



รายงาน Capstone Project

(ชื่อโครงการภาษาไทย ถ้ามี)

(ชื่อโครงการภาษาอังกฤษ).....

โดย

(สมาชิกกลุ่ม คนที่1)

(สมาชิกกลุ่ม คนที่2)

(สมาชิกกลุ่ม คนที่3)

(สมาชิกกลุ่ม คนที่4)

อาจารย์ที่ปรึกษา

รายงานเล่มนี้เป็นส่วนหนึ่งของการเรียนรู้และการพัฒนาศักยภาพผู้เรียน
ภายใต้โครงการพัฒนากำลังคนสมรรถนะสูงด้านปัญญาประดิษฐ์ (บพค.)
ดำเนินการร่วมกับแพลตฟอร์ม 42 Bangkok เพื่อยกระดับขีดความสามารถในการแข่งขัน
และการประยุกต์ใช้เทคโนโลยีของผู้ประกอบการไทย

บทคัดย่อ (Abstract)

รายงานโครงการฉบับนี้จัดทำขึ้นเพื่อแก้ไขปัญหาความซับซ้อนและปริมาณข้อมูลข่าวสารทางการเงินที่มีจำนวนมหาศาล (Information Overload) ซึ่งเป็นอุปสรรคต่อการตัดสินใจลงทุนของผู้ประกอบการไทย โดยมีวัตถุประสงค์เพื่อพัฒนาระบบปัญญาประดิษฐ์สำหรับวิเคราะห์อารมณ์ตลาดและสรุปข่าวสารแบบเรียลไทม์ (AI Market Psychologist) คณะผู้จัดทำได้สร้างกระบวนการทำงานอัตโนมัติแบบครบวงจร (End-to-End Pipeline) เริ่มจากการรวบรวมข้อมูลข่าวสารผ่าน Web Scraping แบบ Incremental Update และนำเข้าสู่กระบวนการประมวลผลภาษาธรรมชาติ (NLP) ด้วยเทคนิคผสมผสาน (Hybrid Approach) ระหว่าง TF-IDF ร่วมกับ Cosine Similarity เพื่อจัดหมวดหมู่ภาคอุตสาหกรรม และประยุกต์ใช้ Large Language Models (LLMs) ประสิทธิภาพสูงหลายโมเดลร่วมกัน (Ensemble) ได้แก่ Qwen-2.5, Llama-3.1 และ Gemma-3 ใน การวิเคราะห์ Sentiment ให้คะแนนความเชื่อมั่น และสรุปใจความสำคัญของข่าว โดยใช้หลักการถ่วงน้ำหนัก ตามเวลา (Time-Weighted Score) และกลไกฉันทามติ (Consensus Mechanism) เพื่อลดความ คลาดเคลื่อนและเน้นแรงส่งของตลาดที่เป็นปัจจุบัน

ผลลัพธ์จากการดำเนินงานคือต้นแบบเว็บแอปพลิเคชัน (Web Dashboard) ที่พัฒนาด้วย Streamlit ซึ่ง ทำหน้าที่เเส่มีอนผู้ช่วยวิเคราะห์การลงทุนอัจฉริยะ ระบบสามารถแสดงผลภาพรวมตลาดในรูปแบบ Market Heatmap ที่ใช้เฉดสีสะท้อนอารมณ์ตลาด (Greed/Fear) และขนาดพื้นที่แทนปริมาณข่าว ช่วยให้ผู้ใช้งาน มองเห็นภาพรวมและเจาะลึกรายละเอียดราย Sector ได้อย่างรวดเร็ว นอกจากนี้ ระบบยังช่วยลดระยะเวลา ในการติดตามข่าวสารด้วยการสรุปเนื้อหาที่กระชับและตรงประเด็น ส่งเสริมให้นักลงทุนและผู้ประกอบการ ไทยสามารถเข้าถึงข้อมูลเชิงลึก (Insights) และตัดสินใจลงทุนโดยอาศัยข้อมูลขับเคลื่อน (Data-driven Decision Making) ได้อย่างมีประสิทธิภาพและทันต่อสถานการณ์ครับ

สารบัญ

หน้า

1. บทนำ (Introduction)	1
1.1 ที่มาและความสำคัญของโครงการ	1
1.2 ปัญหาและขอบเขตของโครงการ	2
1.3 วัตถุประสงค์ของโครงการ	3
1.4 ประโยชน์ที่คาดว่าจะได้รับ	4
2. เครื่องมือ เทคโนโลยี และแพลตฟอร์มที่ใช้ในโครงการ (Tools, Technologies, and Platforms)	5
3. วิธีการดำเนินงาน (Methodology)	8
3.1 ขั้นตอน/วิธีการดำเนินงาน	8
3.2 การบูรณาการกับเนื้อหาที่เรียน	9
3.3 การพัฒนาและการทดสอบ	10
4. ผลการดำเนินงานโครงการ (Results)	13
4.1 ผลของการดำเนินงาน	13
4.2 การประเมินผลงาน	13
5. สรุปผลและข้อเสนอแนะ (Conclusion & Future Work)	16
5.1 สรุปผลลัพธ์	16
5.2 ปัญหาอุปสรรคในการดำเนินโครงการ	16
5.3 แนวทางการแก้ไข และข้อเสนอแนะ	17

ภาคผนวก (ถ้ามี)

ภาคผนวก ก (รูปภาพ,ตาราง,งานวิจัยต่างๆที่เกี่ยวข้อง)
ภาคผนวก ข (สไลด์ที่ใช้สำหรับการนำเสนอ)

บทนำ (Introduction)

1.1 ที่มาและความสำคัญของโครงการ

ในยุคปัจจุบันที่เทคโนโลยีดิจิทัลเข้ามายึด主导ทางเศรษฐกิจโลก ตลาดการเงินและการลงทุนมีการเปลี่ยนแปลงอย่างรวดเร็วและมีความผันผวนสูง (High Volatility) ปัจจัยสำคัญที่ขับเคลื่อนการเคลื่อนไหวของราคางานต์ต่าง ๆ ไม่ได้ขึ้นอยู่กับผลประกอบการทางบัญชีเพียงอย่างเดียวอีกต่อไป แต่ยังขึ้นอยู่กับ "อารมณ์ของตลาด" (Market Sentiment) ซึ่งถูกกำหนดโดยกระแสข่าวสาร (News Flow) ที่เกิดขึ้นทั่วโลกตลอด 24 ชั่วโมง ไม่ว่าจะเป็นการประกาศนโยบายทางการเงิน ภาวะเศรษฐกิจ หรือการเปลี่ยนแปลงทางเทคโนโลยี

ปัญหาสำคัญที่นักลงทุนและผู้ประกอบการต้องเผชิญในปัจจุบันไม่ใช่การขาดแคลนข้อมูล แต่คือภาวะ "ข้อมูลท่วมท้น" (Information Overload) ในแต่ละวันมีข่าวสารทางการเงินถูกเผยแพร่ผ่านช่องทางออนไลน์ นับพันชิ้น การที่มนุษย์จะติดตาม อ่าน และวิเคราะห์ผลกรอบของข่าวทั้งหมดที่มีต่อภาคอุตสาหกรรม (Sector) ต่าง ๆ ให้ทันท่วงทีนั้นเป็นเรื่องที่เป็นไปไม่ได้ในทางปฏิบัติ นอกจากนี้ การวิเคราะห์ข่าวด้วยมนุษย์ (Human Analysis) ยังมีข้อจำกัดเรื่องความลำเอียงส่วนบุคคล (Bias) ความเห็นอย่างเดียว และความไม่สม่ำเสมอในการประเมิน ซึ่งอาจนำไปสู่การตัดสินใจที่ผิดพลาดหรือล่าช้ากว่าเหตุการณ์ (Lagging Decision)

เทคโนโลยีปัญญาประดิษฐ์ (Artificial Intelligence: AI) โดยเฉพาะอย่างยิ่งความก้าวหน้าของโมเดลภาษาขนาดใหญ่ (Large Language Models: LLMs) และการประมวลผลภาษาธรรมชาติ (Natural Language Processing: NLP) ได้เข้ามายึด主导ทางสำคัญในการแก้ไขปัญหานี้ LLMs สมัยใหม่มีความสามารถในการอ่าน เข้าใจบริบท และสรุปความข้อมูลเชิงซับซ้อนได้ใกล้เคียงกับผู้เชี่ยวชาญ อีกทั้งยังสามารถประมวลผลข้อมูลจำนวนมากทางค่าลได้ในระยะเวลาอันสั้น

ด้วยเหตุผลดังกล่าว คณะผู้จัดทำจึงได้ริเริ่มโครงการ "AI Market Psychologist" หรือระบบวิเคราะห์อารมณ์ตลาดและสรุปข่าวสารการเงินแบบเรียลไทม์ขึ้น โดยมุ่งเน้นการนำเทคโนโลยี AI แบบผสมผสาน (Hybrid Approach) มาประยุกต์ใช้ เริ่มจากการใช้เทคนิค NLP พื้นฐานในการคัดกรองข้อมูล และใช้พลังของ Generative AI (LLMs) หลายโมเดลทำงานร่วมกัน (Ensemble Learning) เพื่อทำหน้าที่เสนอแนะนักวิเคราะห์ การเงินอัจฉริยะ ระบบจะช่วยเปลี่ยนข้อมูลข่าวสารที่ยุ่งเหยิง (Unstructured Data) ให้กลายเป็นดัชนีชี้วัดทางอารมณ์ตลาดที่เป็นตัวเลข (Quantitative Data) และนำเสนอผ่าน Dashboard ที่เข้าใจง่าย ซึ่งจะช่วยยกระดับขีดความสามารถในการแข่งขันของผู้ประกอบการไทย ให้สามารถเข้าถึงเครื่องมือวิเคราะห์การลงทุนที่มีประสิทธิภาพเทียบเท่าระดับสากล

1.2 ปัญหาและขอบเขตของโครงการ

1.2.1 ปัญหาที่ต้องการแก้ไข (Problem Statement)

จากการศึกษาบริบทและสภาพปัญหาปัจจุบัน พบระดับปัญหาหลักที่โครงการนี้มุ่งเน้นแก้ไข ดังนี้

- ปริมาณข้อมูลมหาศาล (Volume) :** ข่าวสารทางการเงินเกิดขึ้นจำนวนมากในแต่ละวัน เกินขีดความสามารถของมนุษย์ที่จะติดตามได้ครบถ้วนทุก Sector
- ความเร็วของข้อมูล (Velocity) :** ตลาดตอบสนองต่อข่าวสารอย่างรวดเร็ว (Real-time) การวิเคราะห์ด้วยกระบวนการเดิมที่ต้องใช้เวลาอ่านและสรุป ทำให้ข้อมูลล้าสมัยก่อนนำไปใช้ตัดสินใจ
- ความไม่แน่นอนและความลำเอียง (Bias & Consistency) :** การตีความข่าวเดียวกันโดยนักวิเคราะห์แต่ละคนอาจให้ผลลัพธ์ที่ต่างกัน และอาจได้รับอิทธิพลจากการมณีส่วนตัว ขณะที่ AI สามารถกำหนดเกณฑ์การให้คะแนนที่เป็นมาตรฐานได้
- ความยากในการมองภาพรวม (Lack of Holistic View) :** การติดตามข่าวรายตัวทำให้นักลงทุนมองไม่เห็นภาพรวมว่า ภาคอุตสาหกรรมใดกำลังเป็นผู้นำตลาด (Bullish) หรือกำลังซบเชา (Bearish) ในขณะนั้น

1.2.2 ขอบเขตการทำงาน (Scope of Work)

เพื่อให้โครงการบรรลุวัตถุประสงค์ภายใต้ระยะเวลาและทรัพยากรที่กำหนด คณะกรรมการผู้จัดทำได้กำหนดขอบเขตการดำเนินงานไว้ดังนี้

1. ด้านข้อมูล (Data)

- รวบรวมข้อมูลข่าวสารการเงินภาษาอังกฤษจากเว็บไซต์ชั้นนำ (เช่น Investing.com) โดยเน้นข่าวที่มีผลกระทบต่อตลาดหุ้นและภาคเศรษฐกิจ
- ข้อมูลครอบคลุมภาคอุตสาหกรรม (Sectors) หลัก เช่น Technology, Financials, Healthcare, Energy, Industrials ฯลฯ
- ใช้วิธีการดึงข้อมูลแบบอัตโนมัติ (Web Scraping) และมีระบบตรวจสอบข้อมูลช้า (Incremental Update) เพื่อจัดการข้อมูลอย่างมีประสิทธิภาพ

2. ด้านการประมวลผล (Processing & AI):

- การจำแนกหมวดหมู่ (Classification) :** ใช้เทคนิค TF-IDF (Term Frequency-Inverse Document Frequency) ร่วมกับ Cosine Similarity ในการจัดกลุ่มข่าวเข้าสู่ Sector ต่าง ๆ
- การวิเคราะห์อารมณ์ (Sentiment Analysis) :** ใช้ Large Language Models (LLMs) จำนวน 3 โมเดล ได้แก่ Qwen-2.5-14B, Llama-3.1-8B และ Gemma-3-12B ทำงานร่วมกันในรูปแบบ Consensus Mechanism เพื่อให้คะแนน Sentiment (-10 ถึง +10) และระบุแนวโน้ม (Outlook)

- การสรุปความ (Summarization) : ใช้ LLM ในการสรุปเนื้อหาข่าวขนาดยาวให้เหลือเพียงใจความสำคัญ (Short Answer)
- การให้คะแนนแบบถ่วงน้ำหนัก (Time-Weighted Scoring) : พัฒนาอัลกอริทึมที่ให้ความสำคัญกับข่าวล่าสุดมากกว่าข่าวเก่า เพื่อสะท้อนโมเมนตัมปัจจุบัน (Real-time Momentum)

3. ด้านการแสดงผล (User Interface)

- พัฒนา Web Application ด้วย Streamlit
- แสดงผลในรูปแบบ Market Heatmap (Treemap) ที่ใช้สี (แดง/เขียว) แทนค่า Sentiment และขนาดแทนปริมาณข่าว
- มีหน้ารายละเอียด (Sector Detail) แสดงกราฟแนวโน้มย้อนหลัง 7 วัน และรายการข่าวที่ผ่านการคัดกรองแล้ว

