสำหรับการประเมินประสิทธิภาพของระบบ

จากตารางที่ 1 ซึ่งจะเห็นได้ว่าชุดข้อมูลทดสอบมีการกระจายตัวของคำถามที่ครอบคลุมหลากหลายประเด็นที่นิสิตและผู้สนใจมักสอบถาม โดยมีการเน้นคำถามเกี่ยวกับปฏิทินการศึกษาและโครงสร้างหลักสูตรเป็นหลัก สะท้อนให้เห็นถึงความสำคัญของข้อมูลดังกล่าวในการตัดสินใจของผู้สมัครและการวางแผนการเรียนและการศึกษา นิสิต การจัดทำชุดข้อมูลทดสอบที่ครอบคลุมและมีความ

หลากหลายนี้ช่วยให้การประเมินประสิทธิภาพของระบบมีความน่าเชื่อถือและสามารถสะท้อนความสามารถในการตอบคำถามในสถานการณ์จริงได้อย่างแม่นยำ

ชุดข้อมูลทดสอบได้รับการออกแบบให้ครอบคลุมลักษณะคำถามที่หลากหลาย เพื่อให้สามารถประเมินความสามารถของระบบใน Retrieval ข้อมูลและสร้างคำตอบได้อย่างรอบด้าน โดยแต่ละคำถามมีองค์ประกอบที่สำคัญสามประการ ได้แก่ Question ซึ่งเป็นข้อสอบถามที่ผู้ใช้ต้องการทราบเกี่ยวกับข้อมูลทางวิชาการ คำตอบมาตรฐาน ซึ่งเป็นข้อมูลที่ถูกต้องและเป็นทางการสำหรับคำถามนั้นๆ และ Retrieved Contexts ซึ่งประกอบด้วยข้อมูลที่ระบบ Retrieval จากฐานความรู้เพื่อนำมาใช้ในการสร้างคำตอบที่ตรงประเด็น โดยการประเมินประสิทธิภาพของระบบได้ดำเนินการบนชุดข้อมูลทดสอบที่ประกอบด้วยคำถามหลากหลายประเภท โดยเฉพาะข้อมูลปฏิทินการศึกษา โครงสร้างหลักสูตรและรายละเอียดรายวิชา และค่าธรรมเนียมการศึกษา ดังตัวอย่างในภาพประกอบ 44

```
"question": [
    "ภาคเรียนที่ 1/2567 เปิดเรียนวันไหน?",
],
"answer": [
    "ภาคเรียนที่ 1/2567 เปิดเรียนในวันจันทร์ที่ 19 สิงหาคม 2567",
],
"contexts": [
    ["ภาคเรียนที่ 1/2567 เปิดเรียนวันจันทร์ที่ 19 สิงหาคม 2567"],
]
```

ภาพประกอบ 49 ตัวอย่างข้อมูลทดสอบ แสดงความสัมพันธ์ของคำถาม คำตอบมาตรฐาน และบริบทที่ Retrieval

ซึ่งแสดงให้เห็นถึงโครงสร้างของข้อมูลทดสอบในรูปแบบ JSON ที่ประกอบด้วยคำถามเกี่ยวกับวันเปิดเรียนภาคเรียนที่ 1 ปีการศึกษา 2567 คำตอบมาตรฐานที่ระบุว่า "วันเปิดภาคเรียนที่ 1 ปีการศึกษา 2567 คือวันจันทร์ที่ 19 สิงหาคม 2567" และบริบทที่ Retrieval ซึ่งมีรายละเอียดเพิ่มเติมเกี่ยวกับการลงทะเบียน ภาคเรียน และวันเดือนปีที่เกี่ยวข้อง ชุดข้อมูลทดสอบนี้ครอบคลุมประเภทคำถามที่หลากหลาย ได้แก่ คำถามเกี่ยวกับปฏิทินการศึกษาและกำหนดการจำนวน 20 คำถาม คำถามเกี่ยวกับโครงสร้างหลักสูตรและรายวิชาจำนวน 12 คำถาม คำถามเกี่ยวกับการสมัครและคุณสมบัติผู้สมัครจำนวน 10 คำถาม คำถามเกี่ยวกับค่าธรรมเนียมการศึกษาและค่าปรับล่าช้า 3 คำถาม คำถามเกี่ยวกับการติดต่อและสถานที่จำนวน 5 คำถาม และคำถามเกี่ยวกับทุนการศึกษาของสาขา ที่จำนวน 5 คำถาม การกระจายตัวของคำถามในลักษณะนี้สะท้อนให้เห็นถึงความสำคัญและความถี่ของประเด็นคำถามที่ผู้ใช้มักสอบถาม ซึ่งคำถามเกี่ยวกับปฏิทินการศึกษาและโครงสร้างหลักสูตรมีสัดส่วนมากที่สุด อัน

