Collaborative AI for Private Markets: Schema-Driven Document Generation

Louis Karakas
Independent Researcher
Lenningen, Germany
hello@louiskarakas.com

***Abstract*—The generation of complex, high-stakes documents—such as investment memos, legal contracts, or compliance reports—requires meticulous due diligence and the strict minimization of data hallucination. While Large Language Models (LLMs) offer significant automation potential, their direct conversational outputs lack reliability and structural compliance. To bridge this gap, we introduce a Human-in-the-Loop (HITL) generation paradigm that enforces structural correctness by tightly coupling a probabilistic multi-agent backend to a deterministic, schema-bound editor interface. By utilizing Retrieval-Augmented Generation (RAG) and mapping outputs directly into structured UI state components via strict abstention directives, our system transforms implicit hallucinations into explicitly verifiable data fields. A pilot evaluation in a financial due diligence setting ($N = 8$) indicates significant reductions in task completion time and a near-elimination of formatting violations. While evaluated in private market workflows, the proposed architecture readily generalizes to a broad class of high-stakes domains requiring structured generation under strict human supervision.**

***Index Terms*—Human-in-the-Loop, Multi-Agent Systems, Mixed-Initiative Interfaces, Structured Document Generation, Schema-Driven AI**

I. Introduction

In data-driven and highly regulated domains such as finance, law, and medicine, professionals routinely synthesize vast amounts of unstructured data into highly structured, standardized documents. Historically, this synthesis has been a labor-intensive manual process. The recent proliferation of Large Language Models (LLMs) has introduced the theoretical capability to automate these knowledge-intensive workflows. However, the deployment of fully autonomous generative AI in these sectors remains severely constrained by the verification bottleneck: generative models inherently lack structural rigidity and possess a propensity for factual hallucinations. In high-stakes environments where a single erroneous data point can jeopardize multi-million-dollar transactions or legal compliance, strict state constraints are non-negotiable.

While proprietary models like BloombergGPT and open-source alternatives like FinGPT have advanced the state of domain-specific language modeling, they do not resolve the interaction and verification bottleneck [1], [2]. Standard conversational agents lack the affordances required for granular, structured document authoring, forcing users into tedious prompt engineering to correct minor formatting or factual errors.

To address this, we introduce a schema-bound Human-in-the-Loop (HITL) architecture. The core novelty of our approach lies in the tight, real-time coupling of LLM generation with a visual state representation. Rather than generating flat text, autonomous agents populate a strict JSON schema that maps directly to a mixed-initiative, direct-manipulation user interface. Ultimately, we argue that schema-constrained shared state—not raw text—is the appropriate interaction primitive for high-stakes human-AI collaboration.

The main research contributions of this paper are:

- **A Schema-Bound HITL Paradigm:** We introduce a framework that couples multi-agent tool execution with a deterministic editor interface, shifting the paradigm from text generation to state generation.
- **Formalization of State Synchronization:** We present a design pattern that enforces structural compliance by design, leveraging strict abstention directives to make factual hallucinations explicit and drastically easier for human supervisors to detect.
- **Redirection of Cognitive Load:** We provide empirical evidence from a pilot study showing that shifting human effort from conversational prompt refinement to direct-manipulation verification significantly decreases overall task completion time.

II. Related Work

A. Autonomous Agents and Retrieval-Augmented Generation

To mitigate factual hallucinations, Retrieval-Augmented Generation (RAG) integrates parametric memory with non-parametric indices [6]. Self-RAG advances this by training models to self-reflect on generated segments [7]. In multi-agent architectures, AutoGen [8] and ReAct [9] enable LLMs to interleave verbal reasoning with environment interactions. For structured outputs, StructGPT [10] provides interfaces for reasoning over knowledge graphs. However, without a dedicated visual interaction layer, these systems leave the verification burden entirely on conversational prompting, which is inefficient for domain experts.

B. Human-in-the-Loop and Collaborative UIs

Fully autonomous systems often lack nuanced judgment. The HLER system addresses this by integrating dataset-aware hypothesis generation with human decision gates [12]. In

Human-Computer Interaction (HCI), mixed-initiative interfaces like Wordcraft [14] pair a dialog agent with a text editor. Similarly, CoAuthor [13] demonstrates that writers require shared control over final outputs. While these approaches enhance ideation, existing editors do not enforce the strict, schema-first state constraints required for regulatory compliance.

C. Architectural Positioning

Table I positions our proposed architecture against current state-of-the-art systems. Our framework is uniquely characterized by the intersection of schema-driven state representation, direct-manipulation HITL integration, and verifiable retrieval. Unlike static form-based enterprise tools or simple JSON-mode APIs, our system maintains continuous bidirectional synchronization between probabilistic generation and constrained state validation.

