Optimizing Human-Agent Interaction: Evaluating Diverse Strategies for Human Input in the OptiMUS LLM Agent System

Stanford CS224N Custom Project

İdil Defne Çekin
Department of Management Science and Engineering
Stanford University
icekin@stanford.edu

Isaiah Hall
Department of Computer Science
Stanford University
isaiah03@stanford.edu

Abstract

Despite the immense potential of large language model (LLM) agent automated systems, their deployment is constrained by critical limitations, notably in error propagation and generalization. Error propagation becomes particularly problematic in workflows where outputs from initial steps feed into later ones, with minor inaccuracies potentially magnifying into significant errors. Moreover, LLMs frequently falter when tasked with deep contextual or domain-specific understanding, often producing seemingly accurate responses that fail under scrutiny. These limitations exist alongside growing concerns about the opaque "black box" nature of such automated systems, increasing demand for more explainable AI systems. This study explores the intersection of reliability and explainability by investigating the relationship between human input and LLM agent system performance. Specifically, we investigate the trade off between output accuracy and human input in identifying mathematical modeling constraints from natural language descriptions of optimization problems. More importantly, we study how the performance of a human-in-the-loop component changes with its insertion point within the system. To explore these areas, we simulate human interaction using a two-tiered approach: Gemini 1.5-Pro emulates human understanding while Gemini Flash and Llama3-8B function as automated agents. While varying interaction frequency and location within the system, we benchmark this hybrid system against fully automated and fully human systems using an LLM-based optimization problem solver for evaluation. Our work assesses the performance improvements from human involvement and identifies differences in how each agent model responds to this involvement.

1 Key Information to include

- TA mentor: Yann Dubois
- External collaborators: No
- External mentor: No
- Sharing project: No

2 Introduction

The emergence of large language models (LLMs) has led many people to explore how these tools can be better leveraged in the real world. While these language models are not particularly powerful for completing complex tasks, they are increasingly being applied to more complex problems through the implementation of autonomous agent systems Wang (2023). Recent iterations of models such as

Llama Touvron (2023), GPT OpenAI (2023), and Gemini Gemini Team (2023) have increased the capability and lowered the cost of powerful LLMs. As a result, it is becoming increasingly possible to scale the size and cabability of autonomous agents and multi-agent systems. One problem solving task that exists within many industries and is ripe for automation is optimization.

Optimization problems serve as the backbone for enhancing efficiency and decision-making in sectors ranging from logistics and manufacturing to finance and energy Singh (2012). The automation of optimization problem solving is crucial for many industries as they move towards increased automation because it enables systems to make optimal decisions rapidly and reliably, which is vital for maintaining competitiveness and operational effectiveness. Automated optimization can lead to significant advancements in objectives such as resource allocation, cost minimization, and output maximization in complex environments. Thus, the ability to automate and solve these problems efficiently is not only a technological achievement but also a strategic imperative for businesses aiming to innovate and excel in the digital age.

OptiMUS Ahmadi Teshnizi (2023) proposes an LLM-based agent to make optimization more accessible. It translates natural language descriptions of real-world problems into Linear Programming (LP) or Mixed Integer Linear Programming (MILP) forms, generates the necessary code for problem-solving, and executes solutions efficiently. While this system is itself a massive step forward in the realm of automated optimization problem solving, it also brings to light a critical issue: the lack of a human-centered approach in its design. While OptiMUS significantly advances the technical capabilities of LLMs in automating complex optimization tasks, it primarily focuses on the efficiency and accuracy of translating and solving these problems, somewhat neglecting the user experience, particularly for non-technical stakeholders. This oversight renders the system somewhat of a "black box," where users may see outputs without understanding the processes leading to them. In real-world applications, where decisions based on optimization can have substantial economic, environmental, and social impacts, the ability for all users to comprehend and interrogate the system's recommendations and workings is crucial.

Thus, for systems like OptiMUS to be successfully integrated into everyday business practices, they need to incorporate more human-centric design principles. This involves not just making the system's operations transparent but also implementing a human-agent collaboration framework that allows the user to modify or question the system's approach to the problem.

