Review

GenAI and LLM for Financial Institutions: A Corporate Strategic Survey

Jun Xu [1,*]

[1] Executive Director of Machine Learning Engineering, A top international bank
* Correspondence: xujun@ieee.org;

Abstract: The integration of Generative Artificial Intelligence (GenAI) and Large Language Models (LLMs) is transforming the landscape of financial institutions (FIs), offering unprecedented opportunities alongside significant challenges. Different from the existing surveys on GenAI and LLMs in financial domain, this paper provides a comprehensive discussion of the strategic adoption of LLMs within FIs, examining their potential to revolutionize various facets of the industry. We begin with an introduction to LLMs in financial contexts, detailing the dual aspects of opportunities — such as enhanced customer service, improved risk assessment, and optimized operational efficiency—and challenges, including data privacy concerns, regulatory compliance, and the need for robust risk management frameworks. Our methodology section delves into the lifecycle and major components of LLMs and the operational practices of LLMOps, highlighting best practices in development, deployment, and maintenance to ensure both efficacy and compliance. We further explore a variety of LLM applications, from personalized banking services and fraud detection to complex regulatory compliance and investment strategies. Finally, the paper outlines strategic implementation guidelines, emphasizing a phased approach that includes stakeholder engagement, rigorous testing, and continuous learning cycles. This survey aims to equip financial leaders and policymakers with the insights needed to navigate the complexities of adopting LLM technologies, ultimately fostering informed decision-making for successful integration.

Keywords: AI; LLM, machine learning; Sustainable Finance; Financial institutions; Banking

1 Introduction

The financial sector has always been at the forefront of adopting innovative technologies to enhance efficiency, accuracy, and customer service satisfaction [1]. With the advent of artificial intelligence (AI), particularly in the form of Generative AI (GenAI) and large language models (LLMs) [1], the industry is witnessing a transformative shift in its operations and services. LLMs like OpenAI's GPT-3.5/4/4o/o1, Google Gemini, and Anthropic Claude have demonstrated remarkable capabilities in understanding, generating, and reasoning about natural languages, which can be leveraged across various financial applications. A discernible trend is evident: GenAI is swiftly progressing from experimental technology to a critical business asset. McKinsey's research indicates a 65% adoption rate of GenAI, underscoring its rapid integration into

[1] GenAI is an extensive domain that includes various techniques and models capable of producing new data, including text, images, audio, or code. In contrast, LLMs are a specialized subset of generative AI that focus on comprehending and generating human-like language. In this paper, we focus on LLMs, while discussing GenAI in general.

financial services [2]. Additionally, Deloitte's findings highlight substantial organizational benefits derived from GenAI, particularly from LLM technologies [3].

LLMs are sophisticated AI systems trained on extensive datasets, enabling them to understand and generate human-like text. These models utilize deep learning techniques and neural network architectures to process and analyze vast quantities of unstructured data, including books, papers, news, and web content. The development of LLMs has been propelled by enhanced computational capabilities, the availability of expansive datasets, and innovations in neural network architectures [1] . While the earlier models such as BERT, T5, and GPT-2/3 also employed the foundational Transformer architecture, the performance improvements in more recent LLMs have sparked considerable debate. Specifically, these enhancements have raised concerns about risks, prompting discussions about their implications, especially in traditionally risk-averse sectors like finance. For financial institutions (FIs), however, the question on LLMs is never to be "use or not use". Known for their cautious approach to risk and leaders in adopting this new wave of GenAI and LLM technologies, the fundamental question is always "how to use LLMs legally, efficiently, and safely", or in other words, "what not to do" to prioritize the applications.

Despite the enthusiasm for GenAI, many FIs face challenges in developing robust and sustainable GenAI solutions. The complexity of transitioning from a basic prototype to a fully functional, deployable application cannot be underestimated. Specific use cases require close attention to detail; for instance, while internal applications may accommodate occasional inaccuracies known as "AI hallucinations," customer-facing applications demand higher standards of accuracy to maintain trust and reliability. Major obstacles, including trust and risk management, accuracy, and the reliability of AI outputs, stand as significant barriers to widespread adoption. Moreover, issues such as data quality and privacy, model interpretability, and overall system reliability continue to pose substantial challenges. Different from the existing survey papers on LLM, e.g., [2] [3] [4] [5] [6] [7] [8] [9] [10] [11], this article focuses more on the practical respective of LLM adaptation in FIs. We leverage insights from industry surveys and expert discussions to delve into these crucial issues, offering AI teams and entrepreneurs an in-depth perspective on the current landscape and the hurdles that must be overcome, specifically for FIs. After further discussing the LLM opportunities and challenges, the LLM foundation will be reviewed at Section 2 for self-contained purpose. Section 3 then summarizes the typical LLM applications in financial industry. In order to implement these applications, we have to consider two major components, data and model, which are discussed in Sections 4 and 5, respectively. Finally, we conclude our findings and propose a strategic approach for LLM integration into FIs' operating process.

