Universal Library of Innovative Research and Studies Research Article

ISSN: 3065-0003 | Volume 3, Issue 2

Open Access | PP: 01-05

DOI: https://doi.org/10.70315/uloap.ulirs.2026.0302001

The Evolution of Human–Machine Interaction Interfaces Based on Large Language Models

Dmitry Masyuk

Chief Executive Officer, Head of the Business Group for Search Services and Artificial Intelligence, Yandex Technologies LLC.

Abstract

The article examines the evolution of human–machine interaction interfaces shaped by the widespread adoption of large language models and by the shift from command- and menu-driven schemes to dialogue, multimodal input, and agent-based scenarios. The study's relevance arises from the profound transformation of user practices in search services, assistants, and recommendation products, in which short-text queries are increasingly being replaced by natural-language tasks. The novelty of the study lies in the analytical integration of several technological trajectories in interface development—conversational search, knowledge retrieval from external sources, tool-based extension, and multi-agent orchestration—into a unified model that makes it possible to describe systematically how model capabilities reshape the design of screens, prompts, and trust mechanisms. The article aims to identify stable patterns in the transformation of interface design as systems move from text generation to action execution. To address this aim, the study employs source analysis, comparative examination of approaches, and the structuring of architectural solutions. The research corpus includes scientific reports on multimodal models, surveys on agents and conversational search, and publications on API tool use and retrieval-based approaches. The article will be useful to executives and architects of digital products engaged in developing and scaling search services, assistants, and advertising platforms.

Keywords: *Human–Machine Interaction, Large Language Models, Conversational Search, Multimodal Interfaces, Agent-Based Systems, RAG, Tool Calling, Orchestration, User Experience, Trust In Generation.*

INTRODUCTION

The relevance of the topic is determined by the rapid shift in user intent from "searching for a document" to "obtaining a ready-made answer and completing an action," which changes the requirements imposed on the interfaces of search services, assistants, mapping applications, and advertising tools. Interaction is no longer confined to a short query format; increasingly, it takes the form of a task description containing clarifications, examples, and constraints. The spread of multimodal models intensifies this process: user input expands through images and other signals, while the output ceases to be purely textual and becomes a structured, controllable sequence of steps and calls to external functions.

The study aims to provide an analytical description of the evolution of human–machine interaction interfaces driven by the adoption of large language models and to identify stable design patterns for the development of large-scale product ecosystems.

The objectives of the study are as follows:

1. to explain the transition from dialogic text interaction to architectures in which the model performs actions through external tools and data sources;
2. to describe how retrieval-based approaches and conversational search reshape the interface of trust through references, confirmation mechanisms, and intent clarification;
3. to systematize agent-based and multi-agent schemes as the next stage in interface development, where the central object of interaction becomes the plan and its execution.

The novelty of the study lies in integrating the following developmental trajectory—conversational search → answer generation → tool integration → retrieval enhancement → agent orchestration—into a single model of interface evolution oriented toward the product needs of large-scale search and advertising systems.

MATERIALS AND METHODS

The empirical foundation of the article consists of scientific

Citation: Dmitry Masyuk, "The Evolution of Human–Machine Interaction Interfaces Based on Large Language Models", Universal Library of Innovative Research and Studies, 2026; 3(2): 01-05. DOI: https://doi.org/10.70315/uloap.ulirs.2026.0302001.

reports and review papers documenting the transition from text generation to multimodal interaction, the integration of external functions, and agent orchestration. J. Achiam et al. described a multimodal model and the practice of its evaluation across heterogeneous tasks [1]. R. Anil et al. presented a family of multimodal models and scenarios for working with different types of input signals [2]. L. Beurer-Kellner et al. formalized prompting as a query language that structures interaction between a human user and a model [3]. S. Chen et al. systematized LLM-based multi-agent solutions and approaches to inter-agent communication [4]. A. Grattafiori et al. described a model family with an emphasis on multilinguality, code, and tool-oriented applicability [5]. Y. Huang and J. Huang summarized retrieval-augmented generation and its evaluation criteria for tasks that rely on external data [6]. F. Mo et al. outlined the architecture of conversational search. They showed how multi-step dialogue transforms search relative to classical retrieval output [7]. T. Schick et al. proposed a method for training a model to call APIs at the appropriate time during an interaction [8]. A. Tsanda and E. Bruches described a Russian-language multimodal dataset and compared models for scientific text summarization [9]. S. Yao et al. proposed a framework combining reasoning and action to improve the controllability of problem-solving by recourse to external sources [10].

