LLMs for Quantitative Investment Research: A Practitioner's Guide

Anna-Helena Mihov[1,2], Nick Firoozye[1] and Philip Treleaven[1]
[1]UCL Computer Science, [2]DWS*

ABSTRACT

Large Language Models (LLMs) and more broadly generative AI (GenAI) are beginning to reshape the portfolio-management landscape, influencing both quantitative and fundamental research practices (Spyrou and Pisaneschi, 2024). Their capacity to process unstructured information, interpret complex narratives, and produce research-ready outputs creates meaningful opportunities for institutional investors, while also introducing technical and governance challenges. This paper provides a practitioner-oriented overview of the evolving role of LLMs in quantitative investment research, outlining current applications, methodological considerations, and organisational implications.

Three areas are examined in which LLMs are changing day-to-day research workflows. First, LLMs function as research assistants ("LLM Assistant") that support literature synthesis, data exploration, code generation, and retrieval of institutional knowledge. Second, in LLM-based quantitative modelling ("LLM Quant"), models extract signals from text and multimodal data, generate features, and interface directly with numerical pipelines. Third, LLM-enabled Augmented Financial Intelligence (AFI) ("LLM Quantamental") deals with multiple applications: Scaling of fundamental reasoning, systemising qualitative insight, and creating a bidirectional link between discretionary and systematic processes. Drawing on practitioner case studies, industry experience, and recent academic work, this review describes both the realised benefits of these systems and the associated risks, including hallucination, temporal leakage, reproducibility constraints, and challenges related to objective alignment.

Practical guidelines are provided for quantitative research teams on model selection, retrieval-augmented grounding, prompt design, and the integration of LLM systems into governed research pipelines. Although LLMs are not reliable stand-alone forecasting tools, they already enhance signal extraction, shorten research cycles, and improve the interpretability of modelling outputs. When deployed responsibly, they act as complementary cognitive tools that extend analytical capacity while preserving the essential role of expert judgment, supporting a more iterative and scalable approach to quantitative innovation.

1. Introduction

LLMs are transforming quantitative investment research, reshaping how information is processed, signals are generated, and human expertise is integrated into systematic workflows. For quantitative researchers and investment professionals, LLMs function as powerful research assistants, accelerating information discovery, enabling sophisticated text analysis, providing coding support, and streamlining operational tasks across the research lifecycle. As firms modernise their data and model infrastructure, LLMs are becoming central to the quant stack, contributing not only to Alpha research but also to the design of next-generation quantitative platforms.

One of the most immediate applications of LLMs lies in text-based alternative datasets, where they mark a structural shift away from traditional dictionary methods. Earlier approaches depended on lexicons, bag-of-words models, and modular natural language processing (NLP) pipelines that decomposed text into sentiment or entities. LLMs introduce a different paradigm: context-aware language understanding, richer semantic extraction, and the ability to map unstructured text directly to economic impact. This transition from modular, task-specific pipelines to end-to-end LLM-driven impact modelling has already begun to reshape quantitative strategies. At the same

* The views expressed in this paper are not necessarily shared by DWS or its subsidiaries.

time, these opportunities introduce new methodological challenges, including forward-looking bias, temporal contamination, model-selection trade-offs, and the need for evaluation frameworks suited to foundation models.

Beyond text analytics, LLMs advance Quantamental research and Augmented Financial Intelligence (AFI) (Mihov, Firoozye and Treleaven, 2022). Quantamental investing traditionally used quantitative models to enhance fundamental investment processes. AFI involves integrating human expertise into algorithms, resulting in a new generation of adaptive financial models. LLMs expand this frontier by enabling human expertise, analyst reasoning, sector knowledge, macro narratives, and judgment, to be converted into structured and scalable datasets that quants can analyse systematically. This creates a new class of expert-derived alternative data and supports deeper integration of qualitative insight into quantitative workflows. Conversely, LLMs enhance communication in the opposite direction from quants to fundamental specialists: by translating complex model outputs into natural-language explanations, they make quantitative signals more interpretable and actionable for fundamental portfolio managers. For clarity, this article uses fundamental and discretionary interchangeably, as well as quantitative and systematic.

LLMs have rapidly evolved from tools that simply generate text into systems capable of interpreting financial language, extracting context, and supporting analytical tasks. A newer class of reasoning models (e.g. Claude 3.7 Sonnet, DeepSeek R1, Gemini 2.5) extends these abilities by performing structured, step-by-step problem solving, enabling more reliable explanations, code generation, and statistical reasoning. Alongside these developments, firms are beginning to deploy agents, LLM-powered components that can take actions such as running scripts, querying data, or producing draft research. When these agents are combined into coordinated agentic workflows, they can automate multi-step processes such as data preparation, backtesting, or documentation while keeping humans in control. In parallel, retrieval-augmented generation (RAG) has become a standard method for grounding LLMs in proprietary financial data, allowing models to incorporate real-time information, reduce hallucinations, and integrate seamlessly with internal research systems. Finally, advances in deep research tools enable models to read and connect information across large volumes of filings, transcripts, reports and academic research. Together, these developments form a new toolkit for quantitative investment researchers, expanding analytical capacity without replacing the judgment, verification, and domain expertise required for real-world decision-making.

The industry's engagement with LLMs in quantitative investment research and portfolio management can be understood as occurring in distinct stages (see Figure 1). Following the logic of the artificial intelligence (AI) hype cycle of Gartner (2024), adoption over the past several years can be divided into multiple phases. For this review the phases were adjusted to focus on four core stages. This pattern mirrors the diffusion of other general-purpose technologies, where early experimentation evolves into broad enthusiasm and is later followed by rationalisation, consolidation, and selective integration. The years associated with each stage are approximate and reflect broad industry tendencies rather than precise adoption rates.

Taken together, LLMs are multi-purpose models that can enhance multiple tasks associated with quantitative investment research within organizations: improving efficiency, enhancing Alpha discovery, systemising expert reasoning, and strengthening collaboration between quantitative and fundamental teams. This practitioner's guide reviews these developments, evaluates the academic evidence, and outlines practical approaches for integrating LLMs into modern investment research workflows.

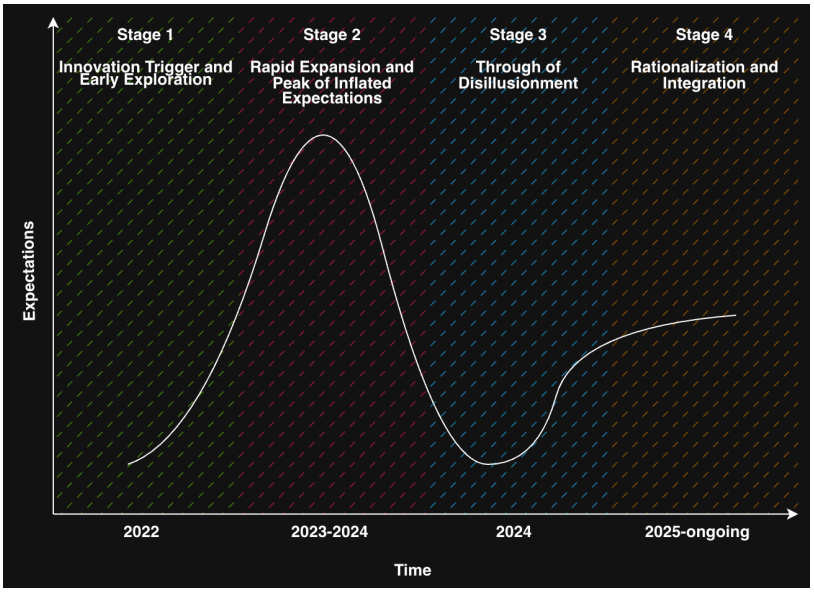

Figure 1: Evolution of LLM adoption in quantitative investment research resembles the Hype Cycle for AI. The curve illustrates the progression from innovation trigger (2022), through rapid expansion (2023-2024) and through of disillusionment (2024), to a more stable and selective integration phase (2025 onward).

1.1. Stages of LLM Adoption in Quantitative Investment Research

Stage 0 (Pre-ChatGPT): Transformer-Based Models in Quantitative Investment Research

Before the emergence of ChatGPT and the proliferation of generative LLMs, quantitative investment researchers were already engaging with Transformer architectures, mostly with interpretative (discriminative) models, significantly different from current generative AI (GenAI) models. Between 2018 and 2022, Transformers entered finance primarily through task-specific, encoder-based architectures such as BERT (Devlin et al., 2019) and finance-specific fine-tuned models like FinBERT (Araci, 2019). These models support amongst others the core tasks of sentiment classification and named-entity recognition. Their adoption marked a meaningful shift away from dictionary-based sentiment and bag-of-words methods, establishing Transformers as the new baseline for financial NLP. Additional encoder models, including RoBERTa (Liu et al., 2019), ELECTRA (Clark et al., 2020), DeBERTa (He et al., 2021), and finance-specific adaptations such as FinBERT (Araci, 2019) finetuned on the Financial PhraseBank dataset (Malo et al., 2014) and FinBART (Yang, Zhang & Gao, 2020), expanded analytical capabilities while maintaining deterministic and classification-oriented structures suited financial text analysis tasks.

In parallel to developments in language modelling, Transformer architectures began to influence quantitative forecasting, particularly in financial time-series applications. Models such as the Temporal Fusion Transformer (Lim et al., 2021), Informer (Zhou et al., 2021), Autoformer (Wu et al., 2021), and FEDformer (Zhou et al., 2022) were applied to forecasting returns, volatility, and macroeconomic indicators. Wood et al. (2022) introduced the Momentum Transformer, an attention-based architecture to improve benchmark time-series momentum and mean-reversion trading strategies, and compared it to a LSTM-based Deep Momentum Network. Transformer models can capture longer-term dependencies and show greater robustness to non-stationarity and regime shifts, while also offering improved interpretability.

These developments are technically significant but to some degree confined to research settings as Transformer-based time-series models are computationally intensive and more complex with slightly more variance and such more sensitive to hyperparameter tuning. Consequently, special care is necessary to ensure robust backtesting and to leverage the available interpretability (e.g. through features like the Variable Selection Network (Wood et al., 2022) or analysis of attention patterns). These contributions nevertheless provided an important technical

foundation for the attention-based architectures that power todays LLMs. The remainder of this review focuses on LLMs and text-based applications.

Thus, in the pre-ChatGPT era, financial AI research was already shaped by Transformers, but in specialised, narrow, and non-generative forms. NLP deployments relied on small, fine-tuned, deterministic Transformer models, while forecasting research explored the application of attention mechanisms to numerical data. LLMs today are general purpose models capable of broad text understanding, reasoning-pattern extraction, and interactive analysis. Stage 0 therefore represents the technical foundation on which subsequent stages of LLM adoption were built: advanced in architecture, but fragmented, specialised, and far from the unified capabilities introduced by large generative models.

Stage 1 (2022): Innovation trigger and early exploration

The November 2022 release of ChatGPT, based on GPT-3.5, marked the industry's first large-scale exposure to conversational AI and dramatically lowered the entry barriers to experimentation with LLMs. Although pretrained Transformers were already known within quantitative research circles, ChatGPT catalysed a broader phase of familiarisation across investment teams, prompting researchers, quants, and portfolio managers to probe the capabilities of LLMs in a low-friction environment.

In this initial phase, applications centred on generic natural-language tasks rather than domain-specific modelling. Practitioners used LLMs for text generation, summarisation, explanation, and knowledge retrieval, often comparing them informally to existing tools such as FinBERT or bespoke sentiment classifiers. Most experimentation occurred within the familiar boundaries of classical NLP workflows, sentiment scoring, news classification, entity extraction, and earnings-call or 10-K summarisation, tasks that earlier models were already designed to perform.

For quantitative investment teams, this stage was characterised by exploration rather than deployment**.** Researchers investigated whether LLMs could match or exceed the performance of supervised financial NLP models trained on domain-specific corpora, while beginning to develop the early operational skills of prompt engineering, reliability testing, and workflow integration. Across the industry, LLMs were viewed primarily as research assistants or prototyping tools, not yet as components of production pipelines.

This exploratory period laid the groundwork for the subsequent surge of enthusiasm and model adoption, as firms recognised that LLMs could extend beyond generic text manipulation to more sophisticated financial reasoning and data-extraction tasks.

Stage 2 (2023–2024): Rapid Expansion and Peak of Inflated Expectations

The release of GPT-4, Claude, and other frontier models in 2023 initiated a period of expansion across all industries. As API access widened and infrastructure such as vector databases, orchestration frameworks, and turnkey RAG pipelines became widely available, the technical barriers to experimentation fell sharply. Tasks that previously required considerable engineering expertise, document indexing, semantic search, structured extraction, and text-to-SQL, became accessible to quantitative researchers and non-specialist analysts. This rapid lowering of the entry threshold fuelled the belief that LLMs could be applied across nearly every research or operational task. During this phase, practitioners experimented with using generative models for data cleaning, memo drafting, macro commentary, automated report writing, thematic idea generation and exploratory coding.

Industry enthusiasm was reinforced by empirical work suggesting that LLMs could mimic elements of expert judgment. Studies such as Pelster and Val (2024), Glasserman and Lin (2023), and Ko and Lee (2024) showed that LLMs could produce analyst-style assessments, derive predictive sentiment from news headlines, or improve portfolio-choice quality in controlled settings. These findings encouraged the view that GenAI might augment or even automate significant components of the investment research workflow. At the same time, the limitations of raw model outputs became increasingly visible, prompting the widespread adoption of RAG systems, validation

layers, and prompt-engineering heuristics designed to stabilise outputs. Nevertheless, the overarching narrative during this period was highly optimistic, with LLMs perceived as general-purpose cognitive copilots for both discretionary and systematic teams.

Stage 3 (2024): Through of Disillusionment

By the second half of 2024, many practitioners began to confront the structural limitations of large language models, marking the onset of a disillusionment phase. A central realisation was that LLMs are probabilistic token generators whose outputs vary across runs, making them fundamentally different from deterministic analytical engines. Despite improvements in grounding and structured prompting, hallucinations remained difficult to eliminate, and response quality continued to depend on prompt formulation and contextual cues, as surveyed by Ji et al. (2023).

Concerns about forward-looking bias and temporal contamination also became prominent. Glasserman and Lin (2024) highlighted that pretrained models may incorporate information from future periods, posing risks of leakage in financial backtests unless stringent temporal controls are applied. Meanwhile, although Lopez-Lira and Tang (2025) demonstrated that LLMs can interpret news in an analyst-like manner, later discussions in the literature emphasized the need for caution when using pretrained models for prediction in temporally constrained settings.

Operational constraints further dampened expectations. Inference costs and latency limited the use of large models in real-time environments, and non-determinism complicated integration into risk-sensitive workflows. Compliance and governance teams raised concerns about auditability, reproducibility, and explainability, noting that LLMs' stochastic nature is difficult to reconcile with regulatory expectations for model-risk management. As in other safety-critical domains, such as autonomous driving, machine errors were judged more harshly than human ones, despite comparable error rates. By late 2024, firms increasingly questioned whether LLMs were suitable for certain quantitative tasks, particularly those requiring stable, repeatable outputs, such as sentiment scoring, factor construction, or forecasting. As a result, LLMs remained widely used as research assistants, summarisation engines, and coding tools, but their role in decision-critical components of the investment process was scaled back, with human oversight remaining essential.

Stage 4 (2025-ongoing): Rationalization and Integration

Entering 2025, the industry is moving toward a phase of rationalisation and sustainable integration. The initial belief that LLMs could serve as universal problem-solving engines has given way to a more pragmatic understanding of where these systems add distinct value. Firms increasingly deploy LLMs in domains where linguistic comprehension, unstructured-text processing, and knowledge retrieval are central, including coding assistance, compliance checks, documentation analysis, internal knowledge-base search, and the extraction of structured signals from textual sources. These applications complement rather than replace traditional quantitative models.

The prevailing approach is one of augmentation: LLMs accelerate exploratory analysis, streamline operational workflows, and assist analysts and quantitative researchers in synthesising information, while domain experts remain responsible for validation, interpretation, and judgment. Hybrid architectures that integrate LLM-derived narrative features with econometric, factor-based, or machine-learning models have gained traction. These models allow textual and numeric information to coexist without relying on the stochastic reasoning behaviour of the LLM itself. In parallel, governance and explainability frameworks have matured, with firms implementing deterministic production pipelines, retrieval-based grounding to reduce hallucinations, versioned model checkpoints, and audit trails aligned with regulatory model-risk standards.

A related development, though still in early exploratory stages in quantitative pipelines, is the rise of agentic AI workflows, in which LLMs are embedded in multi-step task pipelines that can plan, call tools, query databases, write and execute code, or iteratively refine outputs. In quantitative research, these systems are being evaluated for their potential to automate routine modelling steps like backtesting new signals, orchestrate data-cleaning or exploratory pipelines, and assist in research documentation. However, given unresolved issues around reliability, state tracking, safety constraints, and reproducibility, current adoption remains cautious. Agentic systems are treated as experimental productivity enhancers rather than autonomous decision-making components, consistent with the industry's broader shift toward governance-conscious deployment.

Overall, this stage represents a form of mean reversion within the LLM hype cycle. Exuberant expectations have stabilised, and practitioners now emphasise specific, economically grounded applications rather than universal automation. LLMs are recognised as powerful linguistic and productivity tools, most effective when embedded within a broader quantitative infrastructure and supervised by human expertise. The focus is no longer on the technology's revolutionary potential but on its targeted, reliable contribution to modern investment research and workflow efficiency.

As such, LLMs increasingly complement rather than replace quantitative systems. Over the long term, their role is likely to resemble other pervasive but background technologies (e.g., spreadsheets, version-control systems, and statistical programming languages), widely embedded in research and reporting pipelines, central to workflow efficiency, yet integrated in a stable, unremarkable manner within the broader investment process.

1.2. Overview GenAI Use-Cases for Portfolio Management

The portfolio management value chain benefits from a diverse range of GenAI use cases (see Figure 2), organized by implementation difficulty and strategic impact. At the foundational level, operational tasks such as answering requests for proposals (RFPs), client reporting, and compliance checks represent the most accessible and deterministic applications. These systems typically employ LLMs or standardized document generation, information extraction, and summarization, thereby enhancing efficiency and reducing manual workloads. While technically straightforward, these implementations still require careful consideration of data security, regulatory compliance, and the integration of firm-specific knowledge to ensure reliability and institutional alignment.