1.2.3 ข้อจำกัด (Constraints)

- ทรัพยากรการประมวลผล : การรันโมเดล LLM ขนาดใหญ่ (14B Parameters) ต้องใช้หน่วยความจำการ์ดจอ (GPU VRAM) สูง จึงต้องมีการบริหารจัดการ Memory (VRAM Clearing) และจำกัดขนาด Context Window ในบางขั้นตอน
- ความถูกต้องของโมเดลภาษา : เมื่อจะใช้ระบบฉันทามติ (Consensus) แต่ผลลัพธ์ยังชี้อยู่กับความสามารถพื้นฐานของ Pre-trained Model ที่เลือกใช้
- ความล่าช้า (Latency) : ระบบไม่ได้ออกแบบมาสำหรับการเทรดความเร็วสูง (High-Frequency Trading) แต่ออกแบบมาเพื่อการวิเคราะห์เชิงกลยุทธ์ (Strategic Analysis) ที่มีการอัปเดตข้อมูลเป็นรอบเวลา

1.3 วัตถุประสงค์ของโครงการ

เพื่อให้การดำเนินงานเป็นไปในทิศทางเดียวกัน คณะผู้จัดทำได้กำหนดวัตถุประสงค์หลักและวัตถุประสงค์ย่อยดังนี้

- เพื่อพัฒนาระบบรวมและวิเคราะห์ข่าวสารการเงินอัตโนมัติ (Automated Pipeline) ที่สามารถทำงานได้ตั้งแต่การดึงข้อมูล การประมวลผล จนถึงการแสดงผล โดยลดภาระงานของมนุษย์
- เพื่อประยุกต์ใช้เทคโนโลยีปัญญาประดิษฐ์แบบผสมผสาน (Hybrid AI) ระหว่างเทคนิคทางสถิติ (TF-IDF) และโมเดลภาษาขนาดใหญ่ (LLMs) ในการจำแนกหมวดหมู่และวิเคราะห์อารมณ์ของตลาดได้อย่างแม่นยำ
- เพื่อสร้างกลไกการให้คะแนนความเชื่อมั่นที่มีมาตรฐาน (Consensus Scoring) โดยใช้โมเดลหลายตัวช่วยกันประเมิน เพื่อลดความคลาดเคลื่อนและความลำเอียงในการวิเคราะห์
- เพื่อพัฒนาเครื่องมือแสดงผลข้อมูล (Dashboard) ในรูปแบบ Visual Data ที่เข้าใจง่าย ช่วยให้นักลงทุนสามารถมองเห็นภาพรวม (Big Picture) และเจาะลึกรายละเอียด (Drill-down) ได้ในแพลตฟอร์มเดียว

1.4 ประโยชน์ที่คาดว่าจะได้รับ

การพัฒนาโครงงาน "AI Market Psychologist" คาดว่าจะก่อให้เกิดประโยชน์ต่อภาคส่วนต่าง ๆ ดังนี้

1. ต่อผู้ใช้งาน (นักลงทุนและผู้ประกอบการ)

- ลดระยะเวลาในการทำงาน : สามารถรับรู้ข่าวสารสำคัญและทิศทางตลาดได้ภายในเวลาไม่กี่นาที แทนการเสียเวลาอ่านข่าวหลายชั่วโมง
- สนับสนุนการตัดสินใจ : มีข้อมูลเชิงปริมาณ (Sentiment Score) รองรับการตัดสินใจลงทุน (Data-Driven Decision Making) ลดการใช้อารมณ์ตัดสินใจ
- ไม่พลาดโอกาสสำคัญ : ระบบแจ้งเตือนแนวโน้มตลาด (Bullish/Bearish) แบบเรียลไทม์ ช่วยให้ปรับเปลี่ยนกลยุทธ์ได้ทันสถานการณ์

2. ต่องค์กรและภาคธุรกิจ:

- เพิ่มขีดความสามารถในการแข่งขัน : องค์กรธุรกิจไทยมีเครื่องมือวิเคราะห์ตลาดที่มีประสิทธิภาพ ทัดเทียมกับเครื่องมือในต่างประเทศ
- การประยุกต์ใช้ AI ในองค์กร : เป็นต้นแบบ (Prototype) ในการนำ AI มาใช้ในกระบวนการ Business Intelligence (BI) ซึ่งสามารถขยายผลไปสู่การวิเคราะห์ข้อมูลภายในองค์กรอื่น ๆ ได้

3. ต่อผู้พัฒนาและการเทคโนโลยี

- องค์ความรู้ใหม่ : ได้เรียนรู้และเข้าใจกระบวนการสร้าง End-to-End AI Pipeline การจัดการทรัพยากร GPU และการทำ Prompt Engineering สำหรับงานเฉพาะทางด้านการเงิน
- นวัตกรรม : เป็นการนำเสนอแนวทางใหม่ในการวิเคราะห์ Sentiment ที่ใช้ Time-Weighted Score และ Consensus Mechanism ซึ่งสามารถต่อยอดเป็นงานวิจัยหรือผลิตภัณฑ์เชิงพาณิชย์ได้ในอนาคต

เครื่องมือ เทคโนโลยี และแพลตฟอร์มที่ใช้ในโครงการ

(Tools, Technologies, and Platforms)

ในการพัฒนาโครงการ "AI Market Psychologist" คณะกรรมการจัดทำได้คัดเลือกเครื่องมือและเทคโนโลยีแบบผสมผสาน (Hybrid Stack) โดยเน้นความสามารถในการประมวลผลภาษาธรรมชาติ (NLP) ที่ซับซ้อน และการแสดงผลข้อมูลเชิงลึกแบบเรียลไทม์ เครื่องมือที่ใช้สามารถจำแนกออกเป็น 4 ส่วนหลัก ดังนี้

2.1 ภาษาโปรแกรมและสภาพแวดล้อมการพัฒนา (Programming Language & Environment)

- Python (Version 3.10+)
 - รายละเอียด : เป็นภาษาหลักในการพัฒนาระบบทั้งหมด ตั้งแต่กระบวนการดึงข้อมูล (ETL) ไปจนถึงส่วนติดต่อผู้ใช้งาน
 - เหตุผลในการเลือก : Python เป็นภาษามาตรฐานสำหรับงานด้าน Data Science และ AI มีไลบรารีรองรับจำนวนมาก (Rich Ecosystem) โค้ดอ่านง่าย และมีความยืดหยุ่นสูงในการเขียนโมเดล Deep Learning เช้ากับเว็บแอปพลิเคชัน
- Jupyter Notebook
 - รายละเอียด : ใช้สำหรับการทดลอง (Experimentation) พัฒนาโมเดล และรันกระบวนการวิเคราะห์ข้อมูลในเบื้องต้น (main.ipynb)
 - เหตุผลในการเลือก : เหมาะสมสำหรับการวิเคราะห์ข้อมูลแบบโต้ตอบ (Interactive) สามารถดูผลลัพธ์ของ Dataframe และกราฟได้ทันทีที่ลากขึ้นตอน ซึ่งจำเป็นมากในช่วงการจูน Prompt ของ LLM

2.2 การประมวลผลข้อมูลและปัญญาประดิษฐ์ (Data Processing & AI Frameworks)

2.2.1. ไลบรารีจัดการข้อมูล (Data Manipulation)

- Pandas & NumPy
 - รายละเอียด : ใช้จัดการข้อมูลในรูปแบบตาราง (DataFrame) การคำนวณ Time-Series และการจัดการค่าทางสถิติ เช่น การคำนวณ Time-Weighted Score
 - เหตุผลในการเลือก : มีประสิทธิภาพสูงในการจัดการข้อมูลจำนวนมาก (High Performance) และมีฟังก์ชันจัดการข้อมูลเวลา (Datetime) ที่ครบถ้วน

2.2.2. โมเดลภาษาขนาดใหญ่และเฟรมเวิร์ก (LLMs & Deep Learning Frameworks)

- PyTorch
 - รายละเอียด : เป็น Backend Framework หลักสำหรับการรันโมเดล Deep Learning
 - เหตุผลในการเลือก : รองรับการทำงานบน GPU (CUDA) ได้ดีเยี่ยม และเป็นพื้นฐานของไลบรารี Transformers
- Hugging Face Transformers

- รายละเอียด : ใช้สำหรับเรียกใช้โมเดล Pre-trained, Tokenizer และจัดการ Pipeline การประมวลผลข้อความ
- เหตุผลในการเลือก : เป็นมาตรฐานกลางในการเข้าถึงโมเดล Open Source ระดับโลก ช่วยให้สามารถสลับเปลี่ยนโมเดล (Model Swapping) ได้ง่าย
- Selected Large Language Models (LLMs)
 - Qwen-2.5-14B-Instruct : โมเดลหลักที่มีความฉลาดสูง ใช้สำหรับการวิเคราะห์เชิงลึกและการให้คะแนน Sentiment
 - Meta-Llama-3.1-8B-Instruct : โมเดลรองที่มีความเร็วสูง ใช้เพื่อสอบถามผลลัพธ์ (Consensus)
 - Google Gemma-3-12B-IT: ใช้เป็นความเห็นที่สามเพื่อสร้างความสมดุล
 - เหตุผลในการเลือก : การใช้สถาปัตยกรรม Multi-Model Consensus ช่วยลดความลำเอียง (Bias) ของโมเดลเดียว และโมเดลเหล่านี้เป็น State-of-the-art (SOTA) ในกลุ่ม Open Weights ที่สามารถรันบน GPU ระดับองค์กรได้

2.2.3. การประมวลผลภาษาธรรมชาติแบบดั้งเดิม (Classical NLP)

- Scikit-learn (TF-IDF & Cosine Similarity)
 - รายละเอียด : ใช้แปลงข้อความข่าวเป็นเวกเตอร์ทางคณิตศาสตร์เพื่อคำนวณความเหมือน
 - เหตุผลในการเลือก : ทำงานได้เร็วกว่า LLM มาก จึงเหมาะสมสำหรับใช้เป็นด่านแรกในการคัดกรอง Sector (First-pass Classification) เพื่อลดภาระงานของ GPU

2.3 การรวบรวมข้อมูล (Data Acquisition)

- Cloudscraper
 - รายละเอียด : ใช้สำหรับทำ Web Scraping ดึงข้อมูลจากเว็บไซต์ข่าวการเงิน
 - เหตุผลในการเลือก : สามารถจัดการกับการป้องกันบอท (Anti-bot Protection) ของเว็บไซต์เป็นอย่างมากโดยใช้ライบรารี Requests ธรรมดा
- BeautifulSoup4
 - รายละเอียด : ใช้สำหรับแก้โครงสร้าง HTML (Parsing) เพื่อดึงเฉพาะเนื้อหาข่าว หัวข้อ และวันที่
 - เหตุผลในการเลือก : ใช้งานง่าย แม่นยำ และทนทานต่อโครงสร้าง HTML ที่อาจผิดเพี้ยน

2.4 ส่วนติดต่อผู้ใช้งานและการแสดงผล (User Interface & Visualization)

- Streamlit
 - รายละเอียด : เฟรมเวิร์กสำหรับสร้าง Web Application (Home.py, pages/)
 - เหตุผลในการเลือก : ช่วยให้เปลี่ยนสคริปต์ Python เป็นเว็บแอปพลิเคชันได้รวดเร็ว (Rapid Prototyping) โดยไม่ต้องเขียน HTML/CSS/JavaScript เองทั้งหมด เหมาะสมสำหรับการนำเสนอ Data Product
- Plotly Express
 - รายละเอียด : ใช้สร้างกราฟิกเชิงโต้ตอบ โดยเฉพาะ Treemap (Heatmap) และกราฟเส้น
 - เหตุผลในการเลือก : รองรับการทำ Interactive Chart (Zoom, Hover, Click) ซึ่งช่วยให้ผู้ใช้งานสามารถเจาะลึกข้อมูล (Drill-down) ในแต่ละ Sector ได้อย่างสะดวก

2.5 โครงสร้างพื้นฐานฮาร์ดแวร์ (Hardware Infrastructure)

- GPU Computing (CUDA)
 - รายละเอียด : ใช้หน่วยประมวลผลกราฟิก (GPU) ของ NVIDIA (เช่น A100 หรือ RTX 3090/4090)
 - เหตุผลในการเลือก : จำเป็นในการรัน LLM ขนาด 8B-14B Parameters การใช้ CPU เพียงอย่างเดียวจะช้าเกินไปสำหรับการประมวลผลแบบเรียลไทม์ นอกจากนี้ยังมีการใช้เทคนิค VRAM Management เพื่อบริหารจัดการทรัพยากรให้คุ้มค่าที่สุดอีกด้วย

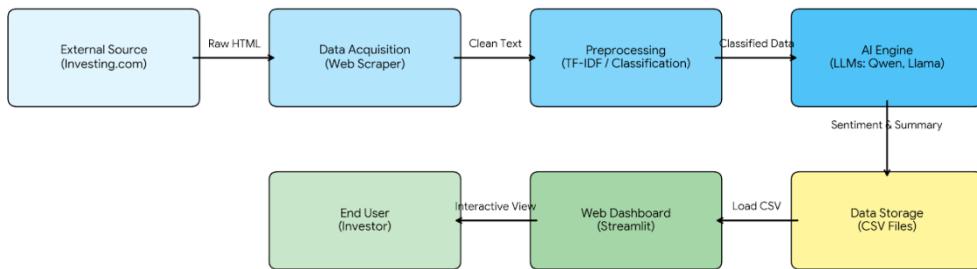
วิธีการดำเนินงาน (Methodology)

3.1 ขั้นตอนและวิธีการดำเนินงาน

3.1.1 ภาพรวมระบบ (System Overview)

ระบบถูกออกแบบภายใต้สถาปัตยกรรมแบบ End-to-End Data Pipeline ซึ่งแบ่งการทำงานออกเป็น 3 ส่วนหลัก ได้แก่ (1) ส่วนรวบรวมข้อมูล (Data Ingestion), (2) ส่วนประมวลผลอัจฉริยะ (AI Processing Core), และ (3) ส่วนแสดงผล (Visualization Layer) โดยมีการไหลของข้อมูลดังแผนภาพด้านล่าง

