R^3 Project Proposal - AIthena

Raahil Dhingra, Rohan Kalani, Ria Mittal

11/17/2025

Question 1: A major challenge in today's media landscape is how younger audiences consume news. About 40% of adults under 30 now rely on social media, where information is brief, visual, and algorithmically curated. Traditional outlets like Morning Brew require subscriptions and deliver uniform content that isn't tailored to individual interests, leaving a gap for concise, personalized updates that match users' attention patterns. Our project addresses this by building a customized AI-generated "Morning Brew" that provides short-form, digestible summaries aligned with each user's professional and personal interests. The target audience includes professionals and students in fast-moving fields such as finance, politics, and technology, who must stay informed but lack time for long-form content. This matters now as shrinking attention spans and social media algorithms distort information quality. Our goal is content curation and personalization, not creating or fact-checking news or offering investment advice. Our service helps users stay efficiently informed without the bias and clutter of social feeds.

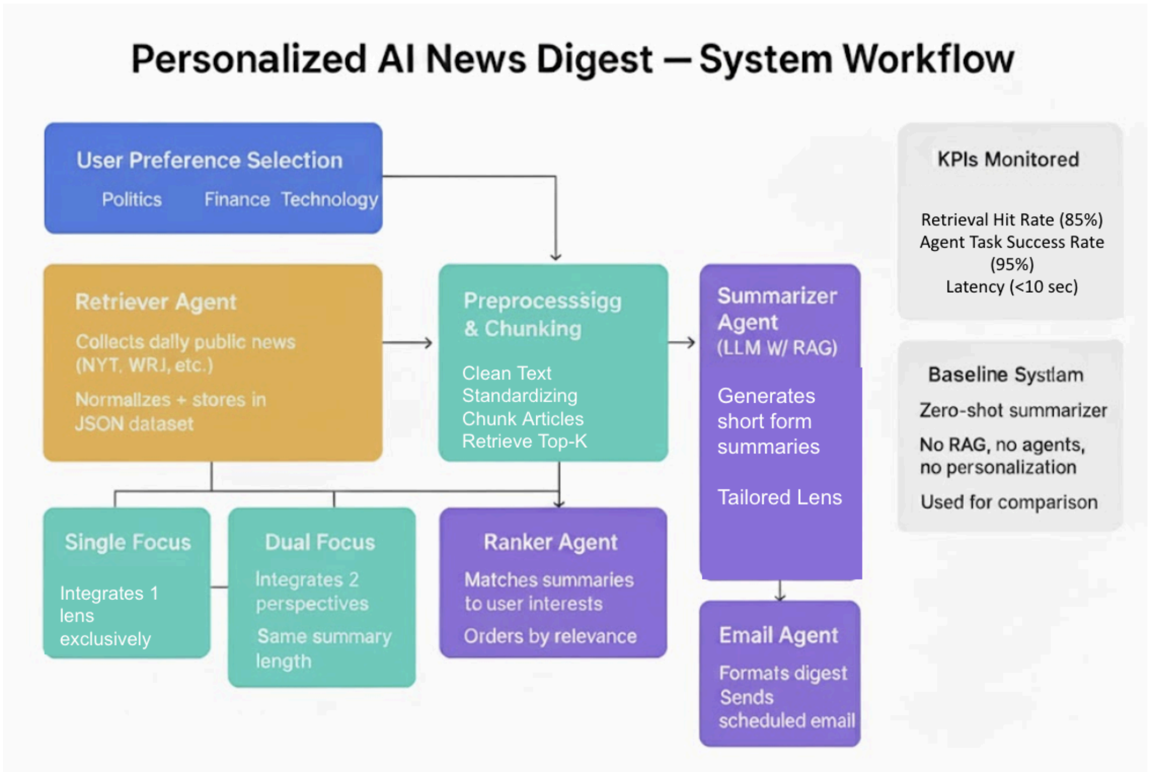
Question 2: This project uses two techniques: Retrieval-Augmented Generation (RAG) and a multi-agent automated workflow. RAG retrieves and grounds summaries in current news sources, ensuring factual accuracy and relevance. A coordinated agent system manages the full pipeline: the Retriever Agent collects new articles, the Summarizer Agent condenses them into concise summaries, the Ranker Agent personalizes content to user interests, and the Email Agent assembles and delivers the digest. Key parameters include the embedding model and chunk size for retrieval accuracy, agent handoff rules, and email scheduling frequency. Together, RAG and multi-agent automation enable timely, personalized news with minimal human input. If time allows, we may add Direct Preference Optimization (DPO) to improve alignment with individual user preferences.

