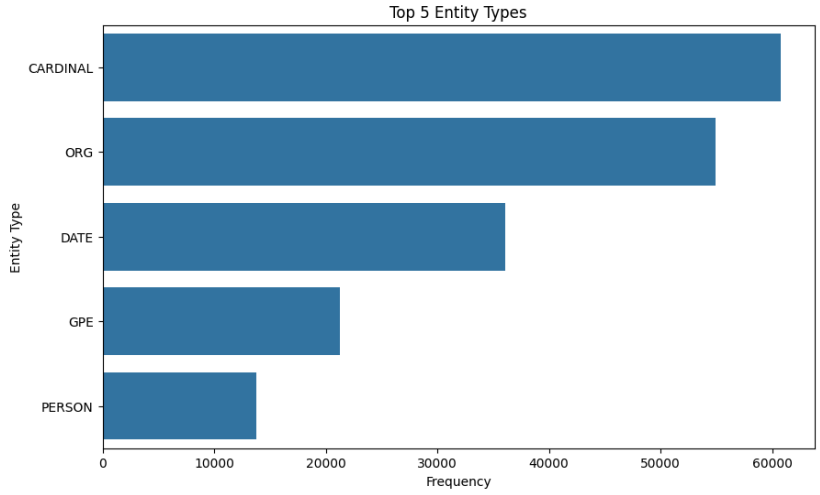
NLP Evaluation Task

Name:Adane Moges
Email : Adanemoges6@gmail.com
Date: June 20, 2025

Overview of Methods

This project implements an NLP pipeline in a Jupyter Notebook (NLP_evaluation.ipynb) to process economic news articles from the NLTK Reuters Corpus (~10,788 documents). **Preprocessing** involves tokenizing text with spaCy, lowercasing, removing stop-words using NLTK, and lemmatizing words to their base forms (e.g., "running" to "run"). A subset of 1000 documents was preprocessed for efficiency, enabling exploratory data analysis (EDA) on document lengths and word frequencies.

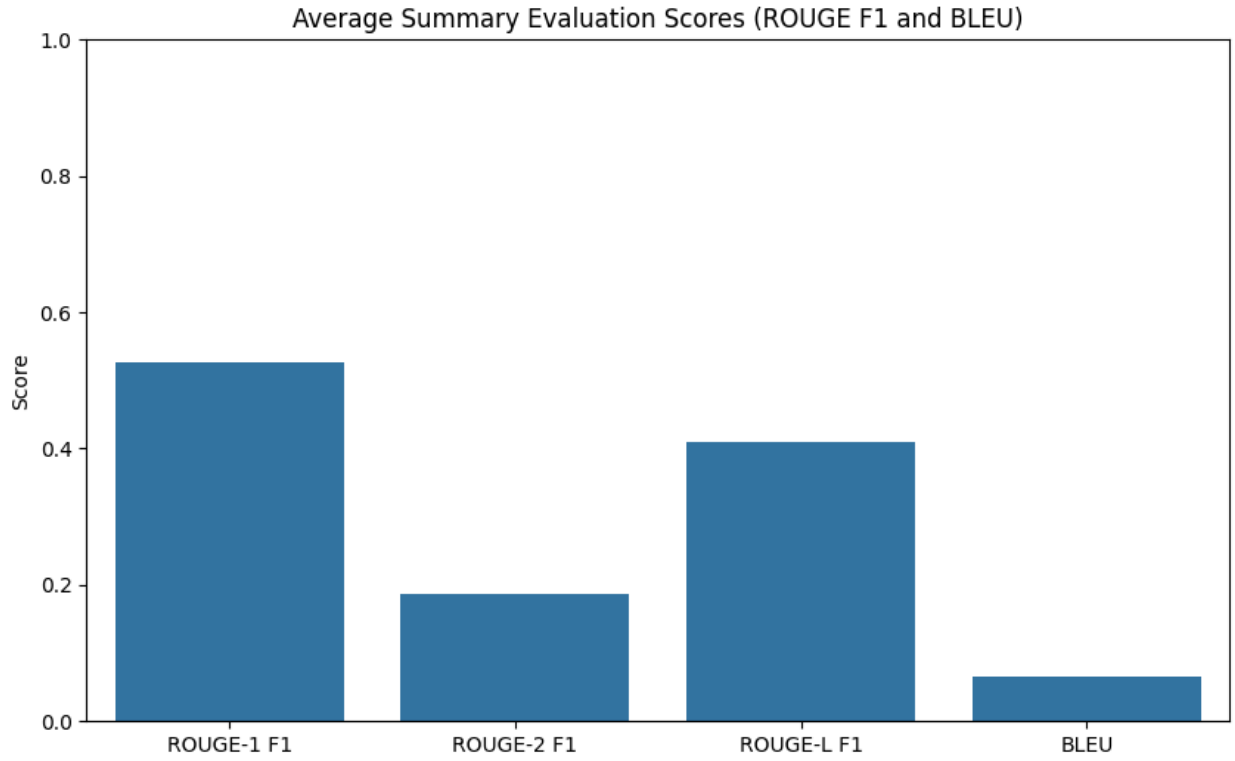
Entity extraction combines rule-based and named entity recognition (NER) approaches. Rule-based extraction uses regex to capture dates (e.g., "15 March 2025") and metrics (e.g., "4.5 billion baht"). NER, via spaCy's en_core_web_sm model, identifies PERSON, ORGANIZATION, GPE, and DATE entities, outputting structured JSON (e.g., {"ORGANIZATION": ["U.S. Federal Reserve"], "metrics": ["0.25%"]}). Extraction was applied to five documents, identifying entities like "Thailand" and "7-12%."