1.1 LLM Motivations, Impacts and Opportunities

In the intricate and ever-evolving landscape of the finance sector, the integration of LLMs and other AI technologies is not just beneficial but crucial. The capacity of LLMs to process and understand vast amounts of unstructured financial data and regulatory documents is indispensable in addressing the multifarious challenges and seizing the opportunities within this domain. For example, LLMs can analyze extensive financial reports and market data, offering FIs insights to evaluate investment risks and

[2] McKinsey & Company, The state of AI in early 2024: Gen AI adoption spikes and starts to generate value, https://www.mckinsey.com/capabilities/quantumblack/our-insights/the-state-of-ai, May 2024, Retrieved 20/07/2024

[3] J. B. Ucuzoglue and D. Dutt, New Deloitte survey finds expectations for Gen AI remain high, but many are feeling pressure to quickly realize value while managing risks, https://www.deloitte.com/global/en/about/press-room/gen-ai-survey.html, Jan 2024. Retrieved at 20/07/2024

opportunities with unprecedented depth and precision. This analytical prowess supports more informed decision-making in investment strategies, enhancing portfolio performance and risk management. For example, the study in [12] evaluates ChatGPT and GPT-4's financial reasoning using CFA exam questions, exploring their impressive performance in Zero-Shot, Chain-of-Thought, and Few-Shot scenarios. Another example on the financial sentiment analysis (FSA) in [13] demonstrates how LLMs, through a novel design framework based on Minsky's theory, can enhance FSA without fine-tuning. By leveraging heterogeneous LLM agents and their discussions, it achieves superior accuracy on FSA tasks, challenging traditional data-heavy approaches and suggesting new directions for applying LLMs in business and management contexts.

Moreover, in terms of regulatory compliance and fraud detection, LLMs can sift through complex datasets to identify patterns indicative of non-compliance or fraudulent activities. This capability allows FIs to proactively address potential regulatory issues and minimize financial losses due to fraud. Additionally, LLMs can enhance customer experience and service personalization in finance by analyzing customer behavior and preferences, thereby enabling FIs to offer customized financial products and advice. To expand on the additional benefits of adopting LLMs besides the existing AI framework within enterprise environments, we can delve into each point more thoroughly:

1. Enhanced Productivity and Streamlined Operations:

LLMs can significantly enhance productivity by automating routine tasks that traditionally require human linguistic input, such as drafting emails, generating reports, and creating statements. By automating these tasks, employees can focus on more complex and strategic activities. Due to the strong emerging and reasoning capability, it also makes the development work simpler, e.g., project planning and assistant coding. Meanwhile, streamlining operations also refers to the ability of LLMs to integrate and synthesize information across various platforms, leading to more efficient workflow management. For example, LLMs can be programmed to extract key information from large volumes of data, summarize insights, and even suggest actions, thereby speeding up decision-making processes. As a programming and planning assistant, it also improves the product quality and innovation. This shift not only reduces operational costs but also improves employee satisfaction by focusing on higher-value activities.

2. Potential to Create New Value for Customers and Stakeholders:

LLMs can offer new ways to engage with customers and personalize interactions at scale with high personality. For instance, they can power advanced chatbots and virtual assistants that provide immediate, high-quality customer service across multiple languages and time zones. This capability not only enhances customer experience but also opens up opportunities for enterprises to expand into new markets more seamlessly. Moreover, LLMs can analyze customer feedback and interaction data to generate insights into consumer behavior, thereby helping businesses tailor their offerings more precisely and effectively.

3. Improvement in Efficiency Across Sectors, Including Business and Governance:

In the business sector, LLMs can streamline operations by automating complex processes such as supply chain communications and financial forecasting. Their ability to process and analyze large volumes of data with high accuracy enables businesses to make more informed decisions, reduce errors, and enhance overall operational efficiency. In governance, LLMs can play a crucial role in improving regulatory compliance and public service delivery. By automating routine tasks such as responding to both internal feedback/paperwork and common public inquiries, LLMs can reduce the administrative burden on government agencies. This not only speeds up service delivery but also

increases accessibility and responsiveness, ensuring that citizens receive timely and accurate information. Ultimately, the integration of LLMs into governance processes can lead to more efficient and effective public services, benefiting both government operations and the public at large.

4. Transformational Potentials in Addressing Challenges in Underserved Regions and Domains:

In regions where resources are scarce, LLMs can provide substantial benefits. LLMs are transforming Human Resources (HR) by automating and enhancing tasks such as resume screening, employee onboarding, and personalized training. The integration of LLMs enables HR professionals to efficiently handle large volumes of inquiries and documents, ensuring consistency and accuracy in communications and processes. Risk and Compliance department can reduce much of their manual documentation processing works and proactively understand the regulation changes and detect the internal potential risks. In education, LLMs can offer personalized learning experiences and support to students in remote or underserved areas by providing tutoring or language translation services. This can help bridge educational gaps and bring high-quality educational resources to areas where traditional educational infrastructure is lacking.

Overall, the integration of LLMs into the financial sector empowers FIs to navigate the complex financial landscape with greater accuracy, efficiency, and strategic insight. It fosters a more sophisticated understanding of financial risks and opportunities, paving the way for more innovative and customer-centric financial services. The adoption of LLMs in enterprises holds the promise not only to enhance operational efficiencies and stakeholder value but also to drive significant social and economic improvements, particularly in areas most in need of innovative solutions. Thus, LLMs become an indispensable component of FIs' digitalization by enhancing customer interactions, automating complex document processing, and providing insights through advanced data analysis. In an era where the pace of change in the financial sector is accelerating, leveraging LLMs and AI becomes a strategic imperative to stay competitive and meet the evolving needs of the market. Section 3 will delve into more specific examples and case studies.

1.2 LLM Challenges

Despite the promising opportunities offered by LLMs, their adoption in financial institutions is not a straightforward process. To better understand the challenges involved, we will explore them from multiple perspectives, providing a more detailed explanation of the challenges these institutions face in integrating LLMs.

First, from **technical** point of view, the complexity of LLMs bring the significant barriers for adoptions.

