Steven Morrisroe
CS Independent Research
Professor Daniel Mitchell
April 28th, 2025

Abstract Limitations to Grounded Simulation:

Improving LLM-MAS Realism via Data-Driven Personas and Tool Use

Abstract

Multi-Agent Systems (MAS) struggle to simulate complex scenarios requiring human-like reasoning and adaptation, while Large Language Models (LLMs) offer advanced natural language and inference capabilities but often lack grounding in specific environmental contexts. This research investigates the potential and challenges of integrating LLMs as decision-making engines within MAS to enhance simulation realism, focusing on addressing issues of situational awareness and causal consistency. We employ a two-phase experimental methodology. First, we conduct limit testing on an LLM-based multi-agent planning system in an abstract domain, revealing significant difficulties in maintaining causal realism and state tracking without robust grounding mechanisms. Second, leveraging these findings, we develop and evaluate an adapted MAS simulating eBay buyer-seller interactions. This system incorporates grounding through data-driven buyer personas derived from historical purchase data (using TF-IDF and NMF) and seller agents equipped with tools for real-time interaction with the eBay API. Results from the grounded simulation demonstrate substantially improved interaction plausibility and adherence to persona-specific behaviors compared to the abstract system. However, the simulation's success and the reliability of extracted behavioral patterns were critically constrained by the completeness of information available through the external API tools, highlighting a significant "information bottleneck." We conclude that while LLM-driven MAS grounded with data-derived personas and external tools show significant promise for realistic simulation, their effectiveness is fundamentally dependent on the quality and accessibility of grounding information.

1. Introduction

1.1. Background and Motivation

Real-world systems are driven by countless autonomous actors whose interactions give rise to unpredictable, emergent patterns. Traditional Multi-Agent Systems (MAS) excel at simulating rule-based behaviors, but they struggle when agents must interpret subtle cues, adapt to new objectives on the fly, or negotiate in ambiguous environments. As a result, MAS models often lag in domains where human-style reasoning and natural language communication are key.

Large Language Models (LLMs) excel at understanding context, generating coherent dialogue, and performing common sense inference. By embedding LLMs as the decision engines within MAS agents, we can endow each agent with on-the-fly planning, flexible goal revision, and richer interaction protocols. This hybrid approach promises simulations that not only follow predefined rules but also improvise when agents encounter novel scenarios. In this paper, we investigate how LLM-driven agents can enhance both the realism and adaptability of MAS.

1.2. Problem Statement

Despite the promise of LLM-powered MAS, significant challenges hinder their application to complex simulation tasks, particularly those requiring robust situational awareness and causal reasoning beyond constrained conversational contexts. Key problems addressed in this research include:

Situational Awareness and Grounding: LLMs, primarily trained on vast text corpora, often struggle to maintain awareness of dynamic states and adhere to the specific constraints and physics of a simulated environment. Their outputs can become ungrounded, due to a lack of training data within a niche, leading to unrealistic or impossible actions.
Causal Realism: Simulating plausible cause-and-effect relationships is crucial for meaningful simulation. LLMs may generate sequences of events that lack logical consistency or fail to accurately model the consequences of actions within the simulation's rules.
Coordination and State Management: Effectively coordinating multiple LLM agents and maintaining a consistent, shared understanding of the environment's state over time is non-trivial, especially as complexity increases.
Knowledge Reliability: If simulations are intended to inform planning, the results must reliably reflect plausible dynamics, rather than evidence of the simulation's limitations.

2. Core Concepts and Background

This section outlines the fundamental concepts underpinning the research, including Multi-Agent Systems, Large Language Models, grounding techniques, and their application in simulation and planning workflows.

2.1. Multi-Agent Systems (MAS)

A Multi-Agent System (MAS) provides a computational framework for modeling complex environments populated by multiple autonomous agents. Each agent possesses distinct goals, knowledge, and sensing abilities, interacting through communication or shared environmental actions to achieve individual or collective objectives. A key characteristic of MAS is the potential for complex, emergent behaviors to arise from these local interactions, patterns which may not be apparent from analyzing individual agent rules or requiring central control. MAS methodologies are widely applied in fields like swarm robotics, traffic modeling, and market simulation to study decentralized coordination and system dynamics (Wooldridge, 2009).

2.2. Large Language Models (LLMs)

Large Language Models (LLMs) represent a class of deep neural networks, predominantly based on the Transformer architecture introduced by Vaswani et al. (2017). This architecture utilizes self-attention mechanisms, replacing traditional recurrence and convolution to efficiently process sequential data and capture long-range dependencies. Consequently, modern LLMs demonstrate remarkable capabilities in natural language understanding, generation, translation, summarization, and instruction following through in-context learning. However, significant challenges persist; LLMs can generate factually incorrect or nonsensical outputs ("hallucinations"), often lack robust common-sense reasoning, and typically require grounding in external data or environmental context to maintain accuracy and relevance (He et al., 2024; Vaswani et al., 2017).

2.3. Embedding Models, Semantic Search, and RAG

Addressing the grounding challenge often involves techniques built upon semantic understanding. Embedding models transform text or other data into dense vector representations where semantic similarity corresponds to proximity in the vector space (Cer et al., 2018; Mikolov et al., 2013). This enables semantic search, which retrieves information based on conceptual meaning rather than exact keyword matches. Retrieval-Augmented Generation (RAG) leverages this by first performing semantic search to find relevant external information and then providing this context to an LLM during generation. This process aims to reduce hallucinations and improve the factual accuracy and contextual relevance of the LLM's output

(Lewis et al., 2020).

2.4. Multi-Agent LLM Workflows

Integrating LLMs as the core reasoning engine ("brain") for agents within a MAS facilitates the creation of more dynamic and sophisticated multi-agent workflows. This integration enhances agent capabilities in several key areas:

- **Coordination and Communication:** Agents can leverage LLM natural language capabilities to exchange more nuanced information, negotiate complex states, and coordinate actions based on shared understanding.
- **State Management:** The system must track both the global environment state and each agent's internal state (beliefs, goals), allowing agents to react coherently to evolving conditions over time.
- **Tool Invocation:** LLM agents can be equipped with tools, enabling them to call external APIs, databases, or other functions. This grounds their reasoning in real-time data and extends their abilities beyond text generation.
- **Framework Orchestration:** Libraries such as LangGraph provide abstractions (e.g., nodes for agents, edges for control flow, persistent state objects) that simplify the development and management of these complex, potentially cyclic, multi-agent interactions.