เนื่องมาจากความสำคัญของข้อมูลดังกล่าวในการวางแผนการเรียนและการศึกษา และการตัดสินใจของผู้เรียนและผู้สนใจสมัครเข้าศึกษา ทั้งนี้ การจัดทำชุดข้อมูลทดสอบที่มีความครอบคลุมและหลากหลายนี้ช่วยให้การประเมินประสิทธิภาพของระบบมีความน่าเชื่อถือและสามารถสะท้อนความสามารถในการตอบคำถามในสถานการณ์จริงได้อย่างแม่นยำและมีประสิทธิภาพ

2. Retrieval Evaluation

ผลการประเมินประสิทธิภาพ Retrieval ด้วยเมตริก LLM Context Precision Without Reference พบว่าระบบมีค่าความเกี่ยวข้องของบริบทที่ Retrieval มาได้อยู่ที่ 0.9500 หรือคิดเป็น 95.00% ซึ่งเป็นการวัดความสามารถของระบบในการดึงข้อมูลที่เกี่ยวข้องกับคำถามโดยไม่ต้องอาศัยการเปรียบเทียบกับคำตอบมาตรฐาน Ground Truth โดยตรง ค่าที่ได้นี้สะท้อนให้เห็นว่าประสิทธิภาพของกลไก Retrieval อยู่ในระดับยอดเยี่ยม โดยบริบทที่ระบบดึงมาใช้ในการสร้างคำตอบมีความเกี่ยวข้องกับคำถามได้ถึงร้อยละ 95 ผลลัพธ์นี้แสดงให้เห็นว่าระบบมีประสิทธิภาพสูงมากในการค้นหาข้อมูลที่เกี่ยวข้อง อย่างไรก็ตาม ยังมีช่องว่างเล็กน้อยสำหรับการปรับปรุงเพิ่มเติม โดยอาจพิจารณาการปรับแต่งพารามิเตอร์ในการค้นหาหรือการปรับปรุงวิธีการสร้างเวกเตอร์ เพื่อให้ระบบสามารถบรรลุประสิทธิภาพที่สมบูรณ์แบบมากยิ่งขึ้น แม้ว่าในระดับปัจจุบันระบบจะมีความสามารถในการสร้างคำตอบที่มีคุณภาพสูงจากข้อมูลที่เกี่ยวข้องได้อย่างน่าประทับใจแล้วก็ตาม

```
dataset = Dataset.from_dict(data)
context_precision = LLMContextPrecisionWithoutReference()

result = evaluate(
    dataset=dataset,
    metrics=[context_precision],
    llm=evaluator_llm
)

print(result)

answer_relevancy': 0.6000
```

ภาพประกอบ 50 ผลการประเมินประสิทธิภาพ Retrieval ข้อมูลด้วยเมตริก LLMContextPrecisionWithoutReference

