© APR 2026 | IRE Journals | Volume 9 Issue 10 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV9I10-1717107

AI Social Media Manager

UTKARSH BARBHAI[1], SOHAM DHAPATE[2], HARSHAD POTDAR[3], SARVESH MOHITE[4]
[1,2,3,4]*Computer Science Engineering, MIT Art, Design & Technology University, Pune, India*

Abstract- ***The management of social media for brands and individuals is a critical but highly inefficient process. Professionals rely on a fragmented suite of tools for content creation, formatting, and distribution, a disconnection that hinders the ability to execute a cohesive strategy. This paper introduces Phase 2 of PostHero, an AI-powered, multi-platform social media manager designed to solve these challenges at scale. Building upon an initial single-platform proof-of-concept, the system's core contribution is a novel, stateful agentic workflow built with LangGraph, which dynamically manages the entire content lifecycle across both Instagram and Facebook simultaneously. This architecture integrates generative AI for text and image creation (via Groq and Pollinations AI), automated cross-platform API routing via FastAPI, and real-time market trend ingestion powered by Pytrends. Furthermore, it introduces a native, zero-dependency asynchronous scheduling engine to manage concurrent API publications without external message brokers. By creating a proactive, data-driven workflow, the system automatically aligns generated multi-platform content with current public interest. Results demonstrate that unifying these core functions through a multi-agent AI significantly streamlines the multi-platform workflow, maintaining a rapid time-to-publish efficiency of under 3 minutes while successfully automating distribution and formatting across distinct social network APIs.***

Keywords—Social Media Management, Generative AI, Agentic Workflow, LangGraph, Multi-Platform Integration, Real-Time Trends.

I. INTRODUCTION

In the modern digital economy, a dynamic social media presence is a non-negotiable asset for businesses and content creators. With over 5.24 billion active users globally and a market size for social media management (SMM) software valued at over $24.47 billion in 2024, these platforms have become the primary channel for marketing, service delivery, and audience engagement. However, the effectiveness of this presence is throttled by a workflow that remains fundamentally fragmented and labor-intensive.

A typical social media manager's day is characterized by constant context-switching between disparate, single-function tools. The workflow is siloed:

1. Ideation & Creation: Content is conceived and then manually created in graphic design platforms like Canva or standalone generative AI tools (e.g., DALL-E, Gemini).
2. Scheduling & Publishing: This finished content is then downloaded and re-uploaded into a separate scheduling application, such as Hootsuite or Buffer.
3. Analysis: Performance metrics are tracked in yet another dashboard, either on the native platform or a third-party analytics tool.

This disconnection is not just a minor inconvenience; it represents a significant drain on resources, with studies indicating that inefficient business processes can cost companies 20-30% of their annual revenue. The most critical gap in this workflow is the lack of an automated, intelligent feedback loop. The insights gleaned from analytics must be manually interpreted and translated by a human operator back into the ideation phase.

While our initial Phase 1 research demonstrated that a unified system could resolve these bottlenecks for a single platform, real-world social media strategies are inherently multi-channel. Expanding distribution across multiple networks, such as integrating Facebook alongside Instagram, exponentially increases the cognitive load and workflow fragmentation. Distinct API constraints, varying content formatting requirements, and isolated analytics ecosystems make cross-platform management highly complex.

To overcome these escalating challenges, we present Phase 2 of PostHero. This paper introduces an expanded, unified, high-performance platform that consolidates the entire lifecycle of a cross-platform social media campaign into a single system. The primary contributions of this work are:

- A Unified Multi-Platform API Backend: The expansion of our single, asynchronous API using fastapi that serves as the central hub, engineered to concurrently handle authentication, strict data validation via pydantic, and payload routing for both the Instagram and Facebook Graph APIs.
- An Agentic AI Core: The refinement of a novel, stateful agentic workflow using langgraph to dynamically route user prompts, manage the content lifecycle, and format assets for multi-platform deployment. This engine leverages high-speed LLMs via the groq API for real-time text and prompt generation.
- Proactive Trend Integration: A major architectural enhancement utilizing pytrends to ingest real-time search interest and trending topics directly into the AI's Graph State, allowing the system to autonomously align generated content with current market trends.
- A Native Asynchronous Scheduling Engine: The development of a custom, lightweight background worker utilizing Python's built-in asyncio integrated within the FastAPI lifecycle. This module independently manages a persistent JSON-based queue to automate cross-platform publication at precise intervals without the computational overhead of external message brokers.
- An Aggregated Feedback Loop: The creation of an advanced analytics-to-recommendation pipeline that processes combined engagement data from multiple platforms, feeding holistic strategic insights directly back into the content generation agent.