2.5. Simulation and Planning in LLM-MAS

Within the context of this research, *simulation* refers to executing a system of LLM-driven agents within a controlled environment over time to observe emergent dynamics under specified constraints and interaction rules. *Planning*, conversely, involves the generation and execution of action sequences by these agents to achieve predefined objectives. Our approach utilizes LLMs in a dual capacity: as actors within the simulation, aiming to model realistic, adaptive behaviors, and potentially as planners that can leverage simulation feedback to refine strategies. This coupling seeks to produce richer simulation insights and explore how these insights can inform more adaptive, context-aware planning

3. Literature Review

This section reviews existing research relevant to MAS, LLMs in simulation and planning, and grounding techniques, setting the stage for the current work.

3.1. Traditional MAS for Simulation and Planning

Traditional Multi-Agent Systems (MAS) utilize frameworks like NetLogo or Repast for agent-based modeling (ABM), where autonomous agents follow predefined rules, leading to emergent system-level behaviors (He et al., 2024). Architectures such as the Belief-Desire-Intention (BDI) model endow agents with rudimentary mental states to guide planning and action. These classical approaches excel in well-defined environments, offering transparency and control over agent logic. However, they typically rely on simplified cognitive models or reactive behaviors to remain computationally tractable, limiting their ability to simulate complex human-like reasoning, adapt to truly novel situations outside their programming, or generalize across diverse domains without significant re-engineering (He et al., 2024). Consequently, achieving simultaneous accuracy, adaptability, and reliability under changing conditions remains a challenge for rule-based MAS.

3.2. LLMs in Simulation

To overcome the limitations of traditional MAS in modeling nuanced behavior, researchers are increasingly incorporating Large Language Models (LLMs) as the decision-making core of agents (He et al., 2024). A landmark example is the "Generative Agents" work by Park et al. (2023), where 25 LLM-powered agents in a sandbox environment exhibited believable, emergent social behaviors (e.g., autonomously organizing a party) by integrating LLMs with memory, reflection, and planning modules. Similar studies have demonstrated LLM agents showing sophisticated social dynamics, simulating disease spread based on social ties, reenacting historical conflicts, or generating credible macroeconomic patterns in market simulations where traditional models failed (He et al., 2024). These results suggest LLMs enable a higher fidelity of simulation, particularly for open-ended social interactions. Despite this promise, challenges remain: LLM simulations face high computational costs, inherent LLM issues like factual errors or bias, difficulties maintaining long-term consistency due to context window limits, and a lack of guaranteed causal realism (He et al., 2024; Park et al., 2023). Ensuring robust, scalable, and controllable LLM agent behavior is an ongoing research focus.

3.3. LLMs for Planning and Task Execution

LLMs' capacity for multi-step reasoning has spurred interest in their use for planning and task execution. Techniques like Chain-of-Thought prompting improve complex problem-solving by eliciting intermediate reasoning steps (Wei et al., 2022, as cited in He et al., 2024). Frameworks such as ReAct (Yao et al., 2022) build on this by interleaving reasoning ("thought") with actions ("act"), enabling LLMs to use tools (e.g., APIs, search engines), observe outcomes, and refine plans dynamically. This feedback loop enhances performance on interactive tasks. Conceptual frameworks like AutoGPT further explore LLM autonomy, creating loops where the LLM generates sub-goals and actions towards a high-level objective, demonstrating potential for open-ended task completion (He et al., 2024). However, these autonomous systems often suffer from reliability issues, including getting stuck in loops, generating inefficient plans, or failing without oversight (He et al., 2024). Research is exploring mitigation strategies like reflection, where agents analyze failures to improve subsequent attempts (e.g., Reflexion concept, cited in He et al., 2024). While LLMs offer new capabilities for decomposing and executing complex tasks, ensuring the robustness, efficiency, and validity of their plans remains critical.

3.4. Grounding LLMs in External Context

A key factor for reliable LLM agent performance is *grounding*—connecting the LLM's outputs to external facts or the current state of the simulated environment. Retrieval-Augmented Generation (RAG) (Lewis et al., 2020) is a prominent technique, allowing LLMs to query external knowledge sources (documents, databases) and incorporate retrieved information into their responses, reducing hallucinations and improving factual accuracy. This is directly applicable in MAS, where agents can retrieve relevant context about the world state or past events to inform decisions, as demonstrated by the memory retrieval mechanism in Generative Agents (Park et al., 2023). Beyond RAG, other grounding methods include using structured knowledge (e.g., knowledge graphs) to enforce consistency, employing structured prompts to guide LLM output formats, implementing feedback loops where actions are validated against environment rules, and fine-tuning models on domain-specific data (He et al., 2024). These techniques aim to enhance situational awareness and ensure that agent actions are causally plausible and factually consistent with the simulation's reality.

3.5. Synthesis and Research Gap

Existing literature reveals a spectrum: traditional MAS offer reliability within defined rules but lack cognitive depth and adaptability, while LLM-based agents provide human-like flexibility but face challenges with reliability and grounding (He et al., 2024). The convergence of these approaches is promising, yet current LLM-based multi-agent simulations often struggle to maintain robust *causal realism* and *situational awareness*. Agents may generate plausible-sounding actions based on training data patterns that are inconsistent with the actual simulated environment state or underlying rules (He et al., 2024). This research gap – the need for LLM-driven MAS that combine adaptability with rigorous grounding to ensure true understanding of cause-and-effect within the simulation – is a primary focus of current work. Developing agents that are both creative and consistently tethered to the simulation's reality is crucial for building trustworthy simulations capable of yielding reliable insights, which motivates the experimental investigations in this paper.

4. Methodology: Overall Experimental Approach

This research employs a two-phase experimental methodology to investigate the efficacy of

LLM-powered MAS for simulation and planning, focusing on addressing challenges related to causal realism and situational awareness.