Complexity of the Technology. LLMs are advanced pieces of AI technology that rely on vast amounts of data and sophisticated algorithms to function. They necessitate a deep understanding of both machine learning principles and the specific mechanics of NLP. This complexity requires potential adopters to have or develop expertise in AI and machine learning, and to commit to ongoing learning and adaptation as the technology evolves. This can be a significant barrier for organizations without the resources to invest in specialized knowledge or training. That means the integration of these advanced AI technologies requires a thorough understanding of evolving AI terminologies and frameworks, demanding ongoing learning and adaptability from enterprise teams. This leads to lack of explicability of the Model, which is required by some applications, although (partial) explainability is possible. This dynamic landscape

of AI development marks a significant shift from traditional models, pushing organizations to constantly update their knowledge and tools.

Hallucination and Algorithmic Bias: LLMs demonstrate remarkable capabilities in recognizing and generating language patterns. However, these models often lack a grounded understanding of factual content, leading to errors known as "hallucinations" where outputs appear plausible but are not factually accurate. This issue is particularly critical in enterprise applications that demand high accuracy. Furthermore, LLMs can inherit and potentially amplify existing biases from their training data, resulting in outputs that may be unfair or harmful.[4] This leads to the concern of reliability for LLMs.

- Hallucinations in LLM Outputs: As one of the major concerns from practitioners and regulators, LLMs are susceptible to generating hallucinations, which are outputs that, while seemingly credible, do not align with factual reality. This tendency can undermine their utility in contexts where precision and reliability are paramount. In response to this challenge, industries are emphasizing the importance of rigorous data curation and model training processes. For example, LLMs are directed to derive answers from specifically curated, high-quality financial documents, such as quarterly filings and earnings transcripts, as well as from real-time data, which results in more accurate and reliable outputs.[5] Another method is Retrieval-Augmented Generation (RAG), which will be discussed later. Some recent studies show the unlearning technique is also helpful [6].
- Algorithmic Bias in LLMs: Bias in LLMs can manifest in various forms, including racial, social, and recency biases—the latter influencing the model's outputs based on the most recent data it has processed. To combat these biases, it is beneficial to combine LLMs with data models and rule engines that provide a balanced view and contextual grounding. This integration helps in diluting the skewed perspectives that LLMs might develop from biased training datasets and enhances the fairness and applicability of their responses. See more bias in Table 1. Another potential issue is the drift, which is listed in Table 2 [7].

[4] The data quality issue is crucial for all AI models, including LLMs. While LLMs, like earlier NLP models, are susceptible to the "rubbish in, rubbish out" problem, they differ by producing more convincing and natural-sounding content, even when it's factually incorrect. This poses a significant risk, as individuals without specialized knowledge may be misled by these erroneous yet plausible outputs.

[5] In the paper [66], the authors further describe bullshit as a disregard for the truth, where the speaker is indifferent to the accuracy of their statements. This is contrasted with hallucinations, which imply a false perception or belief about reality. By distinguishing between “hallucinations” and “bullshit” in LLM outputs, developers can gain a clearer understanding of the types of inaccuracies they are dealing with and devise targeted solutions. Note that LLM generally has the limitations in incorporating real-time or dynamic information.

[6] Unlearning in LLMs is a critical capability that allows for the selective removal or modification of specific information within a pre-trained model. This process is essential for enhancing the reliability, security, and adaptability of AI systems. The benefits of unlearning for LLM performance include:

- Privacy and Ethics: Ensures compliance by removing sensitive data.
- Data Accuracy: Updates models to eliminate outdated information.
- Bias Mitigation: Reduces discriminatory outputs by eliminating biased data.
- Adaptability: Allows models to adjust to regulatory and data changes.
- Efficiency: Provides a less resource-intensive alternative to full retraining.
- Customization: Tailors models to specific needs by focusing on relevant data.
- Robustness: Improves model reliability by refining the training data set.

[7] Bias refers to systematic errors in an AI model caused by inherent flaws in the training data or algorithms, leading to unfair or incorrect outputs. Drift, on the other hand, occurs when external changes over time, such as shifts in data distribution or user behavior, cause the model's performance to degrade because it no longer aligns with the current environment. In essence, bias is

Type of Bias	Description	Potential Solutions
Data Biases	**Encompasses biases arising from the training data**, such as non-representative samples, overrepresentation of certain groups, temporal imbalances, or sentiment skew. This includes: - Sample Bias / Selection Bias - Availability Bias - Frequency Bias - Exposure Bias - Recency Bias - Positivity/Negativity Bias	- Use diverse and representative datasets. - Balance data to include underrepresented groups and rare instances. - Regularly update training data to reflect current and varied information. - Apply data augmentation and resampling techniques.
Confirmation Biases	Biases where the model or users favor information that confirms existing beliefs, ignoring contradictory evidence. This includes: - Confirmation Bias - Confirmation-of-Common-Knowledge Bias	- Implement mechanisms to challenge assumptions. - Use counterfactual reasoning and adversarial examples. - Regularly update models with new insights. - Encourage critical evaluation of outputs.
Social and Demographic Biases	Biases stemming from stereotypes or prejudices related to social groups, demographics, cultures, or languages, leading to unfair or discriminatory outputs. This includes: - Group Attribution Bias - Social Bias - Implicit Bias - Relational Bias - Gender Bias - Age Bias - Economic/Socioeconomic Bias - Cultural Bias - Geographic Bias - Linguistic Bias	- Include diverse and balanced data representing various social and demographic groups. - Use fairness and debiasing techniques in training and post-processing. - Regularly audit model outputs for biases. - Apply fairness constraints and metrics.
Cognitive Biases	Biases resulting from the model's processing strategies, such as overreliance on initial information or failure to correctly interpret context. This includes: - Anchoring Bias - Contextual Bias	- Design models to consider the full context. - Implement mechanisms to reduce overreliance on initial inputs. - Improve contextual understanding through advanced training methods. - Use reinforcement learning from human feedback (RLHF).
Automation Bias	The tendency of users to overtrust AI-generated outputs without critical evaluation, potentially accepting incorrect or biased results.	- Encourage model transparency and explainability. - Educate users to critically assess AI outputs. - Implement confidence scores or uncertainty indicators to prompt users to verify results. - Provide clear disclaimers and guidance.