Previous work has shown that human-agent collaboration methods outperform both human-only and agent-only methods on relatively complex task solving datasets while at the same time achieving an optimal balance between effectiveness and efficiency Feng (2024). However, the study also highlights the importance of determining the stages in the task-solving process where human intervention is most beneficial and effective. This motivates our study on the impact of human interaction on the performance of the OptiMUS system.

Our approach features three experiments, each evaluated with both LLama3-8B and Gemini Flash. First, we assess the effectiveness of human feedback prior to constraint extraction, where human expertise is used to clarify ambiguities and enhance the contextual framework of the problem descriptions. This stage is critical for setting a solid foundation for the automated processes that follow. In the second experiment, we explore the impact of human interventions immediately following the constraint extraction phase. Here, the focus is on identifying any potential errors or omissions in the constraints list and generating clarifying questions to rectify them. The third and final experiment investigates the benefits of human input after the constraint validation phase. This phase allows a human expert to review and provide a second opinion on the constraints deemed invalid by the agent. The objective here is to prevent the exclusion of viable constraints that might have been incorrectly flagged by the system.

By systematically introducing human feedback at these strategic points, our experiments aim to show the benefits of more collaborative and transparent problem-solving systems. This not only leverages the strengths of both human expertise and LLM capabilities but also addresses the critical need for more user-friendly and understandable AI systems in practical applications. The results from these experiments hope to offer insights into the optimal integration of human interaction within LLM agent systems, potentially influencing the design and implementation of future autonomous agents in various sectors.

3 Related Work

Our experiments are inspired by and reliant on the work of Ahmadi Teshnizi (2023), who created the OptiMUS system. We leveraged this optimization problem solving LLM agent throughout all three of our experiments, modifying the system pre-processing step for the implementation of our human-agent interaction testing.

For our evaluations, we leveraged the work done by Ramamonjison et al (2022) on the NL4Opt dataset, which provided a robust framework for assessing the efficacy of LLM agents in generating optimization problem formulations from natural language descriptions.

Additionally, Wu (2024) investigated the communication capabilities of LLMs in code generation tasks, underscoring the critical role of enabling LLMs to ask clarifying questions as a means to improve code accuracy. Concurrently, another study Feng (2024) illuminates the significance of strategically integrating human inputs at optimal moments, further supporting the notion that timely human-LLM collaboration can substantially enhance task outcomes.

4 Approach

4.1 Building on existing work

Our research is supported by the advancements made by the OptiMUS system Ahmadi Teshnizi (2023), which focused on increasing the accessibility of complex optimization solutions across various industries by creating an LLM agent system to translate natural language descriptions into Linear Programming (LP) and Mixed Integer Linear Programming (MILP) models. Specifically, we conduct experiments by introducing changes to the pre-processing stage of the system. During this stage, the agent processes an optimization problem description to extract constraints and objectives. Following this, the generated constraints undergo a validation process where potential inaccuracies or misinterpretations are reviewed by the agent. We conduct our baseline testing using the original pre-processing stage, and introduce modifications for each of our experiments to support adding a Gemini 1.5 Pro agent which acts as the "human". For each experiment, we generated an in-context learning (ICL) synthetic dataset using Gemini 1.5 Pro to teach the agent how to ask relevant clarifying questions based on the specific step in the system.

4.2 Preliminary Experiments

To gain a more general understanding of how the system responds to human feedback before running our main experiments, we ran a number of exploratory experiments. Some of these experiments leveraged modified datasets with introduced ambiguities or retracted information, hoping to illicit more substantial question-asking behavior from the agent. However, these experiments failed to provide relevant or interesting results. We concluded that these inconsistent results occurred because we were introducing the modified problem descriptions after the system had already assigned variable names from the parameters in the original problem description, and as a result the system could easily recover redacted information from the variable names, skewing the results. As a result, we decided to conduct our final evaluations using the original dataset.

4.3 Main Experiments

Our main experiments are designed to critically assess the impact of human intervention placement on the performance of the OptiMUS system. We measure the performance of the system when introducing human feedback in three different locations: Before constraint extraction, to clear up ambiguities in the problem description and add additional context; after constraint extraction, to identify any potential missing or incorrect constraints; and after constraint validation, to let the agent get a second opinion on constraints that it has marked as invalid. This systematic placement of human feedback aims to enhance the precision of the system's outputs at each stage, allowing us to measure downstream performance increases and reflect on the efficiency of human interaction at each point.