GitHub repo: https://github.com/stevenmorrisroe/UT_CS_Research

Phase 1: Exploratory Limit Testing: An initial experiment was designed to assess the capabilities and inherent limitations of an LLM-based multi-agent planning system operating on abstract tasks. This involved defining a structured planning schema and using the LangGraph framework, powered by gpt-4o-mini, to orchestrate LLM agents attempting to achieve goals based on evolving and flexible state. The primary goal was to identify fundamental failure modes related to causal reasoning, state tracking, and sensitivity to initial conditions in a less constrained environment. Evaluation in this phase was primarily qualitative, observing system behavior and identifying points where simulations diverged from plausible reality, such as "causal spiraling" where initial flawed assumptions led to unrealistic tangents, diluting the prompt context and causing loops, or demonstrating extreme sensitivity to minor changes in assumptions or prompts.

Phase 2: Adapted Domain-Specific Simulation: Leveraging the insights gained from Phase 1 – specifically the difficulties LLMs faced in maintaining causal realism and state consistency without a robust world model or strong grounding – a second experiment focused on developing and evaluating a MAS simulation within a specific, more grounded domain: eBay sales interactions. This domain was chosen for its relatively structured conversational nature, the potential for clear agent roles, defined state attributes, and the availability of external grounding mechanisms like the eBay API & Open E-commerce 1.0 dataset. This phase emphasized incorporating techniques aimed at mitigating the limitations identified earlier. Grounding was pursued through two primary mechanisms: 1) Data-driven buyer persona generation using TF-IDF and NMF on historical purchase data (via Scikit-learn) to create specific, behaviorally-informed system prompts for the buyer agent, and 2) Equipping the seller agent with tools to interact with external information sources. The objective was to assess whether these adaptations, orchestrated again using LangGraph but with Google's Gemini models, could improve simulation realism and enable the extraction of meaningful behavioral patterns. Evaluation in this phase involved a mix of quantitative metrics (e.g., transaction success rates, persona relevance scores based on simulated sales) and qualitative analysis (e.g., assessing conversation plausibility and persona adherence).

Across both phases, Python was the primary implementation language, utilizing libraries such as LangChain and LangGraph for agent and workflow orchestration, Pydantic for data modeling, OpenAI and Google APIs for LLM access, Gemini embedding models, vector databases (Qdrant) for similarity checks (Phase 1), standard data science libraries (Pandas, Scikit-learn) for data processing and analysis (Phase 2) and PostgreSQL for logging.

5. Experiment 1: Multi-Agent System Limit Testing and Learning

5.1. Objective

The primary objective of this experiment was to evaluate the capability of an LLM-based multi-agent system, orchestrated via LangGraph, to generate and execute complex, multi-step plans based on a defined schema and input state. A key goal was to identify fundamental failure modes and limitations concerning causal reasoning, situational awareness, and robustness when operating in abstract, less grounded environments.

5.2. System Design

The system was designed as a stateful graph using LangGraph, centered around a Plan state object. (See Appendix Exhibit A for the full Pydantic model and graph diagram).

State Representation: Pydantic models defined the structure of the simulation state. The core Plan model tracked the overall goal (goal_state), the sequence of inputs (input_state), generated steps (steps), and resulting outcomes (outcomes). Crucially, it also maintained cumulative state throughout the plan's execution, including cumulative_assumptions (aggregating ground_truth and vulnerabilities from all steps) and two generic metrics (metric_count_1, metric_count_2) often representing resources like budget and time. (See Appendix Exhibit B for the full Plan model definition).

- **LangGraph Workflow:** A StatefulGraph (visualized in Appendix Exhibit A) managed the Plan state. The execution flow involved several key nodes orchestrated by conditional logic:
    - ○ generate_next_idea: An LLM agent prompted to propose the next step (NextStep) towards the goal, given the current InputState and potentially the cumulative_assumptions. To encourage novelty and prevent simple loops, proposed ideas were checked for semantic similarity against previously generated ideas stored in a Qdrant vector database.
    - ○ decider: Another LLM agent evaluated the proposed NextStep. It was prompted to act as a "world-class expert" in the plan's topic, carefully evaluating the proposed idea and its assumptions against the current cumulative_assumptions (specifically ground_truth) and resource status (metric_count_1, metric_count_2). It was instructed to weigh "tangible evidence and unpredictable elements" to determine if the step would "realistically succeed or fail," outputting only 'success' or 'failure'.
    - ○ generate_good_outcome / generate_bad_outcome: LLM agents generated

structured Outcome objects reflecting the consequences (new truths, vulnerabilities, resource increments) based on the decider's verdict.
    - summarize_assumptions: Consolidated the ground_truth and vulnerabilities in cumulative_assumptions based on the latest Outcome.
    - goal_check: Determined if the current state (potentially evaluating cumulative_assumptions or metrics against the goal_state) satisfied the overall goal.
    - abandon_check: Assessed whether the plan should be terminated, likely by comparing cumulative resource usage (metric_count_1, metric_count_2) against predefined limits or potentially detecting prolonged lack of progress.
    - Conditional Edges: Routing logic directed the flow based on the decider's output and the checks for goal achievement or abandonment.

5.3. Implementation Details

The system was implemented in Python using LangGraph for workflow orchestration. The OpenAI gpt-4o-mini model was primarily used for the agent nodes (generate_next_idea, decider, generate_outcome). Specific prompts guided each agent, enforcing output formatting according to the Pydantic models. Resource tracking (cost, time) was implemented via the metric_count_1 and metric_count_2 fields, updated by the Outcome objects and checked by the abandon_check node against implicit or explicit thresholds.

5.4. Evaluation and Results

Evaluation was primarily qualitative, observing the system's behavior across different types of goals, supplemented by analysis of execution traces like the comparison report provided earlier.

- **Simple, Constrained Tasks:** For goals with limited variables and well-defined steps (e.g., "fix a leaking sink"), the system often performed adequately and consistently across runs, identifying logical steps and reaching the goal efficiently.
- **Complex, Open-Ended Tasks:** When presented with more complex goals (e.g., "build an A-frame house in the woods"), the system exhibited significant limitations, as illustrated by the Frame House test runs:
    - **Causal Spiraling/Meandering:** Plans often failed to converge efficiently. Even when individual steps were deemed "successful" by the decider, the overall plan could meander without achieving the final goal, eventually hitting recursion limits or resource exhaustion. For instance, in one Frame House run, numerous steps like implementing QC, training, and weather plans were marked 'success', but the goal_check node consistently failed, indicating the plan was executing tasks but not effectively progressing towards completion. This highlights how the plan can

spiral based on the sequence of generated ideas and decisions.