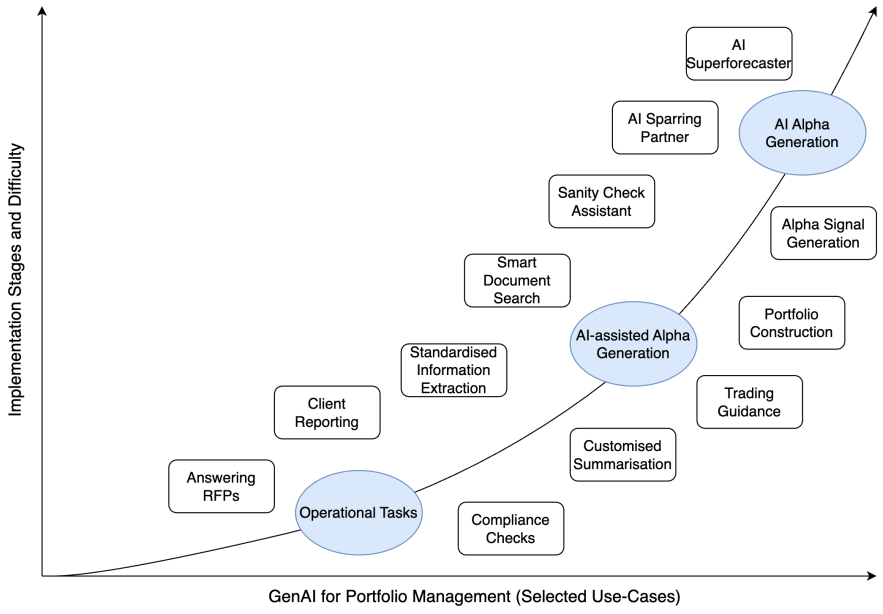

Figure 2: Overview of GenAI use-cases for portfolio management with three distinct categories of operational tasks, AI-assisted Alpha Generation and AI Alpha Generation.

Progressing up the complexity curve, AI-assisted and AI-driven Alpha Generation represent increasingly advanced stages of GenAI adoption. The intermediate stage includes tools such as smart document search and customized summarisation, where RAG systems enhance the analytical capabilities of investment professionals. At the frontier, AI Alpha Generation, most relevant for quantitative investment research, encompasses predictive and decision-support applications such as Alpha signal generation, trading guidance, and AI Superforecaster models (Mihov, Firoozye and Treleaven, 2022) that aim to extract and operationalize market-relevant insights from unstructured data. Although these use cases promise substantial competitive advantages, they introduce greater challenges related to validation, explainability, and robustness. Addressing these limitations is essential for advancing GenAI in portfolio management from experimental pilots to production-grade decision systems. More specifically, within the context of quantitative investment research, three core areas are being transformed most profoundly: research assistance, analysis of text-based alternative datasets, and the quantification of fundamental human decision-making, each of which will be discussed in the following sections.

1.3. Key Design Choices of Building AI Systems

In general, building GenAI systems requires a series of connected design decisions. For simplicity, and to provide a clear conceptual overview, this section focuses on the three most fundamental choices. These choices ultimately shape model performance, governance, and usability. As illustrated in Figure 3, they can be organised into three concentric layers: (1) the selection of the core LLM, (2) the optimisation techniques used to adapt the model to the investment domain, and (3) the balance between user customisation and organisational control through guardrails.

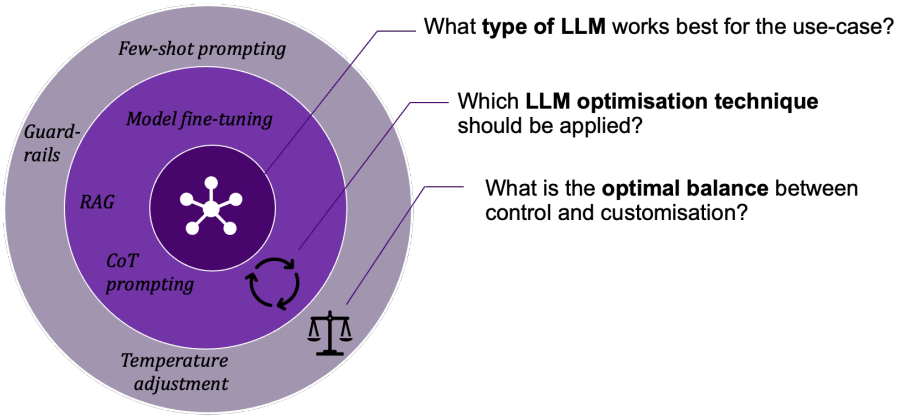

Figure 3: Key design choices of building AI systems. They consist of the three layers of the type of LLM used, LLM optimisation techniques applied and the optimal balance between control and customisation.

The innermost layer deals with the choice of core LLM**,** which includes selecting the model family, size, and hosting strategy. Different model providers offer distinct strengths: some models excel in reasoning, others in long-context processing, while open-source models often provide greater transparency and flexibility for in-house adaptation. Model size also introduces trade-offs, larger models tend to demonstrate stronger generalisation and problem-solving capabilities, whereas smaller models may offer advantages in latency, cost efficiency, and ease of deployment. There is not a one solution fits all since there is a stark inverse relationship between model scale and output determinism and a "fit-for-purpose" deployment strategy is essential. Hereby, for mission-critical quantitative applications with mandatory audit trails (e.g. automated credit assessment or risk analysis), smaller models are preferred (Khatchadourian and Franco, 2025).

For quantitative investment research, where tasks may include analysing corporate guidance, extracting event signals from text, or interpreting multi-document narratives, the chosen model must perform well on factual grounding, numerical consistency, and extended-context retrieval. Selecting an appropriate core model therefore forms the foundation for any downstream system design.

The middle layer involves deciding which optimisation techniques to apply in order to adapt the base model to the firm's analytical environment. Options range from non-gradient methods such as lightweight approaches including prompt engineering and chain-of-thought prompting, to more structured methods like RAG and finally to deeper techniques such as fine-tuning that are included in gradient methods (Sharky and Treleaven, 2025). Each approach offers different benefits: prompt-based methods allow quick adaptation but can be unstable; RAG improves factual accuracy by grounding outputs in verified documents such as filings or transcripts; and fine-tuning introduces firm-specific language, domain conventions, or proprietary datasets directly into the model. For quantitative research teams, this middle layer is critical, as the optimisation choices determine whether LLM outputs can be reliably incorporated into hypothesis generation, signal analysis, or textual factor construction.

The outermost layer addresses the balance between control and user customisation, a design dimension that is particularly important in institutional research environments. On the one hand, analysts require flexibility to explore ideas, interrogate datasets, and iterate on research questions. On the other hand, organisations must impose guardrails to ensure output reliability, regulatory compliance, and protection of proprietary information. Effective systems therefore incorporate mechanisms such as controlled prompt templates, validation layers to detect hallucinations or unsupported claims, detailed logging for auditability, and access restrictions aligned with data-governance policies. The challenge is to provide sufficient freedom for researchers to make productive use of the system, while maintaining a level of oversight that supports reproducibility and institutional accountability.

In summary, the design of GenAI systems for quantitative investment research rests on a coherent alignment between the choice of core model, the degree and method of optimisation, and the guardrails that shape user interaction. These three layers jointly determine whether an LLM system becomes a robust, scalable component of the research process or introduces risks that undermine its analytical value. Importantly, in an organisational setting these design decisions are rarely made in isolation; they typically require close collaboration between technology teams, quantitative researchers, and other stakeholders to ensure that the resulting system fits both the firm's analytical objectives and its operational and governance constraints.

2. "LLM Assistant": Research Assistant for information retrieval and processing, and implementation

LLMs are increasingly serving as research assistants for quantitative finance professionals, automating and accelerating tasks that are traditionally time-consuming, such as literature review, exploratory coding, documentation, and client communication. Their value lies in facilitating faster knowledge acquisition, supporting complex analytical workflows, and maintaining domain-specific contextual reasoning. This development reflects a broader shift toward augmented intelligence, a concept with origins in Ashby's (1956) notion of "intelligence amplification," which emphasised the enhancement rather than replacement of human cognitive capabilities. Under this paradigm, AI systems improve the quality, speed, and consistency of human decision-making instead of functioning as autonomous substitutes for expert judgement. This idea has been extended to financial analysis through the framework of AFI**,** in which AI models and tools are positioned as complements that enhance the analytical processes of financial professionals and quantitative processes rather than displace them (Mihov, Firoozye and Treleaven, 2024). In practice, LLM-based systems support three main areas of the quantitative research workflow: first, information processing and knowledge discovery, including rapid familiarization with new topics and continuous monitoring of relevant publications; second, coding assistance, such as code generation, debugging, and documentation; and third, optimization of operational and communication-related non-research tasks. Together these functions increase researcher productivity and allow domain experts to focus on conceptual development and methodologically robust empirical research processes.

2.1. Information Processing and Knowledge Discovery

LLMs have become particularly influential in quantitative investment research due to their capacity to digest extensive textual material and distil it into analytically useful insights. Quantitative researchers often need to familiarize themselves quickly with new research domains, whether novel asset classes or investment regions, recently developed econometric methods, or advances in financial machine learning. This process of research

coverage initiation within investment settings, has traditionally required labour- and time-intensive literature review, reading of academic studies, and integration of findings across heterogeneous and large array of sources. LLMs now enable substantial automation of these tasks by conducting semantic summarization, extracting core concepts, and applying structured reasoning to large collections of unstructured text. As a result, researchers can replace days of manual reading with targeted queries to a LLM-based assistant capable of identifying principal methodologies, underlying assumptions, and relevant empirical evidence, thereby materially shortening the initial learning curve for new research topics.

At a technical level, the advantage arises from the semantic understanding and contextual reasoning enabled by Transformer-based models, in contrast to traditional keyword-driven retrieval, which has motivated hybrid approaches such as those first explored in Dense Passage Retrieval (Karpukhin et al., 2020) and later expanded in hybrid RAG systems (e.g., Arivazhagan et al., 2023; Sawarkar et al., 2024). When paired with RAG architectures, now commonly combining semantic and keyword retrieval for higher recall, these systems maintain factual grounding by restricting outputs to verified source materials. Put differently, while off-the-shelf LLMs function like probabilistic knowledge bases that "remember" information in their parameters, integrating them with RAG enables a more deliberate search process, allowing the model to shift attention toward relevant external documents rather than relying solely on internal memory. In addition, because LLMs are trained on large collections of human-written text, they can approximate an understanding of how concepts connect, which supports more coherent interpretations of the retrieved material.

The ecosystem around RAG has expanded quickly, producing a wide range of models, frameworks, and infrastructure tailored to different retrieval and generation needs. Because RAG systems rely on the interaction between retrieval quality, embeddings, vector databases, and generative reasoning, quantitative researchers increasingly benefit from clarity on how these components fit together. Current offerings range from LLMs with built-in grounding features to enterprise platforms that provide managed retrieval pipelines, alongside mature embedding models, hybrid retrievers, and vector databases for high-recall search across financial documents. For an overview of representative tools and system components, see Appendix A**.** Collectively, these developments offer practical pathways for automating literature review and building research assistants capable of producing document-grounded insights.

Using AI Tools for Academic Research and Literature Review

Recent advances in AI have introduced a diverse ecosystem of research tools designed to streamline literature discovery, synthesis, and critical analysis. Many of these systems incorporate RAG workflows, enabling researchers to upload or query collections of papers and receive structured, citation-linked outputs that maintain traceability to original sources. This represents a significant shift from earlier generative models by prioritising grounded synthesis over unsupported generation.

A growing set of specialised platforms now support different stages of the research process. Tools such as SciSummary (SciSummary, 2025) automate the extraction of key points and references, allowing a rapid assessment of large numbers of papers. Others, including SciSpace (SciSpace, 2025) and Anara (Anara, 2025), offer more comprehensive research workflows by integrating database search, cross-document synthesis, and reference organisation; Anara additionally incorporates multi-agent support for tasks such as systematic review and manuscript preparation. Sourcely (Sourcely, 2025) uses semantic search to retrieve conceptually relevant papers and supports reverse literature review, while Liner (Liner, 2025) applies task-specific AI agents to identify keywords, evaluate relevance and draft structured review outlines. Visual discovery tools such as Litmaps (Litmaps, 2025) help researchers understand the structure of a field by mapping citation networks and highlighting influential studies. Consensus AI (Consensus AI, 2025) is another specialised tool that organises literature around yes/no questions, visualises levels of empirical support and produces short, structured reviews, offering rapid early-stage analysis but still requiring manual verification due to limited linking and occasional irreproducibility.

Elicit (Elicit, 2025) exemplifies widely adopted assistants for paper discovery and systematic extraction. It combines semantic search with automated screening and table-based evidence extraction, improving

transparency and reproducibility. Nevertheless, its reliance on a single database and its focus on structured extraction make it less suitable for projects that require multi-source synthesis or full workflow automation, where multi-agent platforms such as Anara provide broader coverage (Anara, 2025). Similarly, Google's NotebookLM supports literature synthesis by allowing users to upload corpora of documents and generate mind-maps, timelines, and cross-document analyses. These are useful for identifying themes and research gaps. In addition, NotebookLM can be used as a learning tool with a Q&A interface as well as podcast generation capabilities. ScholarGPT demonstrates the potential of domain-specialised language models, outperforming general-purpose models in discipline-specific tasks and illustrating the benefits of training on large, curated academic corpora. Balel (2025) found that ScholarGPT produced more consistent and higher-quality responses compared to ChatGPT specifically for oral and maxillofacial surgery research.

General deep-research tools, such as Perplexity's (Perplexity, 2025), ChatGPT's (OpenAI, 2025) and Gemini's (Google, 2025) deep research models, are built to autonomously run iterative queries, aggregate information from large numbers of sources and produce extended reports that integrate search, reasoning and synthesis across platforms. While these systems provide breadth and speed, they often rely on incomplete retrieval pipelines and therefore exhibit limitations in accuracy and reproducibility. Specialist tools typically outperform such general systems because they draw on curated academic databases and domain-specific workflows, reducing common problems such as source gaps, inconsistent retrieval, and hallucinations.

For practitioners in quantitative finance, the implications are substantial. Analysts and portfolio researchers must routinely integrate insights from macroeconomics, statistics, machine learning, and behavioural finance to design new Factor models and trading strategies. LLM-based systems support this process by summarising the historical foundations of a research topic, reviewing recent working papers, identifying methodological developments, and generating conceptual maps that connect related strands of the literature. In a typical new-knowledge-acquisition workflow, a researcher examining a topic such as regime detection with machine learning can use LLM tools to extract theoretical and historical context, tracing, for example, the progression from early Markov-switching models to recent deep-learning architectures. The resulting synthesis functions much like a coverage-initiation note, but focused on a research theme rather than a specific firm or sector, helping teams keep pace with the rapidly expanding volume of academic publications and industry white papers that has become difficult to manage through manual review alone.

Beyond initial familiarisation with new topics, LLMs also play an important role in deepening expertise and maintaining domain awareness once a research area is established. Modern research workflows require continuous tracking of new arXiv submissions, SSRN working papers, and updates in econometrics and machine-learning journals, at volumes that exceed what individual researchers or teams can monitor manually. LLM-driven monitoring systems, typically built on RAG pipelines using frameworks such as LangChain and vector databases, can scan new publications, filter them for relevance according to team-specific criteria, and extract key methodological and empirical developments. An established research team can thus use the same tools to track advances in its primary area, receiving concise summaries of newly released studies and highlights of methodological innovations. Systems such as Google NotebookLM exemplify this capability: given an uploaded corpus of research papers, they can generate structured literature overviews with citations and distil the main experimental methods and results into a format that supports ongoing, rather than one-off, knowledge maintenance.

Across these systems, a consistent challenge remains. AI tools can struggle to judge source quality, may reinforce assumptions implicit in user prompts and typically process only a subset of available literature due to computational constraints. Moreover, hallucinations, plausible but incorrect statements produced with unwarranted confidence, albeit improving, still remain a recognised limitations of LLMs (e.g. Huang et al., 2025). For these reasons, it is important to emphasise that AI should not replace expert reading or critical reasoning. Instead, these tools are best viewed as accelerators that help researchers navigate large bodies of information more efficiently, while final interpretation, evaluation and learning remain firmly human responsibilities.

LLM-based research assistants signal a shift from manual literature review and knowledge curation toward a system of continuous, AI-supported knowledge discovery. Rather than substituting for domain expertise, LLMs operate as extensions of human analytical capacity, allowing quantitative researchers to manage complexity and information availability more effectively and maintain wider awareness of methodological and market developments. In this way, LLMs function not only as tools for summarization but as contributors to knowledge synthesis, linking large information sets with the focused analytical reasoning required in quantitative investment research.

2.2. Coding Assistance Landscape

LLMs increasingly support the coding and engineering tasks that underpin quantitative investment research, where productivity, consistency, and reproducibility are central requirements. Quantitative researchers routinely translate conceptual ideas into executable code, maintain large and evolving codebases, and generate diagnostic analyses. Each of these tasks can benefit from targeted automation. Contemporary models can translate pseudocode or mathematical descriptions into working implementations, refactor and document legacy libraries, and produce visualisations or diagnostic checks that facilitate exploratory analyses.

A related development is the use of natural-language interfaces for structured data through text-to-SQL tools and models. BIRD (BIg Bench for LaRge-scale Database-grounded Text-to-SQL Evaluation) has become a standard benchmark, with more than 12,500 question–SQL pairs across 95 databases totalling 33 GB of data (Li et al., 2023a). Its Single-Trained-Model Track evaluates a model's inherent SQL-generation capability by restricting the use of preprocessing, retrieval mechanisms, or agentic orchestration that might otherwise inflate performance. Strong results in this setting signal genuine improvements in a model's ability to map natural-language prompts to accurate executable SQL.

Recent systems illustrate how quickly this area is advancing. Google Cloud's Gemini-SQL (Multitask SFT + Gemini-2.5-Pro) currently leads the Single-Trained-Model Track based on a design emphasising data filtering, multitask learning, and test-time scaling (Google Cloud, 2025). Other providers have focused on reinforcement-learning-based methods. Databricks' RVLR (Ali et al., 2025) and Snowflake's Arctic-Text2SQL-R1 are specialised reasoning models tailored to their respective SQL dialects. Arctic-Text2SQL-R1 employs a lightweight reward signal based solely on execution correctness to train a family of models that reliably generate executable SQL (Yao et al., 2024). These systems streamline database querying and reduce time spent on repetitive query construction, particularly valuable in environments with extensive historical or proprietary datasets where efficient data access is a persistent constraint.

Beyond database access, an expanding set of tools provides LLM-based coding assistance in routine development workflows. GitHub Copilot, OpenAI's code-oriented tools in ChatGPT (including the Python "Code Interpreter" environment), and Amazon Q Developer offer real-time code generation, debugging, and documentation support within mainstream IDEs and cloud development platforms. In practice, they serve as autocomplete-plus assistants that scaffold new functions, propose test cases, and surface relevant API examples directly in the development environment.

In notebook-centric workflows, tools such as Jupyter AI and Anaconda Assistant integrate LLMs into JupyterLab and cloud-hosted notebooks. Researchers can generate, explain, and debug code; analyse data; and synthesise entire notebooks from natural-language prompts while maintaining execution history and outputs in place. These extensions align with the broader Project Jupyter ecosystem, which is increasingly incorporating generative-AI support into interactive scientific computing. Across both IDE and notebook environments, AI-assisted development has been shown to increase programming speed, reduce syntax and logic errors, and improve developer satisfaction (Ziegler et al., 2022; Vaithilingam et al., 2022). For quantitative finance teams, such gains support faster prototyping of trading strategies and risk models, more consistent adherence to internal coding standards, and improved maintainability of research infrastructure.