Question 3: "Good enough" means the system consistently delivers accurate, personalized, and timely news digests with minimal human input. It must retrieve the right articles, summarize them faithfully, tailor content to user preferences, and send emails on schedule. Our first KPI is Retrieval Hit Rate, comparing retrieved sources to a labeled set of on-topic articles; ≥80% indicates effective retrieval. The second KPI is Agent Task Success Rate, measuring whether the Retrieve → Summarize → Rank → Email chain completes without intervention; ≥95% is acceptable. We also target <10 seconds per full digest. Our baseline is a zero-shot prompt with no RAG, no agents, and no personalization, which leads to hallucinations, outdated info, and generic tone. Clear improvements over this baseline on both primary KPIs define success.

Question 4: Users begin with a "select all that apply" interface choosing political, financial, and technology as focuses. These preferences guide the Retriever Agent, which feeds aligned content into the RAG pipeline. If a user picks only one focus, summaries reflect that lens alone (e.g., purely financial). If two focuses are selected, summaries integrate both perspectives while staying the same length. If all three are chosen, the emphasis is evenly balanced across them. In all cases, the RAG pipeline keeps summaries factual, current, and consistent by grounding generation in high-quality retrieved context matched to user preferences.

Question 4 (REDONE AFTER FEEDBACK): Users begin with a “select all that apply” interface choosing political, financial, and technology as focuses. These preferences guide the Retriever Agent, which feeds aligned content into the RAG pipeline. If a user picks only one focus, summaries reflect that lens alone (e.g., purely financial). If two focuses are selected, summaries integrate both perspectives while staying the same length. If all three are chosen, the emphasis is evenly balanced across them. In all cases, the RAG pipeline keeps summaries factual, current, and consistent by grounding generation in high-quality retrieved context matched to user preferences. To ensure this multi-agent chain actually works end-to-end, we will define explicit acceptance tests: for example, we will set up test cases with a fixed user preference set and a labeled article corpus, specify the correct retrieved articles and ranked order, and check whether the Retriever → Summarizer → Ranker → Email workflow produces a digest that matches this “gold” outcome. A digest passes if all four agents complete their steps without intervention and the retrieved, summarized, and ranked content aligns with the labeled ground truth. These acceptance tests give us a concrete way to validate the Agent Task Success Rate beyond high-level KPIs.

Question 5: Our system uses public news articles from sources like the New York Times and the Wall Street Journal, collected daily and stored in JSON with consistent fields and tags. These are combined into a unified dataset for the RAG pipeline. We will start with data preparation, then build the RAG pipeline, and finally add agents for retrieval, summarization, ranking, and email delivery. We will use GPT-4 via OpenAI or Azure APIs and tune retrieval parameters such as top-K and metrics like NDCG or Hit Ratio. All processing runs through API calls to manage cost. Main risks include inconsistent formats, outdated data, and mismatched preferences. We mitigate by adjusting chunking and K, adding survey-based preference labels if needed, and normalizing article structure across sources.


Personalized AI News Digest – System Workflow
User Preference Selection
Politics
Finance
Technology
Retriever Agent
Collects daily public news (NYT, WRJ, etc.)
Normalizes + stores in JSON dataset
Preprocesssigg & Chunking
Clean Text
Standardizing
Chunk Articles
Retrieve Top-K
Summarizer Agent (LLM W/ RAG)
Generates short form summaries
Tailored Lens
KPIs Monitored
Retrieval Hit Rate (85%)
Agent Task Success Rate (95%)
Latency (<10 sec)
Baseline Systlam
Zero-shot summarizer
No RAG, no agents, no personalization
Used for comparison
Single Focus
Integrates 1 lens exclusively
Dual Focus
Integrates 2 perspectives
Same summary length
Ranker Agent
Matches summaries to user interests
Orders by relevance
Email Agent
Formats digest
Sends scheduled email

ChatGPT 5.1 Use Cases:

1) "Take these ideas and format them into a cohesive paragraph, do not use any information that is not provided" (Used this for each question, it did not generate ideas)

   → To make sure our proposal sounds neat and structured

2) "How feasible is it to build a agent on VS code and integrate that with frontend?"

   → To understand what our front end would look like

3) "How to connect frontend to email?"

   → This is our final output