II. RELATED WORK

The problem of social media management exists at the intersection of three distinct domains: traditional SMM platforms, standalone generative AI tools, and the underlying software architectures that can integrate them. Our review of the literature focuses on these three areas to identify the evolving critical research gap.

A. Commercial Social Media Management (SMM) Platforms

The SMM platform market is mature, dominated by established players like Hootsuite, Sprout Social, and Buffer. These platforms are highly optimized for multi-channel scheduling, team collaboration, and reactive performance analytics. Their primary value lies in providing a unified dashboard for publishing content across diverse networks and aggregating historical performance metrics to demonstrate ROI. However, these tools were architected before the advent of high-fidelity generative AI. Their recent integrations of AI are often limited to text-based assistance, such as AI caption writers (e.g., Hootsuite's OwlyWriterAI or Buffer's AI Assistant). These features are typically wrappers for large language models (LLMs) like GPT. They do not possess native, high-quality image generation capabilities, thereby requiring the user to perform the fragmented workflow of creating visual assets externally before distribution. Crucially, their analytics are purely reactive—they report on past performance but cannot proactively ingest real-time market trends to autonomously inform the generation of new content.

B. Standalone Generative AI in Content Creation

Concurrently, a separate revolution has occurred in content creation, led by generative models like OpenAI's DALL-E, Google's Gemini, and open-source models like Stable Diffusion. These tools offer unprecedented power to marketers, automating the creation of novel text and images and drastically reducing the time and cost of content production. Academic research has increasingly explored their use for brainstorming and generating marketing campaigns. However, the limitation of these tools is that they are "creative black boxes," completely disconnected from the strategic SMM lifecycle. They lack multi-platform API integration, native scheduling functions, or awareness of real-time search trends. This forces the user to manually bridge

© APR 2026 | IRE Journals | Volume 9 Issue 10 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV9I10-1717107

the cognitive gap, translating external trend research and analytic insights into effective text prompts.

C. Agentic Workflows for System Integration

To bridge the gap between these two siloed domains, a new architectural paradigm is required. Traditional, linear software pipelines (often built with early iterations of tools like LangChain) are insufficient, as they follow a rigid, predefined sequence of steps and cannot easily loop back on themselves. A proactive social media engine requires dynamic data ingestion (such as pulling live search trends) and cyclical refinement. This is where agentic workflows, built using frameworks like LangGraph, present a novel solution. Unlike simple chains, LangGraph is designed to build stateful, multi-agent systems where workflows are represented as graphs. This allows for complex, dynamic behavior, such as a central router delegating tasks to specialized trend-analysis or content-generation nodes, and looping back for error correction if API formatting fails.

D. The Evolving Research Gap

The literature reveals a clear dichotomy: SMM platforms excel at multi-channel distribution but fail at proactive, AI-driven generation; GenAI tools excel at generation but fail at distribution and strategic alignment. In our Phase 1 research, we addressed the initial gap by creating a unified pipeline for a single network. However, the true requirement of modern SMM is multi-platform distribution.

The new, unaddressed research gap for Phase 2 is scaling this stateful agentic workflow to autonomously manage API formatting and concurrent distribution across multiple platforms (specifically navigating the distinct technical constraints of both the Instagram and Facebook Graph APIs) while simultaneously integrating real-time market trend data. Furthermore, existing research lacks examples of executing these complex AI pipelines utilizing native, zero-dependency asynchronous scheduling architectures. Our work demonstrates how a multi-agent AI can be proactively aligned with current public interest and deployed across multiple networks without relying on heavyweight external message brokers.

III. SYSTEM ARCHITECTURE AND IMPLEMENTATION

A. High-Level Architecture

The AI-Powered Social Media Manager is designed using a modular, microservices-based architecture to provide a seamless, high-performance user experience. The system's workflow integrates all core functions—from proactive trend analysis and multi-platform content creation to cross-network distribution—into a single cohesive platform.