The article relies on the analysis of scholarly sources, comparative examination of approaches, and analytical modeling of interface architectures.

RESULTS

The evolution of HMI interfaces in the age of large language models is better described as a shift in the unit of interaction. In conventional interfaces, that unit was a command or a completed form; in dialog systems, it became the user's utterance; in contemporary LLM interfaces, it takes the form of a task specification containing a goal, constraints, examples, and quality criteria. The formalization of the prompt as a query language has reinforced this shift: the prompt has ceased to function as mere "text for the model". It has become an instrument for specification and process control through which the user defines the response format, the rules of data transformation, and the verification procedure [3]. At the interface level, this leads to the emergence of new elements: task templates, structured constraint fields, example blocks, response modes, and mechanisms for reformulating the task without losing semantic continuity across steps.

From the standpoint of interface design, this change shifts the design object itself. Instead of optimizing the input field and ranking the output, the focus moves to managing the task specification. The user is provided with tools for formulating requirements, while the system is expected to preserve the goal and response rules as a stable interaction frame across multiple steps.

The next stage is associated with the growth of multimodality. In the GPT-4 and Gemini reports, multimodality is described as the capacity to accept image and text inputs and generate text-based output grounded in visual features [1; 2]. In interface design, this transition is a redistribution of user intent from text to images. The user shows an object, a diagram, a screen, or a fragment of a document, and then formulates an interpretation or transformation task. In product scenarios involving search and maps, this produces a sequence of the form "multimodal input → clarifying dialogue → result in the form of an answer and action." At the same time, the cost of interpretive error increases, since a mistaken reading of the image changes the trajectory of the dialogue and all subsequent steps.

Multimodal input shifts interpretive error from the level of an "imprecise formulation" to that of an "incorrect problem-solving trajectory." Accordingly, an interface oriented toward images and screenshots requires ambiguity-reduction procedures: clarifying questions about the focal object, confirming interpretations, and explicitly separating observed features from the model's inferences.

In parallel, the transition from "answer generation" to "execution through tools" took shape. The Toolformer approach establishes the principle that a model learns when to call an API, which arguments to pass, and how to incorporate the returned result into subsequent reasoning [8]. At the interface level, this implies transparent tracing: the user sees the external operations performed, such as search, calculation, translation, calendar use, or reference lookup. Within the ReAct framework, interaction is organized as an alternation of reasoning and action, which reduces the share of errors caused by reliance on "plausible generation" [10]. In interface design, this leads to the emergence of a "plan–execution" mode: the user approves steps or constraints, after which the system performs actions and returns verifiable artifacts, including retrieved documents, excerpts, calculated values, and generated query parameters.

Tool-based extension changes user expectations. Value shifts away from the "quality of the text" toward the controllability of operations and the reproducibility of results. For that reason, interface design in such systems rests on two foundations: observability of actions, meaning what was done and with which parameters, and step control, meaning confirmation of conditions before an operation is executed. Together, these form the basis of operational transparency.

A third line of development is the integration of dialogue with search, where the interface does not end the interaction by presenting a list of links. The survey on conversational search describes an architecture in which dialogue is used to reformulate the query, ask clarifying questions, retrieve relevant documents, and generate an answer based on the retrieved material [7]. Unlike classical search, multi-step dialogue requires preserving semantic continuity across user turns: pronouns, ellipses, and implicit agreements established within the dialogue alter the interpretation of

subsequent queries. At the interface level, the clarification mechanism becomes more significant. The system does not merely answer; it asks a question that reduces uncertainty about intent and thereby lowers the share of irrelevant results [7]. For large search platforms, this logic shifts competition away from link ranking and toward dialogue management: quality is determined by how accurately and economically the system clarifies intent, how many steps the user must take, and how quickly the goal is reached.