Table 1 LLM Bias

Type of Drift	Description	Potential Solution

about internal issues within the model's design or data, while drift is about the model's declining relevance due to evolving external conditions.

Temporal Drift	The knowledge base becomes outdated as new information emerges.	Regularly update the external knowledge sources.
Contextual Drift	The model fails to retrieve relevant information for specific contexts or domains. Domain difference also falls in this category.	Improve context-awareness in retrieval mechanisms. Regularly monitor the stability and resiliency metrics of the pipeline.
Semantic Drift	The meaning or usage of words and phrases changes over time.	Incorporate mechanisms to adapt to linguistic changes.
Data Distribution Drift	The distribution of data the model was trained on differs from the distribution of data it encounters in deployment.	Use techniques like domain adaptation and continuous learning.
Quality Drift	The quality of information in the knowledge base degrades, possibly due to the accumulation of errors or low-quality inputs.	Implement quality control measures and feedback loops for the knowledge base.
Model / Algorithmic Drift	Changes in the model's algorithms or parameters over time can lead to performance changes. Or Over time, the pre-trained language model (LLM) used in RAG may experience drift due to changes in its training data or fine-tuning process.	Monitor model performance and adjust algorithms as needed.
User Behavior	User behavior can evolve, leading to different patterns of interaction with the RAG system. This can cause drift in the types of queries and context provided.	Regularly analyze user interactions and adapt the RAG system to evolving user needs. Consider retraining the model based on recent user behavior
Pipeline Drift	The pipeline may contain multiple steps of processes, e.g., data parse, text chunk, query intent detection, vector similarity and re-ranking, and prompt engineering, which may cause the drift	Apply internal efficacy metrics on an ongoing basis. These metrics could be about context efficacy, anti-hallucination scores, answer relevancy, racial bias scores, and so on.

Table 2 LLM drifts

Second, from a strategic point of view, LLMs face significant challenges when transforming the operational models of financial institutions.

Mismatch with Enterprise Planning Cycles: The field of AI, particularly technologies like LLMs, is advancing at a rapid pace. However, traditional enterprise planning and budget cycles often move more slowly, usually structured on annual or biennial bases. This discrepancy can lead to difficulties in aligning the fast-evolving capabilities and applications of LLMs with the strategic planning and investment timelines of traditional enterprises. Organizations may find it challenging to keep up with technological advancements, risking obsolescence or missed opportunities if they cannot adjust their planning mechanisms to be more flexible and responsive to technological change.

Strategic Technology Choices and Technology Infrastructure: Enterprises face critical strategic decisions in choosing the appropriate operational infrastructure, such as opting between on-premise, cloud-based, or AI-as-a-Service models. Each choice carries its own set of cost implications and operational impacts, making it a pivotal decision that can greatly influence the efficiency and success of LLM implementation. Additionally, the market currently experiences a scarcity of robust, enterprise-grade LLM solutions that can seamlessly integrate into existing business processes. This scarcity is not due to a lack of interest but rather reflects the complex and novel nature of LLM technologies.

Need for Customization: Each enterprise has unique needs and use cases for technology like LLMs. For LLMs to be effectively integrated into business processes, they must be customized to fit specific organizational requirements. This might involve tailoring the model's training process to understand industry-specific jargon, configuring the model to interface with existing IT infrastructure, or even ensuring compliance with industry regulations regarding data use. Customization is crucial for maximizing the utility of LLMs but requires additional time, expertise, and resources, adding to the complexity of their adoption.

Productivity Paradox: Just as early computer adoption didn't immediately lead to productivity gains, the introduction of AI and LLMs may not instantaneously reflect in improved productivity statistics. The real benefits of AI might only manifest after significant adjustments in business processes and workforce skills, highlighting the need for a strategic approach to AI integration. In addition, the ROI methodology may be revisited accordingly based on reality.

Third, the **compliance, regulation and governance** challenges may slow down the adoption, although they provide a safer environment.

Regulation and Governance: Regulatory bodies worldwide are working on frameworks to govern AI. The EU's AI Act, the U.S. AI Bill of Rights, Canada's Draft Artificial Intelligence and Data Act, Singapore's Model AI Governance Framework for GenAI and the U.K.'s AI White Paper all signal a move towards greater accountability and oversight in AI. Enterprises must adapt to these changes to avoid penalties and ensure compliance.

Control Challenges: As AI systems become more autonomous, the need for robust control measures increases. Key aspects include steerability (directing AI towards desired outcomes), alignment (ensuring AI objectives match organizational goals and regulatory requirements), interpretability (understanding AI decision-making), observability (monitoring AI impact and compliance), and testability (evaluating AI performance under various conditions).

Data management and privacy also present significant challenges. Effective knowledge management requires the creation and maintenance of up-to-date, consistent knowledge bases, yet managing these with the required level of data privacy is complex. Enterprises must navigate the risks of data poisoning and ensure robust data security and intellectual property, particularly when sensitive information is involved. Harmful or toxic contents can be inadvertently generated due to biased training data, and potentially perpetuate stereotypes or spreading misinformation. Thus, they shall be proactively processed with proper safeguards.