This architecture is layered into five primary components:

1. Frontend Layer: A React-based user interface providing dashboards, content calendars, and an AI chat interface.
2. Backend API Layer: A robust Python backend server handling incoming HTTP/REST requests, routing, and lifecycle management.
3. Orchestration Layer: A LangGraph-based central routing system that determines the optimal workflow path for any given request.
4. AI Agent Layer: A suite of specialized AI agents (Strategy, Content, Publishing, Analytics, Compliance, and General) that execute decomposed tasks.
5. External Services & Storage: Integrations with the Instagram and Facebook Graph APIs, image generation via Pollinations AI, high-speed LLM inference via Groq, and persistent local storage.

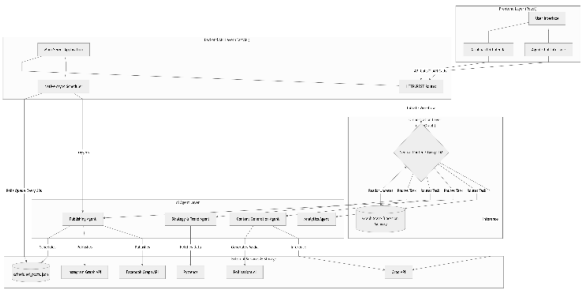

Fig. 1. System architecture of the multi-platform PostHero system, detailing the LangGraph orchestration layer, multi-agent workflow, and native asynchronous scheduling integration.

© APR 2026 | IRE Journals | Volume 9 Issue 10 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV9I10-1717107

B. Technology Stack Justification

The selection of a modern, asynchronous technology stack was a deliberate choice to ensure performance, scalability, and real-time data ingestion.

- Backend (API Server): We chose fastapi as the core web framework. Its asynchronous nature is critical for handling I/O-bound operations, such as concurrent requests to external AI models and social media APIs, without blocking the server. Native integration with pydantic enforces strict data validation for complex cross-platform JSON payloads.
- Core AI Engine: The system's intelligence is built on langgraph. A simple linear chain is insufficient for complex SMM tasks. LangGraph was chosen for its ability to create stateful, cyclical, and agentic graphs, allowing our central router to dynamically delegate tasks. To power this engine, we use the groq API, selected for its exceptionally high-speed, low-latency inference.
- Trend Integration: The analytics module utilizes pytrends to ingest real-time Google search trends directly into the AI workflow, transforming the system from a reactive publisher into a proactive content strategist.
- Analytics and Data Processing: The analytics module relies on the industry-standard pandas and numpy libraries. pandas DataFrames are perfectly suited for ingesting, cleaning, and performing time-series analysis on the JSON-based response data from social media APIs. This allows for the efficient calculation of key metrics, such as the normalized engagement rate, which forms the basis of our recommendation engine.

C. The Multi-Platform Graph State and Session Memory

The entire agentic workflow is built around a central, persistent Graph State. This state functions as a "digital passport" for each request, passed to and updated by every agent in the graph. For Phase 2, this state was significantly expanded to accommodate multi-platform parameters and multi-step agent queues. The state explicitly tracks: user_request, current_agent, workflow_type, agent_queue, generated_content, and the final_response.

To maintain conversational continuity, this Graph State interfaces with a Session Memory System (session_memory.json). This centralized repository stores historical exchanges, allowing the AI to recall user preferences, brand guidelines, and prior context across multiple independent sessions.

D. The Orchestration Layer: Central Router

The LangGraph workflow begins at the Central Router (central_router.py), the intelligent entry point of the system. Utilizing the robust openai/gpt-oss-120b model via the Groq API, this node's sole purpose is to analyze the semantic intent of the user request and make complex routing decisions.

The router dynamically dictates the execution path by defining the workflow_type:

- Direct: A single agent handles the entire task (e.g., answering a simple question).
- Sequential: Multiple agents operate in a specific order (e.g., researching trends, then writing content, then scheduling).
- Parallel: Independent tasks are executed simultaneously to reduce latency. The router compiles an agent_queue and passes the Graph State to the first assigned specialist.

