1. Project Summary

Problem & Motivation

In complex cognitive tasks such as academic writing and structured reasoning, users often suffer from cognitive fixation-early commitment to a single framing or idea. While Large Language Models (LLMs) are intended to assist in such tasks, dominant chat-based interfaces frequently reinforce this issue. By optimizing for the most probable continuation, LLMs encourage passive consumption and maximization bias, leading to homogenized outputs and shallow exploration. Consequently, users are often reduced to editors of AI-generated text rather than active participants in a reasoning process.

Solution: Lantern

Lantern is a collaborative decision-support system designed to shift Human-AI interaction from passive generation to active exploration. Integrated as a sidebar within a document editor, Lantern functions as a Thinking Partner rather than a ghostwriter. The system employs a Tree-of-Thoughts (ToT) interaction framework, externalizing reasoning as multiple distinct paths instead of producing a single linear output. Users navigate, compare, and prune these alternatives, while synthesis and authorship remain entirely human-driven.

Unique Approach: Bridging "Wants" vs. "Needs"

Lantern prioritizes what users need for high-quality reasoning over what they may want for convenience, translating cognitive principles into concrete design constraints.

First, the system introduces productive friction by limiting the number of alternatives shown per interaction and requiring explicit user selection, encouraging deliberate System-2 thinking over autopilot behavior.

Second, Lantern emphasizes challenge over validation through a dedicated Devil's Advocate agent that is structurally constrained to surface counter-arguments and weaknesses, countering sycophancy and automation bias.

Third, the system favors scaffolding over generation: users pin strategic directions that guide their writing, but AI outputs are never inserted directly into the document.

Finally, embedded expertise abstracts prompt engineering by encoding academic standards directly into agent roles, providing expert guidance without increasing user cognitive load.

2. Design Principles

Our system design is guided by three core HAI guidelines derived from the **Google PAIR Guidebook**. These principles were selected to specifically address the problem of cognitive fixation and to ensure the user remains the primary decision-maker.

1. Balance Automation & Augmentation (User Needs)

- **The Principle:** The guidebook suggests favoring augmentation over full automation when the goal is to enhance human creativity and responsibility.

- **Application:** Lantern rejects the "Ghostwriter" paradigm in favor of a "Thinking Partner" approach. The system augments the user's reasoning by expanding the solution space , while the user retains the cognitive responsibility of synthesis and decision-making.

2. N-best Alternatives (Explainability + Trust)

- **The Principle:** In scenarios involving subjective judgment, presenting multiple distinct options prompts users to apply their own judgment rather than blindly trusting the AI.
- **Application:** This is the logic behind our **ToT interface**. By visualizing multiple distinct branches at every node, we mitigate "Automation Bias." This forces the user to engage in System-2 thinking - actively comparing and evaluating options - rather than passively accepting a single, linear default output.

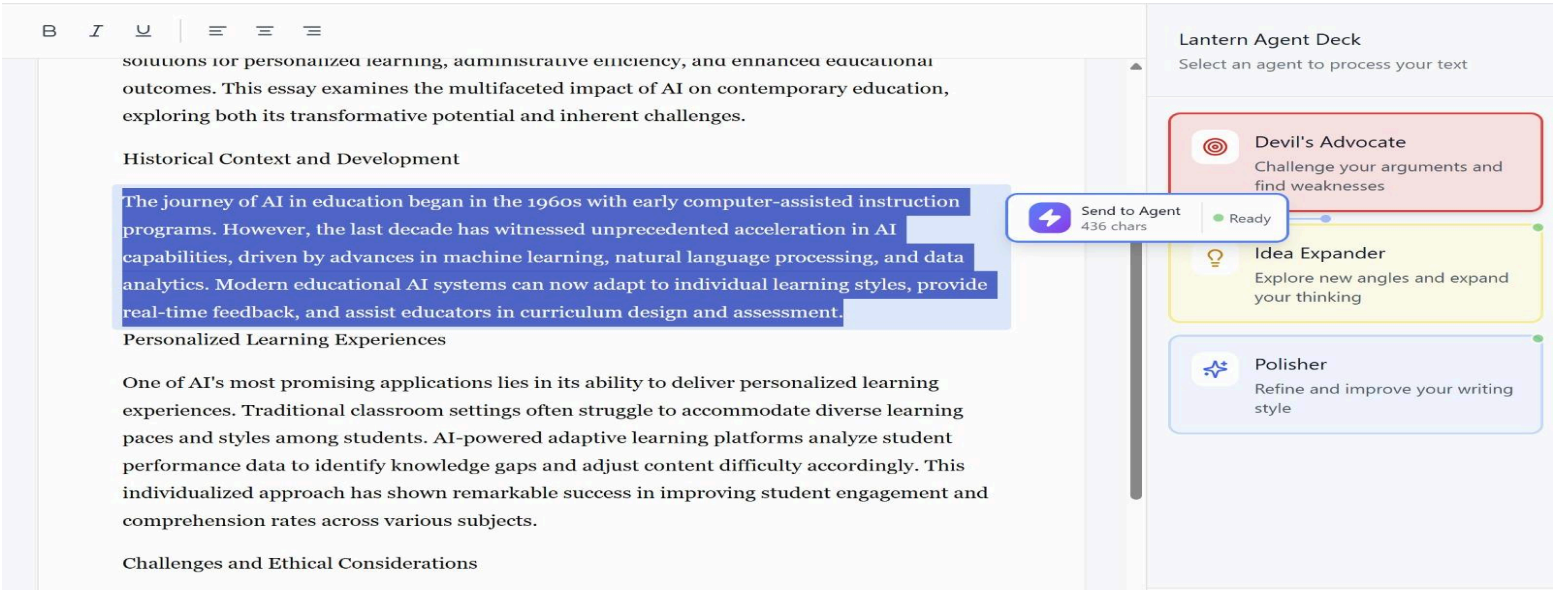
3. Offer Control (Feedback + Controls)