A supporting trend is the rise of reasoning-focused model families, including OpenAI's o-series, DeepSeek-R1, Claude 3.5 Sonnet and Opus, and the Gemini 2.5 Pro series. These systems allocate more compute to multi-step reasoning and are designed for complex code-generation and debugging tasks, particularly in mathematically or data-intensive workflows. At the more autonomous end, agentic coding platforms such as Devin (Wu, 2024) frame software-engineering tasks end-to-end. Given a natural-language specification, they coordinate an entire toolchain, editor, shell, test suite, documentation search, and version control, to reproduce bugs, evaluate candidate fixes, and prepare pull requests with limited human intervention. For quantitative research teams, such systems are particularly relevant for debugging, managing multi-step data pipelines, and enforcing coding and model-validation standards across large, evolving research codebases. See Figure 4 for an overview of the AI coding landscape.

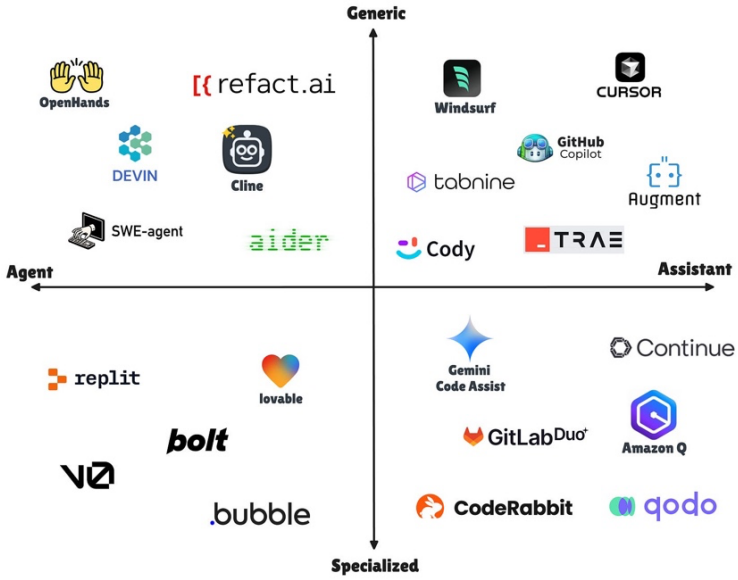

Figure 4: AI Coding Assistants Landscape ranges from assistant to autonomous and generic to specialised solutions, with a diverse set of focus tasks. (Ibryam, 2024)

These tools do not replace rigorous quantitative validation but can serve as an additional reviewing layer, automating early-stage error detection and highlighting potential methodological blind spots. Evidence from software engineering shows that LLMs reduce debugging time and identify logical inconsistencies (Vaithilingam et al., 2022), while empirical work suggests they can flag discrepancies when translating mathematical notation into executable code. For practitioners, LLMs therefore function not only as accelerators but also as testing assistants that enhance analytical reliability and reduce the likelihood of undetected logic errors in strategy pipelines.

LLMs also support reproducibility and governance. Quantitative teams must ensure that documentation is complete, coding styles are consistent across projects, and model variants are fully specified for audit and regulatory purposes. Models can generate documentation for existing codebases, summarise analytical workflows, and produce standardised templates for experiment logs and pipeline configurations. In this capacity, they strengthen internal validation, enhance regulatory transparency, and reduce operational risk, particularly for smaller or more dynamic teams with broad role distributions.

Finally, the choice of coding assistant in a corporate environment is shaped less by user experience or marginal accuracy differences than by infrastructure, vendor dependencies, and governance requirements. Tools must align with the firm's cloud stack, identity and access management, source-control workflows, and continuous integration and continuous deployment (CI/CD) pipelines, while accommodating licensing arrangements and inference costs at scale. Seamless integration with established development environments and security controls, such as single sign-on, data-loss-prevention policies, logging, and auditability, is essential, particularly in regulated settings. These operational and governance considerations often outweigh small differences in model

performance. Appendix B summarises selected tools and systems with notes on deployment and integration patterns relevant for corporate use.

2.3. Coding Assistance Applications for Quantitative Investment Research

LLMs are becoming more widely used as "vibe" coding assistants in quantitative research workflows, especially in settings with large legacy codebases, mixed programming languages, and complex data pipelines. Their impact spans several recurring tasks that typically require considerable engineering time but limited conceptual risk.

A first category involves non-core model coding tasks, such as generating diagnostic visualizations used for early-stage validation. LLMs can produce plots for autocorrelation, rolling exposures, distribution diagnostics, or regime annotations with libraries such as Matplotlib, Plotly, or Seaborn. These tools help researchers run quick, lightweight checks without distracting from conceptual development.

A second area of value concerns code refactoring of legacy code and generating documentation with limited human input (Jelodar, Meymani and Razavi-Far, 2025). Legacy quantitative libraries often built over many years and modified by numerous contributors, frequently exhibit inconsistent coding styles, sparse documentation, and duplicated routines. LLMs can rewrite code to follow established conventions such as PEP8 or Google's Java Style Guide, add docstrings and comments, introduce type hints, and modularize complex functions. This is particularly useful in teams with high turnover, low number of researchers or fragmented ownership of research infrastructure.

LLMs also improve access to structured data through text-to-SQL generation, which converts natural-language requests into optimized SQL statements. This is helpful when working with large proprietary datasets, historical market databases, or factor libraries stored in cloud data warehouses. The ability to generate and explain SQL queries lowers the barrier to exploratory analysis and reduces friction in multi-database workflows.

A fourth and increasingly significant application is legacy system modernization. Many quantitative research environments rely on infrastructure originally written in MATLAB, R, SAS, or early versions of Python and SQL. LLMs can help translate these systems into more modern, modular, and maintainable codebases. A representative case is the modernization of legacy backtesting engines that had been incrementally revised by multiple researchers, resulting in inconsistent logic, duplicated methods, undocumented assumptions, and performance bottlenecks. An LLM-assisted process can translate procedural code into modular Python, propose object-oriented abstractions, generate documentation, identify duplicated logic, and produce diagnostic outputs such as Alpha diagnostics, turnover profiles or residual distributions to confirm behavioural equivalence between old and new implementations. Text-to-SQL capabilities also support validation of data joins, timestamp alignment, and investment factor construction pipelines.

This pattern aligns with current practice in many institutions that have integrated tools such as GitHub Copilot, Google's Gemini Code Assist or Snowflake's Cortex Code into their workflows.

2.3.1. Risks, Controls and Governance Consideration for LLM-Assisted Coding

Important limitations remain. LLMs can generate code that is syntactically correct yet logically flawed, inefficient, or misaligned with domain-specific modelling assumptions. Performance varies across tasks and languages, and generated code typically requires careful human review (Jiang et al., 2025). In quantitative finance, such review is critical, since small implementation errors in backtesting systems, factor-construction routines, or risk-model components can materially distort results. Effective use therefore depends on structured workflows, appropriate oversight, and, where feasible, models that are fine-tuned or otherwise calibrated for financial programming tasks. Sustained human supervision remains essential.

Domain experts are central to this oversight. They are needed to identify conceptual misinterpretations, validate financial logic (including timestamp alignment, corporate-actions treatment, and the avoidance of look-ahead

bias), and confirm that new implementations reproduce the behaviour of established models. When combined with disciplined engineering practices, LLM tools can nonetheless shorten modernisation timelines and improve documentation quality, maintainability, and analytical reliability. This hybrid model, in which automated support is paired with expert verification, reflects an emerging pattern of AI-augmented research engineering in which LLMs accelerate technical execution while methodological standards remain anchored in human judgement.

Integrating LLMs into quantitative research pipelines also introduces new risk categories that require explicit control. Evaluations of code models highlight the possibility of logically incorrect or convention-breaking outputs, even when code compiles successfully (Chen et al., 2021). LLM-generated code exhibits distinctive bug patterns, such as Prompt-biased Code, Wrong Attribute, and Hallucinated Object, that differ substantially from human-written bugs and often arise from prompt ambiguity or the stochastic nature of model generation (Tambon et al., 2024). While many are easy to fix, others are difficult to diagnose.

Reliance on external API-based systems raises additional concerns about data governance, security, and intellectual property, especially when working with proprietary or regulated datasets. Moreover, LLM outputs are non-deterministic and sensitive to prompt formulation, which makes systematic versioning, prompt locking, and careful change management important for auditability and reproducibility.

To manage these risks, quantitative research teams should adopt an expanded model-governance framework tailored to LLM-assisted development. Key elements include mandatory human review of all LLM-generated or edited code; systematic use of unit, regression, and behaviour-equivalence tests; and rigorous versioning and logging of code changes, prompts, and LLM outputs, consistent with established software engineering and quantitative model development controls. In regulated environments, LLM-driven coding must align with existing model risk management standards, with clear documentation of how AI tools support researcher judgement.

Governance also depends on staff training and skills development. Even experienced quantitative researchers benefit from guidance on how to prompt LLMs effectively, evaluate AI-generated code, detect subtle logical inconsistencies, and escalate anomalies. Structured training programmes on prompt strategies, common LLM failure modes, secure data handling, and best practices for reviewing AI-generated code are important for consistent and responsible use. Evidence from software engineering suggests that outcomes improve when users treat LLMs as pair-programming partners rather than authoritative sources (Vaithilingam et al., 2022).

Strong engagement from the technology teams that support quantitative research is similarly necessary, as these groups maintain infrastructure, enforce development standards, and implement access controls. However, responsibility cannot rest solely with technology functions. Effective risk mitigation requires close coordination between technology and quant research teams, including joint training, shared review workflows, and collaborative design of guardrails. Technology teams provide secure environments, enforcement tooling, and architectural controls, while quantitative researchers validate outputs and identify domain-specific risks. Governance thus becomes a shared responsibility between engineering and research.

Organisations should also implement technical and procedural guardrails that constrain unsafe or inappropriate LLM behaviour. Examples include limiting LLM access to production environments, using retrieval-augmented architectures to enforce factual grounding, prohibiting execution of unvalidated code in production, and employing rule-based filters or sandboxed execution environments to limit potential harm.

Finally, effective governance requires mechanisms to ensure that human validation takes place, rather than relying on informal review. These include mandatory code-review checks in Git-based workflows, automated blockers that prevent merging LLM-generated changes without designated reviewer approval, dashboards that track the proportion of LLM-generated code and its review status, and audit trails documenting human-in-the-loop (HIL) decisions. Such processes enforce discipline, reduce incentives for AI-assisted shortcuts, and provide transparency for internal audit and regulators. These controls help ensure that LLMs enhance research-engineering productivity without compromising model integrity, operational robustness, or regulatory compliance, and they support a

culture in which both technology and quantitative research teams contribute to the safe and effective integration of LLM tools.

2.3.2. Best Practices for Using LLM-Based Coding Assistants in Quantitative Research

Table 1 contains best-practice guidelines for using coding assistants tailored for quantitative research teams. It follows principles of reproducible science, regulated model development, and disciplined engineering workflows.

A. Coding and Workflow Best Practices

- **Start from pseudocode or clear mathematical specification** to reduce hallucinations and enforce conceptual alignment.
- **Anchor LLM outputs with reference code** (e.g., NumPy/Numba patterns, vectorized operations) to avoid suboptimal solutions.
- **Actively challenge the efficiency and correctness** of generated code by requesting multiple candidate implementations and refining them, especially in languages or frameworks with relatively limited public training data.
- **Use retrieval-augmented setups** when the model must refer to internal research libraries or coding conventions.
- **Pair LLM-generated code with immediate unit tests**, ideally auto-generated by the LLM itself.
- **Do not grant LLMs direct production-deployment rights**; avoid deploying raw, unreviewed code. Treat all LLM suggestions as drafts that require validation, testing, and code review.

B. Reproducibility and Documentation

- **Document LLM-generated code automatically**, including prompt history and model version.
- **Embed docstring generation and type hints** in every LLM-assisted workflow.
- **Use experiment-tracking tools** (MLflow, Weights & Biases, internal equivalents) to ensure reproducible outputs.
- **Require version-controlled prompts** when teams collaborate using LLMs.

C. Model Governance and Risk Controls

- **Introduce mandatory human review** for all LLM-generated code touching live data, research engines, or production pipelines.
- **Block external LLMs from accessing proprietary data** unless within firm-approved secure environments.
- **Establish "AI development policies"** describing permissible use cases, data boundaries, and review standards.
- **Regularly benchmark LLM outputs,** performance declines may occur with model updates or drift.

D. Quant-Specific Safeguards

- **Check for financial logic errors** (look-ahead bias, survivorship bias, incorrect timestamps, double-counted returns).
- **Cross-check numerical outputs** using simple baseline models (e.g., rolling mean, CAPM regression).
- **Validate assumptions** such as frequency adjustments, volatility scaling, and transaction-cost modelling.
- **Require behavioural equivalence tests** when refactoring legacy backtesting or risk-model code.

E. Cultural and Workflow Adaptation

- **Train researchers in prompt engineering** for coding tasks, focusing on structure rather than creativity.
- **Encourage pair-programming setups** where the LLM is treated as a collaborator.
- **Reserve LLM automation for repetitive or boilerplate tasks**, maintaining human oversight for conceptual modelling.

Table 1: Best-Practice Guidelines for Safe and Effective LLM Use in Quant Research

2.3.3. Future Infrastructure of Quantitative Research Teams

Recent advances in domain-specific LLMs indicate a structural shift in how quantitative investment teams will build research infrastructure, manage research projects and collaborate with AI systems. LLMs have the potential to significantly reduce the time from idea generation to testing to integration in production processes. Across both open-source and private institutional efforts, the literature shows a movement toward LLM-powered research pipelines, domain-adaptive architectures, and HIL systems that require human judgment while scaling analytical capabilities.

RAG Foundations of Next-Generation Research Pipelines

A central element of this emerging infrastructure is the integration of LLMs with RAG. Hybrid retrieval systems, combining semantic embeddings with keyword search, provide the grounding required for reliable financial

research. These systems maintain factual integrity by constraining generation to verified documents, shifting LLMs from probabilistic knowledge bases toward deliberate search engines capable of targeted reasoning. As the ecosystem around RAG expands, quantitative teams increasingly rely on high-recall vector databases, hybrid retrievers, and managed retrieval pipelines that serve as the foundation of reliable AI systems.

Financial LLMs

A key development in financial AI is the emergence of domain-specialized language models designed to handle financial terminology, numerical reasoning, and context-specific data. Such models must cope with the distinctive challenges of financial datasets, high temporal sensitivity, rapid structural shifts, and low signal-to-noise ratios (Yang, Liu, and Wang, 2025). A persistent limitation of standard LLMs is their lack of real-time data integration, which remains essential in financial decision-making and motivates the need for hybrid, latency-aware (and smaller) LLM systems capable of delivering immediate analytical insight (Ali, Zafar, and Aysan, 2025).

In parallel, financial institutions and data providers are investing in private, large-scale models built on proprietary datasets. BloombergGPT, a 50-billion-parameter model trained on Bloomberg's FinPile corpus, demonstrates how firms with substantial data assets and strong technical infrastructure can produce high-performance LLMs for internal analytics, research, and client services (Wu et al., 2023). Smaller, lightweight models remain equally important, particularly for cost-efficient deployment. FinGPT exemplifies this approach by applying a data-centric, low-resource adaptation strategy to open-source LLMs, thereby broadening access to financial AI through open, flexible pipelines (Yang, Liu, and Wang 2025).

Recent research also highlights limitations of very small models, which often exhibit poor generalisation outside their instructed tasks (Yang, Tam, and Tang, 2023). This has encouraged the development of mid-sized open-source financial LLMs. InvestLM, for example, adapts the LLaMA architecture using a curated corpus of financial disclosures, analyst commentary, and earnings call transcripts to enhance investment-focused reasoning (Yang, Tam, and Tang 2023). Despite inherent data constraints, these specialised models illustrate that fine-tuned financial LLMs can meaningfully improve domain relevance, interpretability, and performance over general-purpose alternatives, especially when aligned with financial practitioners' analytical needs.

LLM Architectures for Different Workflows

Beyond domain specialization, LLM architectures are increasingly being applied to specific components of the quantitative research workflow, particularly as linguistic front-ends, analytical instruments, and Alpha-mining systems.

As linguistic front-ends, LLMs support large-scale text analytics by translating qualitative market narratives into alpha formulas and producing interpretable summaries of complex backtests (Yuan, Wang, and Guo, 2024). LLMs also enable natural-language accounting queries to be converted into SQL for corporate reconciliation through RAG-based systems, as demonstrated by Yadav et al. (2024), whose virtual assistant automates SQL generation and improves the efficiency of financial reconciliation workflows.

Evidence of analytical capability is shown in Chou et al. (2025), demonstrating that the reasoning model ChatGPT-4o performs reliably across standard financial tasks, including time-series modelling, risk and return decomposition, and ARMA–GARCH estimation, with accuracy comparable to established econometric software such as Stata. Reasoning models more generally are trained to handle multi-step problems by decomposing them into smaller parts, although expert oversight remains essential in volatile market environments.

LLMs can support robustness analysis by suggesting alternative specifications, diagnostic checks and additional tests (Bridgeford et al., 2025). These recommendations do not replace human oversight, but they can strengthen model design and help reduce blind spots in empirical workflows. In applied settings, LLMs can develop edge-case scenarios, propose diagnostics such as autocorrelation or stationarity tests, generate unit tests for common

backtesting failure points, identify data-leakage risks or logical mismatches, and recommend sensitivity analyses, for example, varying rebalancing frequencies, calibration windows, or transaction-cost assumptions.

Alpha Discovery and Idea Generation

LLMs are also being developed specifically for Alpha discovery. Alpha-GPT proposes an interactive alpha-mining framework that translates natural-language researcher inputs into symbolic alpha expressions, reducing the manual workload traditionally associated with signal engineering (Wang et al., 2025). Alpha-GPT 2.0 extends this paradigm by introducing a multi-agent architecture that supports Alpha generation, modelling, and diagnostic analysis, enabling a continuous human–AI research cycle (Yuan, Wang, and Guo, 2025). At the practitioner level, conversational systems such as GPTQuant (Yue and Au, 2023) further lower technical barriers by converting natural-language instructions into executable Python code for backtesting and strategy development.

Looking forward, the integration of LLMs into idea generation and Alpha sourcing is beginning to reshape research practices in both academia and the industry. Recent research explores how LLMs can support literature-based discovery by surfacing cross-disciplinary relationships that may not be immediately evident. Similar opportunities arise in finance, where these models can highlight connections across subfields, for example, linking volatility modelling to text-based sentiment measures or relating reinforcement-learning approaches to portfolio optimisation.