  - **Lack of Situational Grounding:** Without external knowledge or a robust world model, the agents often proposed ideas or predicted outcomes inconsistent with real-world physics or constraints (e.g., assuming unrealistic resource availability, ignoring environmental factors not explicitly listed in vulnerabilities). The generated truths and vulnerabilities were purely text-based constructs internal to the LLM's reasoning.
  - **Prompt Sensitivity and Path Dependence:** The system showed high sensitivity to initial conditions and the decider's judgments. As seen in the Frame House comparisons, different runs could take entirely different paths based on whether an early step (like implementing a specific tool or training) was deemed a 'success' or 'failure'. A single different judgment by the decider early on could cascade, leading to a completely different sequence of steps and vulnerabilities, dramatically altering the plan's trajectory and outcome.
  - **Evaluation Difficulty (Decider Reliability):** Using the decider LLM to evaluate step success proved unreliable for judging overall progress on complex tasks. While it might reasonably assess the isolated feasibility of a single step (e.g., "implement weather contingency plan" is plausibly a 'success' action), the analysis showed it struggled to weigh the *impact* of that step towards the *final goal*. The Frame House run hitting the recursion limit despite many 'successful' steps suggests the decider might have been overly optimistic or failed to account for the cumulative negative effects of introduced vulnerabilities or resource drain, highlighting the challenge of using an LLM for robust, goal-oriented evaluation within the simulation loop itself.

5.5. Discussion (Takeaways from Experiment 1)

This experiment highlighted critical limitations of relying solely on LLMs within a MAS planning/simulation loop without strong grounding mechanisms:

1. **Causal Realism:** While capable of simple sequential logic, the LLM agents struggled to maintain consistent and realistic cause-and-effect chains in tasks with high uncertainty or many variables. The lack of grounding led to unfeasible plans and unrealistic dynamics.
2. **State Grounding:** Assumptions and state updates (truths, vulnerabilities) generated by LLMs were often untethered from reality. Effective state management requires not just tracking variables but ensuring the *quality* and *groundedness* of those updates.
3. **Internal Evaluation Limits:** Using LLMs to judge step outcomes internally is challenging. The "judge" LLM often lacks the specific domain knowledge or goal-oriented perspective to make consistently accurate assessments of progress, especially for complex goals. More objective metrics or externally grounded evaluation

functions are likely needed.

4. **Sensitivity and Control:** The architecture is highly sensitive to initial assumptions and prompt design, with errors potentially amplifying through the recursive process, making fine-tuning and reliable control difficult.

These findings directly motivated the design of Experiment 2, emphasizing the need for data-driven agent initialization (personas), explicit grounding mechanisms (tool use), a more constrained interaction model (conversation), and a focus on a specific domain where environmental factors could be more concretely represented or accessed.

6. Experiment 2: Adapted Multi-Agent eBay Sale Simulation

6.1. Objective

Building upon the insights from the initial limit testing, this second experiment aimed to construct and evaluate a more realistic Multi-Agent System (MAS) specifically designed for simulating eBay buyer-seller interactions. The primary goals were twofold: first, to implement and assess specific grounding mechanisms—data-driven buyer personas derived from historical purchase data and live external tool use by the seller agent—to improve simulation realism compared to the ungrounded abstract planner. Second, the experiment sought to analyze the resulting simulation outputs, including conversation logs and sale outcomes, to identify potentially actionable patterns in negotiation tactics or persona-specific behaviors relevant to informing real-world sales strategies.

6.2. System Design Enhancements

To address the shortcomings identified in Experiment 1, particularly the lack of grounding and causal realism, this simulation focused on the eBay domain, leveraging its structured interactions and potential for external grounding. Key adaptations centered on enhancing agent grounding and state representation through two primary mechanisms.

First, distinct and behaviorally grounded buyer agents were created using a data-driven persona generation pipeline based on TF-IDF and Non-negative Matrix Factorization (NMF). The process utilized a large-scale, validated dataset detailing the Amazon purchase histories (including product details, dates, and linked demographic/lifestyle survey data) of over 5000 US consumers (Berke et al., 2024). This dataset, while the most robust available in an adjacent domain, was recognized as not being ideal for the specific eBay context, underscoring the importance of procuring domain-specific data early in system design. The raw data underwent extensive preprocessing involving cleaning, handling missing values, standardizing product categories, and concatenating relevant text fields (cleaned product titles) per customer into a single document (Purchase_Doc). This corpus was then transformed into a Document-Term

Matrix (DTM) using Scikit-learn's TfidfVectorizer, employing specific parameters (e.g., max_df=0.90, min_df=5, max_features=5000, custom stop words, bigram preservation) to capture salient terms while controlling dimensionality. NMF (with N_TOPICS=20, chosen a priori for manageability, and specific solver/loss parameters) was applied to the DTM to identify 20 underlying purchasing themes or topics. For each topic, a persona profile was constructed by assigning customers, extracting top keywords, analyzing associated purchase behaviors (frequency, seasonality) and available demographic signals, and calculating value metrics. The final output of this pipeline was a set of 20 structured text prompts, each synthesizing a persona's key characteristics, which served as the direct system prompts to initialize the Buyer agent's behavior in the simulation.

Second, the environment simulation and grounding mechanisms were adapted for the eBay context. The simulation modeled a turn-based conversation between a single buyer and seller concerning actual eBay listings, accessed via the official eBay Browse API. State management was handled by a central SimulationState object (a Python TypedDict) orchestrated by LangGraph, tracking essential variables like the conversation history (messages), sale completion status (sale_completed), sold item details, persona identifiers, and post-analysis metrics (product_avg_rank). This provided more concrete state tracking compared to the abstract, LLM-generated state updates in Experiment 1. Crucially, grounding was primarily achieved through live tool usage by the Seller agent. The Seller was equipped with tools (ebay_search_tool, answer_item_question_tool) that directly interacted with specific eBay API endpoints (/item_summary/search, /item/{item_id}). These tools enabled the Seller to retrieve real-time listing information (e.g., price, description) based on buyer queries, injecting external context and constraints into the conversation, a distinct approach from Experiment 1's reliance on internal LLM reasoning and avoiding static RAG during interactions.