Integration and interoperability issues arise as LLMs must be woven into existing enterprise systems, a process that demands a deep understanding of system architectures and API interactions. This integration must support diverse functionalities while maintaining system reliability and addressing potential security vulnerabilities.

Security risks are another major concern with LLM adoption. New forms of cybersecurity threats specific to AI technologies, such as adversarial attacks, malicious prompt injections, insecure output handling, model denial of services, orchestration vulnerabilities, excessive agency, model theft and other sophisticated social engineering tactics, necessitate advanced security measures. These AI systems, capable of generating human-like text, could potentially manipulate users or breach sensitive data, underscoring the need for heightened cybersecurity vigilance.

Last but not least, from a **human and organizational** perspective, the challenges involve deep implications for human interactions and the overall workplace dynamics.

Depersonalization of Human Connection: As LLM technologies strive to perfect human communication, they sometimes risk creating an uncanny valley effect, where the near-human correctness of AI responses feels unsettling and impersonal. This phenomenon may alienate customers and employees alike, disrupting the authentic

nature of human interactions and potentially harming brand perception and workplace morale. FIs may have to re-design their customer trust and loyalty structure and customer engagement experiences to meet the satisfaction level.

Emotional and Social Impact: The integration of AI into the workplace could fundamentally alter the human work experience, introducing new sources of stress and anxiety. Employees might feel surveilled and judged by omnipresent AI systems analyzing their communications, potentially leading to discomfort and resistance. Additionally, the evolving roles in AI-human interactions could shift power dynamics, making some employees feel subordinate to AI systems, which could lead to emotional and psychological discomfort.

Over-Dependence and Complexity: An increasing reliance on AI for decision-making and problem-solving could atrophy critical human cognitive abilities. As business operations become more AI-driven, employees may find it challenging to navigate an increasingly complex and unfamiliar work environment. This over-dependence on AI risks making the workplace feel impersonal and daunting, potentially leading to a disconnect between employees and their roles.

Shadow AI and Bring Your Own AI (BYOAI): Similar to the concept of Shadow IT, Shadow AI refers to unsanctioned AI initiatives within organizations that could lead to security risks and compliance issues. BYOAI involves employees bringing personal AI tools to work, which while enhancing productivity, also pose significant security risks and challenge corporate governance.

These challenges suggest that for enterprises to successfully adopt and benefit from LLMs, they must not only understand and manage the technological complexity but also adapt their strategic approaches and resource allocation to the dynamic nature of AI development.

2 Theoretical Framework of LLMs

This section serves as an introduction to LLM techniques, aiming to familiarize readers with the fundamental concepts necessary for understanding their applications and to make this article self-contained.

LLMs utilize several core techniques to process, train and generate human-like text. They are primarily built on deep learning architectures, notably and mostly transformer models [8], which enable them to understand and generate text based on patterns learned from vast datasets. Key techniques include attention mechanisms that help the model focus on relevant parts of the input data, fine-tuning on specific tasks to enhance performance, and transfer learning, which allows a pre-trained model to adapt to new tasks with minimal additional training. Additionally, alignment techniques, e.g., Reinforcement Learning from Human Feedback (RLHF), is used for ensuring that the

[8] Although most LLMs are based on transformers models nowadays, there are few other structures, e.g., variational autoencoder (VAE), generative adversarial network (GAN), etc. There are also some new and promising techniques. Mamba, a linear time sequence modeling paradigm with Selective State Spaces (SSMs), is not only garnering attention for its impressive empirical results but also for its unique design [69]. The hybrid Transformer-Mamba structure with a mixture-of-export, Jamba, can deliver high throughput on long context and show competitive quality on some popular benchmarks compared with transformer-based models [70]. Test-Time Training (TTT) layer is another technique declaimed that they can not only process far more data than transformers, but that they can do so without consuming nearly as much compute power [71]. Kolmogorov-Arnold Network (KAN) tries to change the network structure, which is characterized by four main features: activation functions are located on the "edges" rather than the nodes, these activation functions are learnable instead of fixed, nonlinear kernel functions can replace the linear functions on MLP edges, and fine-grained knots can be set to improve approximation accuracy [67]. The recent Liquid Foundation Models (LFM) claimed a better performance than GPT structure with less memory usage in small size models, e.g., 1B and 3B. For image generation, diffusion models are one of the most popular structures.

model's outputs align closely with human values and preferences. These techniques collectively empower LLMs to handle complex language tasks with remarkable effectiveness.

Given the substantial costs associated with training a LLM from scratch, which is usually in the level of millions of dollars, FIs often utilize pre-trained models developed by leading technology companies and major research institutions and universities. In the deployment of these LLMs for domain-specific applications, techniques such as prompt-engineering, model fine-tuning with smaller datasets, Retrieval-Augmented Generation (RAG) and agents are commonly employed to tailor the models' outputs to specific needs. The subsequent sections will provide a detailed discussion of these techniques to provide an LLM foundation.

2.1 General concepts

2.1.1 Pre-trained models

Pre-trained models represent a significant advancement in the field of AI, particularly in NLP and Computer Vision (CV). These models are trained on vast amounts of data before being fine-tuned for specific tasks, allowing them to capture a deep understanding of language structure and semantics. The developers usually need to collect and clean a large amount of data, and choose a proper tokenizer to segment the text into token sequence for modeling. Recently, subword tokenizers are widely used, e.g., Byte-Pair Encoding (BPE), WordPiece and Unigram.