3. Response Quality Evaluation

ผลการประเมินคุณภาพการตอบสนองของระบบเมื่อทดสอบกับชุดข้อมูลที่กำหนด พบว่าระบบมีค่าความเกี่ยวข้องของคำตอบ Answer Relevancy อยู่ที่ 0.8130 จากการวัดด้วยเมตริก ResponseRelevancy ซึ่งเป็นตัวชี้วัดที่ประเมินความสอดคล้องระหว่างคำตอบที่ระบบสร้างขึ้นกับคำตอบมาตรฐาน ค่าที่ได้นี้แสดงให้เห็นว่าระบบสามารถสร้างคำตอบที่มีความเกี่ยวข้องกับคำถามในระดับที่ค่อนข้างดี โดยคำตอบที่สร้างขึ้นมีความสอดคล้องกับข้อมูลที่ถูกต้องประมาณ 81.30% การวัดค่านี้ดำเนินการโดยใช้ชุดเครื่องมือประเมินที่

ประกอบด้วยฟังก์ชัน evaluate(Laun และ Wolff, 2025) ซึ่งรับพารามิเตอร์หลักได้แก่ ชุดข้อมูลทดสอบ dataset เมตริกที่ใช้วัด context_precision และ LLM ที่ใช้ในการประเมิน evaluator_llm ผลลัพธ์นี้สะท้อนให้เห็นว่าระบบ RAG ที่พัฒนาขึ้นมีประสิทธิภาพในระดับที่น่าพอใจสำหรับการตอบคำถามในบริบทของข้อมูลทางวิชาการที่ทดสอบ อย่างไรก็ตาม ยังมีโอกาสในการปรับปรุงระบบให้มีความแม่นยำสูงขึ้นอีกประมาณ 18.7%

```
dataset = Dataset.from_dict(data)
context_precision = ResponseRelevancy()

result = evaluate(
    dataset=dataset,
    metrics=[context_precision],
    llm=evaluator_llm
)

print(result)

answer_relevancy: 0.8130
```

ภาพประกอบ 51 ผลการประเมินคุณภาพการตอบสนองด้วยเมตริก ResponseRelevancy

4. BERTScore Evaluation

ผลการประเมินด้วยเทคนิค BERTScore แสดงให้เห็นถึงประสิทธิภาพของระบบในการสร้างคำตอบที่มีความคล้ายคลึงเชิงความหมายกับคำตอบมาตรฐาน โดยพบว่าระบบมีค่า BERTScore Precision อยู่ที่ 85.19% ซึ่งสะท้อนถึงความแม่นยำของเนื้อหาในคำตอบที่ระบบสร้างขึ้นเมื่อเทียบกับคำตอบอ้างอิง ในขณะที่ค่า BERTScore Recall อยู่ที่ 90.94% แสดงให้เห็นว่าระบบสามารถครอบคลุมเนื้อหาสำคัญจากคำตอบอ้างอิงได้ในระดับที่สูงมาก และค่า BERTScore F1 ซึ่งเป็นค่าเฉลี่ยแบบถ่วงน้ำหนักระหว่าง Precision และ Recall อยู่ที่ 87.83% การวัดค่าเหล่านี้ดำเนินการโดยใช้โมเดล BERT ในการวิเคราะห์ความคล้ายคลึงระหว่างการฝังตัวเชิงความหมาย (semantic embeddings) ของคำตอบที่ระบบสร้างขึ้นกับคำตอบมาตรฐาน ผลลัพธ์ดังกล่าวชี้ให้เห็นว่าระบบ RAG ที่พัฒนาขึ้นมีประสิทธิภาพในระดับที่ดีเยี่ยม โดยเฉพาะอย่างยิ่งความสามารถในการดึงข้อมูลสำคัญจากแหล่งความรู้ (ค่า Recall สูงถึง 90.94%) และสามารถนำเสนอข้อมูลอย่างถูกต้องแม่นยำในระดับที่น่าพอใจ (ค่า Precision 85.19%) แสดงถึงศักยภาพของระบบในการให้บริการข้อมูลที่มีคุณภาพสูงแก่ผู้ใช้งาน

```python
from bert_score import score

hypotheses = ["ภาคเรียนที่ 1/2567 เปิดเรียนในวันจันทร์ที่ 19 สิงหาคม 2567"]
references = ["ภาคเรียนที่ 1/2567 เปิดเรียนวันจันทร์ที่ 19 สิงหาคม 2567"]

P, R, F1 = score(hypotheses, references, lang="th",
                 model_type="bert-base-multilingual-cased", verbose=True)

print(f"BERTScore Precision: {P.mean():.4f}")
print(f"BERTScore Recall: {R.mean():.4f}")
print(f"BERTScore F1: {F1.mean():.4f}")

BERTScore Precision: 0.8519
BERTScore Recall: 0.9094
BERTScore F1: 0.8783
```