Data Flow Diagram: AI Market Psychologist



คำอธิบายแผนภาพ

1. Input Source : ข้อมูลข่าวสารดิบ (Raw News) จากเว็บไซต์ Investing.com
2. ETL Process : scraper Python (Cloudscraper) ดึงข้อมูลและทำความสะอาดเบื้องต้น
3. Knowledge Base : เก็บข้อมูลลงในไฟล์ CSV กลาง (sentiment_final.csv)
4. AI Engine : โมเดล LLMs (Qwen, Llama, Gemma) อ่านข่าว วิเคราะห์ Sentiment และสรุปความ
5. Application : Streamlit อ่านผลลัพธ์ที่วิเคราะห์แล้วมาแสดงเป็น Heatmap และกราฟ

การไหลของข้อมูล (Data Flow) กระบวนการไหลของข้อมูล (Data Flow Pipeline) เริ่มต้นจากแหล่งข้อมูลภายนอกจนถึงมือผู้ใช้งาน มีขั้นตอนดังนี้

1. Data Acquisition (การได้มาซึ่งข้อมูล)
 - ระบบทำการส่ง Request ไปยัง Server ของแหล่งข่าวโดยใช้เทคนิค User-Agent Spoofing เพื่อหลีกเลี่ยงการป้องกันบอท
 - ทำการตรวจสอบลิงก์ที่รับฐานข้อมูลเดิม (Deduplication) เพื่อดึงเฉพาะข่าวใหม่ (Incremental Update)
2. Preprocessing (การเตรียมข้อมูล)
 - จัดดูข้อมูลขยาย (Noise Reduction) เช่น แท็กโฆษณา หรือข้อความที่ไม่เกี่ยวข้อง

- แปลงข้อความเป็นเวกเตอร์ (Vectorization) ด้วย TF-IDF เพื่อเตรียมเข้าสู่กระบวนการจำแนกหมวดหมู่

3. Classification & Filtering

- ระบบใช้ Cosine Similarity เปรียบเทียบเนื้อหาข่าวกับ Keyword ของแต่ละ Sector
- หากค่าความมั่นใจต่ำ ข้อมูลจะถูกส่งต่อไปยัง LLM เพื่อให้ช่วยตัดสินใจ (Fallback Mechanism)

4. Intelligence Processing (การประมวลผลอัจฉริยะ)

- Sentiment Scoring: ข้อมูลถูกส่งเข้าโมเดล LLM พร้อมกัน 3 ตัว (Multi-Model Inference) เพื่อให้คะแนน -1 ถึง +1
- Summarization : โมเดลทำการอ่านเนื้อหาข่าวทั้งหมดและสร้างบทสรุปสั้น (Executive Summary)
- นำข่าวทั้ง Sector ในช่วงเวลา 7 วันย้อนหลังให้ LLM หั่ง 3 วิเคราะห์ว่ามีความเห็นอย่างไรบ้าง จากข้อมูลตลาดที่ได้ศึกษาไป

5. Visualization (การแสดงผล)

- Streamlit ทำการโหลดข้อมูลล่าสุด คำนวณค่า Time-Weighted Score และพื้นที่ Treemap แบบโต้ตอบได้

3.2 การบูรณาการกับเนื้อหาที่เรียน

โครงการนี้ได้นำองค์ความรู้จากหลักสูตร "การพัฒนาがらมคนสมรรถนะสูงด้านปัญญาประดิษฐ์" มาประยุกต์ใช้ในทุกขั้นตอนของการพัฒนา โดยมีการบูรณาการแต่ละ Module เข้ากับฟังก์ชันงานจริง ดังนี้

1. Module : Deep Learning & Neural Networks

- การประยุกต์ใช้ : ความเข้าใจเรื่องสถาปัตยกรรมของโมเดลและการจัดการทรัพยากร
 - การใช้งาน PyTorch เป็น Backend Framework ในการรันโมเดล
 - ความเข้าใจเรื่อง Tensors และ Data Types (bfloat16) ช่วยให้สามารถปรับจูนการใช้ VRAM ของ GPU ให้มีประสิทธิภาพสูงสุด (Memory Optimization)

2. Module : Generative AI & Large Language Models (LLMs)

- การประยุกต์ใช้ : เป็นหัวใจหลักของระบบ (Core Intelligence)
 - Prompt Engineering : การออกแบบคำสั่ง (System Prompt) เพื่อควบคุมให้โมเดลตอบผลลัพธ์ในรูปแบบ JSON ที่ถูกต้อง (Structured Output) และกำหนดบทบาท (Persona) เป็นผู้เชี่ยวชาญการเงิน
 - Model Selection : การเลือกใช้โมเดล Open Source (Qwen, Llama, Gemma) ที่เหมาะสมกับงานวิเคราะห์ข้อความ
 - Inference Pipeline : การสร้างกระบวนการรันโมเดลแบบ Batch Processing เพื่อเพิ่มความเร็วในการประมวลผล

3. Module : AI Deployment & Visualization

- การประยุกต์ใช้ : การนำโมเดลไปใช้งานจริง (Production)
 - การพัฒนา Web Application ด้วย Streamlit เพื่อสร้าง Dashboard
 - การใช้ Plotly สร้างกราฟิกเชิงข้อมูล (Data Visualization) ที่ชับช้อน เช่น Heatmap และ Interactive Charts

3.3 การพัฒนาและการทดสอบ

3.3.1. แผนการดำเนินงาน (Timeline)

ระยะเวลา (สัปดาห์)	กิจกรรมหลัก (Activity)	ผลลัพธ์ (Deliverables)
วันที่ 1-2	Requirement Analysis & Data Exploration <ul style="list-style-type: none">ศึกษาโครงสร้างเว็บ Investing.comกำหนดขอบเขตข้อมูลและ Features	- ศูนย์ต้นแบบการดึงข้อมูล (Scraper Prototype)
วันที่ 3-4	System Design & Core Development <ul style="list-style-type: none">ออกแบบ Pipelineพัฒนาโมเดล TF-IDF และทดสอบ Prompt LLM	- Architecture Diagram - โค้ดส่วน Backend
วันที่ 5-6	AI Integration & Optimization <ul style="list-style-type: none">เชื่อมต่อ LLM 3 โมเดลปรับจูน Consensus Logic และ VRAM Management	- ระบบวิเคราะห์ Sentiment ที่ทำงานได้จริง
วันที่ 7	Frontend Development <ul style="list-style-type: none">พัฒนาหน้าเว็บด้วย Streamlit	- หน้าเว็บ Dashboard (Home.py)

ระยะเวลา (สัปดาห์)	กิจกรรมหลัก (Activity)	ผลลัพธ์ (Deliverables)
	- สร้างกราฟ Treemap และเขียนต่อข้อมูล	
วันที่ 8	Testing & Documentation <ul style="list-style-type: none"> - ทดสอบระบบรวม (Integration Test) - จัดทำรายงานและคู่มือการใช้งาน 	<ul style="list-style-type: none"> - รายงานฉบับสมบูรณ์ - ระบบพร้อมใช้งาน (Final Product)

3.3.2. การวิเคราะห์ความต้องการ (Requirement Analysis)

- ความต้องการด้านฟังก์ชัน (Functional Requirements)
 - ระบบต้องสามารถดึงข่าวใหม่อัตโนมัติและเมื่อใดก็ได้
 - ระบบต้องระบุ Sector ของข่าวได้ถูกต้อง
 - ระบบต้องให้คะแนน Sentiment (-1 ถึง +1) พร้อมสรุปข่าว
 - ระบบต้องสามารถให้ AI ออกความเห็นเกี่ยวกับข่าวหรือตลาดใน Sector นั้นๆ ได้
 - หน้าเว็บต้องแสดง Heatmap ที่อัปเดตตามไฟล์ข้อมูลล่าสุด
- ความต้องการที่ไม่ใช่ฟังก์ชัน (Non-Functional Requirements):
 - Performance : การประมวลผลด้วย LLM ต้องบริหารจัดการ Memory ไม่ให้เกิด Out of Memory (OOM) บน GPU การ์ดเตี้ยรา
 - Usability : ข้อมูลสรุปต้องกระชับ (ไม่เกิน 2-3 ประโยค) และอ่านเข้าใจง่าย

3.3.3. การออกแบบระบบ (System Design)

คณะกรรมการจัดทำออกแบบระบบโดยยึดหลัก Modular Design แยกส่วนการทำงานชัดเจน

- Backend Module : รับผิดชอบเรื่อง Data Pipeline, NLP Processing และการจัดการ Model Weights เก็บผลลัพธ์ลง CSV
- Frontend Module: รับผิดชอบเรื่องการอ่านไฟล์ CSV มาแสดงผล และการคำนวณ Time-Weight บนหน้าเว็บ เพื่อลดภาระการคำนวณซ้ำซ้อน

3.3.4. การพัฒนา (Implementation)

- Data Pipeline : พัฒนาด้วย Python โดยใช้ Cloudscraper และ Pandas มีการเขียน Logic เช็คไฟล์ที่มีอยู่เดิม (`os.path.exists`) เพื่อทำ Incremental Update
- AI Engine : ใช้ Hugging Face Transformers โดยลดโมเดลแบบ 4-bit/8-bit quantization เพื่อประหยัดทรัพยากร และเขียนฟังก์ชัน `clear_resources()` เพื่อคืนค่าหน่วยความจำ GPU ทุกครั้งที่ประมวลผลเสร็จสิ้น

- Dashboard : ใช้ Streamlit ร่วมกับ Custom CSS เพื่อปรับแต่งหน้าตา (UI) ให้มีความทันสมัยและใช้งานง่าย

3.3.5. การทดสอบ (Testing)

การทดสอบแบ่งออกเป็น 3 ระดับ เพื่อตรวจสอบความถูกต้องและประสิทธิภาพ

1. Unit Testing : ทดสอบฟังก์ชันย่อย เช่น การทำความสะอาดข้อความ (Text Cleaning), การคำนวณสูตร Time-Weight ว่าได้ค่าถูกต้องตามมาตรฐานหรือไม่
2. Integration Testing : ทดสอบการทำงานของข้อมูลตั้งแต่ต้นจนจบ (End-to-End) ว่าเมื่อดึงข่าวมาแล้วข้อมูลถูกส่งต่อให้ LLM และบันทึกลง CSV ได้ถูกต้องโดยโปรแกรมไม่หยุดทำงาน
3. User Acceptance Testing (UAT) : ทดสอบความสมเหตุสมผลของผลลัพธ์ (Qualitative Evaluation) โดยให้มนุษย์อ่านข่าวต้นฉบับเทียบกับบทสรุปและคะแนนที่ AI ให้ เพื่อตรวจสอบว่า AI เข้าใจบริบททางการเงินถูกต้องหรือไม่ (เช่น ข่าวการขึ้นดอกเบี้ยครัวเป็นลบต่อหุ้นกลุ่ม Tech เป็นต้น) ซึ่งผลการทดสอบพบว่าระบบสามารถให้เหตุผลได้สอดคล้องกับหลักการเศรษฐศาสตร์ในระดับที่น่าพอใจ

ผลการดำเนินงาน (Results)

จากการดำเนินโครงการ "AI Market Psychologist" คณะผู้จัดทำได้พัฒนาระบบวิเคราะห์อารมณ์ตลาดและสรุปข่าวสารการเงินแบบเรียลไทม์จากเสียงสมบูรณ์ โดยสามารถแบ่งผลลัพธ์การดำเนินงานออกเป็น 3 ส่วนหลัก ได้แก่ ผลลัพธ์ด้านประสิทธิภาพของระบบ (System Performance), ผลลัพธ์ด้านการวิเคราะห์ข้อมูล (Analytical Results), และผลลัพธ์ด้านส่วนติดตอผู้ใช้งาน (User Interface Results) ดังรายละเอียดต่อไปนี้

4.1 ผลลัพธ์ด้านกระบวนการทำงานอัตโนมัติ (Automated Pipeline Results)

คณะผู้จัดทำได้สร้างระบบ End-to-End Pipeline ที่สามารถรวบรวมและประมวลผลข้อมูลได้โดยอัตโนมัติ โดยผลลัพธ์จากการทดสอบระบบแสดงให้เห็นถึงประสิทธิภาพดังนี้

1. การรวบรวมข้อมูล (Data Ingestion)

- ระบบ Web Scraper (Cloudscraper) สามารถดึงข้อมูลข่าวสารจากแหล่งข้อมูลเป้าหมายได้สำเร็จ โดยสามารถจัดการกับระบบป้องกันบอทและดึงเฉพาะข่าวใหม่ (Incremental Update) ได้อย่างถูกต้อง
- จากการทดสอบดึงข้อมูลย้อนหลัง 7 วัน ระบบสามารถรวบรวมข่าวได้ครอบคลุมทุก Sector สำคัญ และจัดเก็บลงในรูปแบบ Structured Data (CSV) ได้อย่างเป็นระเบียบ

2. การบริหารจัดการทรัพยากร (Resource Optimization)

- ระบบจัดการหน่วยความจำ GPU (VRAM Management) ทำงานได้ตามที่ออกแบบไว้ โดยสามารถรับโมเดลขนาดใหญ่ (14B Parameters) บน GPU เดียวได้โดยไม่เกิดข้อผิดพลาด Out of Memory (OOM) ผ่านกระบวนการเคลียร์ Cache (torch.cuda.empty_cache) หลังการประมวลผลแต่ละ Batch

4.2 ผลลัพธ์ด้านการวิเคราะห์อารมณ์ตลาด (Sentiment Analysis Results)

ผลลัพธ์ที่เป็นหัวใจสำคัญของโครงการคือความแม่นยำและความสมเหตุสมผลของการวิเคราะห์โดย AI (Hybrid LLMs) จากการตรวจสอบผลลัพธ์ในไฟล์ sector_daily_history_7days.csv พบว่าระบบสามารถระบุทิศทางตลาดได้สอดคล้องกับเนื้อหาข่าว ดังตัวอย่างต่อไปนี้

ตัวอย่างที่ 1: การระบุภาวะตลาดขาขึ้น (Bullish Case)