Quantitative teams often comprise researchers from diverse academic backgrounds, particularly from physics and mathematics, whose concepts such as Brownian motion and stochastic processes underpin many foundational models in asset pricing and derivatives pricing. Empirical evidence shows that such cognitively diverse groups tend to outperform homogeneous teams on complex analytical tasks (Hong and Page, 2004). LLMs complement this diversity by helping researchers identify related methods, uncover overlapping datasets, and explore alternative empirical strategies, thereby expanding the range of feasible research directions. By mapping conceptual relationships across domains, LLMs can accelerate hypothesis formation and support the development of innovative methodologies in quantitative finance, effectively augmenting the idea-generation processes of modern research teams. The next generation of AI-driven Alpha-discovery systems are expected to draw increasingly novel insights from adjacent disciplines, enabling more out-of-the-box approaches to strategy design.

Despite advances in automation, human judgment remains structurally important. Empirical analyses of ChatGPT-4o show that LLMs cannot make reliable real-time decisions under uncertainty, which underscores the continued need for domain expertise and interpretive oversight (Chou et al., 2025). This requirement is reflected in the design of next-generation systems such as Alpha-GPT 2.0, where human input shapes each stage of strategy development and refinement (Yuan, Wang, and Guo 2025). Related architectures, including FINMEM, integrate human-aligned risk-profiling modules (conservative, balanced, and aggressive) to ensure that agent behaviour remains consistent with institutional risk mandates (Yu et al., 2023). Furthermore, effective deployment depends on grounding these systems in the specific research philosophies and methodological principles that guide quantitative teams. Investment processes are anchored in well-defined theoretical frameworks and empirical standards, and LLM engines must be customized to reflect these foundations. Such alignment is essential for ensuring that model outputs are not only technically sound but also compatible with the practitioner's analytical objectives and decision-making environment.

The next stage of infrastructure development centres on layered memory systems (which separate short-, medium-, and long-term information), self-improving reasoning loops (where models refine their output through feedback), and multi-agent simulations (in which several AI agents interact). FINMEM introduces a biologically inspired hierarchy of shallow, intermediate, and deep memory that enables LLM agents to reason across multiple temporal horizons, from daily news to annual filings, while remaining interpretable for analysts (Yu et al., 2023). QuantAgent incorporates a two-loop self-improvement framework. The inner loop refines domain knowledge, and the outer loop evaluates performance against real environments, which reduces the need for manually curated Alpha libraries (Wang et al., 2024). StockAgent extends this line of work by using LLM-based multi-agent simulations to study how investor behavior responds to macroeconomic conditions, policy adjustments, and social

sentiment, broadening the application of LLMs to scenario analysis and research on market dynamics (Zhang et al., 2024).

Team-specific Personalisation and Customisation

These systems can be calibrated to align with a team's research agenda, methodological preferences, and historical outputs, thereby supporting a more personalized form of research intelligence. Rather than producing broad summaries, such tuning enables targeted retrieval of developments most pertinent to a given research context. For example, recent advances in reinforcement learning relevant to a trading desk or emerging work on LLM-based sentiment modelling within an alternative-data research group. This personalization reduces cognitive load and helps ensure that researchers spend more time on analysis and interpretation and less time searching for relevant material.

In summary, these developments suggest an emerging research environment in which LLMs operate as high-bandwidth cognitive interfaces. They have the potential to translate researcher intent into formal quantitative procedures, integrate large and heterogeneous information streams, and update internal representations without continuous manual intervention. Human involvement, however, remains central. Strategic oversight, contextual interpretation, and risk governance cannot be delegated, and current systems are explicitly designed around these requirements.

Quantitative teams are likely to become leaner, more interdisciplinary, and more cognitively leveraged, working within HIL, agent-based LLM infrastructures that extend analytical capacity while preserving institutional control. Within such workflows, LLMs can support end-to-end processes: they can read raw data, recommend suitable statistical methods, and write or revise code. These capabilities lower the operational cost of incorporating text-based alternative datasets into production-oriented quantitative strategies, as well as more broadly applications in Alpha-model development.

2.4. Optimization of Operational and Non-research Related Tasks

Beyond core quantitative modelling, a substantial share of a corporate quantitative researcher's time is spent on operational activities such as writing reports, preparing presentations, responding to client requests, and producing internal documentation. Although essential for communication, governance, and client servicing, these tasks often consume considerable time relative to their direct analytical contribution. LLMs have become effective tools for automating or accelerating such workflows, allowing quantitative professionals to devote more effort to conceptual research and model development. This shift aligns with findings in organisational science that GenAI can reduce cognitive load and administrative burden in knowledge-intensive roles (e.g. Davenport and Mittal, 2023; Brynjolfsson, Li and Raymond, 2023).

Three core use cases illustrate these effects in quantitative finance:

- **Report generation and templating:** Quantitative researchers producing recurring deliverables, such as monthly factor summaries, performance-attribution notes, or commentaries, can use LLM tools to convert structured analytics into coherent narratives. By connecting the model to internal SQL backends or analytics engines, LLMs can summarise factor returns, volatility diagnostics, or risk decompositions and generate text suitable for LaTeX, Word, or Markdown templates. This workflow parallels developments in data journalism and clinical reporting, where LLM-driven summarisation systems produce templated, data-grounded narratives that domain experts subsequently review for nuance and contextual accuracy. In quantitative finance, embedding such systems in reporting pipelines decreases manual effort and increases consistency in communication on research findings and performance updates.
- **Internal communication and documentation:** LLMs support the drafting of white papers, investment-process updates, research memos, and knowledge-sharing notes. They help harmonise writing styles across contributors, translate technical concepts into formats appropriate for portfolio managers or client-facing teams, and standardise documentation used for governance or compliance. Integrated productivity

tools such as Microsoft 365 Copilot and Google Workspace assistants extend these capabilities to automated slide outlines, meeting-note summaries, and draft presentations based on research text. These patterns are similar to emerging practices in professional services and legal domains, where GenAI has been shown to improve document-drafting efficiency while preserving expert oversight.

- **Client-facing material and tailored communication:** LLMs can assist in producing customised client responses, performance commentaries, and high-level explanations of model behaviour, grounding outputs in underlying data through retrieval-augmented architectures that query internal performance databases or research repositories. These capabilities mirror customer-service applications in other sectors, where LLMs enhance consistency and speed while maintaining personalisation and factual accuracy. In an asset-management context, they enable more timely and targeted communication without requiring each response to be drafted from scratch.

Taken together, these use cases reflect a broader pattern: LLMs streamline operational and communication tasks that, although essential for client service and internal governance, fall outside core quantitative modelling. They reduce administrative overhead, improve the clarity, consistency, and timeliness of research communication, and free capacity for tasks that require human reasoning and judgement. As in adjacent fields such as consulting, medicine, and law, LLM-assisted drafting increases efficiency but still requires human oversight to ensure accuracy, contextual appropriateness, and adherence to institutional communication standards and governance rules; LLM outputs must therefore be treated as drafts and subject to thorough validation.

3. "LLM Quant": Analysis of Text-based Alternative Datasets

The application of LLMs to text-based alternative datasets represents a structural shift in quantitative investment research. Earlier approaches relied on dictionary-based methods and modular NLP pipelines, whereas current practice is moving towards context-aware embeddings and end-to-end LLM architectures that map unstructured text directly into economically meaningful signals. The following section, referred to as "LLM Quant", examines how LLMs are used as core models for extracting information from news, social media, corporate disclosures, and other text-based alternative data, and how this transition from modular to integrated pipelines affects both modelling practice and empirical performance. Equally important are the methodological limitations and risks that currently constrain large-scale deployment of LLM-driven text analysis in production investment settings.

3.1. From Dictionaries to Context-aware Language Models

The evolution of text modelling in quantitative finance follows advances in NLP, progressing through three main methodological stages. The first relied on sparse lexical representations, including bag-of-words, TF-IDF (Salton and Buckley, 1988), and domain-specific lexicons such as those of Loughran and McDonald (2011) and Tetlock (2007). These models treat documents as unordered token counts and cannot encode syntactic structure or contextual semantics, making them brittle in settings involving negation, polysemy, narrative tone, or discourse-level dependencies.

The second stage emerged with contextual embedding models based on the Transformer architecture. Vaswani et al. (2017) introduced self-attention, enabling models to capture token interactions across entire sequences. Encoder-only models such as BERT (Devlin et al., 2019) and RoBERTa (Liu et al., 2019), along with financial variants such as FinBERT (Araci 2019; Huang, Wang and Yang, 2023), produce dense contextual embeddings where token meaning depends on its linguistic environment. The Araci (2019) FinBERT adapts BERT for financial sentiment classification, whereas the Huang, Wang, and Yang (2023) FinBERT model is a domain-pretrained financial language model intended for a broad range of downstream tasks such as information extraction from SEC filings and earnings reports. These contextual embeddings substantially improve robustness and performance in financial sentiment classification, event extraction, and risk analysis relative to sparse methods.

A related development is the growing use of word- and document-level embeddings as quantitative features. LLM-generated embeddings provide numerical vector representations of words, sentences, or documents that capture semantic proximity and contextual similarity. Recent empirical research demonstrates that these embeddings

often outperform traditional NLP representations in accounting and finance. Breitung and Müller (2023), for instance, use LLM-based embeddings to construct global business networks and show that they more accurately reflect economic relationships than standard industry classifications. Kirtac and Germano (2024) show that Transformer-based models, such as OPT (Open Pre-trained Transformers), significantly outperform other models, such as BERT, FinBERT and the traditional Loughran-McDonald dictionary, in sentiment analysis aimed at predicting stock return. Bandara, Flannery and Chandak (2023) find that OpenAI's "ada-002" model is highly effective for financial document classification, while BERT-based embeddings perform best in predicting earnings surprises from earnings call transcripts. These studies highlight that embedding-based representations, particularly when derived from large, pretrained models, have become critical inputs for quantitative models and provide richer signals than earlier lexical approaches.

The current stage is driven by generative autoregressive LLMs, which use decoder-based Transformer architectures trained on massive corpora. Early foundational work includes GPT-2 (Radford et al., 2019) and GPT-3 (Brown et al., 2020), while subsequent research extends these ideas in models such as OPT (Zhang et al. 2022), LLaMA (Touvron et al., 2023), and Mistral (Jiang et al., 2023). These models capture long-range dependencies, narrative structure, and pragmatic cues, enabling advanced capabilities such as instruction following, multi-step reasoning, cross-document synthesis, and structured extraction. For financial applications, this generative paradigm supports interpretation of multi-modal disclosures, reconstruction of event narratives, and hypothesis generation, capabilities not possible with earlier embedding-only architectures. This three-stage progression, from sparse lexical features to contextual encoders to generative LLMs, is consistent with contemporary NLP research and reflects a structural shift in how quantitative researchers are starting to extract information from text-based alternative datasets.

3.2 From Traditional Modular NLP Pipelines to End-to-end LLM-based Pipelines

A second conceptual shift concerns how textual signals are constructed. Prior to the emergence of LLMs, financial text analysis relied on modular and interpretable NLP pipelines, shaped both by computational constraints and the institutional need for transparency and auditability. A typical workflow involved:

1. **Named Entity Recognition (NER):** to identify referenced firms or economic entities;
2. **Sentiment Classification**: often based on domain-specific lexicons (e.g., Loughran and McDonald 2011) or later Transformer-based classifiers such as FinBERT (Araci 2019); and
3. **Downstream Modelling**: including return prediction, event-study analysis, or factor construction.

Under this paradigm, text was processed headline-by-headline or sentence-by-sentence, producing scalar sentiment or relevance scores that were subsequently aggregated at the firm, event, or topic level (See Figure 5 for further information). Although effective for early applications, this approach treated sentences or documents as independent units, even when economic meaning hinged on cross-sentence relationships, narrative structure, or causal explanations. As a result, important contextual information was often lost because each stage of the pipeline, entity resolution, classification, aggregation, and modelling, was executed in isolation. This modularity is characteristic not only of early dictionary-based studies (e.g., Loughran and McDonald 2011; Jegadeesh and Wu 2013) but also of later task-specific Transformer architectures such as FinBERT, where each component remained a separate model within a multi-stage system.

Modern LLMs make such stage-wise decomposition both less necessary and, in some cases, counterproductive. Because LLMs encode entity relationships, temporal structure, event causality, and economic implications within a single, jointly trained representation, mapping text to a single scalar sentiment value risks discarding precisely the information that makes textual data valuable. Rather than using sentiment as a standalone proxy for a simple binary of news polarity (positive or negative), extracting direct, economically meaningful impact estimates is more beneficial.

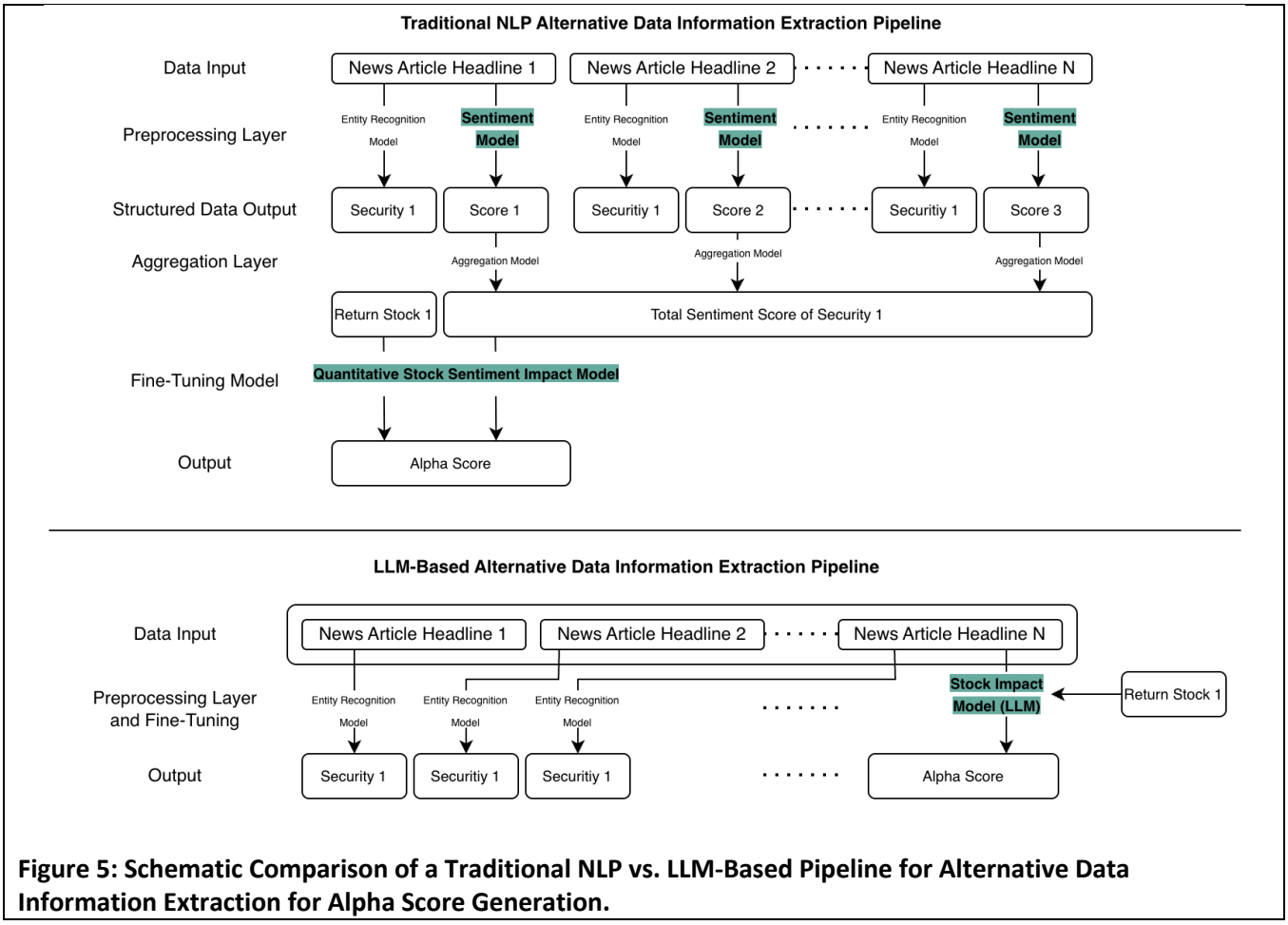

Figure 5: Schematic Comparison of a Traditional NLP vs. LLM-Based Pipeline for Alternative Data Information Extraction for Alpha Score Generation.

Lopez-Lira and Tang (2025) show that LLMs such as GPT-4, using carefully designed zero-shot conversational prompts, can accurately predict immediate market reactions to news, demonstrating emergent financial reasoning despite the absence of task-specific training. Chen, Kelly and Xiu (2022) show that the strengths of LLMs in the context of financial downstream tasks stem primarily from their superior ability to process complex linguistic information (e.g. handling complex narratives and negations) and translate that into profitable predictive signals. Kirtac and Germano (2024) extend this idea by capturing firm-specific risk assessments derived from contextual cues in disclosures. These approaches reflect an emerging end-to-end paradigm, in which LLMs act as context-aware financial interpreters: they map unstructured text directly into structured, multidimensional judgments that align with how human analysts understand firm news, competitive dynamics, and macroeconomic developments.

In this framework, the LLM becomes not a component within a pipeline but the core evaluator linking textual information to economically grounded outputs, mirroring the holistic reasoning used in fundamental analysis.

3.4 Empirical evidence for LLM-driven Quant Strategies

Kong et al. (2024) classify LLM applications in finance into five areas: linguistic tasks, sentiment analysis, financial time series, financial reasoning, and agent-based modelling. For quantitative researchers developing systematic strategies, sentiment analysis and financial time series forecasting are the most relevant. A growing empirical literature shows that LLM-derived signals can be incorporated into trading strategies, risk models, and macroeconomic forecasts. In sentiment analysis, LLMs such as GPT-4 and FinBERT perform well in interpreting domain-specific language, subtle narrative cues, and long documents across news, social media, corporate disclosures, and policy communication. For instance, Lopez-Lira and Tang (2025) find that GPT-4 and ChatGPT-based headline sentiment predicts cross-sectional stock returns, while Su, Mulvey, and Poor (2022) use BERT-based Twitter sentiment to improve covariance estimation and mean–variance portfolios during the COVID-19 period. LLMs have also been applied to a variety of other text-based sources e.g. earnings calls, regulatory filings

and central bank communication. Ernstberger and Nazemi (2025) used a dual-method approach leveraging a general-purpose LLM (o4-mini) with zero-shot prompts and a fine-tuned BERT emotion detection model to extract qualitative and emotional insights from earnings call transcripts and consequently enhance the prediction and understanding of defaulted corporate bond recovery rates and trading dynamics. Christiano Silva, Moriya and Veyrune (2025) introduce an automated classification framework that uses a fine-tuned LLM to systematically analyse central bank communication at the sentence level across four dimensions: topic, communication stance, audience, and sentiment.