Summarization uses Hugging Face's facebook/bart-large-cnn for abstractive summaries (max 100 words). Documents are truncated to 4000 characters (1024 tokens) to fit BART's input limit. Summaries capture key points (e.g., "Thailand's trade deficit widened to 4.5 billion baht in Q1 1987..."). Four of five documents were summarized successfully; Document 1 failed due to an "index out of range" error.

Discussion of Results, Challenges, and Performance

Results: Preprocessing facilitated EDA, revealing varied document lengths and frequent economic terms (e.g., "market", "trade"). Entity extraction was largely accurate, capturing entities like "Japan" and "4.5 billion baht", but missed some metrics due to restrictive regex patterns. Summaries for Documents 2–5 were coherent and accurate, covering key facts (e.g., China's grain losses, Indonesia's CPO prices). Evaluation against hypothetical reference summaries yielded low scores: ROUGE-1 F1: 0.5275, ROUGE-2 F1: 0.1860, ROUGE-L F1: 0.4107, BLEU: 0.0656.



Challenges:

- *Document 1 Error*: An "index out of range" error during summarization likely occurred due to exceeding BART's token limit or malformed text. Stricter truncation to 1024 tokens is needed.
- *Low Evaluation Scores*: Phrasing differences (e.g., "pct" vs. "%", "mln" vs. "million") and BART's abstractive output reduced ROUGE/BLEU scores. BLEU scores of 0 for Documents 2 and 3 reflect no 3-/4-gram overlaps, addressable with smoothing.
- *Entity Extraction*: NER misclassified terms (e.g., "VERMIN EAT 7-12" as ORGANIZATION), and regex missed metrics like "7-12%".

Performance: BART generated coherent summaries, but required low scores for improvement. Standardizing terms (e.g., "pct" to "%") and fine-tuning on news data could enhance scores. Entity extraction was effective but needs refined regex and post-processing for NER errors. The pipeline efficiently handled five documents, but scaling to real-time news requires optimization (e.g., FAISS for retrieval).

News Aggregator Agent: Use-Case, Workflow, and Architecture

Use-Case: The *News Aggregator Agent* provides concise summaries of breaking economic news, reducing information overload for financial analysts. It monitors sources like the Reuters Corpus (or NewsAPI.org, RSS feeds) for daily digests or query responses (e.g., "Summarize today's news about trade policies"). By filtering relevant articles and minimizing duplication, it delivers timely insights.

Workflow:

1. *Query Parsing*: Extracts keywords (e.g., "trade policies") and entities (e.g., "Asia" as GPE) using spaCy.
2. *Document Retrieval*: Fetches 5–10 articles via vector search (all-MiniLM-L6-v2 for Reuters) or NewsAPI.org, clustering to reduce duplicates.
3. *Content Extraction*: Identifies entities (regex for dates/metrics, spaCy for NER) and salient sentences, ranked by query relevance.
4. *Summarization*: Generates abstractive summaries per article using BART.
5. *Synthesis*: Merges summaries and entities into a report (e.g., "U.S.-Japan tariffs increased Thailand's Q1 1987 deficit to 4.5 billion baht..."), with consistency checks.

Architecture:

- *Query Parser*: spaCy for keyword/entity extraction.
- *News Retrieval*: sentence-transformers or NewsAPI.org for relevant articles.
- *Content Extraction*: regex and spaCy for entities/sentences.
- *Summarization*: BART for concise summaries.
- *Filtering/Indexing*: Optional FAISS to deduplicate articles.
- *Reasoning*: Custom logic or Grok 3 API for synthesis.
- *Memory*: JSON/FAISS stores article metadata and user preferences for deduplication and personalization (e.g., "Since last week's trade query...").

The agent ensures relevance via ranking, reduces redundancy with clustering, and personalizes outputs using memory. Limitations include Document 1's error, low evaluation scores, and the need for real-time news integration. Future work involves fine-tuning BART, enhancing regex, and deploying with FastAPI