- วันที่: 2 มกราคม 2026
- Sector : Technology
- ผลการวิเคราะห์ : ได้คะแนน Final Daily Score สูงถึง 8.23 (Bullish)
- เหตุผลจาก AI : ระบบระบุว่าได้รับแรงหนุนจากข่าว "Samsung's record-high stock price" และความต้องการซื้อ AI (AI Demand) ที่ยังคงแข็งแกร่ง ซึ่งสอดคล้องกับความเป็นจริงที่หุ้นกลุ่มเทคโนโลยีมีก ตอบสนองเชิงบวกต่อข่าววัตกรรม
- ความเห็นของโmonte (Consensus) : ทั้ง Qwen (8.3), Llama (8.2), และ Gemma (8.2) ให้คะแนนไป ในทิศทางเดียวกัน แสดงถึงความเชื่อมั่นสูง

ตัวอย่างที่ 2: การระบุภาวะตลาดขาลง (Bearish Case)

- วันที่ : 8 มกราคม 2026
- Sector: Energy
- ผลการวิเคราะห์ : ได้คะแนน Final Daily Score ต่ำเพียง 3.40 (Bearish)
- เหตุผลจาก AI : ระบบตรวจจับข่าวลบเกี่ยวกับ "US government's stance" และปัญหากฎหมายของโครงการพลังงานลม (Revolution Wind) ทำให้ภาพรวมดูแย่ลง
- ผลลัพธ์ : การระบุ Sector นี้เป็นสีแดงใน Heatmap ช่วยเตือนนักลงทุนให้ระมัดระวังได้ทันท่วงที

ตัวอย่างที่ 3: การวิเคราะห์แบบถ่วงน้ำหนักเวลา (Time-Weighted Analysis)

- ระบบสามารถแยกแยะได้ว่า เมื่อมีข่าวดีเมื่อ 5 วันก่อน แต่หาก 24 ชั่วโมงล่าสุดมีข่าวร้ายเข้ามาอย่างหนัก คะแนน Sentiment จะถูกดึงลงอย่างรวดเร็ว (Recency Bias) สะท้อนถึง "Real-time Momentum" ได้อย่างถูกต้อง

4.3 ผลลัพธ์ด้านส่วนติดต่อผู้ใช้งาน (User Interface Results)

ระบบ Web Dashboard ที่พัฒนาด้วย Streamlit สามารถแสดงผลข้อมูลที่ซับซ้อนให้เข้าใจง่าย โดยมีฟีเจอร์หลักที่ทำงานได้สมบูรณ์ดังนี้

- Market Heatmap (Treemap)
 - [ภาพหน้าจอ Heatmap จาก Home.py]
 - แสดงภาพรวมตลาดได้ชัดเจน โดยบล็อกสีเขียว (Bullish) และสีแดง (Bearish) ช่วยให้ผู้ใช้งานระบุ Sector ที่น่าสนใจได้ภายในเวลาไม่กี่วินาที
 - ขนาดของบล็อกแปรผันตามปริมาณข่าว (News Volume) ทำให้ทราบว่า Sector ใดกำลังเป็นกระแสความสนใจ
- Interactive Drill-down
 - เมื่อคลิกที่ Sector บน Heatmap ระบบสามารถนำทางไปยังหน้ารายละเอียด (Sector_Detail) เพื่อแสดงกราฟแนวโน้มย้อนหลังและรายการข่าวที่เกี่ยวข้องได้
- News Summarization
 - ส่วนแสดงผลข่าว (News Feed) มีการแสดงແطبสีด้านข้างตามระดับ Sentiment และแสดงบทสรุปสั้นจาก AI (Short Answer) ทำให้ผู้อ่านจับใจความสำคัญได้โดยไม่ต้องกดเข้าไปอ่านข่าวเต็ม

4.4 การเปรียบเทียบผลการดำเนินงานกับวัตถุประสงค์

จากการดำเนินงานทั้งหมด คณฑ์จัดทำได้เปรียบเทียบผลลัพธ์ที่ได้กับวัตถุประสงค์ที่ตั้งไว้ในบทที่ 1 ดังตาราง เปรียบเทียบต่อไปนี้

วัตถุประสงค์ของโครงการ	ผลการดำเนินงานที่ได้	สถานะ
1. เพื่อพัฒนาระบบรวมและวิเคราะห์ข่าวสารการเงินอัตโนมัติ (Automated Pipeline)	ระบบสามารถดึงข่าว คัดแยก และวิเคราะห์ได้อย่างต่อเนื่องแต่ต้นจนจบ (End-to-End) เพียงแค่รันคำสั่งเดียว ลดภาระงานมนุษย์ได้ 100% ในกระบวนการ Data Processing	ผ่าน
2. เพื่อประยุกต์ใช้เทคโนโลยี AI แบบผสมผสาน (Hybrid AI)	ระบบมีการใช้งาน TF-IDF เพื่อคัดกรองเบื้องต้น และใช้ LLMs (Qwen/Llama) เพื่อวิเคราะห์เชิงลึก เป็นการผสานข้อมูลด้านความเร็วและความฉลาดเข้าด้วยกัน	ผ่าน
3. เพื่อสร้างกลไกการให้คะแนนความเชื่อมั่นที่มีมาตรฐาน (Consensus Scoring)	ระบบใช้ 3 โมเดลช่วยกันให้วัดคะแนน และมีการคำนวณ Weighted Average ที่ลดความลำเอียงของโมเดลเดียวได้จริง ดังที่ปรากฏในไฟล์ Log การวิเคราะห์	ผ่าน
4. เพื่อพัฒนาเครื่องมือแสดงผลข้อมูล (Dashboard) ที่เข้าใจง่าย	ได้ Web Application ที่แสดงผลแบบ Visual Data (Heatmap) ผู้ใช้งานสามารถดูภาพรวมตลาดและเจาะลึกข้อมูลได้ในแพลตฟอร์มเดียว	ผ่าน

สรุปผลและข้อเสนอแนะ (Conclusion & Future Work)

บทนี้เป็นการสรุปภาพรวมของการดำเนินโครงการ "AI Market Psychologist" ตลอดระยะเวลาการดำเนินงาน ตั้งแต่ขั้นตอนการวิเคราะห์ปัญหา การออกแบบสถาปัตยกรรมระบบ (System Architecture) การพัฒนาโมเดลปัญญาประดิษฐ์แบบผสมผสาน (Hybrid AI) ไปจนถึงการทดสอบและการนำไปใช้งานจริง รวมถึงการสะท้อนถึงปัญหาและอุปสรรคที่เกิดขึ้น เพื่อนำเสนอเป็นแนวทางในการพัฒนาต่ออยอดในอนาคต

5.1 สรุปผลลัพธ์ (Summary of Results)

คณะกรรมการได้พัฒนาระบบวิเคราะห์อารมณ์ตลาดและสรุปข่าวสารการเงินแบบเรียลไทม์จนสำเร็จลุล่วงตามวัตถุประสงค์ โดยสามารถสรุปผลการดำเนินงานในแต่ละด้านได้ดังนี้

1. ด้านการพัฒนาระบบ (System Development)

- ได้พัฒนาระบบการทำงานแบบอัตโนมัติ (Automated Pipeline) ที่สมบูรณ์ ตั้งแต่การดึงข้อมูล (Scraping) ไปจนถึงการแสดงผล (Visualization) ทำให้ลดภาระงานของมนุษย์ในการติดตามข่าวสารได้อย่างสิ้นเชิง
- ประสบความสำเร็จในการประยุกต์ใช้เทคโนโลยี AI แบบผสมผสาน (Hybrid Approach) ระหว่าง TF-IDF/Cosine Similarity เพื่อความรวดเร็วในการคัดแยกข้อมูล และ Large Language Models (LLMs) หลายโมเดล (Qwen, Llama, Gemma) เพื่อความฉลาดในการวิเคราะห์

2. ด้านประสิทธิภาพการวิเคราะห์ (Analytical Performance)

- ระบบสามารถให้คะแนน Sentiment (-1 ถึง +1) ที่มีความแม่นยำและสอดคล้องกับทิศทางตลาดจริง โดยใช้กลไกฉันทางติด (Consensus Mechanism) ช่วยลดความลำเอียงของโมเดล
- อัลกอริทึม Time-Weighted Score ที่พัฒนาขึ้น สามารถสะท้อนโน้ม-men ตัมของตลาดในปัจจุบัน (Real-time Momentum) ได้อย่างมีประสิทธิภาพ ทำให้ผู้ใช้เห็นภาพแนวโน้มล่าสุดได้ชัดเจน

3. ด้านการใช้งาน (Usability)

- ได้ต้นแบบเว็บแอปพลิเคชัน (Prototype Dashboard) ที่ใช้งานง่าย สามารถแสดงภาพรวมตลาดผ่าน Market Heatmap และมีระบบสรุปข่าวสั้น (Summarization) ที่ช่วยให้นักลงทุนจับใจความสำคัญได้รวดเร็ว

5.2 ปัญหาและอุปสรรคในการดำเนินโครงการ (Problems and Obstacles)

ในการดำเนินงานตลอดระยะเวลาโครงการ คณะกรรมการได้พบปัญหาและอุปสรรคทางเทคนิคที่สำคัญ ดังนี้

1. ข้อจำกัดด้านทรัพยากรесурс (Computational Resource Constraints)

- การรันโมเดลภาษาขนาดใหญ่ (14B Parameters) บน GPU เดียว ทำให้เกิดปัญหาน่าวัยความจำไม่เพียงพอ (Out of Memory: OOM) ในช่วงแรก โดยเฉพาะเมื่อต้องประมวลผลข่าวจำนวนมากพร้อมกัน

2. ความไม่แน่นอนของโครงสร้างเว็บไซต์ (Web Structure Variability)
 - การทำ Web Scraping พับปัญหาโครงสร้าง HTML ของเว็บไซต์ต้นทางมีการเปลี่ยนแปลงหรือมีความซับซ้อน ทำให้บางครั้งไม่สามารถดึงเนื้อหาข่าวได้ครบถ้วน หรือติดปัญหา Anti-bot Protection
3. ความล่าช้าในการประมวลผล (Processing Latency)
 - แม้จะใช้ GPU ช่วยประมวลผล แต่การวิเคราะห์ข่าวจำนวนมหาศาลด้วย LLM ยังใช้เวลาค่อนข้างนาน (ระดับนาที) ซึ่งอาจไม่ทันต่อการเทรดความเร็วสูง (High-Frequency Trading)
4. ข้อผิดพลาดจากการตีความของโมเดล (Model Hallucination & Parsing Error):
 - ในบางกรณี โมเดลอาจตอบกลับมาในรูปแบบที่ไม่ใช่ JSON ตามที่กำหนด (Malformed Output) ทำให้ระบบไม่สามารถนำค่าคะแนนไปคำนวณต่อได้

5.3 แนวทางการแก้ไขและข้อเสนอแนะ (Solutions and Future Work)

แนวทางการแก้ไขที่ได้ดำเนินการไปแล้ว (Implemented Solutions)

1. การบริหารจัดการหน่วยความจำ (VRAM Optimization)
 - แก้ไขปัญหา OOM ด้วยการเขียนฟังก์ชัน clear_resources() เพื่อบังคับเคลียร์ Cache ของ GPU ทันทีหลังจบแต่ละ Batch และปรับลด Batch Size ให้เหมาะสมกับขนาด VRAM
2. การปรับปรุงกระบวนการดึงข้อมูล (Robust Scraping)
 - เปลี่ยนมาใช้ไลบรารี Cloudscraper แทน Requests เพื่อจัดการกับระบบป้องกันบอท และเพิ่ม Logic การตรวจสอบข้อมูลชั้น (Incremental Update) เพื่อลดปริมาณงานที่ไม่จำเป็น
3. การจัดการข้อผิดพลาด (Error Handling)
 - เพิ่มฟังก์ชัน parse_llm_response ที่ใช้ Regular Expression ในการดึงข้อมูล JSON เพื่อให้ระบบมีความยืดหยุ่นเมื่อโมเดลจะตอบกลับมาผิดรูปแบบเล็กน้อย

5.4 ข้อเสนอแนะสำหรับการพัฒนาต่อไป (Future Work)

เพื่อให้ระบบ "AI Market Psychologist" มีความสมบูรณ์และพร้อมสำหรับการใช้งานในเชิงพาณิชย์มากยิ่งขึ้น คณะกรรมการผู้จัดทำขอเสนอแนะแนวทางพัฒนาต่อไปดังนี้

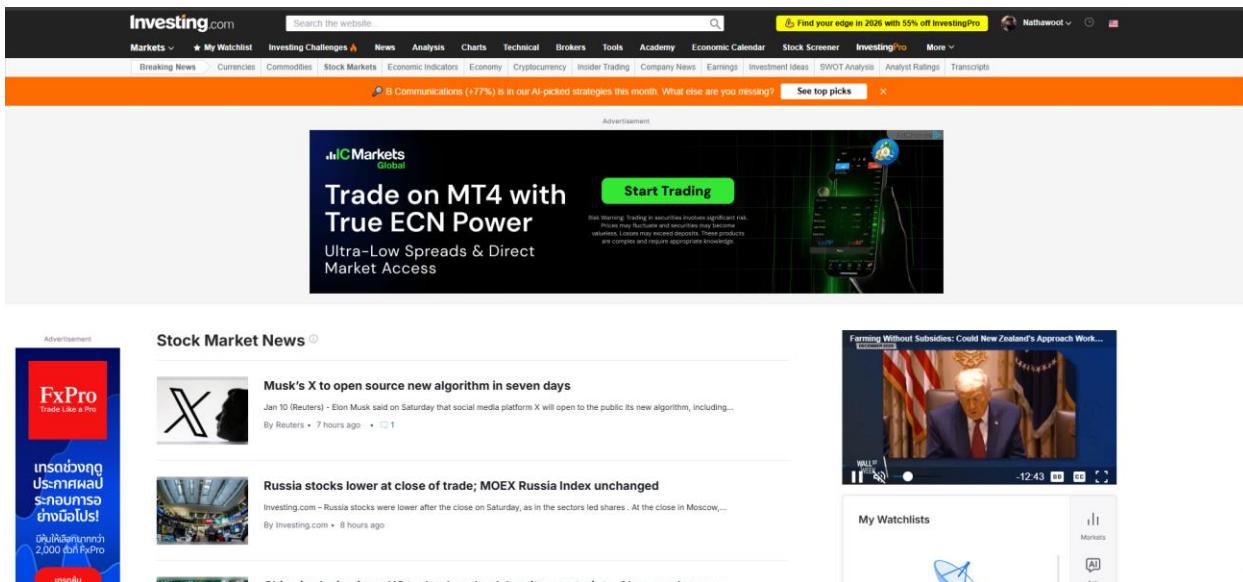
1. การปรับปรุงสถาปัตยกรรมข้อมูล (Data Architecture)
 - ควรเปลี่ยนจากการเก็บข้อมูลลงไฟล์ CSV มาใช้ระบบฐานข้อมูลมาตรฐาน (เช่น PostgreSQL หรือ MongoDB) เพื่อรับข้อมูลจำนวนมากมหาศาลและป้องกันปัญหาข้อมูลชนกัน (Concurrency Issues) เมื่อมีผู้ใช้งานหลายคน
2. การเพิ่มมิติข้อมูล (Multi-Source Integration)
 - ควรขยายแหล่งข้อมูลให้หลากหลายขึ้น เช่น การดึงข้อมูลจาก Twitter (X), Reddit หรือรายงานงบการเงิน (Financial Statements) เพื่อให้การวิเคราะห์ครอบคลุมทุกมิติ
3. การวิเคราะห์ความสัมพันธ์กับราคาจริง (Price Correlation Analysis)
 - ควรเชื่อมต่อ API ราคาหุ้นแบบเรียลไทม์ (เช่น Yahoo Finance API) เพื่อนำมาพล็อตกราฟเปรียบเทียบกับ Sentiment Score ซึ่งจะช่วยยืนยันความแม่นยำของโมเดลและใช้เป็นสัญญาณซื้อขาย (Trading Signal) ได้จริง
4. การปรับจูนโมเดลเฉพาะทาง (Fine-Tuning)