In financial time series analysis, LLMs are applied both as text-derived feature extractors and, more recently, as Transformer-based sequence models that operate directly on numerical data. The motivation for adapting these architectures to structured and numerical series is partly informed by Radford et al. (2019), who show that high-capacity language models trained on broad unlabelled corpora can conduct diverse NLP tasks in a zero-shot setting. This result highlights Transformers as flexible sequence models and motivates their extension beyond text. Empirical forecasting evidence is mixed but encouraging. Yu et al. (2023) integrate heterogeneous data and instruction-tuned LLMs to improve NASDAQ-100 return prediction, and multimodal approaches that combine text, prices, and images outperform statistical and deep-learning baselines in index forecasting. Other work, such as Chen et al. (2023), uses ChatGPT to infer dynamic inter-firm relations from news and incorporates these into graph neural networks that enhance predictive accuracy. At the same time, purely zero-shot or naïve applications to price sequences tend to underperform specialized time-series models, which suggests that LLMs add value primarily when contextual or multimodal information is included rather than as standalone numerical forecasters.

Beyond forecasting, Transformer-based architectures are being examined for anomaly detection, financial classification, and data augmentation. In addition, the latest research developments focus on agentic workflows (e.g. Yu et al., 2023, Wang et al., 2024, Zhang et al., 2024). Multi-agent LLM frameworks are used in applications such as trading decision support and market-surveillance-style tasks, while classifier models assist with sector, ESG, or thematic categorization that can indirectly influence allocation decisions. Generative deep models, not limited to LLMs, are increasingly adopted to simulate order flow and price paths for execution research and reinforcement learning. Transformer-based methods are also explored for augmentation and, in some cases, imputation, although LLM-specific work on the latter remains limited.

Overall, the literature indicates that sentiment analysis is the most empirically established LLM application in systematic investing, with evidence across social media, news, corporate disclosures, and policy documents. Direct financial time series forecasting remains less settled, with the strongest results arising from multimodal and graph-augmented settings rather than from price-only prediction. For practitioners, the evidence suggests that LLMs are most effective as feature generators and interpreters of complex, high-dimensional information when integrated into established quantitative pipelines rather than used as end-to-end forecasting models.

3.3 Methodological Challenges and Model-selection Considerations

Despite their promise, LLMs introduce several methodological risks that require careful handling in quantitative finance. A central concern is temporal contamination, where models inadvertently incorporate future information because they are pretrained on corpora spanning long historical windows. Carlini et al. (2021) demonstrate that modern LLMs can memorize rare or unique sequences from their training data, and recent evidence confirms that this extends to economic and financial variables. Lopez-Lira, Tang, and Zhu (2025) show that LLMs can perfectly recall historical values of key macroeconomic and market indicators, including S&P 500 levels, unemployment rates, and GDP figures, up to their knowledge cut-off. As a result, evaluation inside the training window confounds genuine predictive skill with memorized information. Levy (2024) similarly finds that apparent predictive performance disappears once all post-event data are excluded, indicating that many positive backtests are driven by look-ahead bias.

Attempts to mitigate temporal contamination are only partially effective. Lopez-Lira, Tang, and Zhu (2025) report that explicit temporal instructions do not prevent recall-level accuracy, and masking techniques, such as anonymising firm names or removing dates, fail to block reconstruction. GPT-4o correctly identifies the firm in

100% of anonymised conference calls for the Magnificent Seven and infers the correct quarter and year with 92% accuracy for Apple transcripts, demonstrating that models can infer withheld identifiers from minimal contextual clues. Their perturbation tests further show that LLM forecasts collapse when memorisation is disrupted: randomising only the least significant digit of financial statement items reduces GPT-4's accuracy from 60% to ≈51%, statistically indistinguishable from chance. In contrast, gradient-boosted trees trained on tabular data remain stable under the same perturbation, confirming that LLM forecasting within the training window relies heavily on memorised quantities rather than generalisable numerical reasoning.

Beyond temporal leakage, LLMs exhibit intrinsic informational and cognitive biases tied to their training distributions. Lee et al. (2025) document systematic sector and size biases, with strong preferences toward technology firms and large-cap equities, reflecting popularity skews in online corpora. They also identify a momentum-related contrarian bias and a confirmation bias that becomes pronounced when external evidence conflicts with internally stored knowledge. Models often behave like "stubborn sloths," refusing to update views even when presented with substantial contradictory information, and showing elevated entropy (uncertainty) when forced into conflict situations. A further dimension is foreign bias: Cao, Wang, and Xiang (2025) find that US-based ChatGPT is systematically more optimistic about Chinese firms than China-based DeepSeek, overestimating prices by 11.6% on average. Crucially, this bias disappears once ChatGPT is provided with Chinese news articles, indicating that foreign bias arises from asymmetric information in the training corpus rather than genuine modelling insight.

A final methodological challenge relates to reproducibility and stability. Commercial LLMs have only become widely available in 2022–2023, which constrains backtests to relatively short out-of-sample periods encompassing few market regimes. For instance, Ko and Lee (2024) employ a 13-month out-of-sample window, chosen to respect the LLM training cut-off date, in order to evaluate the effectiveness of LLMs for asset selection in portfolio management. Because API-based models are black boxes that providers update regularly, often without version transparency, researchers cannot guarantee that a given model corresponds to a fixed knowledge cut-off. Together with nondeterminism inherent in probabilistic token sampling, floating-point variation, and post-training alignment steps, this makes reproducibility difficult.

Recent research highlights the importance of Transformer model selection and choice of LLM architecture for financial tasks. Local, fixed-weight open-source models (e.g., LLaMA variants) or domain-specific classifiers such as FinBERT and RoBERTa are recommended for tasks requiring stability and deterministic outputs, while open-source LLMs also allow customization and enhanced privacy (Li et al., 2023b). Chen, Kelly, and Xiu (2022) show that LLMs excel at processing complex linguistic structures, such as narratives and negations, translating them into predictive signals. Bidirectional BERT-type models (FinBERT, RoBERTa) perform best for structured sentiment classification, such as earnings transcripts, whereas autoregressive GPT-type models (GPT-4, LLaMA) are more effective for human-like language generation, enabling real-time sentiment interpretation, dynamic forecasting, and market narrative summarization (Chen, Kelly, and Xiu, 2022; Kirtac and Germano, 2025).

Taken together, these findings underscore that evaluating LLMs in quantitative finance requires rigorous out-of-sample testing beyond the model's knowledge cut-off, careful control for memorisation artefacts, and an awareness that intrinsic biases embedded in pretraining corpora can materially distort economic inference and investment decisions. The empirical evidence also highlights a broader methodological principle: model complexity is not a free good. When large LLMs introduce confounding risks, such as privacy leakage, predictive bias, distraction effects, or numerical fragility, simpler, domain-specific models often provide more reliable and interpretable solutions. In many tasks, especially those grounded in deterministic classification or structured tabular relationships, smaller architectures such as Gradient Descent Boosting-based Decision Tree (GDBT) or specialised Transformer encoders (e.g., FinBERT), or even anonymised linguistic models, outperform LLMs precisely because they eliminate the pathways through which these systematic errors arise.

These considerations reflect established principles in quantitative finance and robust machine learning. In high-noise, low signal-to-noise environments, the principle of parsimony, or Occam's Razor, suggests selecting the simplest model that adequately solves the task. As López de Prado (2018) emphasizes, models with fewer

parameters tend to generalize better out-of-sample, mitigating overfitting that can compromise complex models despite strong in-sample performance. This is consistent with the bias-variance trade-off, where reducing model variance improves predictive stability even at the cost of slightly increased bias. Simpler models also enhance interpretability and human control, which is critical in risk-sensitive applications such as asset management. These principles apply broadly across machine learning and AI, including generative models, highlighting the importance of balancing robustness, complexity, and generalization.

Large foundation models therefore warrant selective use: they are valuable for research, idea generation, cross-document synthesis, and producing training data, but they should not automatically replace smaller, deterministic architectures in production-grade Alpha pipelines. If a large LLM delivers only marginal performance gains on a task such as sentiment extraction or accounting classification, the simpler model remains the more robust and principled choice. Refer to Appendix C. for a checklist for evaluating LLM suitability in quantitative research tasks.

4. "LLM Quantamental": LLMs as Engines for AFI

AFI proposes that investment performance can be improved by incorporating human expert reasoning into quantitative models (Mihov, Firoozye, and Treleaven 2022). AFI treats the judgment of Superforecasters, analysts, and portfolio managers as a distinct, information-rich source that is qualitative, contextual, and forward-looking. Such insight complements the pattern-recognition strengths of purely data-driven models, helping to stabilise or correct algorithmic predictions when markets undergo structural changes.

Recent advances in LLMs significantly expand this vision. As Chin (2025) argues, LLMs are dissolving the long-standing divide between discretionary and systematic investing by giving discretionary investors unprecedented breadth and providing systematic investors with new qualitative depth (Chin, 2025). LLMs can read financial narratives, decompose analyst-style reasoning, and express economic logic in structured form, thus capturing fundamental judgments at scale. Conversely, they can translate model-driven signals back into natural language, enabling discretionary investors to engage more effectively with systematic insights. This two-way interpretative bridge positions LLMs as the technical foundation of a modern Quantamental investment process consistent with AFI principles.

AFI's objective of unifying fundamental and quantitative research rests on three pillars that are strengthened by LLM capabilities: capturing expert judgement, systemizing and scaling those insights and enabling two-way explainability. LLMs can scale the interpretative tasks of discretionary analysts, reading earnings calls, filings, regulatory disclosures, and macro commentary. Chin (2025) emphasises that LLMs act as research force-multipliers, allowing human analysts to apply their qualitative frameworks across a much broader set of companies and information sources than previously possible. Representing expert reasoning in structured, machine-readable form allows qualitative assessments to be integrated into systematic models. Chin describes how LLMs enable quant teams to "quantify the qualitative", extracting signals from text, audio, and multimodal sources that were historically inaccessible to traditional NLP or numeric factor models. LLMs can transform narrative inputs into quantitative features while converting model outputs back into intuitive explanations. This aligns with Chin's argument that AI will merge the toolkits of both discretionary and systematic teams, making systematic approaches more interpretable and discretionary processes more data-rich.

By fulfilling these functions, LLMs operate as the core engine of AFI: scaling human reasoning, enriching quantitative signals, and creating a unified framework where discretionary intuition and systematic discipline reinforce one another. In this sense, LLMs support the emergence of what Chin (2025) calls the „iron-person investor**"**, a human–machine hybrid that achieves more than either could alone**.**

4.1. LLMs as Scalable Interpretive Engines of Fundamental Insights

LLMs advance AFI by enabling, at scale, the interpretive tasks traditionally carried out by fundamental analysts, strategists, and macro researchers. Earlier NLP methods reduced text to sentiment indicators or bag-of-words features, whereas modern LLMs can capture nuance, context, narrative structure, and managerial tone in a way

that more closely resembles human analysis. Chin (2025) shows that this capability deepens systematic models by allowing quantitative researchers to incorporate qualitative inputs that were previously difficult to process, including earnings calls, news, social media, and multimodal signals such as vocal characteristics and presentation style. Complementary evidence from Livnat, Pozharny, and Suslava (2025) indicates that accounting-anchored NLP, built around conventions from the accounting profession, can yield economically grounded extraction of information from alternative datasets. LLMs extend and generalise these interpretive functions.

In practice, LLMs replicate several components of analyst-style reasoning, including:

- **Themes:** identifying economically relevant themes and causal mechanisms,
- **Guidance:** extracting forward-looking guidance and managerial intent,
- **Structure:** recognising industry structure and competitive dynamics,
- **Textual clues:** linking textual cues to valuation, profitability, or risk implications,
- **Highlights:** highlighting catalysts, risks, and potential inflection points, and
- **Shifts:** detecting shifts in market narratives across firms or sectors.

For systematic investors, these capabilities provide a degree of qualitative depth that was previously associated with discretionary research. For discretionary managers, LLMs offer substantial breadth by reading and summarising disclosures, detecting tone changes, and surfacing potential concerns across the full coverage universe, allowing discretionary approaches to scale to far more firms without losing analytical coherence.

An expanding empirical literature supports the view that LLM-generated outputs contain economical meaningful predictive information. Lopez-Lira and Tang (2025) show that ChatGPT's interpretations of news predict returns beyond traditional sentiment models. Pelster and Val (2024) report that ChatGPT-based attractiveness ratings forecast firm performance. Wang (2025) demonstrate improved detection of managerial obfuscation in narrative disclosures.

Taken together, this evidence suggests that LLMs approximate key interpretive functions of human analysts. They formalise and scale qualitative assessments that were previously embedded in fundamental expert judgment and make them available for integration into systematic modelling frameworks, advancing AFI's objective of linking discretionary insight with quantitative rigor.

4.2. Methodologies for Systemising Human Decision-making with LLMs

A central goal of AFI is to translate elements of human expert reasoning into systematic workflows that preserve domain insight while meeting the requirements of quantitative modelling. Several methodological advances underpin this transition. Most notably, chain of thought prompting (CoT), structured templates, RAG, HIL and agentic workflows. CoT prompts encourage structured, multi-step reasoning aligned with analyst workflows to extract events, their drivers and the resulting risk. This improves economic coherence and enables LLMs to approximate the "depth" of fundamental research in a form suitable for downstream modelling. Templates impose formal analytical structures such as thesis summaries, valuation frameworks, and bull/base/bear scenarios. As Chin (2025) notes, such standardisation allows LLM-generated narratives to be treated as model inputs, reducing variance in reasoning and enabling auditability. RAG enables grounding ensures that LLM outputs reflect verifiable facts drawn from filings, transcripts, or structured data. This mitigates hallucinations, an especially important concern in systematic settings where LLM errors can propagate through automated pipelines, and aligns LLM narratives with the empirical foundations of AFI. HIL is essential for every step and agentic workflows are the latest developments. LLMs do not replace human judgment; they structure and scale it. Chin (2025) emphasises that LLMs function as copilots rather than autopilots, augmenting but not replacing human decision-making. Agentic workflows illustrate how LLMs can support ideation, feature engineering, backtesting, and scenario analysis under expert oversight and in a semi-automated system.

Despite these advances, current LLMs still fall short of true expert reasoning. Chin (2025) notes risks such as hallucination, signal decay, and overreliance on model outputs. AFI therefore advocates a disciplined approach:

LLM outputs should be treated as hypotheses or structured signals requiring validation, not as autonomous decision engines.

4.3. Creating New Alternative Datasets of Human Expertise

A foundational objective of AFI is the construction of expert-knowledge datasets that capture the reasoning, conviction, and forward-looking insight of highly skilled financial professionals. Unlike traditional market or fundamental data, expert judgment is not available in structured or quantitative form. It is dispersed across unstructured materials such as analyst reports, portfolio-manager commentaries, corporate discussions, and macroeconomic outlooks. As a result, expert knowledge has traditionally remained tacit, idiosyncratic, and difficult to incorporate into systematic models. Large language models provide a solution by converting these qualitative sources into structured, machine-readable variables, effectively creating a new category of alternative data rooted in professional expertise rather than market transactions or crowd activity.

Recent advances in financial NLP show that professional investment texts contain substantial forward-looking information. Wang (2025) shows that LLMs successfully quantify managerial obfuscation across multiple, complex semantic dimensions in corporate reports, demonstrating that these measures capture meaningful variations in disclosure quality relevant to financial risk that were previously obscured by focusing only on text complexity. Kirtac and Germano (2024) find that LLM-based sentiment extracted from high-quality financial journalism provides stronger predictive signals than lexicon-based alternatives. Taken together, these studies confirm that professional discourse embeds structure, expertise, and implicit reasoning that LLMs can extract with high fidelity.

AFI extends these insights by emphasizing that the objective is to create a Superforecaster dataset of selected forecasts and not to crowdsource consensus signals from retail forums or social media, which often reflect noise, attention cycles, or behavioural biases. Within AFI, GenAI models can add value in different areas. One is the formalization of analytical reasoning, which involves systematizing key elements of forecasters' thought processes, analogous to the Superforecaster framework of Tetlock and Gardner (2016). Another one is the identification of highly skilled forecasters and analysts, which can be supported by constructing structured datasets from text-based narratives, enabling systematic recognition of individuals with expert reasoning. For instance, traditional analyst outputs, such as target prices, provide point estimates but often omit indicators of conviction, which are critical for distinguishing high-confidence insights from low-certainty views. Direct sourcing of conviction scores is challenging without extensive calibration. GenAI, particularly LLMs, can infer conviction from linguistic cues, argument depth, evidentiary quality, and logical coherence, producing reproducible and scalable measures that enhance both the formalization of reasoning and the predictive value of analyst-based signals.

By transforming expert-generated text into structured variables, conviction scores, thesis components, risk assessments, thematic classifications, valuation logic, and macro scenarios, LLMs create a new class of human-insight factors. These can be integrated with traditional quantitative models, econometric forecasts, and machine-learning pipelines, enabling AFI to incorporate the underlying, forward-looking, and interpretative dimensions of human expertise in a systematic and scalable manner. In this way, LLMs convert expert judgment from a narrative form into a formal alternative dataset, fulfilling one of AFI's central aims: embedding high-quality human reasoning directly within quantitative investment models.

4.4. Making Systematic Models Explainable to Fundamental Portfolio Managers

A central requirement for integrating fundamental and quantitative investment processes is that both teams operate with a shared interpretative language. Human decision-makers, including fundamental portfolio managers, typically reason using narratives, causal stories, and qualitative mental models (Kahneman, 2011; Klein, 1998). In contrast, systematic models generate high-dimensional numerical outputs such as factor loadings, cross-sectional z-scores, and optimisation weights, which are often unintuitive and easily misinterpreted when presented without context. For example, a cross-sectionally standardised signal may be mistaken for an absolute valuation metric, and a short-horizon statistical anomaly may be read as a structural conviction.

LLMs can help bridge the cognitive gap between quantitative models and fundamental analysis by translating model outputs into natural-language rationales that align with investors' interpretative frameworks. This approach is consistent with research in explainable AI that emphasizes concept-based explanations for complex models (Gilpin et al., 2018), as well as work on foundation models showing that LLMs can express intricate statistical relationships in human-interpretable language (Bommasani et al., 2022). LLMs can also generate self-explanations, such as chain-of-thought reasoning, which make intermediate decision steps more transparent. However, while these explanations are often coherent and persuasive, they may be unfaithful to the model's underlying reasoning, raising reliability concerns in high-stakes domains such as finance and healthcare (Agarwal, Tanneru, and Lakkaraju, 2024). Research in human-AI collaboration similarly shows that explanations aligned with human reasoning styles improve decision quality, reduce over- or under-reliance on algorithms, and support calibrated trust (Buçinca et al., 2021; Tao et al., 2024).

In investment settings, these capabilities allow LLMs to address the areas of:

- **Explanation:** explain long or short positions in terms of economically meaningful themes such as margin trends, cyclicality, or balance-sheet risk rather than raw statistical coefficients,
- **Clarity:** clarify how a signal should be interpreted, for example cross-sectional versus time-series or short- versus long-horizon,
- **Translation:** translate factor exposures into intuitive narratives, such as "the model is overweight quality and underweight leverage due to improving profitability trends",
- **Contextualization:** contextualise model views within macroeconomic or sector narratives, for instance "signals favour defensives under expected volatility regimes", and
- **Rationales:** generate portfolio-manager-style rationales that mirror the explanatory structures used in discretionary decision-making (Pelster and Val, 2024; Ko and Lee, 2024).