5 Experiments

5.1 Data

We randomly selected 62 instances from the NL4Opt dataset, which was also used to test the performance of the OptiMUS framework in the original paper. This dataset consists of natural language descriptions of problems, with system parameters identified by the OptiMUS system to be consistent with the data files, which indicate the values of the parameters. We did not include parameter identification in our preprocessing pipeline because the newly generated parameter names were not always consistent with the parameter names from the data files, causing errors in the system. Additionally, during our preliminary experimentation, we created two separate datasets by asking an LLM agent to modify these 62 instances, generating descriptions with added ambiguities and another set with missing information. We used in-context learning (ICL) examples to demonstrate to the agent how to create these datasets. However, the experiments with these datasets did not yield meaningful results and were therefore disregarded.

5.2 Evaluation method

The evaluation of our system centers on the primary metric of accuracy (Acc), defined as the percentage of problems in the dataset for which the system obtains the correct solution. To gain deeper insights into the impact and behavior of the human-in-the-loop system, we also consider several secondary metrics.

- **Number of Human interventions (#QA):** The number of times a question is asked to the "human" agent. The maximum number of questions ranges from 0 to 5, and while this number is fixed for most experiments, in one specific experiment, the total number of questions asked will depend on the number of times the agent needs validation. This metric helps assess the impact of the number of questions on performance and provides valuable insights into the model's behavior during validation.
- **Number of agent calls in the main system (#AC):** Higher quality preprocessing should result in fewer agent calls needed to solve the instance, as a well-preprocessed problem is easier for the system to solve. This metric, calculated as an average among correctly solved instances, indicates whether the human-in-the-loop system has made it easier for the main system to solve the problems.

5.3 Experimental details

In this experimental study, three distinct strategies for human input are investigated, with variations in timing and purpose of human interaction. Throughout these experiments, the effect of the amount of human input is examined by varying the number of questions asked, ranging from 1 to 5, with the baseline being no questions asked. The human is simulated using Gemini 1.5 Pro, and each experiment is conducted with two configurations, utilizing different Language Model Models (LLMs) as the question-asking agent: Llama3 8B, simulating the scenario where the human is more intellectually capable; and Gemini 1.5 Flash, supporting the scenario where the human model has a much less significant intellectual advantage.

- **Experiment 1 - Requesting clarification on problem definitions:** This strategy involves taking the problem description as input, posing clarifying questions about the problem description to the human, and updating the description based on received feedback. This experiments assess the agents capability in anticipating the information it may need for its next task and ask questions to retrieve that information from the human before ever attempting its original task.
- **Experiment 2 - Requesting clarification on constraints:** Here, the identified constraints are taken as input, and the system asks clarifying questions about them, subsequently updating the constraints according to the human's feedback. This experiments assess the agents capability in addressing its possible confusions after attempting some part of the task, and if it is able to identify the points it is not certain of and ask clarifying questions to fix potential mistakes.

- **Experiment 3 - Requesting clarification during constraint validation:** The constraint validation step involves taking the identified constraints and labeling them as valid or invalid, and the constraints labelled invalid are then disregarded. In this experiment the agent has the option to request validation from the human for the constraints it is planning to label as "invalid". The labels are then updated based on the received feedback. The responsibility of the agents during this experiment is to ask relevant questions which will provide human validation for their decisions. Since they get a limited number of questions, being able to identify and ask the correct questions is significant at this stage too. However, the feedback they get at this stage is much simpler than at the other stages.

5.4 Results

The quantitative results revealed varied impacts of human input on system performance, and they were less consistent than we anticipated. Overall, these results suggest that while human input can enhance performance, particularly for simpler models, the effectiveness varies based on the model's complexity and task requirements. The inconsistency in results indicates a need for further refinement in our human-in-the-loop strategies.