- ควรนำข้อมูลประวัติข่าวและคะแนนที่รวมไว้ด้วย (sector_daily_history.csv) ไปทำการ Fine-Tune โมเดลขนาดเล็ก (เช่น Llama-3-8B) ให้มีความเชี่ยวชาญด้านการเงินโดยเฉพาะ ซึ่งจะช่วยเพิ่มความเร็วในการประมวลผลและลดต้นทุนทรัพยากรลงได้

ภาคผนวก ก

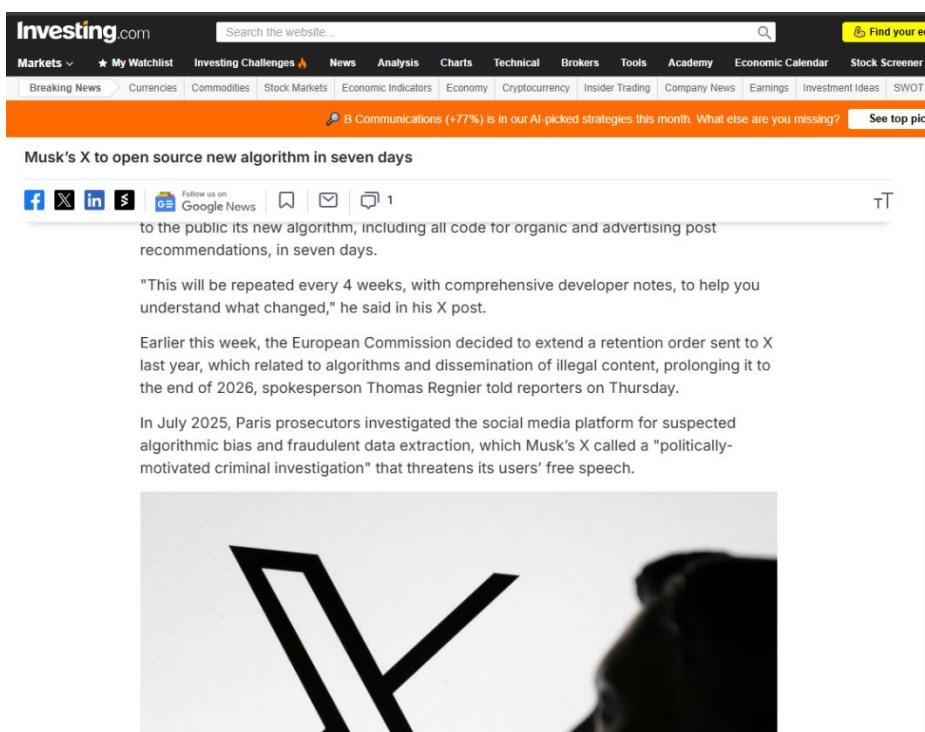
รูปที่ ก-1 : แหล่งข้อมูลข่าวสารทางการเงินจากเว็บไซต์ Investing.com

เป็นหน้าแสดงรายการข่าวล่าสุด (Stock Market News) บนเว็บไซต์ Investing.com ซึ่งใช้เป็นแหล่งข้อมูลต้นทาง (Data Source) หลักของระบบ โดยระบบ Web Scraper จะทำการรวบรวมหัวข้อข่าว เวลาที่เผยแพร่ และลิงก์ข่าวจากหน้านี้เพื่อนำเข้าสู่กระบวนการคัดกรอง



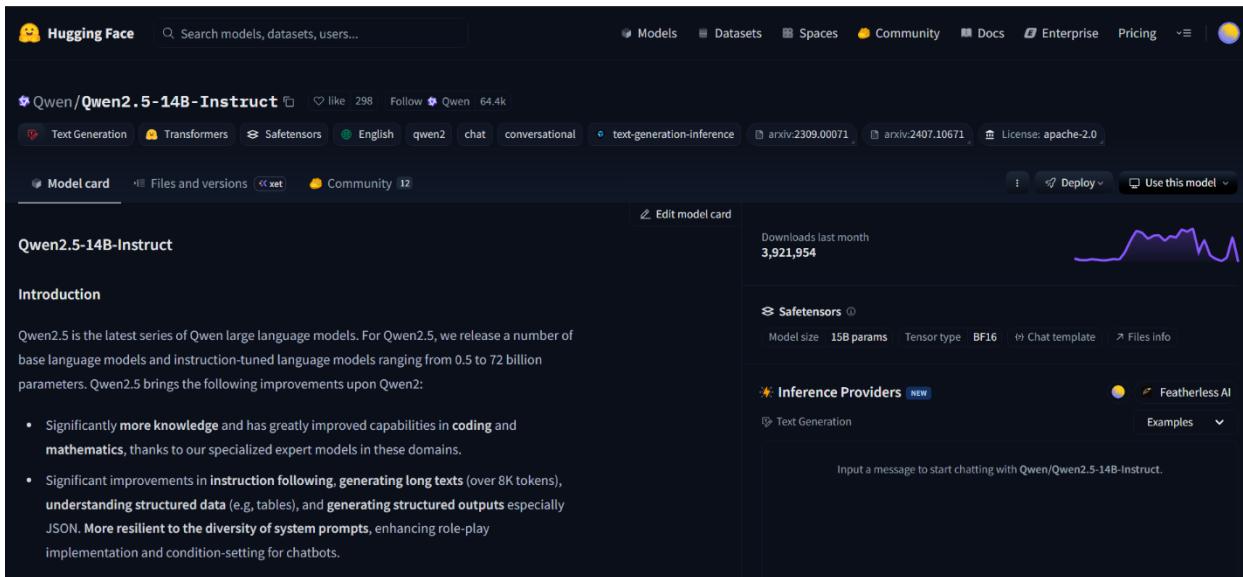
รูปที่ ก-2 : ตัวอย่างรายละเอียดเนื้อหาข่าวที่ใช้ในการวิเคราะห์

ตัวอย่างหน้าแสดงเนื้อหาข่าวฉบับเต็ม (Full Article) ที่ระบบเข้าถึงเพื่อทำการดึงข้อมูล (Extract) เนื้อหาข่าวทั้งหมด (Text Body) โดยตัดส่วนที่ไม่เกี่ยวข้องออก ก่อนนำส่งให้โมเดลภาษาขนาดใหญ่ (LLMs) ทำการสรุปความ (Summarization) และวิเคราะห์อารมณ์ตลาด (Sentiment Analysis)



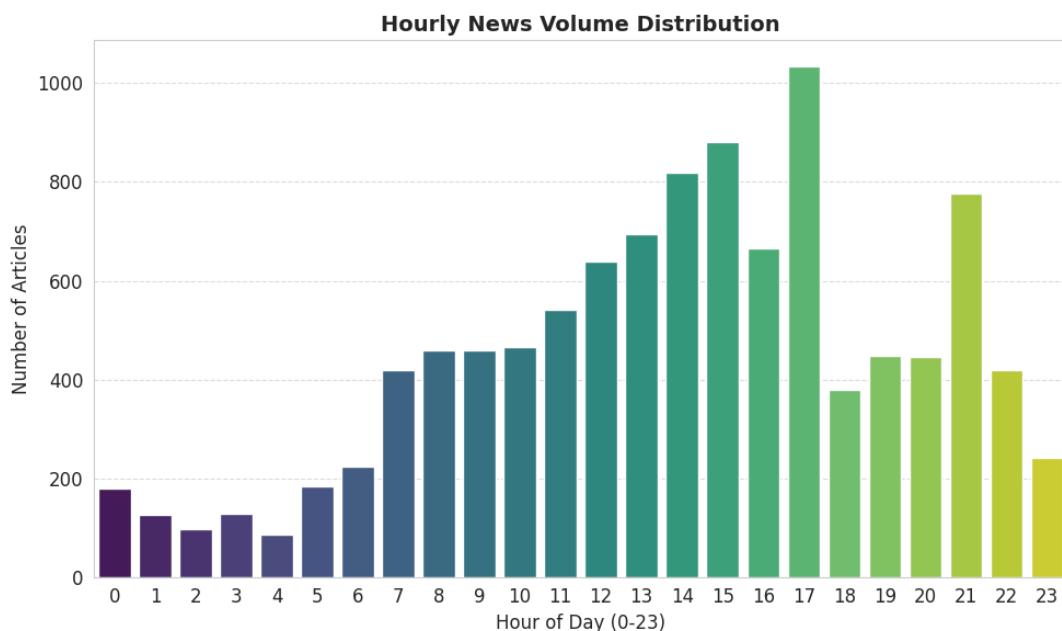
รูปที่ ก-3 : รายละเอียดโมเดลภาษาขนาดใหญ่ Qwen-2.5-14B-Instruct

หน้า Model Card ของ Qwen-2.5-14B-Instruct บนแพลตฟอร์ม Hugging Face ซึ่งเป็นหนึ่งในโมเดลหลักที่เลือกใช้ในโครงการนี้ เนื่องจากมีประสิทธิภาพสูงในการปฏิบัติตามคำสั่งที่ชัดเจน (Instruction Following) และรองรับการประมวลผลข้อมูลที่มีโครงสร้าง (Structured Data Generation) ซึ่งจำเป็นต่อการสร้างผลลัพธ์คุณภาพ Sentiment

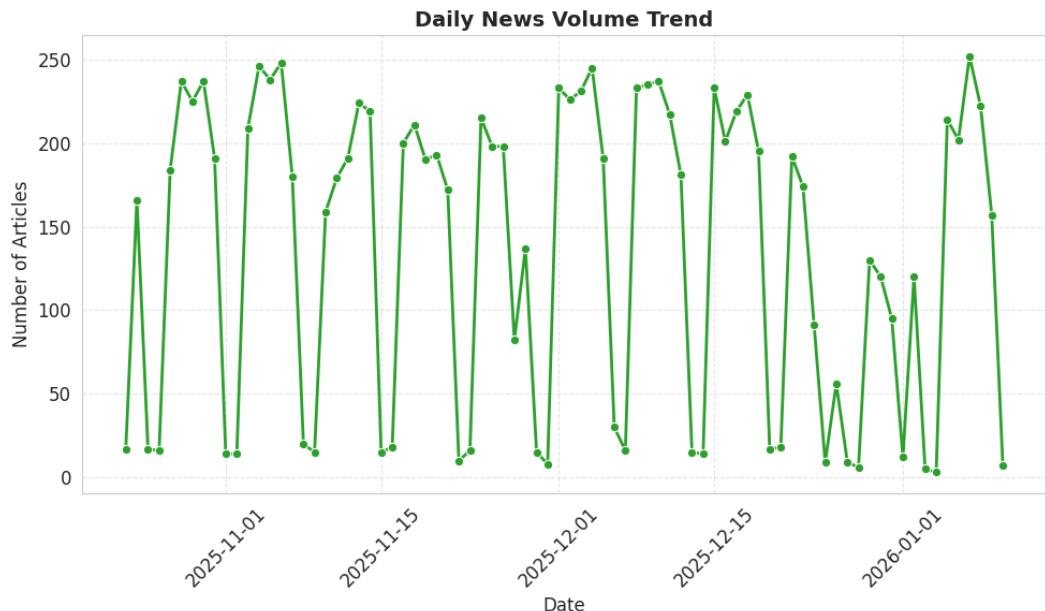


รูปที่ ก-4 : การกระจายตัวของปริมาณข่าวตามช่วงเวลาของวัน (Hourly News Volume Distribution)