The next step is to choose a suitable model architecture. Transformers have become the backbone of many state-of-the-art LLM models, offering superior performance and flexibility in handling various language tasks. Table 2 compares the typical transformers' structure: encoder-only, decoder-only and encoder-decoder. Note that decoder-only structure can be further differentiated as Causal Decoder and Prefix Decoder. The former uses unidirectional (only past tokens) attention mechanism, while the latter applies bidirectional on prefix, unidirectional on generated tokens and uses shared parameters for prefix and output generation.[9]

The training step is usually classified three types of tasks. The first task of language modeling (LM) predicts the next token based on a given sequence of tokens. The second task of denoising autoencoding (DAE) tries to recover the relaced or deleted tokens from a piece of tokens. Usually, DAE is more complex than LM with additional optimization policies, e.g., token replacement, segment size, replacement ratio, etc. The third task of mixture-of-denoiser (MoD) unifies LM and DAE as a denoising process. Note that some models combine multiple pre-training objectives, such as token masking, deletion, and permutation, to enhance performance. Other scalable training techniques include 3D parallel training (i.e., data, pipeline and tensor parallelism), zero redundancy optimizer (ZeRo by DeepSpeed library, or fully shared data parallel, FSDP by PyTorch; reduce duplication of model parameters and optimizer states across GPUs), activation re-computation (gradient checkpointing to optimize the memory usage of backpropagation), mixed precision training (use 16-bit (FP16) and 32-bit (FP32) floating-point precision at the same time), and etc.

The next steps are instruction tuning and alignment. The former is also named supervised fine-tuning (SFT) or multitask prompted training. This process first requires collecting or constructing instruction-based instances, then fine-tuning the parameters of the large language model through a supervised approach. After instruction tuning, the

[9] Commonly, we might only consider causal models. The Prefix encoder/decoder models, e.g., UniLM, XLM and GLM, might be also called as Unified Transformer Models, as their encoder-only language models also leverage the advantages of auto-regressive (decoder) models for model training and inference. For example, UniLM actually uses three types of modeling tasks: unidirectional, bidirectional, and sequence-to-sequence prediction. [68]

LLM can demonstrate strong instruction-following abilities and can address various downstream tasks through zero-shot learning. The latter ensures that the behavior of LLM aligns with human values, genuine human intentions, and social ethics, which is also crucial. For FIs, it usually requires some financial-specific data (See Table 3) to enhance the model's professional capabilities and security in the financial domain to meet specific values and objectives. A common approach to achieve this alignment is RLHF [10]. This technique involves collecting human feedback, training reward models, and developing reinforcement learning training strategies to guide the model's behavior. Additionally, there are non-reinforcement learning methods for alignment, which utilize high-quality alignment datasets and specific supervised learning algorithms to fine-tune the model. One such algorithm is Direct Preference Optimization (DPO), which establishes a relationship between the decision function and the reward function within the objective function of reinforcement learning, thereby avoiding the process of reward modeling; instead, uses human preference data to fine-tune the model directly.

Model type	Definition	Typical models	Scenarios
Encoder only	Focuses on understanding and processing input text to extract meaningful representations.	Masked LM: BERT, RoBERTa, ALBERT Prefix LM: UniLM, XLNet	Primarily used for feature extraction or dimensionality reduction in tasks such as image compression, anomaly detection, and as a preprocessing step before feeding data into another model.
Decoder only	Generates output sequences by predicting the next token based on previous tokens. Typically starts with a learned initial state or a prompt and then produces an output sequence.	Left-to-Right LM: GPT, GPT-2, GPT-3, Llama, BLOOM Prefix Decoder: GLM130B, U-PaLM	Commonly used in tasks that require sequence generation from an internal state, such as text generation, language modeling, and some types of time series forecasting.
Encoder-decoder	The encoder processes an input sequence and encodes it into a context vector, which the decoder then uses to generate an output sequence, often of a different length. Sequence-to-sequence.	T5, MASS, BART, mT5, MarianMT, FLAN-T5	Ideal for tasks that involve mapping an input sequence to an output sequence, such as machine translation, chatbots, question-answering systems, and summarization.

Table 3 Typical Transformer structural comparison

Type of data	Explanation
Financial Knowledge	Introduce financial data and cases to strengthen the model's understanding and mastery of financial knowledge.
Financial Capabilities	Construct task-specific data (such as financial forecasting, risk assessment, etc.) to improve capabilities in financial scenarios
Financial Security	Reinforce data security and privacy capabilities through human preference learning, ensuring safety and compliance when processing financial data

Table 4 Data for model alignment

[10] RLHF is used for GPT-3.5/4, but only RL is applied in GPT-o1 for automation.

2.1.2 *Prompt Engineering*

An intuitive definition of prompt engineering is that "Prompt is a cue given to the pre-trained language model to allow it better understand human's questions.", while a More Technical Definition can be stated as "Prompt is the technique of making better use of the knowledge from the pre-trained model by adding additional texts to the input."[11]

From the magic simple prompts "Take a deep breath and work on this problem step-by-step" and "Calm yourself down" to the complex graph-of-thoughts, there are many types of prompts techniques. Table 4 lists some typical methods, e.g., zero-shot, few-shot and template-based methods. [12] We can choose the proper prompt-based tool learning for different scenarios. For example, in the stock price query task, our aim is to equip the model with the capability to use external tools, e.g., Alpha Vantage API for market information. After employing a prompt-based learning technique, we can assess the model's ability in generating the correct Python function call for a tailored wrapper of APIs based on given natural language instructions. The model is briefed about the function parameters, encompassing the ticker, date, and price type.