Experiment 1		**Baseline**	**Q = 1**	**Q = 2**	**Q = 3**	**Q = 4**	**Q = 5**
Gemini	#AC	4.45	3.81	**3.55**	3.82	*3.38*	3.93
	Acc	50%	46.77%	**53.23%**	53.23%	38.71%	46.77%
Llama	#AC	*4.22*	4.36	4.92	**4.24**	4.35	4.73
	Acc	43.55%	40.32%	41.94%	**46.77%**	32.26%	41.94%

Table 1: Results of Experiment 1 showing the number of agent calls (#AC) and accuracy (Acc) for different numbers of questions (Q).

Experiment 2		**Baseline**	**Q = 1**	**Q = 2**	**Q = 3**	**Q = 4**	**Q = 5**
Gemini	#AC	4.45	**3.3**	3.93	4.29	4.47	3.97
	Acc	50%	48.39%	46.77%	45.16%	48.39%	**51.61%**
Llama	#AC	4.22	4.54	4.92	4.43	3.9	**3**
	Acc	**43.55%**	41.94%	41.94%	33.87%	32.26%	33.87%

Table 2: Results of Experiment 2 showing the number of agent calls (#AC) and accuracy (Acc) for different numbers of questions (Q).

Experiment 2		**Baseline**	**Q = 1**	**Q = 2**	**Q = 3**	**Q = 4**	**Q = 5**
Gemini	#AC	4.45	3.56	4.5	3.84	**3**	3.56
	#QA	0	0.06	0.04	0.08	0.05	**0.09**
	Acc	50%	**51.61%**	48.39%	**51.61%**	46.77%	**51.61%**
Llama	#AC	4.22	4.29	3.93	**3.84**	4.29	**3**
	#QA	0	0.33	0.54	0.54	0.75	**0.9**
	Acc	43.55%	45.16%	45.16%	46.77%	**51.61%**	45.16%

Table 3: Results of Experiment 3 showing the number of agent calls (#AC), number of human interaction (#QA), and accuracy (Acc) for different numbers of questions (Q).

6 Analysis

For the Gemini Flash model, experiment 1 results show that the use of human input through clarifying questions before attempting any tasks, generally improved accuracy and reduced the number of agent

calls when asking up to 3 questions, with best performance being observed with 2. This shows that the Gemini Flash agent was able to ask meaningful questions on the problem description and make use of feedback in the form of clarifying answers to those questions. However, further increasing the number of questions resulted in diminishing returns and even a decrease in accuracy. This was mainly due to the simplicity of the problem descriptions, and that not many questions were necessary to be able to complete the tasks. We observed at this stage the agent resorted to some unnecessarily complex questions which resulted in confusions rather than clarifications. In contrast, the Llama model only showed improvement in performance for 3 questions, and for other cases, performance worsened, sometimes significantly. We observed that this model struggled in both identifying the correct questions to ask and in integrating the received feedback into the description. This agent needed more tries to find the correct question to ask and to be able to integrate the feedback, these struggles showed themselves as repeating the same questions when the feedback was not integrated appropriately, or asking redundant questions that are available in the description already. The performance peak in the experiment with 3 questions shows that the agent probably required approximately 3 questions to overcome these issues. These results also show that the effectiveness of human clarification may depend on the capabilities of the underlying model. Also, the fact that both models observed a performance drop for 1 question shows that the first question they asked was usually not spot-on.

Experiment 2, which focused on requesting clarification on a part of the completed task, presented mixed results. This experiment involved a more complex task for the agents because they had to consider both the previous steps and the further steps needed to complete the task. This complexity resulted in a significant drop in performance for Llama, evident in both accuracy and the number of agent calls. For the Gemini Flash model, the number of agent calls for the solved instances consistently decreased, indicating some improvement in preprocessing quality. However, accuracy slightly dropped in all but one experiment.

Experiment 3, which involved requesting validation on the constraint elimination decision, showed the most promising results for the Llama model. It achieved consistent accuracy improvements at various question levels, with a notable peak at 51.61% accuracy for 4 questions, reaching the best performance obtained with the Gemini model. The number of agent calls and human interactions also showed positive trends, with fewer calls required as the number of clarifications increased. This improvement of performance was mainly because of this agent's tendency to mistakenly disregard relevant information. When making the decision of labeling constraints, it usually decided to remove the ones that were necessary. This experiment results show that this agent was able to identify the good questions to ask when it has less and more straightforward options, and it was more successful in accommodating simpler feedback in the form of a validation. The Gemini Flash model also managed to improve its performance in the number of agent calls and accuracy, however, less consistently and significantly. We also need to note that the Gemini agent asked significantly less questions at this experiment compared to Llama, indicating that it was making better decisions regarding eliminating necessary constraints in the first place.