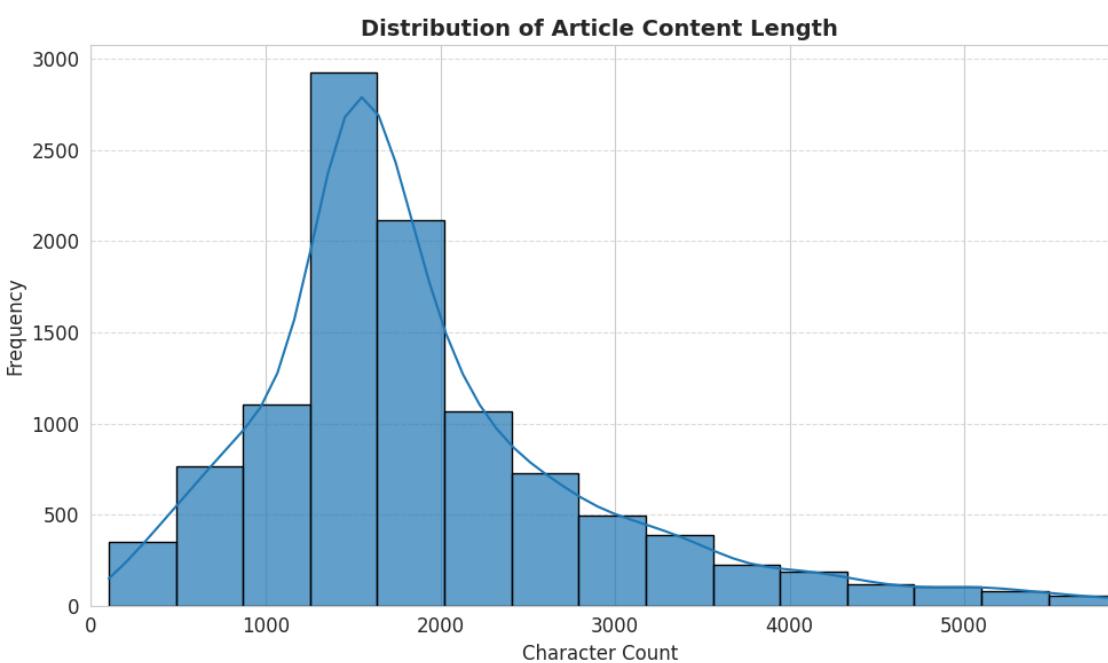
เป็นกราฟแสดงจำนวนข่าวที่ถูกเผยแพร่ในแต่ละชั่วโมงของวัน ข้อมูลแสดงให้เห็นว่าปริมาณข่าวมีความหนาแน่นสูงสุดในช่วงเวลาทำการของตลาดหลัก (Market Hours) ข้อมูลนี้มีความสำคัญต่อการกำหนดตารางการทำงานของระบบ (Scheduler) เพื่อให้จัดสรรทรัพยากรการประมวลผลได้อย่างมีประสิทธิภาพในช่วงที่มีปริมาณข้อมูลสูง



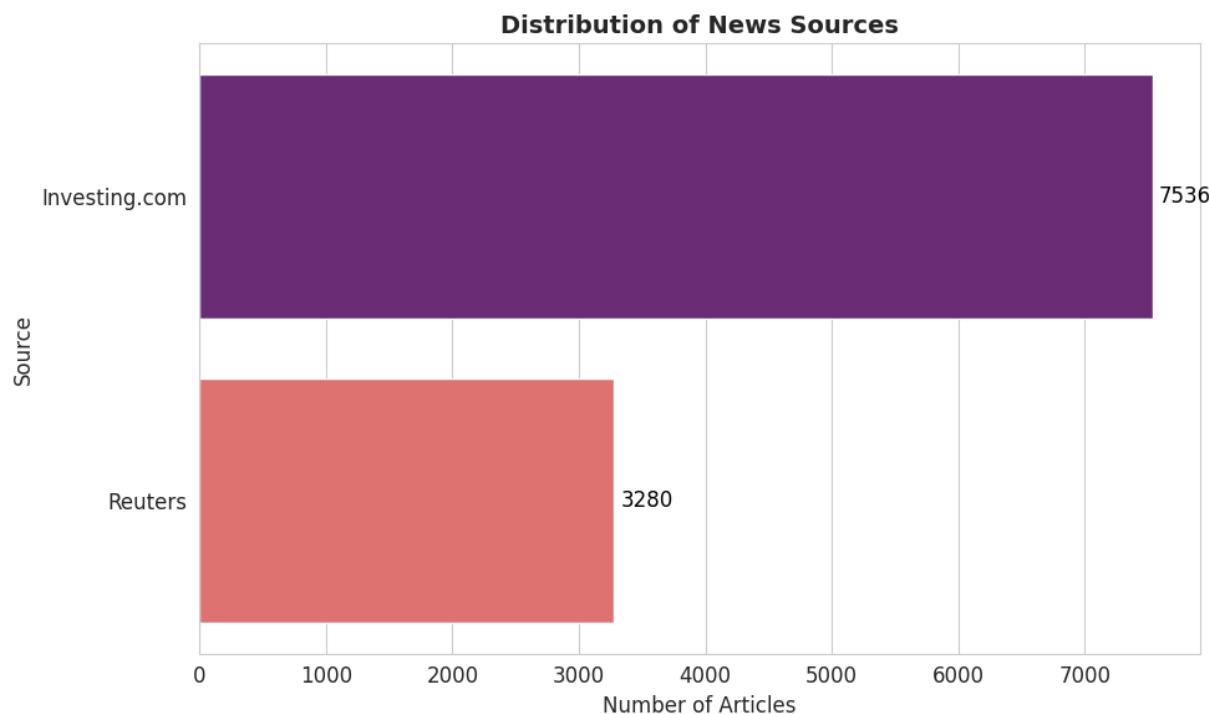
รูปที่ ก-5 : แนวโน้มปริมาณข่าวรายวันตลอดช่วงระยะเวลาการเก็บข้อมูล (Daily News Volume Trend)
 กราฟนี้สื้นแสดงปริมาณข่าวที่รวมมาได้ในแต่ละวัน และงให้เห็นถึงความผันผวนของข้อมูล (Data Volatility)
 ซึ่งมักเพิ่มสูงขึ้นในช่วงวันที่มีเหตุการณ์สำคัญทางเศรษฐกิจ หรือช่วงวันทำการ (Weekdays) และลดลงในช่วง
 วันหยุดสุดสัปดาห์ ช่วยให้สามารถประเมินขีดความสามารถในการรองรับข้อมูล (Scalability) ของระบบได้



รูปที่ ก-6 : สัดส่วนความยาวของเนื้อหาข่าวและการกระจายตัว (Content Length Statistics)
 กราฟฮิสโตแกรมแสดงการกระจายตัวของความยาวเนื้อหาข่าว (จำนวนตัวอักษร) พบร้าข่าวส่วนใหญ่มีความยาว
 อยู่ในช่วงระดับกลาง ซึ่งหมายความว่าการกำหนด Context Window ของโมเดล LLM ที่ 4096 Tokens
 ช่วยยืนยันว่าโมเดลสามารถอ่านเนื้อหาได้ครบถ้วนโดยไม่ต้องตัดตอนส่วนสำคัญ



รูปที่ ก-7 : สัดส่วนจำนวนข่าวจำแนกตามแหล่งที่มา (Distribution of News Articles by Source)

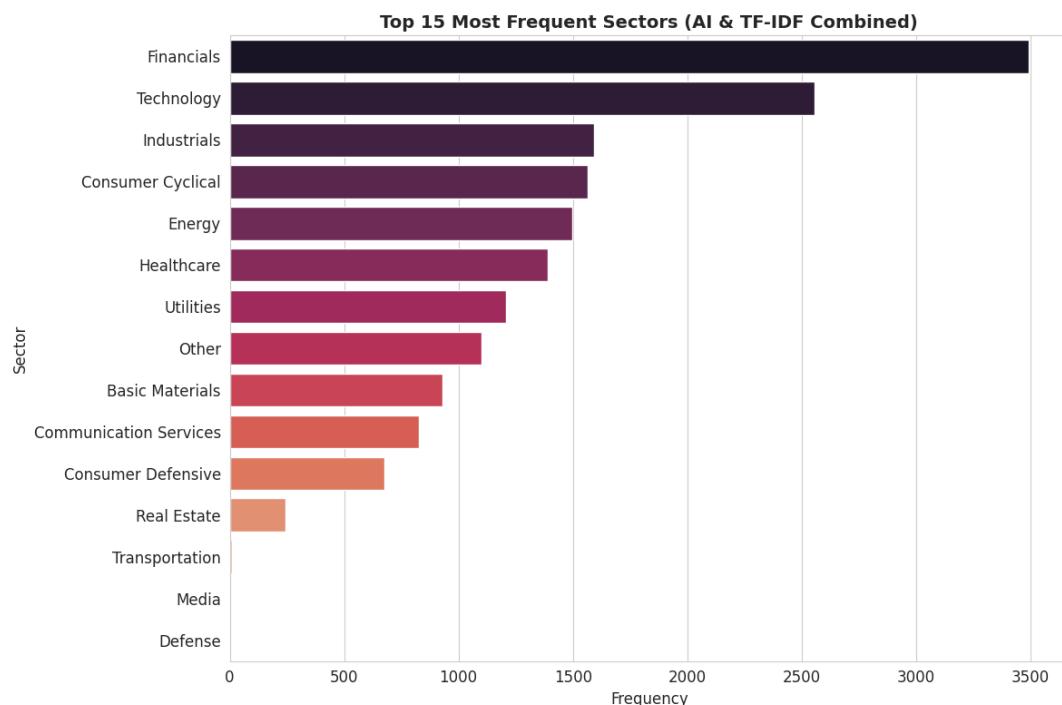


ตารางที่ ก-8 : สถิติจำนวนและร้อยละของบทความแบ่งตามแหล่งที่มา

แหล่งข้อมูล (Source)	จำนวนบทความ (Articles)	ร้อยละ (Percentage)
Investing.com	7,536	69.7%
Reuters	3,280	30.3%
รวมทั้งหมด (Total)	10,816	100.0%

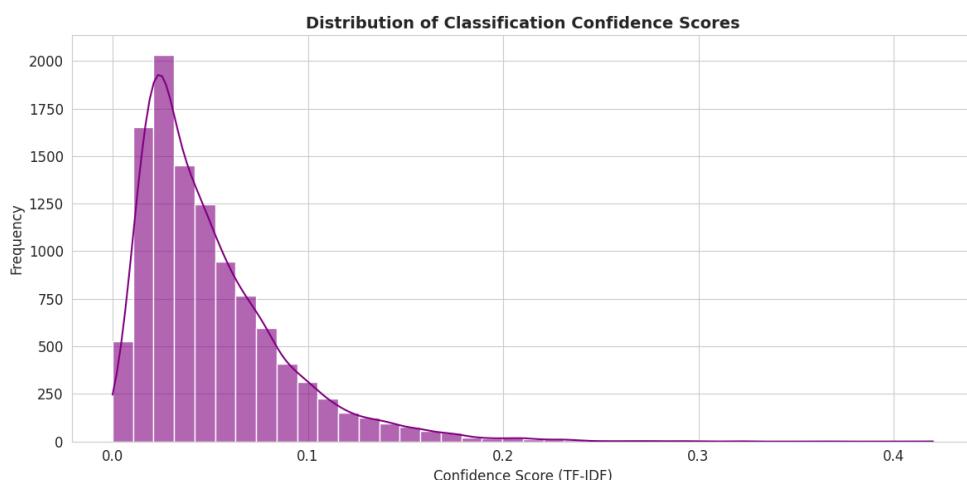
รูปที่ ก-9 : 15 อันดับกลุ่มอุตสาหกรรมที่ปรากฏบ่อยที่สุด (Top 15 Most Frequent Sectors)

ผลลัพธ์จากการจำแนกหมวดหมู่ข่าว (Sector Classification) โดยระบบ AI และ TF-IDF พบว่าข่าวส่วนใหญ่อยู่ในกลุ่ม Technology, Financials และ Energy ซึ่งสอดคล้องกับสภาพตลาดปัจจุบันที่ถูกขับเคลื่อนด้วยกระแส AI และนโยบายดอกรเบี้ย

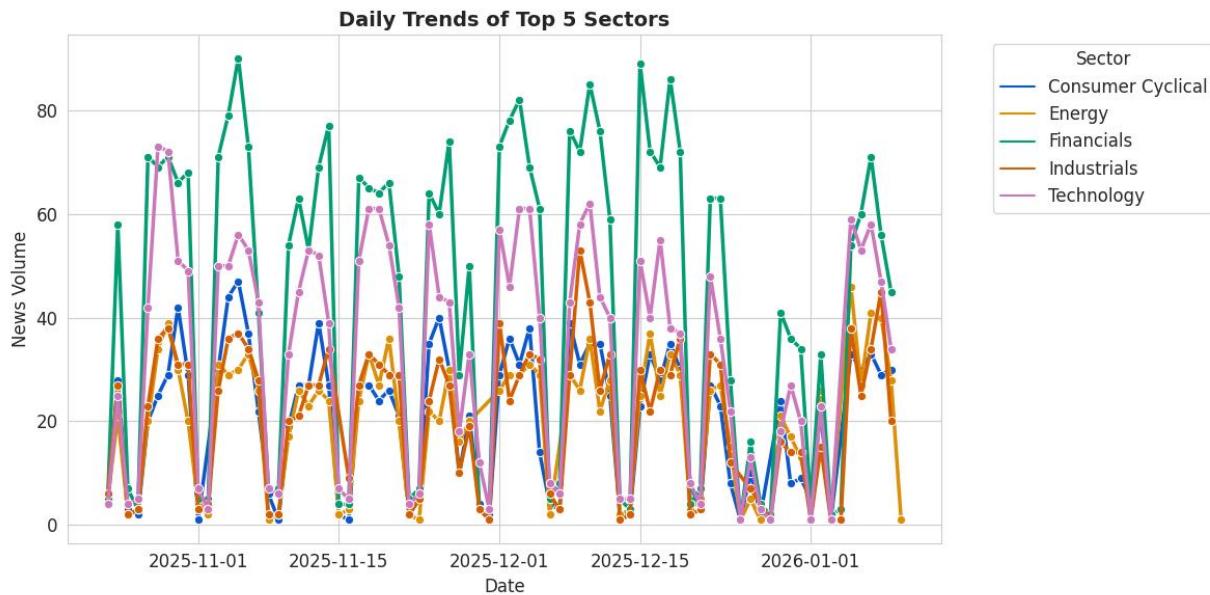


รูปที่ ก-10 : การกระจายตัวของคะแนนความเชื่อมั่น (Distribution of Classification Confidence Scores)

กราฟแสดงค่าความเชื่อมั่น (Confidence Score) ในการระบุหมวดหมู่ข่าว ข้อมูลส่วนใหญ่เป็นไปทางค่าต่ำจากการใช้ TFIDF และให้เห็นว่าระบบ Hybrid AI (TF-IDF ร่วมกับ LLM) เป็นสิ่งที่ควรทำอย่างยิ่ง โดยหาก Confidence Score < 0.02 จะส่งต่อข่าวที่เหลือให้ LLM ทำงานต่อ เพื่อเพิ่มความมั่นในการคัดแยกข้อมูลมากยิ่งขึ้น