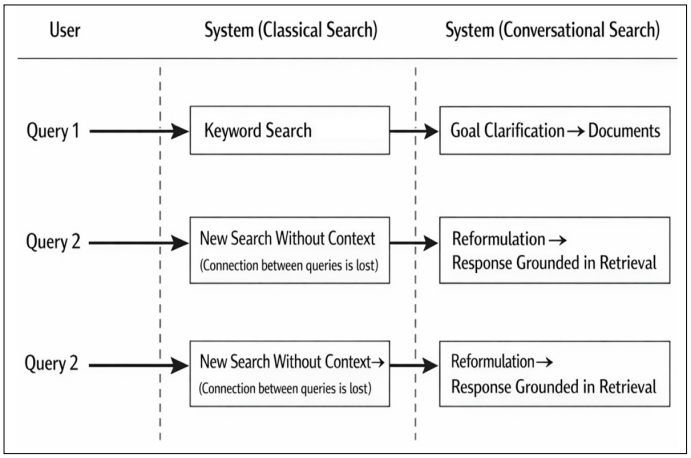

Figure 1. Comparison of classical search and conversational search with LLM-based answer generation (based on [7])

The comparison shown in the figure captures a design shift. In classical search, the interface serves document retrieval; in conversational search, it guides the user along a problem-solving trajectory through a sequence of clarifications and intermediate supports. The practical implication for the product is that the quality metric cannot be reduced to link ranking. Priority shifts to the precision of clarification questions, the preservation of the goal across multiple steps, and the minimization of unnecessary iterations while maintaining answer verifiability.

A fourth line of development is retrieval-augmented generation. The survey on retrieval-augmented text generation describes RAG as a scheme in which answer generation relies on externally retrieved data, thereby increasing verifiability and reducing the share of plausible but incorrect statements [6]. For the interface, RAG changes the trust contract: the user expects an answer grounded in sources. This gives rise to interface requirements such as displaying the provenance of facts through links and excerpts, distinguishing between "retrieved from sources" and "formulated by the model," introducing mechanisms for confirmation and clarification, and, in business platforms, recording which data were used and under what permissions. Feedback grows more complex as well: the user evaluates the correctness of retrieval and citation, which requires dedicated interface elements for quality control, such as source selection, filtering, and domain restriction.

At the UX level, the retrieval layer turns trust into a verifiable procedure. The user evaluates where the factual material comes from, how accurately retrieval has been performed, and where the boundary lies between what was found and what was formulated. As a result, an interface that does not explicitly display its evidentiary support is perceived as "non-verifiable," even when the generated answer appears outwardly convincing, which directly affects perceived product reliability.

A separate result is associated with the growth of "agency" as an interface property. Surveys on autonomous agents and multi-agent systems describe schemes in which the LLM acts as a coordinator of subtasks, distributes work across specialized modules, maintains task memory, and manages calls to external tools [4]. Under this arrangement, dialogue ceases to be the only interaction channel: the user interacts with a plan, execution status, and result artifacts. This changes interfaces in search and assistant systems. Instead of a single answer, a panel of steps shows what has been found, what has been verified, and what still requires clarification; instead of "rewrite the query," the interface offers "adjust the goal" or "modify the constraint." In map services and local search, this is especially evident: the task is often formulated as "find and choose," followed by comparing options, refining conditions, constructing a route, and turning the result into action. Such chains are more naturally described through agent orchestration than through a single response [4; 10].

Agent-based scenarios move user communication from a "question–answer" mode to a "process management" mode. The user edits goals and constraints, observes status, and accepts results step by step. For product architecture, this implies the emergence of a full-fledged state and logging layer; for the interface, it requires displaying the execution process itself, including intermediate decisions and verification points.