ภาพประกอบ 52 ผลการประเมินประสิทธิภาพ BERTScore Evaluationl ข้อมูลด้วยเมตริก bert_score

บทที่ 5
สรุปผลการวิจัย

จากผลการวิจัยที่มุ่งเน้นการยกระดับบริการข้อมูลของมหาวิทยาลัย โดยการที่ประยุกต์ใช้ Generative AI ร่วมกับสถาปัตยกรรม RAG ได้ทำการประเมินประสิทธิภาพของระบบใน 2 ด้านหลัก คือ Retrieval Evaluation และ Generation ผลการวิเคราะห์เชิงปริมาณแสดงให้เห็นว่า ระบบมีประสิทธิภาพในส่วนของ Retrieval Evaluation อยู่ที่ร้อยละ 95 ในขณะที่ส่วนของ Generation สำหรับผู้ใช้มีประสิทธิภาพอยู่ที่ร้อยละ 81.30 ข้อค้นพบดังกล่าวสะท้อนให้เห็นถึงความสำเร็จอย่างโดดเด่นในการพัฒนาส่วน Retrieval ของระบบ ซึ่งสามารถดึงข้อมูลที่เกี่ยวข้องได้อย่างแม่นยำสูงมาก ในขณะที่แบบจำลองภาษาขนาดใหญ่ LLM ที่นำมาใช้ในส่วนของการสร้างคำตอบยังมีโอกาสในการพัฒนาเพื่อให้สามารถประมวลผลข้อมูลและบริบทที่ได้รับได้อย่างมีประสิทธิภาพมากขึ้น

นอกจากนี้ ผลการประเมินด้วย BERTScore ซึ่งเป็นเมตริกที่วัดความคล้ายคลึงเชิงความหมายระหว่างคำตอบที่ระบบสร้างขึ้นกับคำตอบอ้างอิง พบว่าระบบมี BERTScore Precision ที่ 85.19% แสดงถึงความแม่นยำในระดับสูงของเนื้อหาที่สร้างขึ้น BERTScore Recall ที่ 90.94% ซึ่งสะท้อนถึงความสามารถในการครอบคลุมเนื้อหาสำคัญได้อย่างดีเยี่ยม และ BERTScore F1 ที่ 87.83% บ่งชี้ถึงประสิทธิภาพโดยรวมในระดับที่น่าพึงพอใจ ผลการประเมินดังกล่าวยืนยันว่าระบบมีความสามารถสูงในการสร้างคำตอบที่มีความหมายใกล้เคียงกับคำตอบมาตรฐาน โดยเฉพาะในแง่ของการดึงข้อมูลสำคัญจากแหล่งข้อมูลมาใช้ในการตอบคำถาม

การวัดค่าประสิทธิภาพทั้งสามด้านดำเนินการโดยใช้ชุดเครื่องมือประเมินมาตรฐานตามแนวทางที่นำเสนอโดย Laun และ Wolff 2025 ผลการประเมินประสิทธิภาพดังกล่าวชี้ให้เห็นว่าระบบ RAG ที่พัฒนาขึ้นสามารถเป็นเครื่องมือสนับสนุนการให้บริการข้อมูลของมหาวิทยาลัยได้อย่างมีประสิทธิภาพสูง โดยเฉพาะอย่างยิ่งในส่วนของการค้นหาและดึงข้อมูลที่เกี่ยวข้องซึ่งมีความแม่นยำถึงร้อยละ 95 อย่างไรก็ตาม ผลการวิจัยยังบ่งชี้ถึงโอกาสในการพัฒนาที่สำคัญในส่วนของ Generation ที่ยังมีประสิทธิภาพอยู่ที่ร้อยละ 81.30 ซึ่งควรได้รับการปรับปรุงเป็นลำดับแรก ข้อจำกัดนี้อาจเกิดจากความท้าทายในการประมวลผลข้อมูลที่มีความซับซ้อนหรือการเชื่อมโยงความรู้จากหลายแหล่งข้อมูลเข้าด้วยกัน