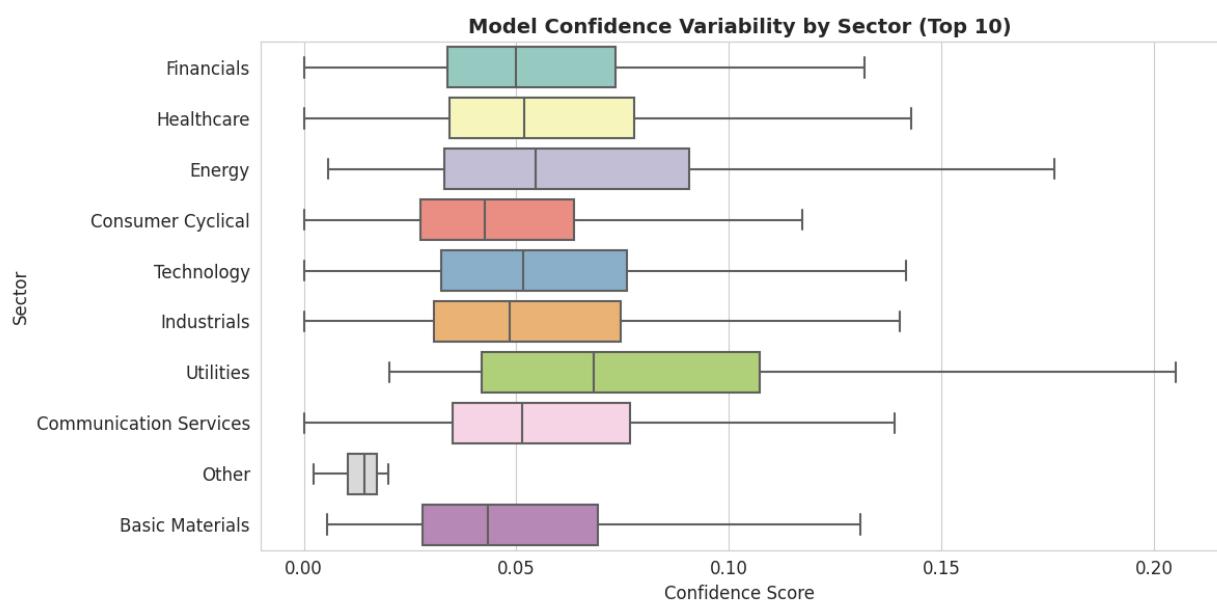


รูปที่ ก-11 : แนวโน้มความสนใจรายวันของ 5 กลุ่มอุตสาหกรรมหลัก (Daily Trends of Top 5 Sectors)
 กราฟนี้แสดงเส้นแนวโน้มปริมาณข่าวรายวันของ 5 อุตสาหกรรมหลัก ซึ่งสะท้อนให้เห็นถึง "การหมุนเวียนกลุ่มลงทุน (Market Rotation)" ในแต่ละช่วงเวลา ว่าความสนใจของตลาดกำลังไปที่กลุ่มใด



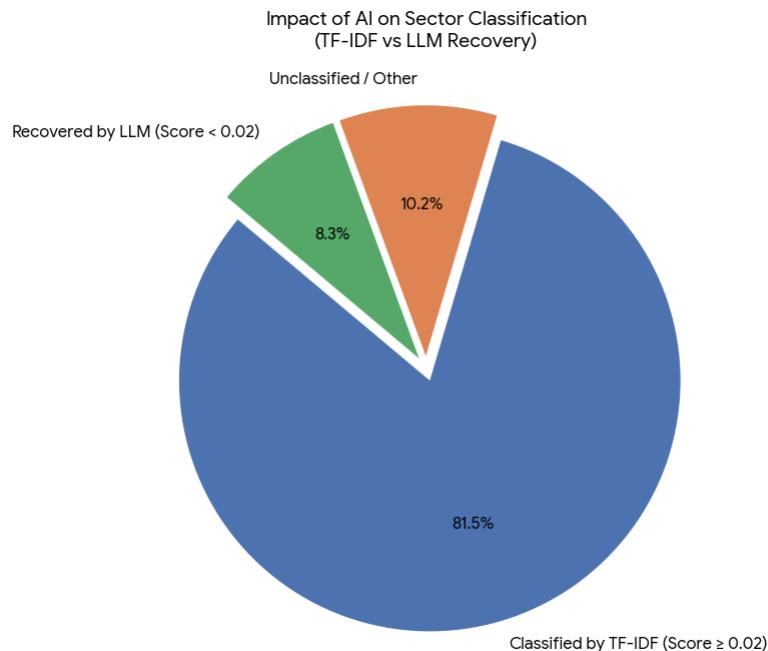
รูปที่ ก-12 : ความแปรปรวนของความเชื่อมั่นจำแนกตามรายอุตสาหกรรม (Model Confidence Variability by Sector)

กราฟ Boxplot แสดงระดับความมั่นใจของโมเดลในแต่ละ Sector พบรากุล Technology และ Financials มีค่าความเชื่อมั่นที่เกากลุ่มกันแน่นในระดับสูง ขณะที่บางกลุ่มอาจมีความผันผวนมากกว่า ขึ้นอยู่กับความชัดเจนของเนื้อหาข่าว



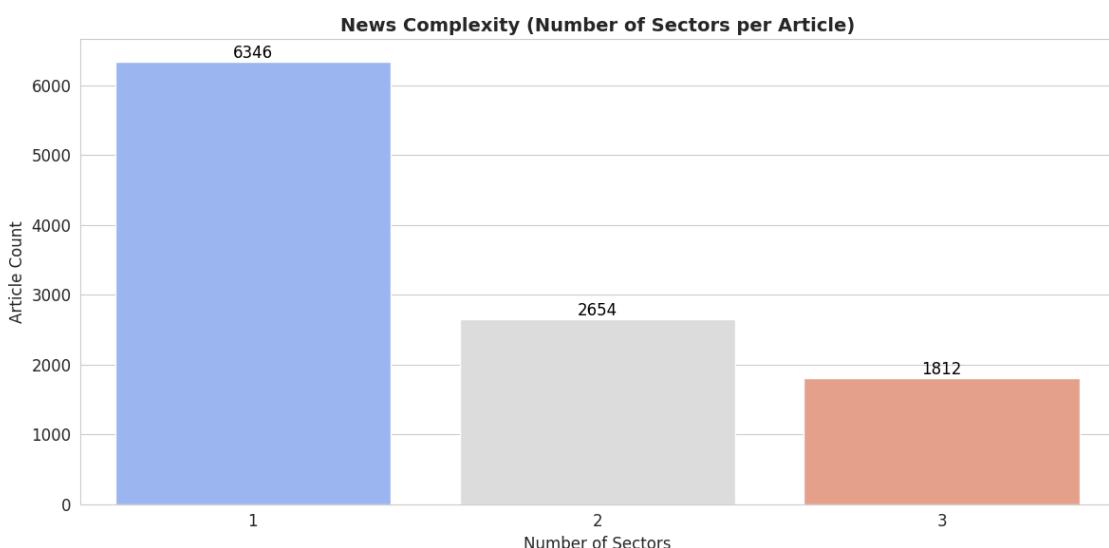
รูปที่ ก-13 : ผลกระทบของ AI ต่อการปรับปรุงการจำแนกหมวดหมู่ (Impact of AI on Sector Classification)

แผนภูมิวงกลมแสดงสัดส่วนบทบาทของเทคโนโลยีในการจำแนกหมวดหมู่ พบร่วมกัน TF-IDF จะเป็นกลไกหลัก (Classified by TF-IDF) แต่ AI (LLM) มีบทบาทสำคัญในการเข้ามา กู้คืนข้อมูล (AI Recovered) ในส่วนที่คะแนนความเชื่อมั่นต่ำกว่าเกณฑ์ (< 0.02) ให้กลับมาใช้งานได้ ซึ่งลดอัตราการสูญเสียข้อมูล (Data Loss) จากการคัดกรองผิดพลาดได้อย่างมีนัยสำคัญ



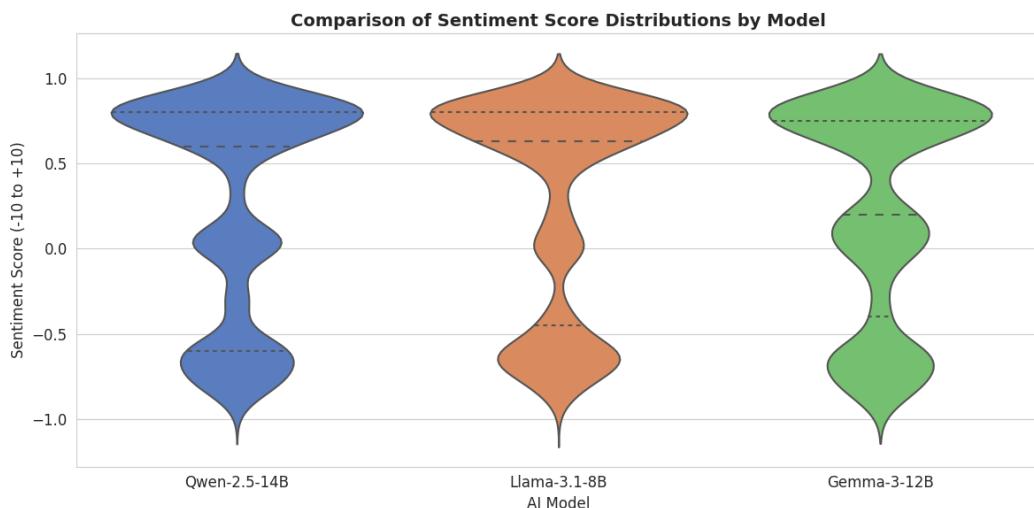
รูปที่ ก-14 : ความซับซ้อนของเนื้อหาข่าว (News Complexity)

กราฟแสดงจำนวน Sector ที่เกี่ยวข้องต่อหนึ่งข่าว พบว่าข่าวการเงินส่วนใหญ่มักมีความซับซ้อนและเกี่ยวข้องกับหลายอุตสาหกรรมพร้อมกัน (Multi-sector Relevance) ซึ่งระบบถูกออกแบบมาให้รองรับลักษณะข้อมูลนี้ได้อย่างถูกต้อง



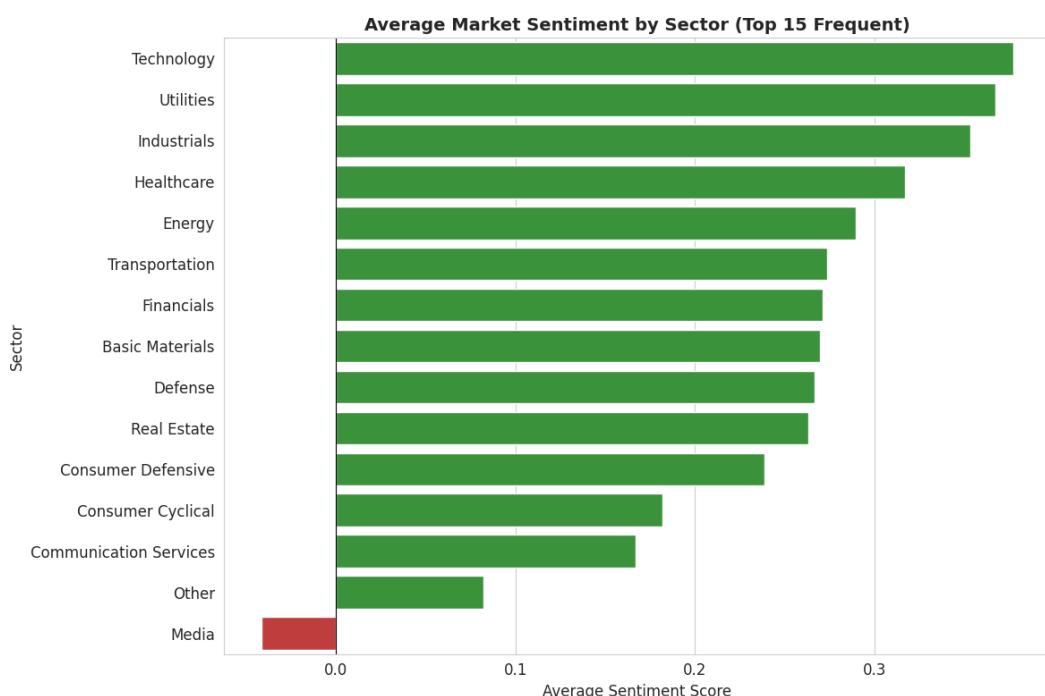
รูปที่ ก-15 : การเปรียบเทียบการกระจายตัวของคะแนนอารมณ์จาก 3 โมเดล (Comparison of Sentiment Score Distributions by Model)

กราฟ Violin Plot แสดงลักษณะการกระจายตัวของคะแนน Sentiment จากโมเดลภาษาขนาดใหญ่ 3 ตัว (Qwen-2.5, Llama-3.1, Gemma-3) พบว่าโมเดลส่วนใหญ่มีการกระจายตัวของคะแนนที่คล้ายคลึงกัน (Consistent Distribution) โดยมีค่ามัธยฐานอยู่ใกล้ศูนย์ แต่ Qwen-2.5 มีแปรโน้มที่จะให้คะแนนที่มีความหลากหลาย (Variance) สูงกว่าเล็กน้อย ซึ่งช่วยเพิ่มมิติในการวิเคราะห์เมื่อนำมาหาค่าเฉลี่ยร่วมกัน