The simulation workflow itself was managed by a LangGraph stateful graph, orchestrating the interaction between the two agents. The Buyer agent (using Google's gemini-2.0-flash-001) was configured with one of the NMF-derived persona prompts, aiming to inquire about and potentially purchase items aligned with its profile. The Seller agent (also gemini-2.0-flash-001), acting as a helpful eBay seller, utilized the aforementioned API tools to answer questions factually and facilitate a sale. Agents exchanged messages turn-by-turn, with the history maintained in SimulationState, until either a sale was detected or a maximum conversation length (approx. 21 messages total) was reached. Sale detection was triggered by specific keywords in recent messages or reaching the message limit, invoking an LLM call (analyze_conversation_for_sale function with gemini-2.0-flash-001 and a specific prompt) to analyze the conversation and return a structured SaleAnalysisOutput. A sale was recorded if the LLM indicated sale_detected as true, with lower confidence scores flagging the transaction for potential review in further research. Separate from the main loop, a post-simulation analysis calculated persona relevance if a sale occurred. This involved embedding the sold item's description (models/text-embedding-004) and comparing its similarity (cosine) to the pre-embedded descriptions of the top 100 historically purchased items for that buyer's persona (stored in a dedicated index file). The average rank (product_avg_rank) of the top K=3 most similar historical items provided a quantitative measure of the transaction's relevance to the persona. While crude, it was ultimately chosen due to differences between eBay and Amazon sales data as well as chat modality analysis difficulty.

6.3. Implementation Details

The simulation was implemented in Python. Google's gemini-2.0-flash-001 model (via LangChain's ChatGoogleGenerativeAI, with temperature=0.7) served as the LLM for both Buyer and Seller agents, as well as the sale detection analysis. Google's models/text-embedding-004 (via GoogleGenerativeAIEmbeddings) handled product embedding for the relevance analysis. Buyer agent behavior was guided by the NMF-generated persona prompts, while Seller prompts likely focused on helpfulness and tool usage instructions. A specific prompt (SALE_ANALYSIS_PROMPT) directed the sale detection LLM to return structured JSON output.

LangGraph orchestrated the stateful graph, managing the SimulationState, agent turns, conditional logic, and tool calls. LangChain provided abstractions for LLMs, embeddings, message history, prompts, and tools. Scikit-learn (TfidfVectorizer, NMF) was used for persona generation, Pydantic defined data models (SimulationState, SaleAnalysisOutput), and Pandas facilitated data manipulation. Simulation outputs, including full conversation history, final state, and exceptions, were logged to a PostgreSQL database (via psycopg2), with LangSmith used for detailed execution tracing. Each run involved one buyer (with one of the 20 personas) and one seller, running for a maximum of ~21 messages or until sale detection, using NMF parameters as described and K=3 for relevance analysis.

6.4. Simulation Evaluation

Evaluating the eBay simulation involved analyzing quantitative metrics from the logged state and qualitative assessment of interaction dynamics, informed by per-run "Sales Wisdom" summaries. Quantitatively, across 100 runs (5 per persona), an overall transaction success rate of 18% was observed, though rates varied dramatically by persona (e.g., 100% for Topic 15 - bulk household goods, 0% for Topic 0 - books/organization), often correlating with the availability of specific product details via the API. For the 18 successful sales, the average post-hoc persona relevance score (product_avg_rank) was 31.74. This score, representing the average rank (1-100, lower is better) of the top 3 most similar items in the buyer's historical purchase index, suggests moderate alignment between simulated sales and persona profiles, although individual persona averages also varied widely. It was noted that further metrics like sale detection confidence scores or turn counts were not analyzed due to time constraints.

Qualitatively, analysis of transcripts indicated generally plausible conversational flows, with agents engaging in typical sales interactions like clarifying needs, presenting options, and handling objections. Persona adherence was evident, as buyers asked relevant questions and exhibited priorities consistent with their NMF-derived prompts (e.g., focusing on tech specs, ingredients, or budget). The effectiveness of tool use was clear, grounding seller responses in real-time API data, a significant improvement over Experiment 1. However, this also highlighted the limitations of API summaries, as missing crucial details (ingredients, compatibility, dimensions) frequently led to buyer frustration and abandoned sales, demonstrating a key bottleneck. While the LLM-based sale detection identified clear agreements, its accuracy in ambiguous cases remains an area requiring further scrutiny, as validation via manual review was not performed. Overall, compared to Experiment 1, the grounded nature of Experiment 2 produced substantially more realistic and contextually relevant simulation dynamics, with failures often stemming from plausible information gaps rather than ungrounded reasoning.

6.5. Knowledge Extraction

The primary goal of knowledge extraction was to identify actionable patterns in buyer behavior and potential sales strategies from the 100 simulation runs. This involved analyzing logged conversation transcripts, structured simulation outcomes (sale status, item details, persona ID, relevance score), detailed LangSmith execution traces, and qualitative per-run "Sales Wisdom" summaries. Qualitative theme analysis of these sources revealed several recurring patterns. A dominant theme was the criticality of specific information; interactions frequently failed because the Seller agent, limited by API data, could not provide details crucial to the buyer persona (e.g., ingredients, material composition), leading directly to abandoned purchases. The analysis also confirmed the value of persona-driven interaction, as buyers acted according to their profiles, and seller success often depended on recognizing and adapting to these specific needs. Effective seller tactics observed included confirming understanding, managing expectations about information limits, and offering alternatives. Common failure points, beyond information gaps, included misinterpreting buyer intent and limitations of API keyword search. While detailed statistical correlation analysis was not performed, per-persona results suggested potential links between persona type and success rate, likely tied to information availability (e.g., high success for bulk goods vs. low success for niche food items). Tool usage analysis primarily highlighted the data limitations of the API tools rather than issues with the invocation logic itself. The extracted knowledge, mainly qualitative insights and observed behavioral patterns, provides a richer understanding of the simulated dynamics than quantitative metrics alone.

6.6. Results

The eBay sales simulation successfully executed 100 runs, pairing 20 unique NMF-derived buyer personas with a seller agent equipped with live eBay API tools. The LangGraph framework effectively orchestrated the turn-based dialogues, state management, and tool integration, successfully generating multi-turn conversations grounded in external data. Key quantitative outcomes included an overall transaction success rate of 18%, with significant variation across buyer personas linked to the availability of relevant product information. The average post-hoc relevance score for sold items (product_avg_rank) was 31.74, indicating moderate alignment with buyer persona purchase histories, again with considerable per-persona variance.