E. The Specialist AI Agents

We implemented a Multi-Agent System where each "agent" is a specialized node equipped with specific tools and localized LLM access (utilizing the meta-llama/llama-4-scout-17b-16e-instruct model). The primary operational agents include:

- Strategy & Trend Agent: Triggered during ideation, this agent utilizes pytrends to analyze live market data and competitor benchmarks, updating the Graph State with highly relevant, proactive context before content generation begins.
- Content Generation Agent: Responsible for the system's creative output. It digests the trend context to generate highly optimized, platform-specific captions for both Facebook and Instagram simultaneously. Concurrently, it triggers an external API call to Pollinations AI (flux model) to synthesize high-fidelity

© APR 2026 | IRE Journals | Volume 9 Issue 10 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV9I10-1717107

accompanying visual assets, storing the media URLs within state.generated_content.
- Publishing Agent: The execution node for distribution. This agent parses user commands (e.g., "schedule for tomorrow at 5 PM") to create formalized entries in the scheduling database, or immediately executes formatting logic to publish concurrent payloads to the Instagram and Facebook Graph APIs.

F. Native Asynchronous Scheduling System

A major architectural contribution of Phase 2 is the custom, zero-dependency background scheduling service (scheduler_service.py). Rather than relying on heavyweight external message brokers (e.g., Celery or Redis), the system employs a native asyncio background loop integrated directly into the FastAPI application lifespan.

- Persistent Queue: The system utilizes a streamlined, file-based JSON architecture (scheduled_posts.json) as a durable, low-latency queue, completely decoupling the act of scheduling from publishing.
- Execution Logic: Operating independently of the main API request cycle, a background worker polls this JSON queue at precise 30-second intervals. It compares the scheduled_time of each pending post against the current server time utilizing a 30-second tolerance window.
- State Machine and Auto-Pruning: The scheduler relies on a strict status flow. Items enter as scheduled. Once the tolerance window is met, the status updates to publishing, and the Publishing Agent autonomously triggers the concurrent API calls. Upon successful publication, the worker automatically removes the record from the active queue, ensuring the persistence layer remains lightweight and highly performant over time.

IV. CORE AI ENGINE

While the system architecture provides the infrastructure for multi-platform distribution, the core intelligence of PostHero resides in a stateful, agentic workflow constructed using langgraph. This architectural choice shifts the system from a rigid, linear script to a cyclical AI engine capable of dynamic decision-making.

A. The Multi-Platform Graph State

The workflow operates around a central, persistent GraphState. Unlike stateless chatbots, this state functions as a digital ledger that is passed to and dynamically updated by every node in the graph, ensuring context is preserved across complex operations. To accommodate Phase 2's multi-platform routing, the GraphState strictly tracks:

- user_request: The raw input string.
- current_agent: The active specialist node executing the task.
- workflow_type: The routing structure (Direct, Sequential, or Parallel).
- agent_queue: The ordered list of agents pending execution.
- session_context: Injected historical interactions and user preferences.
- generated_content: The finalized media URLs and distinct platform-specific text payloads.

B. The Central Router and Decision Logic

The entry point of the graph is the Central Router (central_router.py). Powered by the openai/gpt-oss-120b LLM via the Groq API, this node acts as an intelligent orchestrator.

When a user submits a prompt, the Central Router does not generate content. Instead, it analyzes the semantic intent to formulate an execution plan. It dynamically determines the primary agent, breaks down complex requests to populate the agent_queue, and defines the execution path:

- Direct: A single agent resolves the request.
- Sequential: Agents execute in a strict order (e.g., researching trends, then writing content, then formatting).
- Parallel: Independent agents operate simultaneously to minimize system latency.

C. Specialist Agent Logic

Once routed, execution is handed to a Multi-Agent System (MAS), where each node utilizes localized

© APR 2026 | IRE Journals | Volume 9 Issue 10 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV9I10-1717107

LLM access (meta-llama/llama-4-scout-17b-16e-instruct). The core cognitive agents include:

- Strategy Agent: Executes _analyze_trends() via Pytrends, algorithmically appending proactive market data to the Graph State before generation begins.
- Content Agent: The creative node. It digests the enriched state to execute _generate_content(), applying distinct length and hashtag constraints for Instagram versus Facebook. Concurrently, it triggers _generate_image() to synthesize visual assets via Pollinations AI.
- Publishing Agent: The logic node for distribution. It intercepts the generated_content state and maps it to the respective Graph API payloads or formats it for the asynchronous scheduling queue.