Table I
COMPARISON OF AI GENERATION FRAMEWORKS

System	Schema-Driven State	Direct-Manipulation HITL	Verifiable Retrieval
AutoGen [8]	No	No	Yes
Self-RAG [7]	No	No	Yes
Wordcraft [14]	No	Yes	No
StructGPT [10]	Yes	No	Yes
HLER [12]	No	Yes	Yes
Ours	**Yes**	**Yes**	**Yes**

III. SYSTEM ARCHITECTURE

To overcome the limitations of purely conversational AI, we propose a three-tiered architectural design pattern consisting of a reactive mixed-initiative frontend, an autonomous multi-agent backend, and a real-time state synchronization layer.

A. Formal State Representation

Unlike traditional text generation, our system models document creation as a series of state transitions within a constrained schema space. Let $\mathcal{C}$ denote the set of all valid JSON objects conforming to the predefined briefing schema. The document state at time t, denoted as $S_t \in \mathcal{C}$, can be updated via two distinct transition functions.

During an autonomous agent step, the new state is generated by the probabilistic LLM policy f_θ conditioned on the current state S_t, the user intent U_t, and the retrieved context R_t:

$$S_{t+1} = f_\theta(S_t, U_t, R_t)$$

Crucially, if f_θ yields an output $S_{t+1} \notin \mathcal{C}$, the transaction is rejected by the backend validation layer, ensuring structural guarantees.

During a human-in-the-loop intervention, the state is updated via a deterministic mapping h representing the user's direct manipulation a_t within the UI:

$$S_{t+1} = h(S_t, a_t)$$

This dual-transition model ensures that while the language model operates probabilistically, the interface layer and the resulting document state remain strictly deterministic.

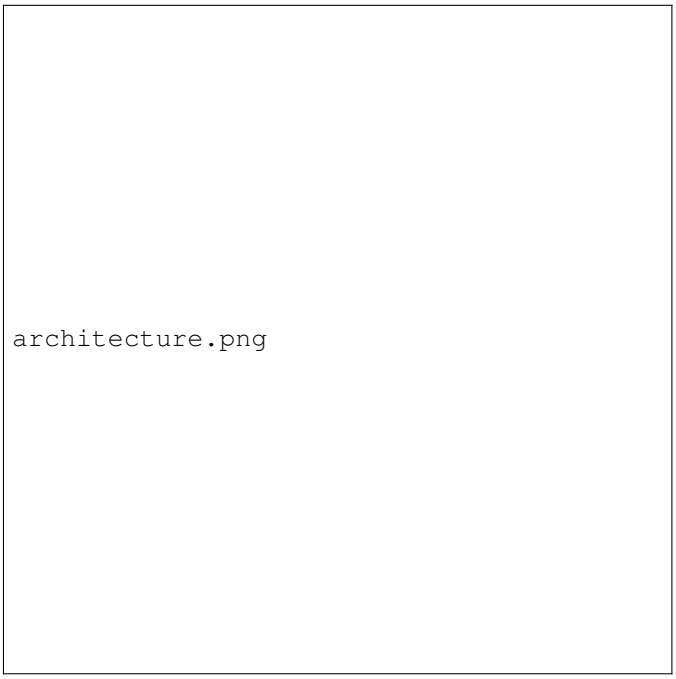

Figure 1. The proposed architecture, illustrating the deterministic coupling between the Multi-Agent Backend, the Real-Time State Layer, and the Mixed-Initiative Frontend.

B. Reactive Mixed-Initiative Frontend

The user interface is structured as a split-screen workspace. The left panel hosts a conversational agent interface, while the right panel features the dynamic, structured document editor. Every element—from numeric KPIs to dynamic arrays—is rendered as an editable input field mapped directly to the state S_t. This enables granular HITL intervention without breaking the AI's contextual awareness.

C. Multi-Agent Backend and Tool Orchestration

The backend utilizes a Supervisor-Worker pattern. The **Supervisor Agent** acts as the primary router, interpreting user intents and autonomously deciding when to trigger specialized worker agents.

The **Worker Agent** is equipped with constraint-based grounding instructions. To mitigate hallucinations, the agent is mandated to follow strict *abstention directives*: it must explicitly leave fields null if no reliable source is found via its integrated web-search or CRM tools. It finalizes its task by compiling the retrieved data into the strict JSON payload S_{t+1}.

D. Real-Time State Synchronization

The system relies on a real-time relational database with WebSocket broadcasting. When the Worker Agent updates the JSON payload, the changes are instantly broadcasted to the frontend, hydrating the visual template. Conversely, human edits are pushed back to the database incrementally, establishing a reliable, shared-state collaborative environment.