Qualitative findings highlighted the effectiveness of the grounding mechanisms. Live API tool use significantly improved conversational groundedness compared to Experiment 1, with seller responses incorporating real-time data. Buyer agents generally adhered to their assigned personas, demonstrating the utility of the NMF-derived prompts. However, the most significant finding was the critical impact of information gaps; reliance on incomplete API summaries frequently prevented sales by failing to provide crucial details, representing a key barrier. Despite this limitation, the simulation produced plausible interaction dynamics mirroring real-world e-commerce challenges. In summary, Experiment 2 demonstrated a more grounded and realistic MAS simulation than Experiment 1 but underscored the critical dependence on the quality and completeness of external tool data for successful outcomes.

6.7. Discussion (Takeaways from Experiment 2)

Experiment 2 yielded significant improvements over the abstract planner, offering valuable insights into using grounded LLM-based MAS for simulating real-world interactions. The grounding effectiveness was evident; combining data-driven buyer personas (from NMF analysis) with seller agents using live eBay API tools produced substantially more plausible dynamics than the ungrounded approach. Personas provided consistent motivation, while tools injected real-world constraints, preventing the unrealistic spiraling seen previously. Tool use for grounding factual claims proved clearly advantageous over relying on the LLM's internal knowledge.

The persona impact and data-driven initialization were also clear. Using NMF-derived profiles for system prompts effectively initialized buyers with distinct, data-informed behaviors, leading to richer simulation outcomes compared to generic prompts, as evidenced by the varying success rates across personas. However, the experiment starkly highlighted persistent challenges, primarily the information bottleneck. The simulation's success was fundamentally constrained by the information available via the eBay API summaries. When critical details were missing, interactions often failed regardless of agent capabilities, illustrating that sophisticated reasoning cannot compensate for data limitations in grounding tools.

Regarding the implicit research questions, this experiment provides strong evidence (RQ2: Grounding) that combining data-derived persona prompts and live external tool access significantly enhances situational awareness and realism in domain-specific simulations. For RQ3 (Knowledge Value), the simulation generated plausible data and highlighted realistic failure points (information gaps), offering qualitative insights ("Sales Wisdom") into persona challenges and interaction strategies. However, the direct applicability of this knowledge depends heavily on simulation fidelity, which hinges on grounding data quality (API completeness) and component reliability (e.g., sale detection). The patterns are suggestive but require validation.

Overall, Experiment 2 demonstrates that carefully designed LLM-based MAS, incorporating data-driven initialization and robust grounding via external tools, can simulate complex interactions like e-commerce sales with improved realism. Yet, their effectiveness and the reliability of extracted knowledge remain critically dependent on the quality and completeness of information accessible through agent tools. Addressing this information bottleneck is a key challenge for future work.

7. Limitations

While this research demonstrates the potential of grounded LLM-based Multi-Agent Systems, several limitations should be acknowledged, which also highlight avenues for future investigation.

1. **Information Bottleneck via External Tools:** A primary limitation, starkly revealed in Experiment 2, is the simulation's critical dependence on the completeness and granularity of information provided by external tools. The seller agent's ability to satisfy buyer inquiries and facilitate sales was fundamentally constrained by the details available through the eBay Browse API summaries. Crucial information (e.g., specific product ingredients, detailed dimensions, compatibility nuances, expiration dates) was often missing, leading to interaction failures that stemmed from data unavailability rather than agent reasoning flaws. This underscores that the realism and success of tool-grounded MAS are heavily reliant on the quality and scope of the underlying APIs or knowledge sources.
2. **Grounding Data Source Mismatch:** Experiment 2 utilized a rich dataset of Amazon purchase histories to generate buyer personas for an eBay simulation context. While this dataset provided detailed consumer behavior insights, the transferability of these patterns from the Amazon platform to the eBay marketplace is an assumption. Differences in user demographics, typical product categories, purchasing mechanisms (e.g., auction vs. fixed

price), and platform interfaces may limit the fidelity of the generated personas within the specific eBay simulation environment. The lack of readily available, comparable eBay-specific datasets necessitated this approach, but it remains a potential source of discrepancy.

3. **Evaluation Reliability and Metrics:** The evaluation methodologies employed have inherent limitations.
    - **Experiment 1 Internal Evaluator:** The use of an LLM (decider node) to evaluate step success in the abstract planning task of Experiment 1 proved unreliable for complex goals, struggling to assess true progress towards the final objective. This limits the definitive interpretation of plan trajectories and "success" rates in that phase, though it served its purpose for limit testing.
    - **Experiment 2 Sale Detection:** Relying on an LLM to detect sales from conversation logs introduces uncertainty. While keyword triggers and structured outputs were used, the accuracy of this detection, particularly in ambiguous cases or those flagged with low confidence, was not rigorously validated through manual review in this study. This adds a potential error margin to the reported conversion rates.
    - **Relevance Score Interpretation:** The post-hoc product_avg_rank provides a quantitative measure of sale relevance to persona, but interpreting its absolute value requires further context or baseline comparisons to fully gauge the strength of the alignment.
4. **Persona Generation Methodology:** The TF-IDF and NMF pipeline successfully generated distinct buyer personas, but the methodology has constraints. The number of topics (N_TOPICS=20) was chosen a priori for manageability rather than through empirical optimization (e.g., maximizing coherence scores), potentially affecting the granularity or distinctiveness of the resulting personas. Furthermore, TF-IDF/NMF primarily captures thematic patterns based on product titles and may not fully represent more nuanced behavioral aspects.
5. **Simulation Scope and Interaction Model:** The simulations were constrained in scope.
    - **Scale:** Interactions were limited to one buyer and one seller agent per run. This simplifies the dynamics compared to real-world marketplaces with multiple competing actors.
    - **Interaction Protocol:** The turn-based conversational model, while structured, does not capture the full complexity of real-time negotiation, concurrent actions, or non-verbal cues present in human interactions.
6. **General LLM Limitations:** The underlying LLMs (OpenAI and Google models) are subject to inherent limitations, including computational cost, potential latency in responses, susceptibility to generating biased or factually inconsistent statements (though grounding aims to mitigate this), and sensitivity to prompt phrasing. These factors can

influence simulation performance and reproducibility.