D. Session Memory Integration

To achieve strategic continuity, the AI engine incorporates a centralized Session Memory Manager. Before the agent_queue executes, this module loads persistent conversational history into the session_context of the Graph State. This allows the specialist agents to generate content that aligns not only with real-time trends but also with the historical brand voice and specific user preferences established in prior sessions.

V. RESULTS AND DISCUSSION

The primary result of this Phase 2 research is a fully operational, multi-platform integrated system that successfully automates the entire digital content lifecycle. A user can input a single natural language prompt, which the system uses to dynamically analyze real-time market trends, generate visual and textual content, format it according to distinct platform constraints, schedule it via a native asynchronous engine, and publish it concurrently on both Instagram and Facebook.

A. Quantitative Results: Cross-Platform Workflow Efficiency

We evaluated the system's core claim of improving workflow efficiency by comparing the manual steps and estimated time required to complete a single cross-platform campaign (from ideation to scheduling on both Instagram and Facebook) using a traditional, fragmented workflow versus our unified PostHero system.

The following table demonstrates the compounding inefficiency of traditional methods when scaling to multiple platforms, and how our LangGraph-based agentic system maintains a flat, minimal time-cost regardless of the distribution network.

Task	Traditional Workflow (Fragmented, Multi-Platform)	Our AI-Powered System (Unified, Multi-Platform)
1.Ideation	User manually brainstorms an idea or searches external platforms for trending topics.	User brainstorms an idea, or the Strategy Agent automatically ingests live keyword data via pytrends.
2.Image Gen	1. Open GenAI/Canva. 2. Write & refine prompt. 3. Generate image. 4. Download image file.	1. Write a simple prompt in one unified chat interface.
3.Caption Gen	1. Open AI text assistant. 2. Write & refine caption. 3. Research hashtags for IG.	1. AI engine generates images (via Pollinations AI) & formatted caption choices simultaneously for both IG and FB.

© APR 2026 | IRE Journals | Volume 9 Issue 10 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV9I10-1717107

Task	Traditional Workflow (Fragmented, Multi-Platform)	Our AI-Powered System (Unified, Multi-Platform)
	4. Adjust formatting/links for FB.	
4. Scheduling	1. Open SMM Tool (e.g., Hootsuite). 2. Create IG post, upload image. 3. Copy-paste caption & schedule.	1.User selects their preferred content. 2.User types schedule time (e.g., "tomorrow at 9am"). 3. Click "Schedule."
Est. Time	12 - 20 minutes	1 - 3 minutes
Context Switches	4 Applications, Repeated UI loops	1 Application

Table 1: Comparison of manual multi-platform workflow versus the AI-powered unified workflow.

B. Discussion of Limitations

While Phase 2 successfully expands distribution to both Facebook and Instagram and introduces proactive trend analysis, the prototype retains certain limitations that inform our ongoing development roadmap.

- Media Types and Video Constraints: Currently, the system is optimized for static image generation and publication. Managing multi-platform short-form video content (e.g., Instagram Reels versus Facebook Video), which requires distinct multi-modal video-generation models and differing file-size API constraints, remains outside the current scope.
- Image Generation Fidelity: As noted in our Phase 1 research, the visual output is strictly dependent on the baseline generative model (currently Pollinations AI). Integrating commercial-grade, state-of-the-art models (such as Midjourney or OpenAI's DALL-E 3) remains a priority for achieving hyper-realistic visual assets suited for enterprise marketing.

VI. FUTURE WORK

While Phase 2 of PostHero successfully establishes a unified, multi-platform agentic workflow, the system architecture provides a robust foundation for several advanced expansions. Our ongoing research roadmap focuses on four primary vectors of development:

A. Multi-Modal Video Generation

The current generative pipeline is restricted to static image synthesis. However, the dominance of short-form video (e.g., Instagram Reels, TikTok, YouTube Shorts) dictates that a comprehensive SMM platform must support dynamic media. Future iterations will integrate multi-modal video generation models to automate the creation of temporaly consistent video assets, complete with AI-generated voiceovers and auto-synced captions. This will require significant updates to the Publishing Agent to handle the heavy payload constraints and distinct chunking protocols of various video APIs.