Another result concerns the growing influence of model scale and robustness on the interface. The description of the Llama 3 family records large context windows and an orientation toward multilingual and tool-centered scenarios [5]. At the practical level, a large context window changes UX: the user can provide more material, including documents, lengthy requirements, and prior correspondence, and the interface gains a rationale for supporting "batch-style" interaction through material upload and task formulation over that material. Even so, expanding the context window does not eliminate the need for retrieval-based approaches. When external knowledge is involved, value comes from controlled retrieval and source verification, which is precisely what RAG surveys emphasize [6]. Consequently, interfaces of large search services tend toward a hybrid scheme: extended task formulation combined with a retrieval layer and source tracing, so that the user can understand the origin of the answer and adjust the process when needed.

A sixth result is connected with the features of local languages and the Russian-language applied base. The description of the Russian-language multimodal dataset documents the practice of comparing models on material containing text, tables, and figures [9]. For interfaces in Russian-language products, this

means that interaction quality depends on its stability across specialized genres such as scientific documents, technical instructions, user reviews, and advertising texts. In products belonging to search and advertising ecosystems, this leads to the separation of interface modes: a reference-answer mode, a source summarization mode, a draft text generation mode, and a user-data processing mode. Each of these modes requires its own constraints, prompts, and quality criteria.

DISCUSSION

The obtained findings shift interface design for large-scale search and assistant platforms into the domain of a managed process: the user interacts with the procedure through which the answer is produced and verified. With tool-based extension (Toolformer) and the integration of reasoning with action (ReAct), the interface acquires formal control points: which external operations have been executed, which data have been used, and where goal clarification is required [8; 10]. For search services, mapping products, and assistants operating under high-load conditions, this has a practical effect: the number of unproductive user steps decreases, while the share of tasks completed without manual navigation across multiple pages increases. At the same time, demands for transparency and for managing the risk of a "plausible error" become higher, and retrieval-based approaches address this issue directly [6; 7].

In products of this type, the interface begins to function as a risk regulator. It reduces the probability of a plausible but incorrect output through the following procedures: intent clarification, presentation of evidentiary support, recording of completed operations, and preservation of the goal. A direct managerial implication follows from this: interface design and quality engineering become interconnected control domains, since UX elements turn into checkpoints for reliability and executability.

Table 1 shows that interface evolution can be described as expansion of control: from control over response format to control over sources and operations, and further to control over execution sequence. This provides grounds for viewing the LLM interface as a managed system in which the reproducibility of the user path measures quality.

Table 1. Transition of Interface Patterns in the Development of LLM Systems [1–4; 6; 7; 10]

Interaction Pattern	**Technical Foundation**	**Type of Result in the Interface**	**Typical Product Scenarios**
Prompt as a task language	formalization of the query and response rules	task template, examples, response format	text preparation, analytical briefs, work assistance
Multimodal input	processing of image and text	image-based answer, clarifying dialogue	image search, screenshot interpretation, visual material analysis
Conversational search	reformulation, clarification, retrieval, generation	Answer with clarifying questions and grounding in retrieved material	complex reference queries, option comparison, selection
Retrieval enhancement	retrieval of external data for answer generation	links, excerpts, separation of "data/formulation."	factual queries, regulations, product information
Agent-based schemes	reasoning plus action, task orchestration	step-by-step plan, execution status, artifacts	"find–compare–select–complete," routine automation

After the transition to these patterns, the interface of search and advertising systems acquires a new quality dimension. In addition to result relevance, evaluation now includes the cost of interaction, such as the number of clarifications, the volume of manual editing, and the speed of goal completion, as well as the degree of answer verifiability. Multimodal reports document the expansion of input signals [1; 2], while surveys on conversational search and retrieval-augmented generation describe architectural measures that support fact-checking through source retrieval [6; 7].

For managers responsible for product directions in search, assistants, maps, and advertising solutions, the practical conclusion is straightforward: the LLM interface ceases to be a "chat form" and becomes a layer for managing the computational process itself, including goal setting, data collection, verification, execution, and the return of artifacts.

In product practice, the transition to LLM interfaces entails predictable risks related to reliability, goal preservation, action controllability, and orchestration complexity. These risks are summarized below in an engineering map of mitigation measures suitable for defining interface and architectural requirements (see Table 2).