These natural-language explanations serve two functions. First, they make quantitative signals more usable by fundamental portfolio managers by placing them within familiar conceptual frameworks. Second, they support governance and accountability in institutional asset management, where risk committees, compliance teams, and clients increasingly expect interpretable and narrative-based justification for model-driven decisions.

By acting as an interpretive interface between complex models and human investment processes, LLMs help create a shared analytical language. This textual layer is essential for the organisational integration at the core of AFI: quantitative teams gain a means of articulating model insights in human-explainable terms, while fundamental portfolio managers gain transparent and actionable understanding of systematic signals. In this way, LLM-enabled explainability is not an auxiliary feature. It is the operational backbone of a unified Quantamental investment process.

4.5. Use Case for a LLM-Enabled Quantamental System within Investment Firms

A useful way to illustrate the practical implications of LLM-enabled AFI is to consider how an investment firm might deploy a unified, organisation-wide system that systematically captures, formalises, and scales fundamental investment expertise. Figure 6 presents a schematic, end-to-end workflow for such a system. At the input stage, Human Expert Data is curated through structured raw data collection from fundamental portfolio managers, using targeted question prompts, scenario analyses, and iterative HIL feedback mechanisms. These inputs include qualitative assessments, probabilistic forecasts, investment theses, and decision rationales. The curated Human Expert Data is then processed using AFI modelling techniques, such as LLM-based representation learning, calibration algorithms, and performance-weighted aggregation, to generate Superforecasting Data. This intermediate layer translates subjective expert judgments into structured, time-stamped, and empirically evaluated signals. The resulting Superforecasting Data can subsequently be integrated as input features within quantitative investment processes, including factor models, signal generation pipelines, and portfolio optimisation frameworks. Importantly, the system relies strongly on both human and model feedback loops. Fundamental portfolio managers receive AI-generated insights regarding performance attribution, rationale explanations, calibration diagnostics, and counterfactual analysis, enabling continuous improvement in forecasting accuracy and

decision quality. In this way, a LLM-enabled AFI system brings fundamental and quantitative investing closer together and enhances organizational learning by transforming dispersed human expertise into a scalable and systematically exploitable asset.

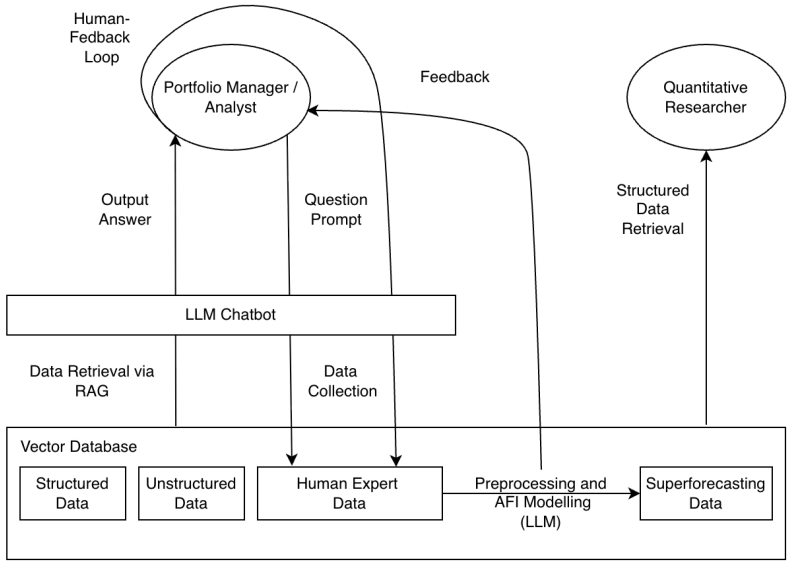

Figure 6: Integrated Portfolio Management AI System for Information Retrieval, Data Collection and Human-Insight Processing

Step 1: Ingesting and Structuring Unstructured Information

The system begins by ingesting unstructured data e.g. annual reports, earnings transcripts, sustainability disclosures, and regulatory filings through a RAG pipeline. Documents are stored in an internal vector database, with embeddings generated by domain-adapted LLMs. Numerical tables and other structured fields are extracted and serialized, following methodologies such as those outlined in Arun et al. (2023), demonstrating that LLMs fine-tuned on FinQA and similar datasets can handle multi-step financial calculations and numerical reasoning. This creates a canonical, query-ready representation of a firm's fundamental data.

Step 2: Analyst-Facing Question-Answering (Q&A) Interface

Fundamental portfolio managers interact with this content through a secure Q&A interface layered on top of the RAG system. The interface supports natural-language queries such as:

- "What explains the decline in operating cash flow relative to last year?"
- "How has management guidance on margins evolved over the last three quarters?"
- "List the main competitive risks mentioned in the MD&A section."

The system retrieves relevant passages, performs context-aware reasoning, and constructs grounded responses with citations to verifiable sources. As portfolio managers explore company disclosures, they can cross-reference answers against internal datasets, broker research, models, ESG assessments, or historical notes, creating a unified information layer.

Step 3: Systemising Analyst Prompts and Analytical Routines

Over time, the system logs the most frequent analyst queries, follow-up questions, and reasoning chains. These interactions reveal the implicit analytical frameworks used by discretionary managers (e.g. patterns in margin analysis, risk framing, competitive structure assessment). These routines can be systemised into reusable

analytical templates analogous to “deep research” workflows but customised to the institution’s investment philosophy. Standardised prompts formalise recurring tasks such as reading transcripts for forward-looking guidance, identifying red flags, or assessing accounting quality.

Step 4: Incorporating Human Feedback and Expertise

Each analyst interaction generates metadata: relevance flags, corrections, preferred answer formats, and signals the analyst deems most important for decision-making. This feedback loop creates a high-value proprietary dataset linking: prompts → retrieved evidence → model responses → analyst corrections → investment actions.

Such HIL feedback is essential for aligning LLM reasoning with institutional standards. It also reflects a growing trend in AI-augmented knowledge systems where organisations “train the model with their workflows” rather than adjusting workflows to the model.

Step 5: Learning From Analyst Behaviour and Creating Expert-Derived Datasets

As the system captures interactions over time, it constructs a unique dataset of expert-labelled reasoning paths and decision-relevant heuristics. This includes:

- which disclosures analysts rely on most;
- how they interpret changes in tone or guidance;
- which financial ratios or narratives trigger concern;
- how they map textual evidence into investment views.

This dataset constitutes an internal representation of the firm’s investment culture, its reasoning style, risk preferences, and informational priorities. It becomes a foundational asset for investment teams.

Step 6: Building Quantamental Features and Hybrid Models

Quantitative researchers can use the accumulated expert-annotated dataset to develop new features and modelling frameworks. Depending on the institution’s focus, this may include:

- fine-tuning LLMs to approximate analyst reasoning on new companies;
- training supervised models to predict analyst “flags” or narrative risk scores;
- building factor models, trading strategies or macro-nowcasting tools using LLM-derived textual signals;
- incorporating analyst-validated reasoning chains into Black–Litterman priors;
- creating interpretable quant signals grounded in expert domain knowledge.

The objective is not to replicate human analysts, but to systemise and scale the aspects of their reasoning that offer durable explanatory or predictive value for the enhancement of existing systematic models.

Step 7: Organisational Benefits and Reduced Overhead

A system of this kind offers two reinforcing advantages:

1. Overheads: reduced operational overhead by consolidating document analysis, information retrieval, and reasoning into a unified workflow.
2. Learning: enhanced organisational learning**,** since every analyst interaction enriches a shared, living knowledge base rather than remaining siloed in individual notes, memories, or teams.

This closes the feedback loop between discretionary judgment and systematic modelling. It also addresses a long-standing challenge in investment organisations: how to retain tacit knowledge and embed institutional expertise into scalable processes.

5. Remarks and Disclaimer

The techniques discussed in this paper are intended to support decision-making rather than to operate as autonomous decision-makers. LLMs can streamline elements of the research process, for example by summarizing literature, suggesting hypotheses, or accelerating code development. However, their outputs remain vulnerable to errors, hallucinations, and gaps in domain knowledge. All generated content should therefore be carefully reviewed, validated, and reproduced by qualified researchers before it is incorporated into quantitative models, investment processes, or client-facing materials. This guide does not recommend delegating core portfolio-construction or risk-management decisions to LLMs. Instead, it identifies contexts in which these tools can serve as useful complements within a well-controlled and transparent research workflow.

Practitioners must also address the governance, regulatory, and data-protection considerations associated with LLM use. The integration of GenAI into investment research should follow established model-risk management standards, including clear documentation of use cases, limitations, and validation procedures. Private, confidential, or regulated data should not be submitted to public LLM interfaces unless appropriate legal, technical, and contractual safeguards are in place. In most cases, institutions should rely on secure enterprise deployments with strict access control, logging, and data minimisation. Users remain responsible for complying with applicable financial-services regulation, data-protection law, and internal governance policies. This paper is not a substitute for regulatory due diligence and does not constitute investment advice.

The authors note that LLMs were used in a limited editorial capacity during the preparation of this manuscript. These tools assisted with refining phrasing, restructuring sections, and improving clarity. All technical content, empirical reasoning, and conclusions are the authors' own and were independently verified.

The aim of this paper is not to position LLMs as universally applicable solutions, but to provide guidance on when and how they can be applied responsibly to enhance research efficiency and to help distil information more effectively. As with any emerging technology, careful application, appropriate controls, and human oversight remain essential.

6. Conclusion and Outlook

LLMs are reshaping quantitative investment research by expanding analytical capacity, deepening interpretability, and enabling new classes of hybrid workflows that blend human judgment with algorithmic rigour. Their impact is unfolding along three fronts: LLMs as research assistants ("LLM Assistants"), LLMs as quantitative modelling tools ("LLM Quant"), and LLMs as enablers of AFI ("LLM Quantamental"). Together, these developments mark a structural shift in how institutional investors generate, validate, and communicate investment insight.

6.1. "LLM Assistants": Outlook for LLMs as Research Assistants

The role of LLMs as research assistants in quantitative finance is likely to grow substantially as institutions move from isolated model experiments to fully integrated AI co-pilots embedded within research pipelines. Early deployments, such as BloombergGPT-enhanced analytics, Refinitiv Workspace AI extensions, and FactSet's AI Copilot, demonstrate how domain-specific models can streamline information retrieval, document review, and exploratory analysis. The most significant efficiency gains, however, are expected from systems that leverage proprietary organisational data, enabling internal LLMs to operate across research, portfolio management, and operational functions.

Several investment firms have reported experimenting with and implementing internal LLM platforms that connect research notes, model documentation, and code repositories, thereby improving navigation of

institutional knowledge. In parallel, others are developing AI-assisted backtesting and research engines that unify data access, modelling workflows, and code generation. These systems support context-aware assistance for literature synthesis, idea generation, diagnostics, code validation, and communication.

A promising direction is the adoption of multi-agent architectures, in which specialised LLMs coordinate tasks, for example, one summarising academic literature, another writing or checking code, and a third validating methodological or governance constraints. Analogous patterns in computational science and automated reasoning that the future of research assistance will involve structured and collaborative agent-based systems.

Realising this potential requires addressing several challenges, including explainability, hallucination control, and output consistency, critical concerns in high-stakes environments such as portfolio construction and risk management. Trusted deployment will depend on integrating RAG for grounding, domain adaptation through fine-tuning, and alignment with version-controlled research processes to ensure traceability and reproducibility. Therefore, LLMs should be embedded within governed workflows rather than used as standalone tools. As these methods mature, LLMs are expected to become reliable cognitive partners that amplify human expertise, accelerate research cycles, and support more iterative, experiment-driven models of quantitative innovation.

6.2. "LLM Quant": Outlook for Quantitative Modelling

The trajectory of LLM use in finance is moving from incremental productivity gains toward a deeper reimagining and reengineering of research processes**.** Many of today's applications, particularly sentiment analysis, extend established workflows. Furthermore, LLM-generated features already contribute to trading signals, risk modelling, and macroeconomic forecasting. However, the field remains early in its adoption cycle, and a consensus has not emerged on which use cases will generate persistent value.

Future progress will depend on addressing foundational methodological challenges, including temporal leakage, memorisation biases, reproducibility constraints, and heterogeneous performance across languages and markets. Advancing LLM Quant will require coordinated innovation across four fronts:

1. **Data:** improving generalisation across assets, geographies, and market regimes.
2. **Methodology:** shifting from backtesting to live validation, strengthening multimodal reasoning, and integrating numerical problem-solving.
3. **Ethics and governance:** embedding explainability, auditability, and bias control into workflows suited for regulated environments.
4. **Professional impact:** understanding how LLMs reshape decision-making, model development, and the organisation of research teams.

In the short-term LLMs are likely to remain most effective as feature generators and interpreters of complex information rather than as standalone forecasting engines. Their value will stem from expanding the set of usable signals and accelerating research iteration, not replacing traditional models. In the long term, durable progress will depend on balancing technological ambition with careful evaluation, principled model selection, and disciplined governance to ensure that LLMs enhance rather than undermine the integrity of quantitative research.

6.3. "LLM Quantamental": Outlook for LLMs as Engines of AFI

LLMs enable AFI to move from conceptual aspiration to operational reality. Across recent academic work, five capabilities consistently support this transition:

1. **Capturing expert reasoning** by interpreting filings, earnings calls, reports, and news with human-level contextual understanding.
2. **Systemising and scaling qualitative insight** through chain-of-thought prompting, decomposition methods, structured templates, and retrieval-augmented grounding.
3. **Creating new alternative datasets** by extracting expert and crowd narratives at scale.

4. **Integrating qualitative reasoning into quantitative models**, with LLM-derived assessments incorporated into factors, Black-Litterman frameworks, and machine-learning forecasting models.
5. **Translating quantitative outputs into natural-language explanations**, closing the communication loop between discretionary and systematic teams.

In this sense, LLMs bring AFI's core objective, bridging fundamental and quantitative research, to maturity. They create hybrid architectures in which human-style interpretability and economic intuition are combined with systematic scalability and rigour.

At the same time, deploying LLM-enabled AFI systems introduces distinctive risks, particularly in institutional contexts where auditability, governance, and explainability are regulatory requirements. Key challenges include temporal leakage due to pretraining on future information (Glasserman and Lin, 2023), reproducibility issues arising from prompt sensitivity and model versioning (Pelster and Val, 2024; Chou et al., 2025), and hallucination or semantic misinterpretation in financial text analysis (Huang et al., 2025). LLMs also face objective-alignment limitations, optimising token likelihood rather than investment performance, and add governance burdens such as model traceability, drift detection, and controls around proprietary or regulated information.

Explainability will remain central. Investment decisions must be defensible to risk committees, clients, compliance, and regulators. While LLMs can generate natural-language rationales, such explanations may be incomplete or post hoc. This underscores the need for hybrid systems in which LLM-based reasoning is grounded in verifiable data through RAG, validated against deterministic models, and overseen by human experts.

Recent empirical evidence indicates that these challenges can be managed and that LLMs already improve signal extraction, research productivity, and decision support. Under the AFI lens, this marks the institutionalisation of Quantamental investing: a unified architecture in which qualitative and quantitative reasoning reinforce one another within scalable, governed, and transparent investment processes.

7. References

Agarwal, Chirag, Sree Harsha Tanneru, and Himabindu Lakkaraju. 'Faithfulness vs. Plausibility: On the (Un)Reliability of Explanations from Large Language Models'. arXiv preprint, 2024. https://arxiv.org/abs/2402.04614.

Ali, Alnur, Ashutosh Baheti, Jonathan Chang, Ta-Chung Chi, Brandon Cui, Andrew Drozdov, Jonathan Frankle, et al. 'A State-of-the-Art SQL Reasoning Model Using RLVR', 2025. https://arxiv.org/abs/2509.21459.

Ali, Hassnian, Muhammad Bilal Zafar, and Ahmet Faruk Aysan. 'Generative AI in Finance: Replicability, Methodological Contingencies, and Future Research Directions'. *Finance Research Letters* 86 (2025): 108797. https://doi.org/https://doi.org/10.1016/j.frl.2025.108797.

Anara. 'Anara', 2025. https://anara.com/.

Araci, Dogu. 'FinBERT: Financial Sentiment Analysis with Pre-Trained Language Models', 2019. https://arxiv.org/abs/1908.10063.

Arivazhagan, Manoj Ghuhan, Lan Liu, Peng Qi, Xinchi Chen, William Yang Wang, and Zhiheng Huang. 'Hybrid Hierarchical Retrieval for Open-Domain Question Answering'. In *Findings of the Association for Computational Linguistics: ACL 2023*, edited by Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki, 10680–89. Toronto, Canada: Association for Computational Linguistics, 2023. https://doi.org/10.18653/v1/2023.findings-acl.679.

Arun, Abhinav, Ashish Dhiman, Mehul Soni, and Yibei Hu. 'Numerical Reasoning for Financial Reports', 2023. https://arxiv.org/abs/2312.14870.

Ashby, William Ross. *An Introduction to Cybernetics*. London: Chapman & Hall Ltd., 1956.

Balel, Yunus. 'ScholarGPT's Performance in Oral and Maxillofacial Surgery'. *Journal of Stomatology Oral and Maxillofacial Surgery* 126, no. 4 (2025): 102114. https://doi.org/https://doi.org/10.1016/j.jormas.2024.102114.

Bandara, Wachi, Brandon Flannery, and Anshuma Chandak. 'Can AI Explain Company Performance: A Horserace'. *SSRN Electronic Journal*, 2023. https://doi.org/10.2139/ssrn.4480665.

Bommasani, Rishi, Drew A. Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S. Bernstein, et al. 'On the Opportunities and Risks of Foundation Models', 2022. https://arxiv.org/abs/2108.07258.

Breitung, Christian, and Sebastian Müller. 'Global Business Networks'. *Journal of Financial Economics* 166 (2025): 104007. https://doi.org/https://doi.org/10.1016/j.jfineco.2025.104007.

Bridgeford, Eric W., Iain Campbell, Zijao Chen, Zhicheng Lin, Harrison Ritz, Joachim Vandekerckhove, and Russell A. Poldrack. 'Ten Simple Rules for AI-Assisted Coding in Science'. arXiv preprint, 2025. https://arxiv.org/abs/2510.22254.

Brown, Tom B., Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, et al. 'Language Models Are Few-Shot Learners', 2020. https://arxiv.org/abs/2005.14165.

Brynjolfsson, Erik, Danielle Li, and Lindsey R. Raymond. 'Generative AI at Work'. Working Paper. Cambridge, MA: National Bureau of Economic Research, April 2023. https://doi.org/10.3386/w31161.

Buçinca, Zana, Maja Barbara Malaya, and Krzysztof Z. Gajos. 'To Trust or to Think: Cognitive Forcing Functions Can Reduce Overreliance on AI in AI-Assisted Decision-Making'. *Proc. ACM Hum.-Comput. Interact.* 5, no. CSCW1 (April 2021). https://doi.org/10.1145/3449287.

Cao, Sean, Charles C. Y. Wang, and Yi Xiang. 'When LLMs Go Abroad: Foreign Bias in AI Financial Predictions'. Harvard Business School, September 2025. https://doi.org/10.2139/ssrn.5440116.