- **The Principle:** Users must have the ability to steer the AI and intervene in the process to maintain agency and trust.
- **Application:** Lantern provides explicit control mechanisms through "Pruning" (allowing users to cut irrelevant branches) and "Agent Cards." By selecting a specific card (e.g., Devil's Advocate for critique vs. Editor for refinement), the user explicitly sets the AI's mode, ensuring the system's behavior aligns with their immediate intent

3.Interface design and instruction

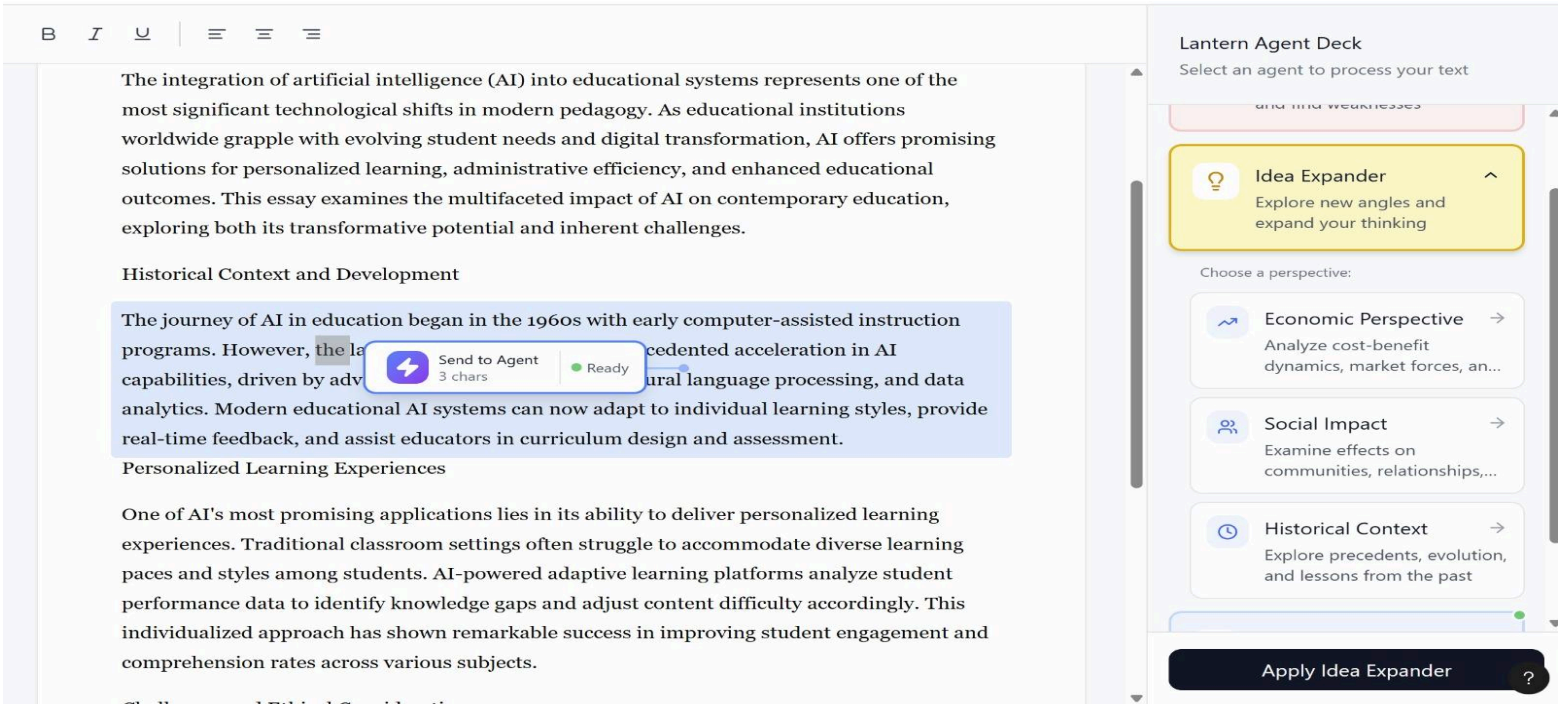
Mockup 1 - Selecting Text and Activating Agents