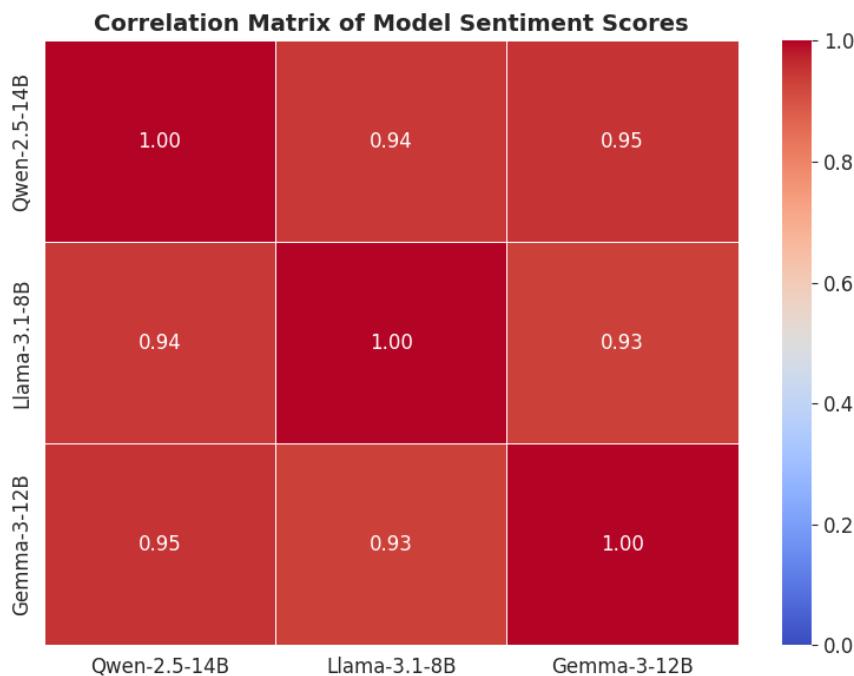
รูปที่ ก-16 : ดัชนีอารมณ์ตลาดเฉลี่ยจำแนกตามรายอุตสาหกรรม (Average Market Sentiment by Sector)

แผนภูมิแท่งแสดงค่าเฉลี่ยคะแนน Sentiment ของ 15 กลุ่มอุตสาหกรรมที่มีการกล่าวถึงสูงสุด โดยจำแนกสีตามทิศทางอารมณ์ (สีเขียว = เชิงบวก, สีแดง = เชิงลบ) ข้อมูลนี้ช่วยบ่งชี้สภาพแวดล้อม (Market Breadth) ว่ากลุ่มอุตสาหกรรมใดกำลังได้รับปัจจัยหนุน (Bullish) หรือถูกกดดัน (Bearish) ในช่วงเวลาที่ทำการวิเคราะห์



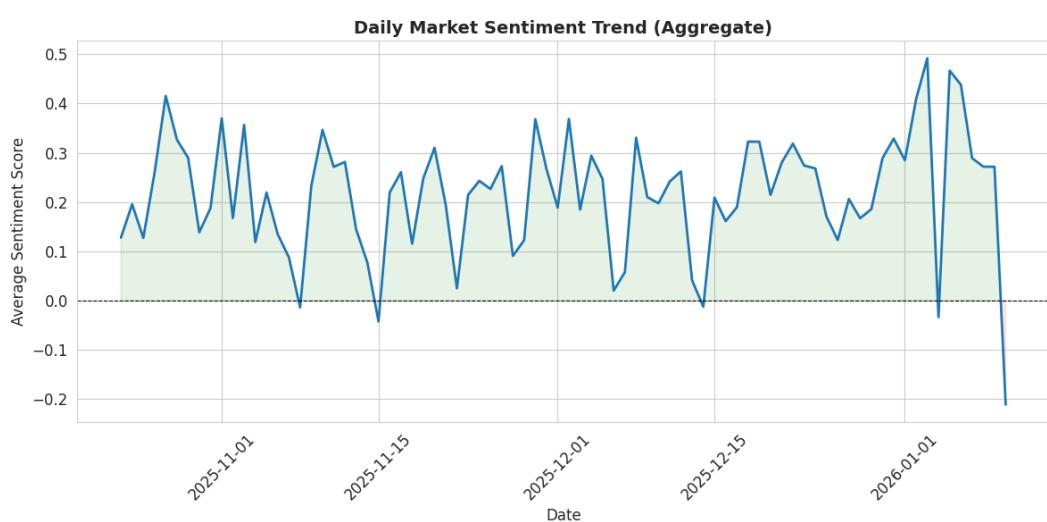
รูปที่ ก-17 : เมทริกซ์สหสัมพันธ์ของคะแนนอารมณ์ระหว่างโมเดล (Correlation Matrix of Model Sentiment Scores)

Heatmap แสดงค่าสัมประสิทธิ์สหสัมพันธ์ (Correlation Coefficient) ระหว่างผลลัพธ์ของโมเดลทั้ง 3 ตัว พบว่า มีค่าความสัมพันธ์เชิงบวกที่สูง (High Positive Correlation) ซึ่งยืนยันถึงความแม่นยำและความสอดคล้อง (Consensus) ของระบบ Multi-Model ในการตีความข่าวสารทางการเงิน



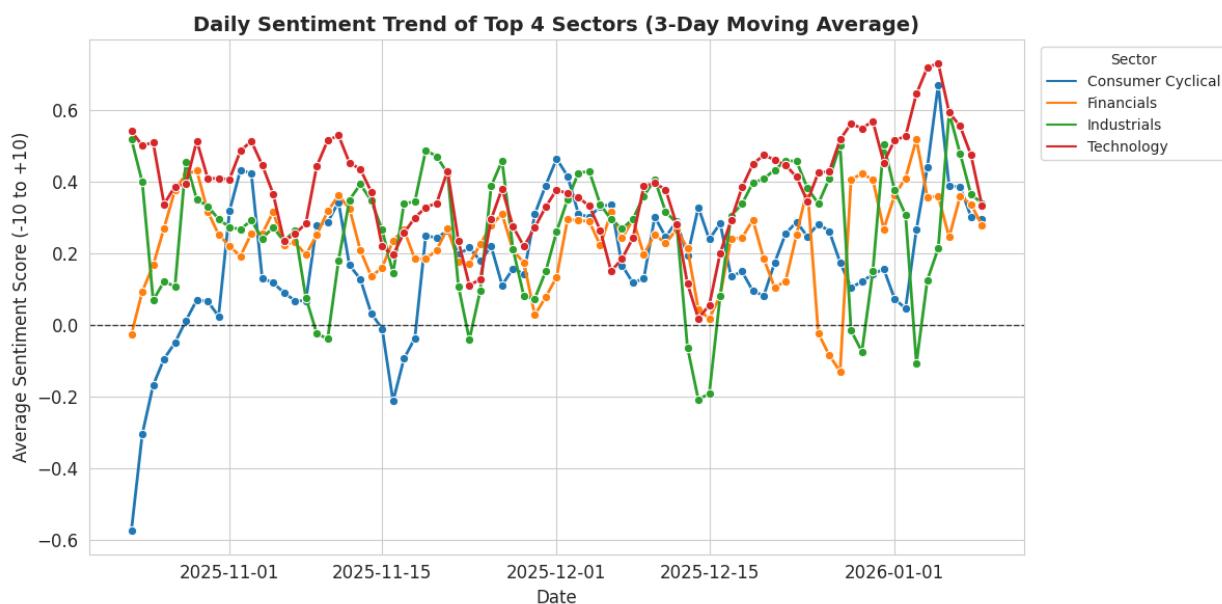
รูปที่ ก-18 : แนวโน้มดัชนีอารมณ์ตลาดรวมรายวัน (Daily Aggregate Market Sentiment Trend)

กราฟแสดงความเคลื่อนไหวของค่าเฉลี่ย Sentiment ตลาดรวมในแต่ละวัน (Aggregate Market Pulse) พื้นที่สีเขียวแสดงช่วงเวลาที่ตลาดมีมุมมองเชิงบวก และพื้นที่สีแดงแสดงมุมมองเชิงลบ ช่วยให้นักลงทุนเห็นภาพรวมวัฏจักรของอารมณ์ตลาด (Market Sentiment Cycle) และจุดเปลี่ยนสำคัญ (Turning Points) ได้อย่างชัดเจน



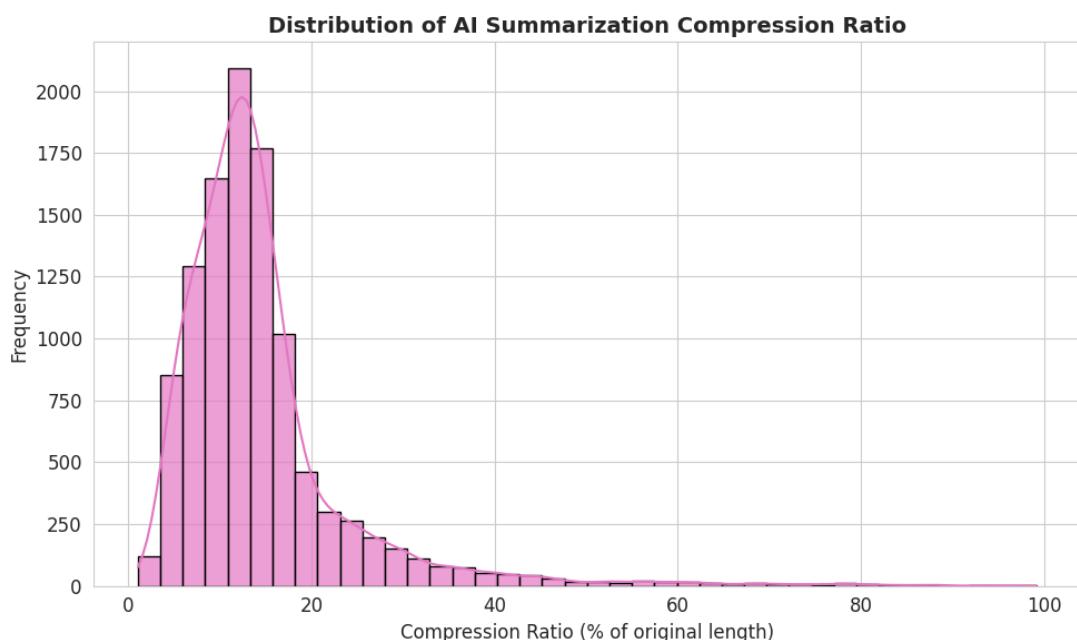
รูปที่ ก-19 : แนวโน้มดัชนีอารมณ์ตลาดรายวันของ 4 กลุ่มอุตสาหกรรมหลัก (Daily Sentiment Trend of Top 4 Sectors)

กราฟเส้นแสดงความเคลื่อนไหวของค่าเฉลี่ยระยะ Sentiment (3-Day Moving Average) ของ 4 อุตสาหกรรม ที่มีบทบาทสูงสุดในตลาด ได้แก่ Financials, Technology, Industrials และ Consumer Cyclical



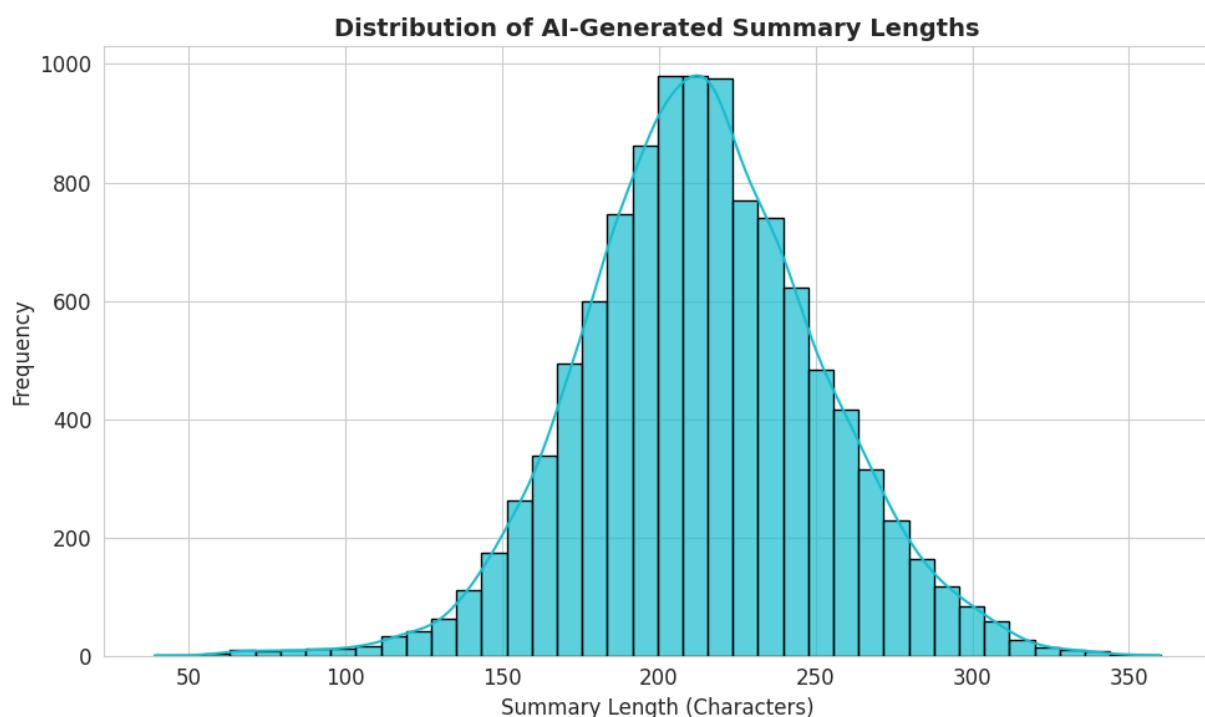
รูปที่ ก-20 : การกระจายตัวของอัตราการย่อความ (Distribution of Summarization Compression Ratios)

กราฟพิสโตแกรมแสดงประสิทธิภาพในการย่อความของโมเดล (Summarization Efficiency) โดยแกนแนวนอนแสดงอัตราส่วนความยาวของบทสรุปต่อบทความต้นฉบับ (Compression Ratio) ข้อมูลชี้ให้เห็นว่าระบบสามารถลดทอนเนื้อหาลงเหลือเพียงประมาณ 10-15% ของความยาวเดิม ซึ่ง有利于ใช้งานสามารถรับทราบใจความสำคัญได้รวดเร็วขึ้นถึง 8-9 เท่า



รูปที่ ก-21 : การกระจายตัวของความยาวที่สรุปโดย AI (Distribution of AI-Generated Summary Lengths)

กราฟแสดงความยาวมาตรฐานของบทสรุป (Summary Length) ที่ระบบสร้างขึ้น โดยส่วนใหญ่มีความยาวอยู่ในช่วง 200 ถึง 400 ตัวอักษร ซึ่งเป็นขนาดที่เหมาะสมสำหรับการแสดงผลแบบ Quick Preview หรือ News Feed โดยไม่สร้างภาระในการอ่าน (Cognitive Load) ให้กับผู้ใช้งานมากเกินไป



ภาคผนวก ข (สไลด์ที่ใช้สำหรับการนำเสนอ)