Carlini, Nicholas, Florian Tramer, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, et al. 'Extracting Training Data from Large Language Models', 2021. https://arxiv.org/abs/2012.07805.

Chen, Mark, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, et al. 'Evaluating Large Language Models Trained on Code', 2021. https://arxiv.org/abs/2107.03374.

Chen, Yifei, Bryan T. Kelly, and Dacheng Xiu. 'Expected Returns and Large Language Models'. SSRN Electronic Journal, November 2022. https://doi.org/10.2139/ssrn.4416687.

Chen, Zihan, Lei Zheng, Cheng Lu, Jialu Yuan, and Di Zhu. 'ChatGPT Informed Graph Neural Network for Stock Movement Prediction'. *SSRN Electronic Journal*, 2023. https://doi.org/10.2139/ssrn.4464002.

Chin, Andrew. 'Leveling the Divide Between Discretionary and Systematic Investing: How AI Enables Breadth and Depth'. *The Journal of Portfolio Management*, 2025. https://doi.org/10.3905/jpm.2025.1.730.

Chou, Wen-Hsiu (Julia), Zifeng Feng, Bingxin Li, and Feng Liu. 'A First Look at Financial Data Analysis Using ChatGPT-4o'. *Journal of Risk and Financial Management* 18, no. 2 (2025). https://doi.org/10.3390/jrfm18020099.

Christiano Silva, Thiago, Moriya, Kei, and Veyrune, Romain M. (2025). From text to quantified insights: A large-scale LLM analysis of central bank communication. IMF Working Papers, No. 109. https://doi.org/10.5089/9798229013802.001.

Clark, Kevin, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 'ELECTRA: Pre-Training Text Encoders as Discriminators Rather Than Generators', 2020. https://arxiv.org/abs/2003.10555.

Consensus. 'Consenus', 2025. https://framer.consensus.app/home/about-us.

Davenport, Thomas H., and Nitin Mittal. *All-In on AI: How Smart Companies Win Big with Artificial Intelligence*. Boston, MA: Harvard Business Review Press, 2023.

Devlin, Jacob, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 'BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding', 2019. https://arxiv.org/abs/1810.04805.
Elicit. 'Elicit', 2025.
Ernstberger, Constantin and Nazemi, Abdolreza. "Deciphering Earnings Calls with Large Language Models: Insights into Defaulted Corporate Bonds," SSRN Working Paper, March 1, 2025, http://dx.doi.org/10.2139/ssrn.5257223.
Gartner, Inc. 'Gartner Hype Cycle for Emerging Technologies, 2024 Highlights Developer Productivity, Total Experience, AI and Security', August 2024. https://www.gartner.com/en/newsroom/press-releases/2024-08-21-gartner-2024-hype-cycle-for-emerging-technologies-highlights-developer-productivity-total-experience-ai-and-security.
Gilpin, Leilani H., David Bau, Ben Z. Yuan, Ayesha Bajwa, Michael Specter, and Lalana Kagal. 'Explaining Explanations: An Overview of Interpretability of Machine Learning'. In *2018 IEEE 5th International Conference on Data Science and Advanced Analytics (DSAA)*, 80–89, 2018. https://doi.org/10.1109/DSAA.2018.00018.
Glasserman, Paul, and Caden Lin. 'Assessing Look-Ahead Bias in Stock Return Predictions Generated by GPT Sentiment Analysis'. In *Social Science Research Network*, 2023. https://api.semanticscholar.org/CorpusID:263310733.
Google. 'Gemini Deep Research', 2025. https://gemini.google/overview/deep-research.
Google Cloud. 'A New Top Score: Advancing Text-to-SQL on the BIRD Benchmark', 2025. https://cloud.google.com/blog/products/databases/how-to-get-gemini-to-deeply-understand-your-database.
He, Pengcheng, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 'DeBERTa: Decoding-Enhanced BERT with Disentangled Attention', 2021. https://arxiv.org/abs/2006.03654.
Hong, Lu, and Scott E. Page. 'Groups of Diverse Problem Solvers Can Outperform Groups of High-Ability Problem Solvers'. *Proceedings of the National Academy of Sciences* 101, no. 46 (2004): 16385–89. https://doi.org/10.1073/pnas.0403723101.
Hongyuan, Dong, Che Wanxiang, He Xiaoyu, Zheng Guidong, and Wen Junjie. 'FinBART: A Pre-Trained Seq2seq Language Model for Chinese Financial Tasks'. In *Proceedings of the 22nd Chinese National Conference on Computational Linguistics*, edited by Maosong Sun, Bing Qin, Xipeng Qiu, Jing Jiang, and Xianpei Han, 906–17. Harbin, China: Chinese Information Processing Society of China, 2023. https://aclanthology.org/2023.ccl-1.77/.
Huang, Allen H., Hui Wang, and Yi Yang. 'FinBERT: A Large Language Model for Extracting Information from Financial Text'. *Contemporary Accounting Research* 40, no. 2 (2023): 806–41. https://doi.org/https://doi.org/10.1111/1911-3846.12832.
Huang, Lei, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, et al. 'A Survey on Hallucination in Large Language Models: Principles, Taxonomy, Challenges, and Open Questions'. *ACM Trans. Inf. Syst.* 43, no. 2 (January 2025). https://doi.org/10.1145/3703155.
Ibryam, Bilgin. 'AI Coding Assistants Landscape (03/2025)', December 2024. https://generativeprogrammer.com/p/ai-coding-assistants-landscape.
Jegadeesh, Narasimhan, and Di Wu. 'Word Power: A New Approach for Content Analysis'. *Journal of Financial Economics* 110, no. 3 (2013): 712–29. https://doi.org/https://doi.org/10.1016/j.jfineco.2013.08.018.
Jelodar, Hamed, Mohammad Meymani, and Roozbeh Razavi-Far. 'Large Language Models (LLMs) for Source Code Analysis: Applications, Models and Datasets'. arXiv preprint, 2025. https://arxiv.org/abs/2503.17502.
Ji, Ziwei, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye-Jin Bang, Delong Chen, Wenliang Dai, Ho Shu Chan, Andrea Madotto, and Pascale Fung. 'Survey of Hallucination in Natural Language Generation.' ACM Computing Surveys 55, no. 12 (2023): 1–38. https://doi.org/10.1145/3571730.

Jiang, Albert Q., Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, et al. 'Mistral 7B', 2023. https://arxiv.org/abs/2310.06825.
Jiang, Juyong, Fan Wang, Jiasi Shen, Sungju Kim, and Sunghun Kim. 'A Survey on Large Language Models for Code Generation'. ACM Computing Surveys, 2025. https://doi.org/10.1145/3747588.
Kahneman, Daniel. *Thinking, Fast and Slow*. New York, NY: Farrar, Straus and Giroux, 2011.
Karpukhin, Vladimir, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 'Dense Passage Retrieval for Open-Domain Question Answering'. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, edited by Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu, 6769–81. Online: Association for Computational Linguistics, 2020. https://doi.org/10.18653/v1/2020.emnlp-main.550.
Khatchadourian, Raffi, and Rolando Franco. 'LLM Output Drift: Cross-Provider Validation & Mitigation for Financial Workflows', 2025. https://arxiv.org/abs/2511.07585.
Kirtac, Kemal, and Guido Germano. 'Sentiment Trading with Large Language Models'. Finance Research Letters 62, Part B (2024): 105227. https://doi.org/10.1016/j.frl.2024.105227.
Kirtac, Kemal, and Guido Germano. 'Large Language Models in Finance: Estimating Financial Sentiment for Stock Prediction'. Elsevier BV, 2025. https://doi.org/10.2139/ssrn.5166656.
Klein, Gary. *Sources of Power: How People Make Decisions*. Cambridge, MA: The MIT Press, 1998.
Ko, Hyungjin, and Jaewook Lee. 'Can ChatGPT Improve Investment Decisions? From a Portfolio Management Perspective'. *Finance Research Letters* 64 (2024): 105433. https://doi.org/10.1016/j.frl.2024.105433.
Kong, Yaxuan, Yuqi Nie, Xiaowen Dong, John M. Mulvey, H. Vincent Poor, Qingsong Wen, and Stefan Zohren. 'Large Language Models for Financial and Investment Management: Applications and Benchmarks'. *The Journal of Portfolio Management* 51, no. 2 (2024): 162–210. https://doi.org/10.3905/jpm.2024.1.645.
Lee, Hoyoung, Junhyuk Seo, Suhwan Park, Junhyeong Lee, Wonbin Ahn, Chanyeol Choi, Alejandro Lopez-Lira, and Yongjae Lee. 'Your AI, Not Your View: The Bias of LLMs in Investment Analysis', 2025. https://arxiv.org/abs/2507.20957.
Levy, Bradford. 'Caution Ahead: Numerical Reasoning and Look-Ahead Bias in AI Models'. *SSRN Electronic Journal*, 2024. https://doi.org/10.2139/ssrn.5082861.
Li, Jinyang, Binyuan Hui, Ge Qu, Jiaxi Yang, Binhua Li, Bowen Li, Bailin Wang, et al. 'Can LLM Already Serve as a Database Interface? A Big Bench for Large-Scale Database Grounded Text-to-SQLs'. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*. NIPS '23. Red Hook, NY, USA: Curran Associates Inc., 2023a.
Li, Yinheng, Shaofei Wang, Han Ding, and Hang Chen. 'Large Language Models in Finance: A Survey'. In *Proceedings of the Fourth ACM International Conference on AI in Finance*, 374–82. ICAIF '23. New York, NY, USA: Association for Computing Machinery, 2023b. https://doi.org/10.1145/3604237.3626869.
Lim, Bryan, Sercan O. Arik, Nicolas Loeff, and Tomas Pfister. 'Temporal Fusion Transformers for Interpretable Multi-Horizon Time Series Forecasting', 2020. https://arxiv.org/abs/1912.09363.
Liner. 'Liner', 2025. https://www.liner.com/.
Litmaps. 'Litmaps', 2025. https://www.litmaps.com/.
Liu, Yinhan, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 'RoBERTa: A Robustly Optimized BERT Pretraining Approach', 2019. https://arxiv.org/abs/1907.11692.
Livnat, Joshua, Jacob Pozharny, and Kate Suslava. 'The Next Accounting Frontier: Bringing Structure and Reliability to NLP'. *The Journal of Portfolio Management* 52, no. 1 (November 2025): 179–201. https://doi.org/10.3905/jpm.2025.1.767.
López de Prado, Marcos. *Advances in Financial Machine Learning*. 1st edn. Hoboken, NJ: Wiley, 2018.

Lopez-Lira, Alejandro, and Yuehua Tang. 'Can ChatGPT Forecast Stock Price Movements? Return Predictability and Large Language Models', 2025. https://arxiv.org/abs/2304.07619.
Lopez-Lira, Alejandro, Yuehua Tang, and Mingyin Zhu. 'The Memorization Problem: Can We Trust LLMs' Economic Forecasts?', 2025. https://arxiv.org/abs/2504.14765.
Loughran, Tim, and Bill McDonald. 'When Is a Liability Not a Liability? Textual Analysis, Dictionaries, and 10-Ks'. *The Journal of Finance* 66, no. 1 (2011): 35–65. https://doi.org/https://doi.org/10.1111/j.1540-6261.2010.01625.x.
Magner, Nicolás, Pablo A. Henríquez, and Aliro Sanhueza. 'Decoding Risk Sentiment in 10-K Filings: Predictability for U.S. Stock Indices'. Finance Research Letters 81 (2025): 107472. https://doi.org/10.1016/j.frl.2025.107472.
Malo, Pekka, Ankur Sinha, Pyry Takala, Pekka Korhonen, and Jyrki Wallenius. 'Good Debt or Bad Debt: Detecting Semantic Orientations in Economic Texts', 2013. https://arxiv.org/abs/1307.5336.
Mihov, Anna-Helena, Nick Firoozye, and Philip Treleaven. 'Towards Augmented Financial Intelligence'. *SSRN Electronic Journal*, January 2022. https://doi.org/10.2139/ssrn.4148057.
Mihov, Anna-Helena, Nick Firoozye, and Philip Treleaven. 'An Augmented Financial Intelligence Multi-Factor Model'. *SSRN Electronic Journal*, July 2024. https://doi.org/10.2139/ssrn.4900781.
OpenAI. 'Introducing Deep Research', 2025. https://openai.com/index/introducing-deep-research/.
Pelster, Matthias, and Joel Val. 'Can ChatGPT Assist in Picking Stocks?' *Finance Research Letters* 59 (2024): 104786. https://doi.org/https://doi.org/10.1016/j.frl.2023.104786.
Perplexity. 'Introducing Perplexity Deep Research', 2025. https://www.perplexity.ai/hub/blog/introducing-perplexity-deep-research.
Radford, Alec, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 'Language Models Are Unsupervised Multitask Learners', 2019. https://api.semanticscholar.org/CorpusID:160025533.
Salton, Gerard, and Christopher Buckley. 'Term-Weighting Approaches in Automatic Text Retrieval'. *Information Processing & Management* 24, no. 5 (1988): 513–23. https://doi.org/https://doi.org/10.1016/0306-4573(88)90021-0.
Sawarkar, Kunal, Abhilasha Mangal, and Shivam Raj Solanki. 'Blended RAG: Improving RAG (Retriever-Augmented Generation) Accuracy with Semantic Search and Hybrid Query-Based Retrievers'. In *2024 IEEE 7th International Conference on Multimedia Information Processing and Retrieval (MIPR)*, 24:155–61. IEEE, 2024. https://doi.org/10.1109/mipr62202.2024.00031.
SciSpace. 'SciSpace', 2025. https://scispace.com/.
SciSummary. 'SciSummary', 2025. https://scisummary.com/.
Sharkey, Ed, and Philip Treleaven. 'Optimising Large Language Models: Taxonomy and Techniques', 2025. https://doi.org/10.2139/ssrn.5278456.
Sourcely. 'Sourcely', 2025. https://www.sourcely.net/.
Spyrou, Alex, and Brian Pisaneschi. 'Practical Guide for LLMs in the Financial Industry'. CFA Institute Research and Policy Center, 2024. https://rpc.cfainstitute.org/research/the-automationahead-content-series/practical-guide-for-llms-in-the-financial-industry.
Su, Di-Jia, John M Mulvey, and H Vincent Poor. 'Improving Portfolio Performance via Natural Language Processing Methods.' *Journal of Financial Data Science* 4, no. 2 (2022).
Tambon, Florian, Arghavan Moradi-Dakhel, Amin Nikanjam, Foutse Khomh, Michel C. Desmarais, and Giuliano Antoniol. 'Bugs in Large Language Models Generated Code: An Empirical Study'. *Empirical Software Engineering* 30 (2024). https://api.semanticscholar.org/CorpusID:268385127.
Tao, Shuchang, Liuyi Yao, Hanxing Ding, Yuexiang Xie, Qi Cao, Fei Sun, Jinyang Gao, Huawei Shen, and Bolin Ding. 'When to Trust LLMs: Aligning Confidence with Response Quality', 2024. https://arxiv.org/abs/2404.17287.
Tetlock, Paul C. 'Giving Content to Investor Sentiment: The Role of Media in the Stock Market'. *The Journal of Finance* 62, no. 3 (2007): 1139–68. https://doi.org/https://doi.org/10.1111/j.1540-6261.2007.01232.x.

Tetlock, Philip E., and Dan Gardner. *Superforecasting: The Art and Science of Prediction*. Random House, 2016.

Touvron, Hugo, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, et al. 'LLaMA: Open and Efficient Foundation Language Models', 2023. https://arxiv.org/abs/2302.13971.

Vaithilingam, Priyan, Tianyi Zhang, and Elena L. Glassman. 'Expectation vs. Experience: Evaluating the Usability of Code Generation Tools Powered by Large Language Models'. In *Extended Abstracts of the 2022 CHI Conference on Human Factors in Computing Systems*. CHI EA '22. New York, NY, USA: Association for Computing Machinery, 2022. https://doi.org/10.1145/3491101.3519665.

Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 'Attention Is All You Need'. In *Advances in Neural Information Processing Systems*, edited by I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Vol. 30. Curran Associates, Inc., 2017. https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

Wang, Richard. 'Beyond Fog: Measuring Managerial Obfuscation Using LLM'. SSRN Electronic Journal, November 2025. https://doi.org/10.2139/ssrn.5702344.

Wang, Saizhuo, Hang Yuan, Leon Zhou, Lionel Ni, Heung-Yeung Shum, and Jian Guo. 'Alpha-GPT: Human–AI Interactive Alpha Mining for Quantitative Investment'. In Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing: System Demonstrations. Association for Computational Linguistics, 2025. https://doi.org/10.18653/v1/2025.emnlp-demos.14.

Wang, Saizhuo, Hang Yuan, Lionel M. Ni, and Jian Guo. 'QuantAgent: Seeking Holy Grail in Trading by Self-Improving Large Language Model', 2024. https://arxiv.org/abs/2402.03755.

Wood, Kieran, Stephen Roberts, and Stefan Zohren. 'Slow Momentum with Fast Reversion: A Trading Strategy Using Deep Learning and Changepoint Detection', The Journal of Financial Data Science, vol. 4, no. 1, pp. 111–129, 2022.

Wu, Scott. 'Introducing Devin, the First AI Software Engineer.' Cognition AI Blog, March 12, 2024. https://www.cognition.ai/blog/introducing-devin.

Wu, Haixu, Jiehui Xu, Jianmin Wang, and Mingsheng Long. 'Autoformer: Decomposition Transformers with Auto-Correlation for Long-Term Series Forecasting', 2022. https://arxiv.org/abs/2106.13008.

Wu, Shijie, Ozan Irsoy, Steven Lu, Vadim Dabravolski, Mark Dredze, Sebastian Gehrmann, Prabhanjan Kambadur, David Rosenberg, and Gideon Mann. 'BloombergGPT: A Large Language Model for Finance', 2023. https://arxiv.org/abs/2303.17564.

Yadav, Daksha, Sabrina Zhang, Tom Jin, Prakash Krishnan, and Des Clarke. 'Generative AI Based Virtual Assistant for Reconciliation Research', 2024. https://www.amazon.science/publications/generative-ai-based-virtual-assistant-for-reconciliation-research.

Yang, Hongyang, Xiao-Yang Liu, and Christina Dan Wang. 'FinGPT: Open-Source Financial Large Language Models', 2025. https://arxiv.org/abs/2306.06031.

Yang, Yi, Yixuan Tang, and Kar Yan Tam. 'InvestLM: A Large Language Model for Investment Using Financial Domain Instruction Tuning', 2023. https://arxiv.org/abs/2309.13064.

Yao, Zhewei, Guoheng Sun, Lukasz Borchmann, Zheyu Shen, Minghang Deng, Bohan Zhai, Hao Zhang, Ang Li, and Yuxiong He. 'Arctic-Text2SQL-R1: Simple Rewards, Strong Reasoning in Text-to-SQL', 2025. https://arxiv.org/abs/2505.20315.