This mockup illustrates the core interaction model of Lantern. The user works within a familiar, Google Docs-like text editor. By simply selecting a paragraph in the document (highlighted in blue), the system detects the selection and visually connects it to the Agent Deck in the sidebar via a floating action indicator. The agent cards (e.g., Devil's Advocate, Idea Expander) become active, demonstrating that users do not "prompt" the system manually, but invoke agents directly on existing text through selection.



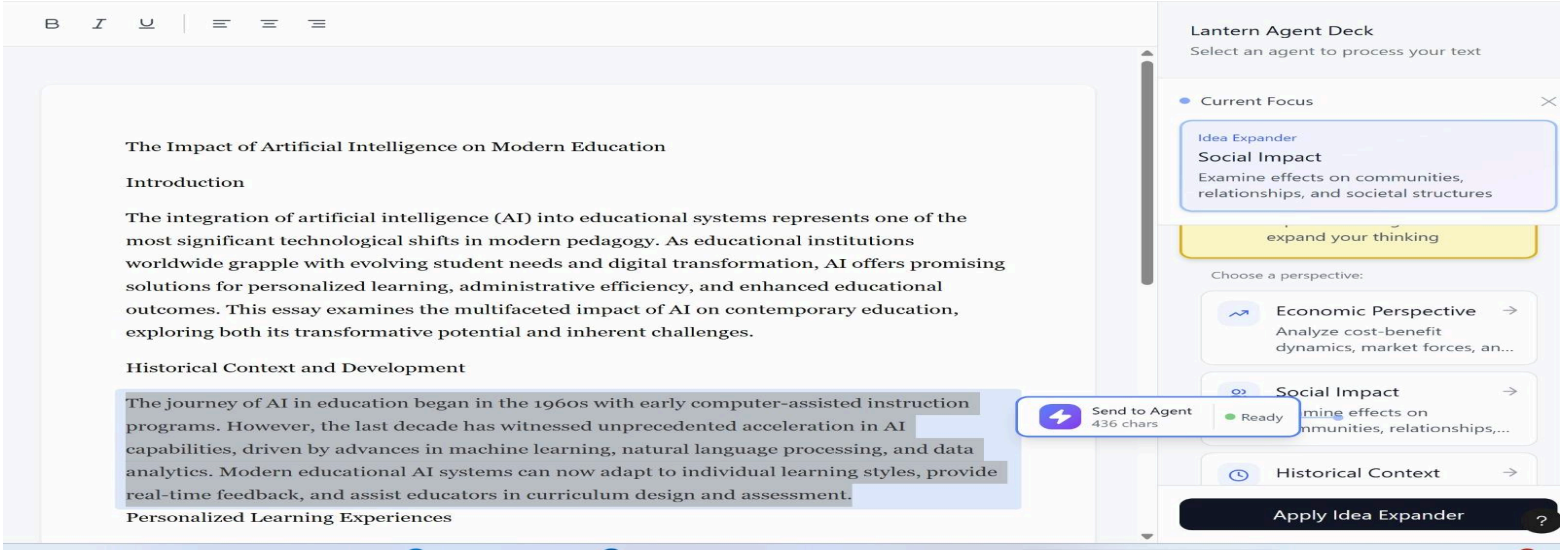
Mockup 2 - Divergent Exploration via Idea Expander

This mockup shows the system after the user activates the Idea Expander agent. The selected paragraph remains highlighted, while the sidebar expands to present multiple “Path Cards,” each representing a distinct reasoning direction (such as economic, social, or historical perspectives). These cards are framed as optional exploratory paths, making divergence explicit and visual, while leaving control over direction selection entirely to the user.