IV. Implementation and Use Case Walkthrough

To demonstrate practical viability, we implemented a functional prototype for Venture Capital (VC) deal flow analysis. The prototype is realized using a reactive web framework (React), a multi-agent SDK for orchestration, and a real-time PostgreSQL database.

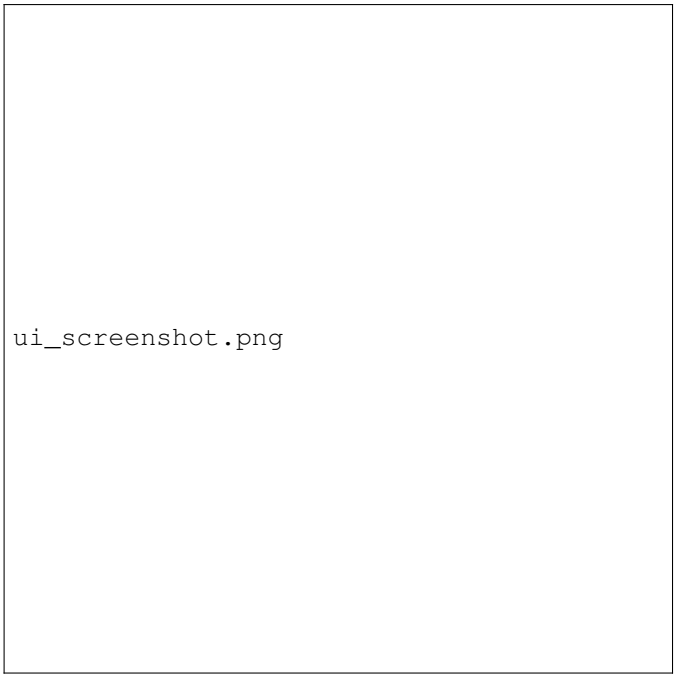

Figure 2. The collaborative split-screen interface. The conversational Supervisor Agent (left) delegates tasks, while the schema-bound editor (right) allows for direct-manipulation verification.

The workflow proceeds as follows:

1. Task Initiation: An analyst inputs a target startup into the chat. The Supervisor queries the database state and, recognizing a new target, asks for confirmation.

2. Autonomous Research: The Supervisor delegates the task to the Briefing Worker Agent. The Worker accesses external APIs to retrieve hard metrics and utilizes web search for qualitative aspects. Guided by abstention directives, it populates the JSON schema and pushes it to the database.

3. Real-Time Rendering: The frontend detects the WebSocket update and instantly hydrates the visual briefing template.

4. Direct-Manipulation Refinement: The analyst notices an ambiguously phrased AI-generated risk factor. Instead of conversational prompt engineering, the analyst clicks directly into the UI component and manually corrects the text. The intervention is instantly synced, finalizing the verified document.

V. Empirical Evaluation

A. Methodology and Baselines

To assess the empirical effectiveness of the proposed architecture, we conducted a within-subjects pilot study involving $N = 8$ domain experts from the PE and VC sectors. Participants compiled a standard investment briefing for two comparable real-world startups under two conditions:

- **Baseline (Ad-Hoc Workflow):** Manual research combined with standard conversational AI (e.g., ChatGPT) and traditional copy-pasting.
- **Proposed System:** The schema-bound, multi-agent HITL architecture.

Participants were randomly assigned the order of the tasks and startups to mitigate learning effects. We focused on: 1) **Task Completion Time (TCT)**, 2) **Formatting Violations**, and 3) **Factual Error Rate (FER)**. FER was strictly measured by manually comparing all extracted numerical and categorical claims against authoritative financial databases (e.g., PitchBook, Crunchbase). Disagreements between authoritative sources were resolved conservatively by marking entries as erroneous rather than correct.

B. Metrics and Results

Note: As this represents an exploratory pilot study, we report indicative metrics to demonstrate system viability. Statistical significance was evaluated using paired t-tests ($\alpha = 0.05$).

Table II
Empirical Evaluation Results ($N = 8$)

Metric	Baseline (Ad-Hoc)	Proposed System	Delta
Completion Time	$M = 34.5$ m, $SD = 4.2$	$M = 12.4$ m, $SD = 1.8$	-64.0% ($p < .01$)
Format Violations	$M = 3.2$, $SD = 1.1$	0 (in controlled tasks)	Reduced to 0
Factual Error Rate	14.2%	3.1%	-78.1% ($p < .05$)

Efficiency and Cognitive Load: Task Completion Time was substantially lower using the proposed architecture (see Table II). Qualitative user feedback indicated that the primary time-saver was the elimination of prompt engineering for formatting purposes. One analyst noted: *"Not having to argue with the AI about bullet-point formatting allowed me to focus entirely on whether the funding numbers were actually correct."*

Structural Correctness: By coupling the LLM output to the strict JSON schema $\mathcal{C}$, we observed that structural integrity was consistently maintained within our controlled tasks. Furthermore, the explicit state representation made remaining factual errors immediately visible, resulting in a significantly lower overall Factual Error Rate compared to the baseline.