7. **Domain Specificity:** The findings of Experiment 2 are situated within the specific context of eBay sales interactions. While the principles of grounding may generalize, the specific challenges and effective strategies identified might differ in other domains with distinct interaction patterns, information availability, or agent objectives.

Acknowledging these limitations is crucial for interpreting the current results and guides the direction of future work aimed at building more robust, scalable, and reliable LLM-driven multi-agent simulations.

8. Future Work

This research highlights several promising avenues for future work aimed at enhancing the capabilities and reliability of LLM-driven Multi-Agent Systems for simulation and planning, with a particular emphasis on overcoming the limitations identified.

1. **Addressing the Information Bottleneck:** The critical dependence on external tool data quality necessitates further research into improving agent access to comprehensive and accurate information. Future work should explore:
    - **Enhanced Tooling:** Equipping agents with more sophisticated tools, such as web browsing capabilities to search beyond specific APIs, multimodal tools to process images or structured data, or integration with knowledge graphs for structured fact retrieval.
    - **Dynamic Information Seeking:** Developing agents that can recognize information gaps and proactively seek clarification or alternative data sources.
    - **API Development:** Investigating the co-design of simulation agents and the APIs they rely on to ensure necessary information granularity is available.
2. **Improving Agent Capabilities:** Enhancing the core reasoning and interaction abilities of the LLM agents remains crucial. This includes:
    - **Advanced Memory and Reasoning:** Implementing more sophisticated memory architectures (beyond simple conversation history or vector retrieval) and incorporating explicit reasoning mechanisms (e.g., causal inference modules, reflection on past failures) to improve planning and decision-making quality.
    - **Persona Dynamics:** Moving beyond static personas to explore dynamic persona adaptation based on interaction history or evolving simulation states. Investigating alternative persona generation techniques beyond TF-IDF/NMF.
    - **Fine-tuning:** Exploring the fine-tuning of LLMs on domain-specific conversational data or simulation traces to improve their performance and alignment within specific contexts like eBay sales.

3. **Increasing Simulation Complexity and Scale:** Future studies should aim to scale the simulations along several dimensions:
    - **More Agents:** Moving beyond 1-on-1 interactions to simulate environments with multiple buyers, sellers, or other agent types, introducing competition and more complex coordination challenges.
    - **Richer Environments:** Incorporating more dynamic environmental factors, temporal elements (e.g., stock levels changing over time), and more complex state representations.
    - **Longer Interactions:** Enabling simulations that run over extended periods, requiring robust long-term memory and state management.
4. **Enhancing Evaluation Methodologies:** Developing more robust methods for evaluating simulation realism and agent performance is essential.
    - **Quantitative Metrics:** Designing more nuanced quantitative metrics beyond simple success rates, potentially incorporating measures of interaction efficiency, negotiation outcomes, or alignment with specific behavioral theories. Establishing clearer baselines for metrics like persona relevance.
    - **Human Evaluation:** Incorporating human judgment systematically to assess the plausibility of conversations, the coherence of agent behavior, and the overall realism of the simulation outputs.
    - **Validation of Internal Mechanisms:** Developing methods to better validate internal evaluation mechanisms like sale detection or step assessment, potentially through comparison with human annotations or objective outcomes where possible.
5. **Domain Exploration:** Applying the grounded LLM-MAS framework developed here to different domains (e.g., customer service interactions, team collaboration simulations, urban planning scenarios, supply chain negotiations) would test the generalizability of the approach and reveal new domain-specific challenges and opportunities.

By pursuing these directions, future research can build upon the findings of this work to create increasingly sophisticated, reliable, and insightful LLM-driven multi-agent simulations.

9. Conclusion

This research addressed the challenge of leveraging Large Language Models (LLMs) to create more realistic and adaptive Multi-Agent System (MAS) simulations capable of capturing complex, human-like interactions. Traditional MAS often lack cognitive depth, while standalone LLMs struggle with environmental grounding and causal consistency. We proposed and investigated an approach integrating LLMs as agent decision-makers within a MAS framework, focusing specifically on enhancing grounding and situational awareness.

Through a two-phase experimental methodology, we first demonstrated the limitations of ungrounded LLM agents in an abstract planning task, observing tendencies towards causal spiraling and sensitivity to initial conditions without robust external context. Subsequently, we developed a grounded MAS simulating eBay sales interactions. This system successfully incorporated data-driven buyer personas (derived via TF-IDF/NMF analysis of consumer purchase data) and seller agents equipped with tools for live interaction with the eBay API. This grounded approach yielded significantly more plausible conversational dynamics and persona adherence compared to the ungrounded system.

However, our findings also revealed a critical dependency: the effectiveness and realism of the grounded simulation were fundamentally limited by the completeness of information accessible through the agents' external tools (the "information bottleneck"). While LLM agents could converse effectively, their ability to successfully complete tasks like sales transactions often hinged on the availability of specific details from the API.

In conclusion, LLM-driven MAS, when carefully designed with data-informed agent initialization and robust grounding mechanisms like external tool use, represent a promising paradigm for simulating complex real-world interactions with enhanced realism. Yet, their practical utility for generating reliable insights or informing planning is currently contingent not only on the sophistication of the LLMs themselves but critically on the quality, completeness, and accessibility of the external information sources they are grounded in. Overcoming this information bottleneck remains a key challenge and a vital direction for future research in realizing the full potential of LLM-based multi-agent simulation.