Methods	Description
Zero-shot Prompting	Asks the model to perform a task without providing specific examples.
One/Few-shot Prompting	Provides the model with one or a few examples of the desired input-output pairs within the prompt.
Self-criticism	Instructs the model to identify and correct self-contradictions in its output.
Confidence-based Answering	Instructs the model to refuse answering if not confident, improving accuracy.
Chain-of-Thought (CoT)	Encourages the model to break down complex problems into step-by-step reasoning. This technique is extended to Tree-of-thought, graph-of-though.
Role-playing Prompts	Asks the model to assume a specific role or persona when responding.
Anchoring	Provides additional context or cues to steer the model's behavior in a specific direction.
Pattern-based Prompting	Uses specific patterns or templates in prompts to elicit desired outputs, especially for tasks requiring background knowledge, e.g., CO-STAR.
Prompt pipeline	Combines a series of connected stages, each stage with a specific function, from refining a user's request to executing the perfect query for the LLM to process.

Table 5 Prompt methods

When Prompt Engineering is a broader concept that includes the strategic design and optimization of prompts as part of a larger framework for working with LLMs, another concept, prompt tuning, is more about making specific adjustments to the wording of the prompt to improve the model's output. [13] See the comparison in Table 5.

Aspect	Prompt Engineering	Prompt Tuning
Definition	The process of finding the most suitable template for each task, manually or algorithmically.	A technique to mold downstream tasks to fit language models (LMs) without altering the LM itself.
Prompt Shape	Considers the placement of [z] in the template, choosing between cloze (middle) or prefix (end) prompts based on the task	Implies a flexibility in integrating prompts with LMs.

[11] Pengfei Liu, The Fourth Paradigm of Modern Natural Language Processing Techniques, https://blender.cs.illinois.edu/course/fall22/lecture9.pdf, Carnegie Mellon University. Accessed at 7/07/2024.

[12] The importance of prompt engineering for the end-users might be lower, when the LLM itself contains some "reflection" mechanism. For example, GPT-o1 doesn't suggest a complex prompt. Instead, a straight and clear instruction might be more helpful.

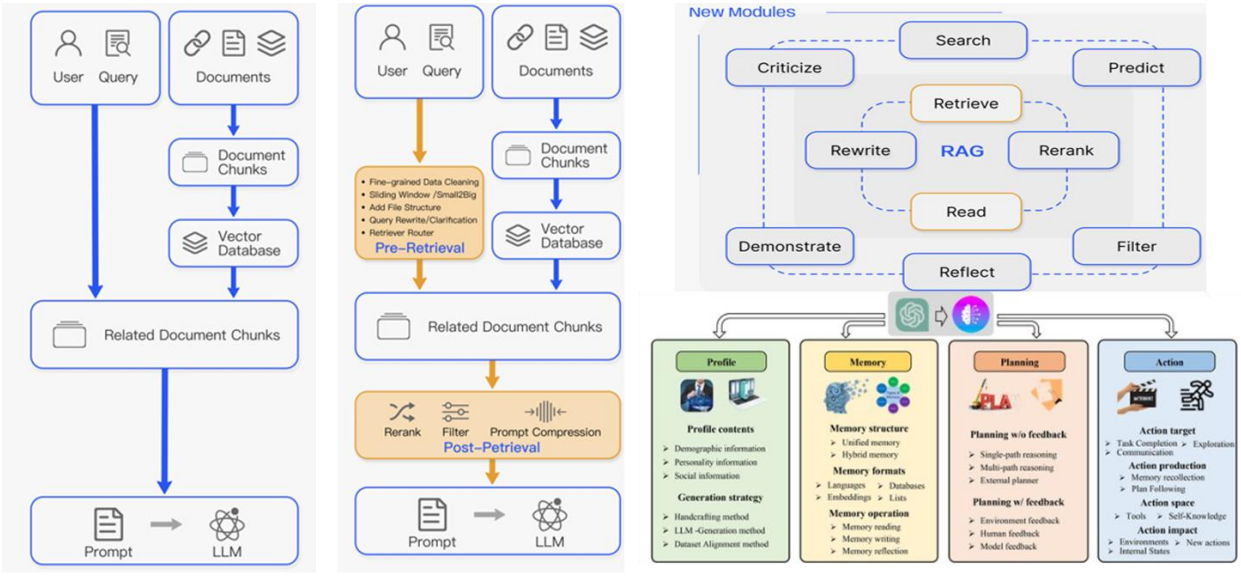
[13] Note that some literatures use prompt tuning as an equivalent term for instruction tuning. Here we have a different meaning.

	and model.	
Design Approach	Divided into manual and automatic design, with automatic design further split into discrete and continuous prompts.	Focuses on adjusting the downstream task to the LM, highlighting an indirect approach to design through task reshaping.
Main Challenges	Over-reliance on human expertise and the possibility of missing optimal prompts even by experienced designers.	The main challenge is to reshape downstream tasks efficiently to align with pre-trained LMs.
Solutions	Automatic template design in discrete (text strings) or continuous (vectors in LM's embedding space) forms to overcome manual design limitations.	Utilizes a minimal number of additional parameters (e.g., 20,480 for a task with "T5 XXL") to adapt the LM to the task, significantly reducing the number of parameters needed compared to fine-tuning.
Advantages	Enables a tailored approach to task-specific LM usage, potentially optimizing interaction without extensive model alterations.	Reduces the cost and complexity associated with fine-tuning large LMs, allows domain transfer, and improves efficiency in small-sample scenarios.