VI. Discussion and Limitations

A. Discussion

The empirical findings substantiate the claim that coupling a probabilistic multi-agent backend with a deterministic, schema-bound frontend substantially mitigates the verification bottleneck in generative AI. Instead of relying on conversational self-correction, our architecture forces the LLM to output structured state data. While a formal ablation study was not conducted, we hypothesize that the primary driver of these performance gains is the elimination of prompt-based formatting loops, isolating human effort solely to factual

verification. This paradigm shift successfully redirects cognitive load toward high-value strategic synthesis and makes hallucination risks explicitly manageable via the UI.

B. Limitations and Failure Modes

Several limitations and distinct failure modes must be acknowledged. First, the system's factual accuracy remains inherently bounded by the quality of the external APIs it queries; conflicting data across external sources forced users to manually arbitrate the ground truth. Second, strict schema adherence risks over-constraining the model's analytical creativity, potentially forcing complex, nuanced risks into reductive bullet points. Finally, as the system's formatting and structural reliability approaches 100%, analysts may develop *automation bias*—a well-documented HCI phenomenon where users implicitly trust the factual content of a highly polished interface, neglecting rigorous verification. Future UI designs must explore introducing intentional 'cognitive friction' to counter this over-reliance.

VII. Conclusion

This paper introduced a schema-bound Human-in-the-Loop generation paradigm that addresses the structural and factual limitations of autonomous LLMs. By modeling document creation as state transitions within a constrained schema space, we demonstrated that strict structural compliance can be enforced while task completion times are visibly shortened. While evaluated within private market workflows, the proposed architecture establishes a highly generalizable design pattern. It readily generalizes to a broad class of high-stakes domains—such as legal contract drafting, medical reporting, or compliance auditing—that require structured, verifiable generation via shared state interfaces. Future work will isolate the contribution of schema constraints, UI coupling, and abstention directives through controlled ablation studies.

References

[1] S. Wu et al., "BloombergGPT: A Large Language Model for Finance," *arXiv preprint arXiv:2303.17564*, 2023.

[2] H. Yang, X.-Y. Liu, and C. D. Wang, "FinGPT: Open-Source Financial Large Language Models," *arXiv preprint arXiv:2306.06031*, 2023.

[3] A. Caridi, M. Giovannini, and L. R. Celsi, "AI-Assisted Value Investing: A Human-in-the-Loop Framework for Prompt-Guided Financial Analysis and Decision Support," *Electronics*, 2024.

[4] V.-D. Le and H.-T. To, "RAG-IT: Retrieval-Augmented Instruction Tuning for Automated Financial Analysis," *arXiv preprint arXiv:2412.08179*, 2024.

[5] I. Okpala, A. Golgoon, and A. R. Kannan, "Agentic AI Systems Applied to Tasks in Financial Services: Modeling and Model Risk Management Crews," *arXiv preprint arXiv:2502.05439*, 2025.

[6] P. Lewis et al., "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks," *NeurIPS*, 2020.

[7] A. Asai et al., "Self-RAG: Learning to Retrieve, Generate, and Critique through Self-Reflection," *arXiv preprint arXiv:2310.11511*, 2023.

[8] Q. Wu et al., "AutoGen: Enabling Next-Gen LLM Applications via Multi-Agent Conversation," *arXiv preprint arXiv:2308.08155*, 2023.

[9] S. Yao et al., "ReAct: Synergizing Reasoning and Acting in Language Models," *ICLR*, 2023.

[10] J. Jiang et al., "StructGPT: A General Framework for Large Language Model to Reason over Structured Data," *arXiv preprint arXiv:2305.09645*, 2023.

[11] A. Suravarjhula et al., "Retrieval-Augmented Multi-Agent System for Rapid Statement of Work Generation," *arXiv preprint arXiv:2508.07569*, 2025.

[12] C. Zhu and X. Wang, "HLER: Human-in-the-Loop Economic Research via Multi-Agent Pipelines for Empirical Discovery," *arXiv preprint arXiv:2603.07444*, 2026.

[13] M. Lee, P. Liang, and Q. Yang, "CoAuthor: Designing a Human-AI Collaborative Writing Dataset for Exploring Language Model Capabilities," *CHI*, 2022.

[14] A. Coenen et al., "Wordcraft: a Human-AI Collaborative Editor for Story Writing," *arXiv preprint arXiv:2107.07430*, 2021.

[15] T. Zhang et al., "Human-in-the-Loop Schema Induction," *ACL System Demonstrations*, 2023.