10. References

Cer, D., Yang, Y., Kong, S.-Y., Hua, N., Limtiaco, N., St. John, R., Constant, N., Guajardo-Cespedes, M., Yuan, S., Tar, C., Sung, Y.-H., Strope, B., & Kurzweil, R. (2018). *Universal sentence encoder*. arXiv preprint arXiv:1803.11175. https://arxiv.org/abs/1803.11175

He, Z., Chen, J., Zhang, Y., Zhu, Q., Tay, W. P., & Cheong, S. A. (2024). Exploring the Frontiers of Large Language Models in Social Science Simulations. *Humanities and Social Sciences Communications*, *11*, 672. https://doi.org/10.1057/s41599-024-03611-3

Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W., Rocktäschel, T., Riedel, S., & Kiela, D. (2020). Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, & H. Lin (Eds.), *Advances in Neural Information Processing Systems* (Vol. 33, pp. 9459-9474). Curran Associates, Inc.

https://proceedings.neurips.cc/paper/2020/hash/6b493230205f780e1bc26945df7481e5-Abstract.html

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). *Efficient estimation of word representations in vector space*. arXiv preprint arXiv:1301.3781. https://arxiv.org/abs/1301.3781

Park, J. S., O'Brien, J. C., Cai, C. J., Morris, M. R., Liang, P., & Bernstein, M. S. (2023). Generative Agents: Interactive Simulacra of Human Behavior. *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology (UIST '23)*. Association for Computing Machinery. https://doi.org/10.1145/3586183.3606763

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, & R. Garnett (Eds.), *Advances in Neural Information Processing Systems* (Vol. 30). Curran Associates, Inc. https://proceedings.neurips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html

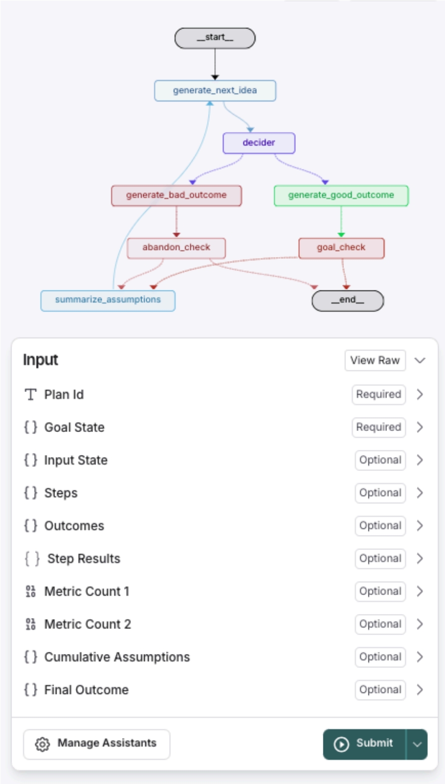
Wei, J., Wang, X., Schuurmans, D., Bosma, M., Ichter, B., Xia, F., Chi, E., Le, Q., & Zhou, D. (2022). *Chain-of-Thought Prompting Elicits Reasoning in Large Language Models*. arXiv preprint arXiv:2201.11903. https://arxiv.org/abs/2201.11903

Wooldridge, M. (2009). *An Introduction to Multiagent Systems* (2nd ed.). John Wiley & Sons.

Yao, S., Zhao, J., Yu, D., Du, N., Shafran, I., Narasimhan, K., & Cao, Y. (2022). *ReAct: Synergizing Reasoning and Acting in Language Models*. arXiv preprint arXiv:2210.03629. https://arxiv.org/abs/2210.03629

11. Appendices

Exhibit A

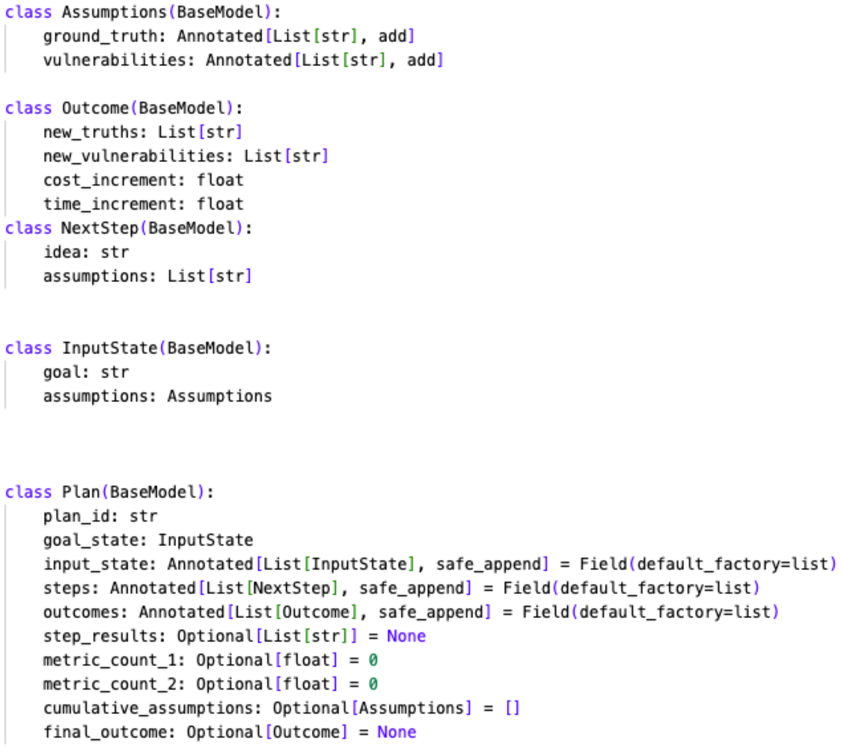


Input

View Raw

Plan Id	Required
Goal State	Required
Input State	Optional
Steps	Optional
Outcomes	Optional
Step Results	Optional
Metric Count 1	Optional
Metric Count 2	Optional
Cumulative Assumptions	Optional
Final Outcome	Optional

Exhibit B



```
class Assumptions(BaseModel):
    ground_truth: Annotated[List[str], add]
    vulnerabilities: Annotated[List[str], add]

class Outcome(BaseModel):
    new_truths: List[str]
    new_vulnerabilities: List[str]
    cost_increment: float
    time_increment: float
class NextStep(BaseModel):
    idea: str
    assumptions: List[str]


class InputState(BaseModel):
    goal: str
    assumptions: Assumptions



class Plan(BaseModel):
    plan_id: str
    goal_state: InputState
    input_state: Annotated[List[InputState], safe_append] = Field(default_factory=list)
    steps: Annotated[List[NextStep], safe_append] = Field(default_factory=list)
    outcomes: Annotated[List[Outcome], safe_append] = Field(default_factory=list)
    step_results: Optional[List[str]] = None
    metric_count_1: Optional[float] = 0
    metric_count_2: Optional[float] = 0
    cumulative_assumptions: Optional[Assumptions] = []
    final_outcome: Optional[Outcome] = None
```