ประการที่สอง ความสามารถในการเข้าใจคำถามที่มีความซับซ้อนยังคงเป็นความท้าทายสำคัญ โดยเฉพาะคำถามที่มีความกำกวมสูงหรือต้องการการตีความในหลายระดับ ซึ่งอาจจะมีการส่งผลต่อ

ประสิทธิภาพในส่วนของการสร้างคำตอบ แม้ว่าการดึงข้อมูลจะมีประสิทธิภาพสูงมากก็ตาม ประการที่สาม Evaluator LLM อาจมีข้อจำกัดในตัวเองซึ่งส่งผลต่อความแม่นยำของการประเมิน โดยเฉพาะในส่วนของการประเมินคุณภาพของคำตอบที่มีความซับซ้อนหรือมีความเฉพาะทางสูง

บรรณานุกรม

1. Takada T, Kawai Y. An analysis of elasticity vector distribution specific to semelparous species using randomly generated population projection matrices and the COMPADRE Plant Matrix Database. Ecological Modelling. 2020;431:109125.

2. Li A, Wang Y, Chen H. AI Driven Cardiovascular Risk Prediction Using NLP and Large Language models for Personalized Medicine in Athletes. SLAS Technology. 2025:100286.

3. Xiang RF. Use of n-grams and K-means clustering to classify data from free text bone marrow reports. Journal of Pathology Informatics. 2024;15:100358.

4. Şen TÜ, Yakit MC, Gümüş MS, Abar O, Bakal G. Combining N-grams and graph convolution for text classification. Applied Soft Computing. 2025;175:113092.

5. Cremaschi M, Ditolve D, Curcio C, Panzeri A, Spoto A, Maurino A. Decoding the mind: A RAG-LLM on ICD-11 for decision support in psychology. Expert Systems with Applications. 2025;279:127191.

6. Zhu M, Yang Q, Gao Z, Yuan Y, Liu J. FedBM: Stealing knowledge from pre-trained language models for heterogeneous federated learning. Medical Image Analysis. 2025;102:103524.

7. R N, Khan SB, kumar AV, T R M, Alojail M, Sangwan SR, et al. Enhancing drug discovery and patient care through advanced analytics with the power of NLP and machine learning in pharmaceutical data interpretation. SLAS Technology. 2025;31:100238.

8. Elhassan SE, Sajid MR, Syed AM, Fathima SA, Khan BS, Tamim H. Assessing Familiarity, Usage Patterns, and Attitudes of Medical Students Toward ChatGPT and Other Chat-Based AI Apps in Medical Education: Cross-Sectional Questionnaire Study. JMIR Medical Education. 2025;11.

9. He G, Huang C. Few-shot medical relation extraction via prompt tuning enhanced pre-trained language model. Neurocomputing. 2025;633:129752.

10. Babin JL, Raber H, Mattingly Ii TJ. Prompt Pattern Engineering for Test Question Mapping Using ChatGPT: A Cross-Sectional Study. American Journal of Pharmaceutical Education. 2024;88(10):101266.

11. Jiang G, Ma Z, Zhang L, Chen J. Prompt engineering to inform large language model in automated building energy modeling. Energy. 2025;316:134548.

12. Hao Z, Liu F, Jiao L, Du Y, Li S, Wang H, et al. Preserving text space integrity for robust compositional zero-shot learning via mixture of pretrained experts. Neurocomputing. 2025;614:128773.

13. Hu M, Chang H, Shan S, Chen X. Inference Calibration of Vision-Language Foundation Models for Zero-Shot and Few-Shot Learning. Pattern Recognition Letters. 2025;192:15-21.