Table 2. Typical Risks of LLM Interfaces and Architectural Mitigation Measures [4; 6–8; 10]

Interface Risk	**Origin**	**Mitigation Measure**
Plausible but incorrect answer	generation without verification	retrieval enhancement and source display
Loss of semantic continuity between turns	multi-step dialogue	clarification, reformulation, goal fixation
Uncontrolled actions in tool chains	calls to external functions	tracing of calls and arguments, step confirmation

Error in step planning	mixing of reasoning and action	alternation of reasoning and action with external verification
Growth of system complexity during orchestration	multi-agent architecture	separation of functions, exchange protocols, execution control

Within this scheme, multimodal models expand the input channel [1; 2], conversational search provides the structure of clarification [7], retrieval-based approaches ensure verifiability [6], and tool-based and agent-based methods form an executable trajectory for solving the user's task.

The reduction of risks through these measures shows that controllability and trust in LLM interfaces are achieved through specific control points embedded into both the screen and the process. In the design of high-load services, this implies a direct prioritization: first, observability of sources and actions; only then, optimization of formulation convenience, since even ideal generation loses value when verification remains unclear and operational execution stays opaque.

CONCLUSION

The transition to large language models fundamentally changes the interface. The user comes for a procedure that leads to a goal. Within that procedure, task formulation takes the form of a specification containing constraints, examples, quality criteria, and an expected result type. For that reason, interface development is most consistently described through a shift in the object of control: from the utterance to the goal, and from text to an operationally executable sequence of steps.

Conversational search and retrieval enhancements shift trust from the realm of psychological expectation into verifiable mechanics. The user perceives an answer as reliable when it is clear how the intention was clarified, on which materials the factual basis rests, which fragments were extracted from sources, and where retrieved data ends, and model formulation begins. This leads directly to concrete requirements for screen design: presentation of evidentiary supports, fixation of the goal, economical clarifying questions, and preservation of semantic continuity across steps.

Tool-based and agent-based scenarios turn the interface into a layer of operational control. The user needs to see which actions have been performed, with which parameters, in which order, and where constraint editing remains available. In products across the search, mapping, and advertising ecosystems, such a mode reduces the "cost of interaction": fewer manual transitions, fewer repeated queries, and more tasks completed. At the same time, the cost of error becomes higher. For that reason, interface design and architectural risk-mitigation measures need to be considered jointly: tracing of calls, confirmation of steps, presentation of sources, state management, and execution logging together form a unified system for ensuring trust and controllability.

REFERENCES

1. Achiam, J., Adler, S., Agarwal, S., et al. (2023). *GPT-4 technical report*. arXiv. https://arxiv.org/abs/2303.08774
2. Anil, R., Dai, A. M., Firat, O., et al. (2023). *Gemini: A family of highly capable multimodal models*. arXiv. https://arxiv.org/abs/2312.11805
3. Beurer-Kellner, L., Fischer, M., & Vechev, M. (2022). *Prompting is programming: A query language for large language models*. arXiv. https://arxiv.org/pdf/2212.06094
4. Chen, S., Zhou, Z., Wang, J., et al. (2024). *A survey on LLM-based multi-agent system*. arXiv. https://arxiv.org/abs/2412.17481
5. Grattafiori, A., Dubey, A., Jauhri, A., et al. (2024). *The Llama 3 herd of models*. arXiv. https://arxiv.org/abs/2407.21783
6. Huang, Y., & Huang, J. (2024). *A survey on retrieval-augmented text generation for large language models*. arXiv. https://arxiv.org/abs/2404.10981
7. Mo, F., Mao, K., Zhao, Z., et al. (2024). *A survey of conversational search*. arXiv. https://arxiv.org/html/2410.15576v1
8. Schick, T., Dwivedi, Y., Dessì, R., et al. (2023). *Toolformer: Language models can teach themselves to use tools*. arXiv. https://arxiv.org/abs/2302.04761
9. Tsanda, A., & Bruches, E. (2024). *Russian-language multimodal dataset for automatic summarization of scientific papers*. arXiv. https://arxiv.org/abs/2405.07886
10. Yao, S., Zhao, J., Yu, D., et al. (2022). *ReAct: Synergizing reasoning and acting in language models*. arXiv. https://arxiv.org/abs/2210.03629