7 Conclusion

In conclusion, our research underscores the delicate balance required for integrating human-in-the-loop systems within LLM-based optimization frameworks. The experiments demonstrated that early human intervention, particularly during problem clarification and constraint clarification, tends to confuse the Llama agent and degrade its performance. While the Gemini Flash agent was better at these tasks and showed some improvements in performance. However, strategically timed human validation at the final constraint validation stage lead to more noticeable improvements, especially when the human agent was more capable than the question-asking agent. These findings suggest that the optimal approach to integrate human expertise in such systems depends highly on the specific capabilities of the models. For models that struggle with complex tasks requiring strategizing and critical thinking, focusing on validation rather than clarification can enhance performance. Conversely, when agents must interact with humans to obtain clarifications, utilizing more intellectually capable models might be more sensible. Additionally, the amount of feedback is a crucial parameter to optimize, as too few iterations can hinder finding the correct questions, while too much feedback can confuse the agents and diminish performance. The study also reveals that smaller LLMs often tend to disregard relevant information, underlining the need for robust validation processes. These insights

contribute valuable knowledge to optimizing human-agent collaboration, emphasizing the need for strategic planning in the integration of human expertise in AI-driven systems.

8 Ethics Statement

Ethical Challenges and Possible Societal Risks. It should be kept in mind that this approach, however advanced it gets, still requires supervision and should not be put in a position to apply the decision outputs of the problems by itself. The results of the optimization problems should be monitored by a human being for accordance and applicability. Blindly following the results of the system does not pose a societal risk by itself, however, depending on the severity of the situation might result in irrevocable or harmful consequences. The most straightforward solution to this problem would be to put a disclaimer in such systems that are developed in this area and to ensure human supervision on all final decisions.

Additionally, as our project aims to foster transparency through human-agent interaction, it is critical to address the potential for future projects to create an over-reliance on such transparency measures. While our system is designed identify and address issues, its inherent limitations and the possibility of errors must be acknowledged, especially in high-stakes scenarios. Over-reliance without adequate human judgment and intervention could lead to critical oversights and misjudgments, potentially endangering lives and resources. Therefore, the project must stress the importance of implementing additional safety protocols within real-world systems to safeguard against these risks.

References

Gao-W. Udell M. Ahmadi Teshnizi, A. 2023. Optimus: Scalable optimization modeling with (mi)lp solvers and large language models.

. Ramamonjison et al. 2022. Augmenting operations research with auto-formulation of optimization models from problem descriptions. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 29–62, Abu Dhabi, UAE. Association for Computational Linguistics.

Chen Z.-Y. Qin Y. Lin Y. Chen-X. Liu Z. Wen J.-R. Feng, X. 2024. Large language model-based human-agent collaboration for complex task solving. In *Journal of Hydrology*.

Google Gemini Team. 2023. Gemini: A family of highly capable multimodal models. In *Google Deepmind*.

OpenAI. 2023. Gpt-4 technical report.

Ajay Singh. 2012. An overview of the optimization modelling applications. In *Journal of Hydrology*.

Lavril T.-Izacard G. Martinet X. Lachaux M.-A. Lacroix T. Rozière B. Goyal-N. Hambro E. Azhar F. Rodriguez A. Joulin A. Grave E. Lample-G. Touvron, H. 2023. Llama: Open and efficient foundation language models. In *Meta AI*.

Ma C.-Feng X. Zhang Z. Yang H.-Zhang J. Chen Z.-Y. Tang-J. Chen X. Lin Y. Zhao W. X. Wei Z. Wen J.-R. Wang, L. 2023. A survey on large language model based autonomous agents. In *Frontiers of Computer Science*.

Fard-F. H. Wu, J.-J. 2024. Benchmarking the communication competence of code generation for llms and llm agent.