14. Geng Y, Chen J, Zeng Y, Chen Z, Zhang W, Pan JZ, et al. Prompting disentangled embeddings for knowledge graph completion with pre-trained language model. Expert Systems with Applications. 2025;268:126175.

15. Chen L, Liu J, Duan Y, Wang R. KG-prompt: Interpretable knowledge graph prompt for pre-trained language models. Knowledge-Based Systems. 2025;311:113118.

16. Steybe D, Poxleitner P, Aljohani S, Herlofson BB, Nicolatou-Galitis O, Patel V, et al. Evaluation of a context-aware chatbot using retrieval-augmented generation for answering clinical questions on medication-related osteonecrosis of the jaw. Journal of Cranio-Maxillofacial Surgery. 2025;53(4):355-60.

17. Bahr L, Wehner C, Wewerka J, Bittencourt J, Schmid U, Daub R. Knowledge graph enhanced retrieval-augmented generation for failure mode and effects analysis. Journal of Industrial Information Integration. 2025;45:100807.

18. Ludwig H, Schmidt T, Kühn M. An ontology-based retrieval augmented generation procedure for a voice-controlled maintenance assistant. Computers in Industry. 2025;169:104289.

19. Wang Z, Liu Z, Lu W, Jia L. Improving knowledge management in building engineering with hybrid retrieval-augmented generation framework. Journal of Building Engineering. 2025;103:112189.

20. Chen L-C, Pardeshi MS, Liao Y-X, Pai K-C. Application of retrieval-augmented generation for interactive industrial knowledge management via a large language model. Computer Standards & Interfaces. 2025;94:103995.

21. Ceccarelli S, Balsalobre A, Cano ME, Canale D, Lobbia P, Stariolo R, et al. Analysis of Chagas disease vectors occurrence data: the Argentinean triatomine species database. Biodiversity Data Journal. 2020;8.

22. Nassif G, Oulabi K, Boules T, Savage K, Kavousi Y, Mansour M, et al. The Utility of Chat GPT in Venous Education. Journal of Vascular Surgery: Venous and Lymphatic Disorders. 2024;12(3):101803.

23. Denny Zhou NS, Le Hou† Jason Wei† Nathan Scales† Xuezhi Wang, Dale Schuurmans† Claire Cui, Olivier Bousquet, Quoc Le, Ed Chi. LEAST-TO-MOST PROMPTING ENABLES COMPLEX

REASONING IN LARGE LANGUAGE MODELS. prompting 2023.

24. Yang VB, Sams CM, Francis SN, Daneshjou R, Lester JC. Uncovering Disparities in Skin Tone Representation Among AI Vision-Language Models (VLMs). Journal of Investigative Dermatology. 2025.

25. Khoboko PW, Marivate V, Sefara J. Optimizing translation for low-resource languages: Efficient fine-tuning with custom prompt engineering in large language models. Machine Learning with Applications. 2025;20:100649.

26. Jung H, Oh J, Stephenson KAJ, Joe AW, Mammo ZN. Prompt engineering with ChatGPT3.5 and GPT4 to improve patient education on retinal diseases. Canadian Journal of Ophthalmology. 2024.

27. van Oers B. Developmental Education: Reflections on a Chat-Research Program in the Netherlands. Learning, Culture and Social Interaction. 2012;1(1):57-65.

28. Ren T, Zhang Z, Jia B, Zhang S. Retrieval-Augmented Generation-aided causal identification of aviation accidents: A large language model methodology. Expert Systems with Applications. 2025;278:127306.

29. Tsai M-L, Ong CW, Chen C-L. Exploring the use of large language models (LLMs) in chemical engineering education: Building core course problem models with Chat-GPT. Education for Chemical Engineers. 2023;44:71-95.

30. Liu H, Yin H, Luo Z, Wang X. Integrating chemistry knowledge in large language models via prompt engineering. Synthetic and Systems Biotechnology. 2025;10(1):23-38.