B. Omnichannel Network Expansion

While the concurrent formatting logic successfully navigates the Meta ecosystem (Facebook and Instagram), enterprise marketing requires true omnichannel presence. Future work will expand the central router and payload parsing logic to integrate professional and micro-blogging networks, specifically the LinkedIn API and the X (formerly Twitter) API, requiring the AI to adapt to radically different tones, character limits, and media ratios.

C. Automated Community Engagement and Sentiment Analysis

As outlined in the system architecture, preliminary modules for community management (sentiment_service.py) have been designed but remain outside the scope of Phase 2 publication. Future research will activate an Engagement Agent within the LangGraph network. This agent will asynchronously pull real-time comments from published multi-platform posts, perform sentiment analysis, and autonomously generate and post

contextual replies, closing the loop between content distribution and audience interaction.

D. State-of-the-Art Model Integration

To transition the system from a high-functioning prototype to an enterprise-ready platform, the baseline image generation dependency (Pollinations AI) will be upgraded. By integrating state-of-the-art commercial APIs (such as Midjourney or OpenAI's DALL-E 3), the visual fidelity of the system will match the sophisticated textual capabilities of the underlying Groq-powered orchestration layer.

VII. CONCLUSION

This paper presented Phase 2 of PostHero, an advanced, multi-platform AI Social Media Manager designed to resolve the deep inefficiencies of traditional, fragmented digital marketing workflows. By evolving our initial single-platform prototype into a robust microservices architecture, we successfully demonstrated the system's capability to automate the entire content lifecycle across both Instagram and Facebook simultaneously.

The system's core innovation lies in its orchestration layer. Utilizing LangGraph, the Central Router dynamically evaluates user intent to trigger direct, sequential, or parallel multi-agent workflows. This intelligent routing allows specialized agents to seamlessly handle proactive trend analysis via pytrends, cross-platform content generation, and strict formatting compliance without manual user intervention. Furthermore, the implementation of a native, zero-dependency background scheduling service—utilizing Python's asyncio and a durable JSON state machine—proved highly effective for executing concurrent API publications, eliminating the computational overhead of heavyweight external message brokers.

Ultimately, the quantitative results validate the multi-agent approach to social media management. By reducing the cross-platform publication workflow from an estimated 20 minutes of disjointed manual effort to a unified, AI-driven process taking under three minutes, PostHero represents a significant step toward truly autonomous, proactive, and multi-platform digital marketing systems.

REFERENCES

[1] [1] Z. Xi et al., "The Rise and Potential of Large Language Model Based Agents: A Survey," arXiv preprint arXiv:2309.07864, 2023.

[2] [2] H. Chase, LangChain: Build context-aware, reasoning applications, Software framework. [Online]. Available: https://github.com/langchain-ai/langchain.

[3] [3] A. I. Touvron et al., "Llama: Open and Efficient Foundation Language Models," arXiv preprint arXiv:2302.13971, 2023.

[4] [4] S. Ramírez, FastAPI: A modern, fast (high-performance), web framework for building APIs with Python 3.8+, Software framework. [Online]. Available: https://github.com/tiangolo/fastapi.

[5] [5] Y. K. Dwivedi et al., "Artificial Intelligence (AI): Multidisciplinary perspectives on emerging challenges, opportunities, and agenda for research, practice and policy," International Journal of Information Management, vol. 57, p. 101994, 2021.

[6] [6] H. Choi and H. Varian, "Predicting the Present with Google Trends," The Economic Record, vol. 88, pp. 2-9, 2012.

[7] [7] Meta Platforms, Inc., Instagram Graph API Documentation. [Online]. Available: https://developers.facebook.com/docs/instagram-api.

[8] [8] Meta Platforms, Inc., Facebook Graph API Documentation. [Online]. Available: https://developers.facebook.com/docs/graph-api.

[9] [9] D. C. Parkes and M. P. Wellman, "Economic reasoning and artificial intelligence," Science, vol. 349, no. 6245, pp. 267-272, 2015. (Note: Useful for supporting the claims about workflow efficiency and resource cost reduction).

[10] [10] Pollinations.ai: Image Generation API. [Online]. Available: https://pollinations.ai/.