Yu, Xinli, Zheng Chen, Yuan Ling, Shujing Dong, Zongyi Liu, and Yanbin Lu. 'Temporal Data Meets LLM – Explainable Financial Time Series Forecasting', 2023. https://arxiv.org/abs/2306.11025.

Yu, Yangyang, Haohang Li, Zhi Chen, Yuechen Jiang, Yang Li, Jordan W. Suchow, Denghui Zhang, and Khaldoun Khashanah. 'FinMem: A Performance-Enhanced LLM Trading Agent With Layered Memory and Character Design'. *IEEE Transactions on Big Data* 11, no. 6 (2025): 3443–59. https://doi.org/10.1109/TBDATA.2025.3593370.

Yuan, Hang, Saizhuo Wang, and Jian Guo. 'Alpha-GPT 2.0: Human-in-the-Loop AI for Quantitative Investment', 2024. https://arxiv.org/abs/2402.09746.
Yue, Thomas, and Chi Au. 'GPTQuant's Conversational AI: Simplifying Investment Research for All'. *SSRN Electronic Journal*, January 2023. https://doi.org/10.2139/ssrn.4380516.
Zhang, Chong, Xinyi Liu, Zhongmou Zhang, Mingyu Jin, Lingyao Li, Zhenting Wang, Wenyue Hua, et al. 'When AI Meets Finance (StockAgent): Large Language Model-Based Stock Trading in Simulated Real-World Environments', 2024. https://arxiv.org/abs/2407.18957.
Zhang, Susan, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, et al. 'OPT: Open Pre-Trained Transformer Language Models', 2022. https://arxiv.org/abs/2205.01068.
Zhou, Haoyi, Shanghang Zhang, Jieqi Peng, Shuai Zhang, Jianxin Li, Hui Xiong, and Wancai Zhang. 'Informer: Beyond Efficient Transformer for Long Sequence Time-Series Forecasting', 2021. https://arxiv.org/abs/2012.07436.
Zhou, Tian, Ziqing Ma, Qingsong Wen, Xue Wang, Liang Sun, and Rong Jin. 'FEDformer: Frequency Enhanced Decomposed Transformer for Long-Term Series Forecasting', 2022. https://arxiv.org/abs/2201.12740.
Ziegler, Albert, Eirini Kalliamvakou, X. Alice Li, Andrew Rice, Devon Rifkin, Shawn Simister, Ganesh Sittampalam, and Edward Aftandilian. 'Productivity Assessment of Neural Code Completion'. In *Proceedings of the 6th ACM SIGPLAN International Symposium on Machine Programming*, 21–29. MAPS 2022. New York, NY, USA: Association for Computing Machinery, 2022. https://doi.org/10.1145/3520312.3534864.

APENDIX.

A. Representative RAG Components and Technology Stack for Retrieval-Augmented LLM Systems

The categories in the first column distinguish between the different functional layers that make up a RAG system. Although all components contribute to the broader retrieval-generation pipeline, they play technically distinct roles:

1. **LLMs with Native RAG Features**
   These are generative models that accept retrieved documents as input and can ground their responses in external evidence. Technically, they sit at the generation layer of a RAG system, performing reasoning, summarization, synthesis, and citation based on the retrieved context. Some models provide built-in grounding APIs or citation metadata, but they do not perform retrieval by themselves, they rely on upstream retrievers or vector databases.
2. **Embedding Models for RAG**
   These models generate dense vector representations of text, enabling semantic similarity search. They operate at the encoding layer, transforming unstructured documents and queries into numerical vectors that make high-recall retrieval possible. They do not generate text themselves and instead, they define the semantic connection that determines which documents are retrieved.
3. **Managed Enterprise RAG Platforms**
   These platforms integrate multiple layers of the RAG stack, data ingestion, indexing, vector search, LLM inference, and sometimes evaluation, into a single managed service. Technically, they focus on the orchestration and infrastructure layer, ensuring scalability, security, compliance, versioning, and performance. They remove the burden of assembling the system from individual components.
4. **RAG Frameworks and Libraries**
   These tools define the application logic layer of a RAG system. They provide modules for document chunking, retrieval pipelines, query engines, prompt templates, multi-step reasoning workflows. While they do not perform retrieval or generation themselves, they connect retrievers, vector stores, and LLMs into reproducible processes.

5. **Vector Databases for RAG**
   These systems store embeddings and support vector similarity search, often combined with keyword or hybrid search. They form the retrieval layer, responsible for efficiently returning semantically relevant documents at query time. Vector databases determine the speed, scale, and recall of RAG systems.
6. **Retrievers and Models Supporting RAG**
   These are specialized retrieval models (e.g., DPR, ColBERT, BM25) that determine how relevant documents are identified. They operate as the retrieval algorithm layer, typically invoked by vector databases or hybrid search engines. They can be dense retrievers (semantic), sparse retrievers (keyword/BM25), hybrid retrievers (late interaction, dense+sparse). They do not generate text, but they strongly influence grounding quality.

Category	Tool / Model	Provider	Core Functionality	Practical Relevance
LLMs with Native RAG Features	**RAG (2020)**	Meta AI	Classical end-to-end RAG model using DPR for retrieval and BART for generation.	Useful academic baseline for RAG architecture understanding and benchmarking.
	GPT Models + RAG / Vector Store APIs	OpenAI	Embeddings, vector stores and retrieval tools enabling external document grounding.	Strong enterprise option; excellent for summarizing research reports and filings.
	Claude 3.5 (Citations API)	Anthropic	Native grounding and citation metadata for document-based answers.	Appeals to compliance and audit workflows requiring source transparency.
	Command R / Command R+	Cohere	RAG-optimized LLMs with multilingual support and citation grounding.	Useful for global macro research and multilingual corpora.
	Mistral SuperRAG 2.0	Mistral AI	Integrated retrieval and generation pipeline built around Mistral models.	Strong for on-prem or cost-sensitive quant environments.
	Gemma 2	Google	Lightweight, open, retrieval-friendly LLM family.	Efficient for self-hosted or GCP-native research workflows.
	LLaMA 3.x Family	Meta AI	High-performance open LLMs widely used in custom RAG setups.	Flexible backbone for proprietary research-assistant systems.
	Snowflake Arctic	Snowflake	Enterprise-grade open LLM family integrated into the Snowflake Data Cloud (with native vector search available).	Fits quant teams already operating in Snowflake's data ecosystem.
Embedding Models for RAG	**OpenAI Embeddings (e.g., text-embedding-3-large)**	OpenAI	Dense semantic embeddings for retrieval and clustering.	Strong baseline for indexing large financial text collections.
	Gemini Embeddings	Google	High-recall embeddings designed for RAG and semantic search.	Suited for multilingual and regulatory corpora.
	Mistral Embed	Mistral AI	Lightweight, efficient embedding model for vector retrieval.	Cost-effective for large internal research corpora.
	Instructor Embeddings	HKUNLP	Instruction-tuned embeddings achieving strong domain retrieval accuracy.	Effective for technical financial language and academic literature.
	BGE Embeddings (bge-large-en)	BAAI	High-performance open embeddings widely used in open-source RAG.	Strong accuracy/cost trade-off for internal search engines.

	E5 / E5-Mistral	Microsoft Research	Universal embeddings optimized for semantic retrieval.	Suitable for broad research search systems in multi-domain quant teams.
Managed Enterprise RAG Platforms	**Databricks Mosaic AI (Vector Search + Agents)**	Databricks	Fully integrated vector search, agents, and RAG evaluation in the Databricks platform.	Ideal for quant shops using Lakehouse architecture for research pipelines.
	Amazon Bedrock (Knowledge Bases)	AWS	Fully managed RAG pipeline with embeddings, vector DB, and grounding.	Easy integration with AWS-native research and alt-data workflows.
	Vertex AI Search & Grounding	Google Cloud	Enterprise document ingestion, retrieval, and grounding for RAG.	Strong for large-scale research corpora and BigQuery users.
	Azure AI Studio (RAG)	Microsoft	RAG pipelines, evaluation tools, vector indexing and governance.	Suitable for enterprise-grade internal research assistants.
	Elastic Search + ELSER	Elastic	Hybrid semantic + keyword retrieval for large document sets.	Effective for high-volume news feeds, regulatory filings, and earnings transcripts.
	Amazon OpenSearch	AWS	Vector + BM25 hybrid retrieval engine.	Common in enterprise setups for long-horizon research corpora.
RAG Frameworks & Libraries	**Haystack**	deepset	Modular indexing, retrieval, and generation pipelines.	Reliable for building internal research search engines.
	LlamaIndex	LlamaIndex	Data ingestion, indexing strategies, and RAG workflow orchestration.	Excellent for handling PDFs, broker notes, and academic papers.
	Weaviate RAG Stack	Weaviate	Vector DB with hybrid search and RAG utilities.	Good performance for financial jargon and heterogeneous corpora.
	DSPy	Stanford	Declarative optimization framework for RAG systems.	Automates retrieval and prompt tuning for higher accuracy.
	Pathway	Pathway	Real-time streaming RAG pipelines with continuous data ingestion.	Useful for intraday, news-driven, or macro event analyses.
	REALM	Google Research	Early retrieval-augmented LM architecture (research-oriented).	Historical reference point for modern RAG systems.
	LangChain	LangChain	LLM orchestration with pre-built RAG chains.	Widely used RAG prototyping tool.
	LangGraph	LangChain	State-machine engine for multi-step retrieval and reasoning tasks.	Useful for multi-stage workflows: summarization → extraction → classification.
	Dust	Dust	Framework for custom LLM assistants with semantic search/RAG support.	Suitable for research-team-specific internal AI tools.
Vector Databases for RAG	**Pinecone**	Pinecone	Managed vector DB with high-scaled retrieval.	Ideal for 10M+ document research universes (filings, transcripts).
	Milvus	LF AI	Open-source vector DB for large-scale AI retrieval.	Good for on-prem, compliant environments.
	Qdrant	Qdrant	Vector DB with dense + sparse hybrid search.	Strong performance for financial text retrieval.
	Weaviate	Weaviate	Vector + keyword hybrid search with metadata filters.	Good for heterogeneous and unstructured research data.

	Deep Lake	Activeloop	"Database for AI" supporting vector storage + LLM integration.	Useful for storing embeddings tied to large alt-data sources.
	Redis Vector Search	Redis	High-performance in-memory vector and hybrid search.	Valuable for latency-sensitive quant applications.
	Chroma DB	Chroma	Lightweight open-source vector DB.	Good for rapid prototyping of research workflows.
	Zep Vector Store	Zep	Vector store with document and conversation memory.	Useful for research assistants requiring persistent context.
Retrievers & Models Supporting RAG	**BM25**	Traditional IR	Sparse lexical retrieval using term weighting.	Strong baseline for regulatory filings, long reports, and legal texts.
	DPR	Meta AI	Dual-encoder semantic retriever.	Foundational retriever in early RAG systems.
	ColBERT / ColBERTv2	Stanford	Late-interaction dense retriever with high recall.	Excellent for long financial documents (10-Ks, transcripts).
	BART + Retrieval Pipelines	Meta AI	Seq2seq generator used in early RAG architectures.	Useful educational model for understanding RAG mechanics.

Table A: presents an overview of representative RAG models, frameworks, vector databases, embedding models, and enterprise platforms relevant for quantitative investment research. The table categorizes each offering by provider, core functionality, and practical applicability to financial research workflows, highlighting tools that support semantic retrieval, hybrid search, document-grounded generation, large-scale indexing, or enterprise-grade deployment. This summary is intended to guide practitioners in selecting appropriate components when designing or evaluating retrieval-augmented LLM systems for tasks such as research synthesis, information extraction, or domain exploration.

B. Selected General-Purpose AI Coding Assistant Tools

Tool (Provider)	**Description**
GitHub Copilot (GitHub / Microsoft)	In-editor code generation, auto-completion, refactoring, and test generation (Speeds up prototyping, modernizes legacy code via refactoring suggestions, aids in enforcing coding standards through customization; tightly integrated with VS Code and JetBrains IDEs). Also powering the Copilot Agent Mode, which handles multi-step tasks across the codebase within supported IDEs.
OpenAI Python / Code Interpreter (OpenAI)	Interactive Python execution, debugging, and visualization (Best for exploratory analysis, diagnostics, and small simulations rather than large production codebases).
Amazon Q Developer (AWS)	A comprehensive enterprise AI assistant that integrates inline code generation, debugging, security scanning, and agentic capabilities for multi-step tasks across AWS environments and codebases. Amazon CodeWhisperer now merged into Q Developer. Context-aware code recommendations with security scanning and AWS-native integration (Well suited for cloud-centric pipelines, infrastructure-as-code, and MLOps workflows).
Databricks AI Assistant (Databricks)	SQL generation, Spark optimization, pipeline debugging inside notebooks (Effective for large-scale data engineering and ML workflows; requires Databricks platform).
Jupyter AI (Project Jupyter / Anaconda)	LLM-powered assistance embedded in Jupyter notebooks (Supports mixed text-and-code reasoning; useful for research and experimentation, less optimized for software engineering at scale).
Google Gemini Code Assist (Google)	Code completion, explanation, and refactoring using Gemini models (Strong support for Python, Java, SQL; integrates with VS Code, JetBrains, and Google Cloud tooling).

Tabnine (Tabnine)	A privacy-first AI assistant offering code completions, generation, and chat, notable for its flexible deployment options (SaaS, on-prem, or air-gapped) and strict code privacy policies (Zero code retention).
Codeium (Codeium)	A popular, free-for-individual-use AI code acceleration toolkit offering rapid, context-aware completions, in-editor chat, and multi-file search capabilities, with broad IDE support.
Sourcegraph Cody (Sourcegraph)	An AI coding assistant specializing in understanding vast codebases via universal code search and graph analysis, providing accurate cross-file answers, code generation, and refactoring across many IDEs.
JetBrains AI Assistant (JetBrains)	A deep, context-aware AI integrated across all JetBrains IDEs that offers code completion, test generation, documentation writing, error explanation, and flexible local/cloud model support.
Replit Ghostwriter/Agent (Replit)	A browser-based AI pair programmer integrated into the Replit platform, excelling at in-line code suggestions, explanations, refactoring, and full end-to-end app generation from natural language prompts.
AI-Native IDEs (Cursor, Windsurf, Zed)	IDEs built around AI-first workflows with whole-codebase context (Enable fast refactoring, cross-file reasoning, and structured edits; require adoption of new editor environments).
CLI / Agent-Based Coding Assistants (Aider, Claude Code, Kilo Code)	Command-line or plugin-based assistants that apply changes across repositories (Powerful for multi-file refactors, test generation, and scripted code modifications; geared toward advanced users).
Devin (Cognition AI)	Autonomous coding agent capable of planning, coding, testing, and debugging tasks end-to-end (Represents a shift toward agentic software development; still emerging and best suited to well-scoped tasks).
Open-Source Models/Tools (e.g., Code Llama, Tabby, Continue, OpenDevin)	Highly customizable AI that runs locally or on self-hosted infrastructure. (Offers low/no cost, data privacy control, and flexibility for specific, custom requirements).

Table B: **Selected General-Purpose AI Coding Assistant Tools and Models**

C. Checklist for Evaluating LLM Suitability in Quantitative Research Use-Case

This exhibit provides a structured checklist for assessing whether a generative large language model (LLM) is appropriate for a quantitative finance application. The checklist is designed to identify methodological risks documented in recent empirical studies, including temporal contamination, memorisation artefacts, numerical fragility, behavioural biases, and reproducibility constraints. Researchers can use this framework to determine whether an LLM is reliable for a specific task or whether a simpler, deterministic model is preferable.

1. Temporal Integrity and Look-Ahead Bias

- ○ Verify that the task lies strictly beyond the model's knowledge cut-off.
- ○ Ensure that no post-event text appears in the input.
- ○ Test for leakage by shifting dates or using artificial cut-offs.
- ○ Check whether anonymised inputs still produce event- or firm-specific predictions.

2. Memorisation and Data Reconstruction

- ○ Mask firm names, dates, and identifiers to assess whether the model reconstructs them.
- ○ Add small perturbations to key inputs (e.g. least significant digit changes) to detect reliance on memorised values.
- ○ Compare behaviour when inputs are partially falsified or intentionally misaligned.

3. Numerical Reasoning Capability

- ○ Evaluate whether the task requires arithmetic, aggregation, or ratio computation.
- ○ Test on basic accounting questions (e.g. line-item tallies, identity checks).
- ○ Assess performance with isolated numbers versus full financial statements.
- ○ Determine whether accuracy depends on external tools (e.g. code interpreters), indicating weak native numerical reasoning.

4. Intrinsic and Behavioural Biases

- ○ Examine whether the model displays systematic preferences for specific sectors (e.g. Technology) or size categories (e.g. large-cap equities).
- ○ Test for contrarian or momentum-related biases.
- ○ Present conflicting evidence and observe whether the model updates beliefs; lack of adjustment indicates confirmation bias.

5. Foreign and Cross-Border Bias

- ○ Compare predictions across firms from different geographic regions.
- ○ Assess for systematic optimism or pessimism linked to asymmetric information availability.
- ○ Provide region-specific text (e.g. local news sources) to test whether biases disappear when information gaps are closed.

6. Robustness and Perturbation Sensitivity

- ○ Re-run identical prompts to check for stability across executions.
- ○ Modify prompt phrasing to assess prompt sensitivity.
- ○ Introduce minor semantic or numerical noise and evaluate consistency of outputs.

5. Foreign and Cross-Border Bias

- ○ Compare predictions across firms from different geographic regions.
- ○ Assess for systematic optimism or pessimism linked to asymmetric information availability.
- ○ Provide region-specific text (e.g. local news sources) to test whether biases disappear when information gaps are closed.

6. Robustness and Perturbation Sensitivity

- ○ Re-run identical prompts to check for stability across executions.
- ○ Modify prompt phrasing to assess prompt sensitivity.
- ○ Introduce minor semantic or numerical noise and evaluate consistency of outputs.

7. Reproducibility and Model Stability

- ○ Determine whether the LLM is accessed via API or as a fixed local model.
- ○ Assess the extent to which the model version is subject to unannounced provider updates.
- ○ Check whether identical queries yield stable outputs across time.
- ○ Prefer local, fixed-weight open-source models where reproducibility is essential.

8. Alignment with Task Requirements

- ○ Use LLMs when the task requires contextual interpretation, cross-document synthesis, long-range dependencies, or narrative reasoning.
- ○ Prefer smaller specialised models (e.g. FinBERT, RoBERTa classifiers, GBDTs) for deterministic tasks such as sentiment classification, tabular prediction, and accounting-based signal extraction.
- ○ Compare performance gains to simpler baselines; marginal improvements do not justify increased complexity.

Decision Rule

An LLM is appropriate only when it delivers substantial, demonstrable improvements out-of-sample**,** with no evidence of temporal leakage, memorisation artefacts, systematic bias, or numerical fragility. In all other cases, particularly when the task is deterministic, arithmetic, or classification-based, simpler, task-specific models remain the more reliable and principled choice.

Table C: LLM Evaluation Checklist