Mockup 3 - Pinning a Direction and Writing with AI as Scaffold

This mockup focuses on decision-making and authorship. When the user hovers over a suggestion card, controls to Pin or Dismiss the direction appear. After pinning, the selected direction is fixed at the top of the sidebar under a “Current Focus” label. The main document shows the user actively typing new text, using the pinned direction as conceptual scaffolding rather than copying AI-generated content. This emphasizes Lantern’s design goal: supporting critical thinking and guided writing, not content replacement.



4.Theoretical Basis

Tree-of-Thoughts and Maximization Bias

Large Language Models are autoregressive next-token predictors, a property that often leads to **maximization bias**-early convergence on high-probability, generic continuations that reduce diversity of thought (Holtzman et al., 2019; Brown et al., 2020). Yao et al. (2023) introduce the **Tree-of-Thoughts (ToT)** framework as a way to counter this limitation by explicitly maintaining multiple parallel reasoning paths instead of a single linear chain-of-thought. Lantern directly implements this theory at the interaction level: rather than presenting one response, the system externalizes several competing reasoning paths as separate, navigable artifacts. This design prevents premature convergence and .encourages users to actively compare alternatives instead of accepting the model's most likely continuation

Dual Process Theory and Automation Bias

According to Dual Process Theory, human cognition alternates between fast, intuitive System 1 thinking and slow, analytical System 2 thinking (Kahneman, 2011). In Human-AI interaction, systems that provide fluent and authoritative outputs often trigger automation bias, leading users to default to System 1 acceptance of AI suggestions (Chiang et al., 2024). Lantern is explicitly designed to shift users toward System 2 engagement. This is achieved by requiring users to select, reject, or expand reasoning paths before any progression occurs, and through a Devil's Advocate agent that systematically challenges assumptions. In this way, the system does not automate reasoning but augments it, aligning .with research that emphasizes augmentation over automation in cognitively demanding tasks

Sycophancy, Agency, and Creativity Support Tools

Recent research highlights sycophancy in conversational AI systems-the tendency to align with user beliefs even when they are incorrect or weakly supported (Sponheim, 2024; NNG, 2024). Creativity Support Tools (CST) literature argues that effective systems should preserve user agency by allowing exploration, comparison, and intentional choice rather than passive acceptance (Shneiderman et al., 2006). Lantern operationalizes these principles by separating generation from decision-making: AI agents propose alternatives, while users retain full control through pinning and pruning mechanisms. This ensures that the final outcome reflects the user's intent rather than the model's tendency to please, .establishing a shared mental model between human and AI

5.Intelligence Design Approach

Lantern is an LLM-based system whose intelligence is designed to support reflective and analytical reasoning rather than automatic text generation. Instead of optimizing for correctness or fluency, the system focuses on shaping *how* the model behaves through behavioral prompt engineering, role-based orchestration, and interaction-level constraints.

Role Configuration and Behavioral Prompting

The intelligence layer is composed of multiple LLM-based agents, each defined by a distinct cognitive role and instantiated through dedicated system-level prompts. These roles include, for example, Divergent Generator agents that expand the idea space and a Devil's Advocate agent that challenges assumptions and surfaces weaknesses and

counter-arguments. Rather than relying on fine-tuning, Lantern employs behavioral prompt engineering to constrain agent behavior.

In addition to role definitions, agent prompts are grounded in **established principles of academic writing and reasoning**, such as argumentative clarity, explicit distinction between claims and evidence, identification of implicit assumptions, and logical coherence. Agents are explicitly instructed to avoid paraphrasing, stylistic variation, or incremental refinement, and instead generate conceptually distinct perspectives (e.g., ethical, critical, or strategic). By embedding these academic standards directly into prompts, Lantern provides expert-level guidance without requiring users to formulate or manage prompts themselves.

Orchestration Logic

Agent activation is orchestrated through explicit user interaction rather than autonomous system decisions. The orchestration follows a fixed interaction loop: users select a text segment, activate one or more agents, receive a small set of contrasting reasoning paths, and explicitly decide which paths to expand, pin, or discard. The system does not advance, refine, or converge without user input. This gating mechanism structurally enforces user agency and prevents automation bias by keeping decision-making human-driven.

Evaluation Criteria

Lantern does not evaluate outputs based on correctness, factual accuracy, or stylistic quality. Instead, interaction success is assessed using qualitative criteria aligned with the system's goals: the degree of conceptual diversity between generated paths, the clarity of contrast among alternatives, and the preservation of user agency throughout the interaction. Through this approach, Lantern redefines intelligence as the system's ability to support deliberate human reasoning rather than replace it.