Table 6 Prompt engineering vs prompt tuning

2.1.3 *Retrieval Augmented Generation (RAG) and Knowledge Graph*

The native (or traditional) RAG usually contains three steps, i.e., the content retrieval via embedding models, the content refinement via reranking and augmentation, and the content generation via LLMs. When this technique can reduce hallucination and bring time freshness, it also brings some other side effects. Table 6 lists some of them. For example, in the first step, the embedding models for text retrieval suffer from three limitations: (1) the number and diversity of samples in a batch are too restricted to supervise the modeling of textual nuances at scale; (2) the high proportional noise is detrimental to the semantic correctness and consistency of embeddings; and (3) the equal treatment to easy and difficult samples would cause sub-optimum convergence of embeddings with poorer generalization. To address these limitations, there are many variants and enhancements proposed from native RAG to advanced RAG and modularized agentic RAG [14].



[14] https://github.com/junxu-ai/RAG-Collections

Figure 2 RAG evolution [9]

For example, a hybrid approach integrating vector databases and keyword retrieval is proposed to mitigate inaccuracies in purely vector-based retrieval, incorporating self-attention mechanisms to reduce model hallucinations and enhance problem-solving effectiveness [14] . This method prioritizes documents with key terms and filters irrelevant ones, enriching the retrieval process. In specialized fields like law and finance, leveraging semi-structured domain data enables efficient long-context handling for question-answering systems, showing superior performance in domain-specific tasks [15] . A multi-stage framework is also introduced, improving decision-making transparency and accuracy by generating and refining rationales [16] . Additionally, progressively learned embeddings (PEG) enhance text retrieval across various domains, using dynamic attention modulation and large-scale datasets for tasks like question-answering and similarity matching [17].

Here is the table with an additional column for potential solutions:

Category	Description	Potential Solutions
Conflict between LLM's Internal Knowledge and Retrieved Information	RAG systems may provide incorrect or misleading responses when retrieved information contradicts the LLM's knowledge, especially in critical domains.	Implement consistency-checking mechanisms between retrieved data and LLM's internal knowledge, and prioritize up-to-date, high-confidence sources.
Quality and Accuracy of Retrieved Information	RAG systems' effectiveness relies on accurate, relevant documents. Inaccurate or outdated information can cause errors, misinformation, and erode trust.	Regularly update and vet data sources, and use filtering algorithms to prioritize the most accurate, reliable documents.
Lack of Retrieval Quality Assessment	Many RAG systems incorporate retrieved information without assessing its quality or relevance, potentially spreading errors and misinformation.	Incorporate retrieval quality assessment methods that rank or flag sources based on credibility and relevance before incorporating them into the generation process.
Vulnerability to Biased or Manipulated Information	RAG systems can produce biased outputs due to biases or manipulation in the retrieved documents.	Use bias detection algorithms to identify and filter out biased content, and diversify data sources to minimize the risk of biased retrieval.
Inefficient Utilization of Retrieved Information	RAG systems may use entire documents, including irrelevant parts, which introduces noise, reduces output quality, and increases computational costs.	Implement summarization techniques and document segmentation to extract only the most relevant sections of documents for use in response generation.
Scalability and Efficiency Constraints	RAG systems struggle with scalability and efficiency when handling large datasets, complex queries, or real-time needs.	Optimize query algorithms, use parallel processing, and implement indexing techniques (e.g., vector or graph databases) to improve scalability and processing speed.
Limited Adaptability to Diverse Domains and Tasks	RAG systems may have limited adaptability to complex domains, reducing their applicability in real-world scenarios.	Fine-tune RAG models on domain-specific data and employ modular architectures that allow for easy adaptation to different tasks or industries.
Evaluation and Benchmarking Challenges	Evaluating RAG systems is challenging due to dynamic retrieved content and lack of standardized metrics.	Develop standardized benchmarking metrics and protocols tailored to RAG systems that account for dynamic retrieval and varying content accuracy.

Lack of Transparency and Explainability	The opaque decision-making process of RAG systems can hinder adoption in sectors valuing trust and accountability.	Incorporate explainability frameworks such as model interpretability tools or traceability features that show how retrieved data impacts the final response.
Alignment with User Expectations and Interaction	User expectations may not align with RAG systems' actual capabilities, risking overreliance or misuse, emphasizing the need for clear communication.	Clearly communicate system limitations and capabilities, and set user expectations through detailed user guides or real-time feedback on system performance.

Table 7 Challenges and Limitations of RAG

Using a graph structure in RAG can significantly improve both accuracy and efficiency by enhancing the model's ability to retrieve and reason over complex relationships. Traditional RAG systems rely primarily on unstructured text or vector-based retrieval, which may struggle with understanding intricate relationships between data points. By integrating graph databases, RAG can incorporate structured knowledge that captures relationships between entities, allowing for better multi-hop reasoning, contextual understanding, and reduced irrelevant information retrieval [18] [19]. Graphs enhance RAG models by allowing them to traverse relationships between entities, leading to more accurate results in complex reasoning tasks like finance, ESG analytics, and social network mapping. This approach improves inferential reasoning and reduces irrelevant information by connecting data points through structured relationships. However, the integration of graph databases requires significant computational power for building, updating, and querying large graphs, especially in resource-intensive applications.

Although RAG is effective in data freshness and completion, a general-purpose LLM may misunderstand many financial-specific terms. For example, some translation tasks may provide non-sense outputs. Therefore, we may still require the knowledge embedding via model fine-tuning.

2.1.4 Fine Tuning

Fine-tuning large-scale pretrained models (e.g., embedding and/or generation models) on downstream tasks has become a common training paradigm for a myriad of tasks in NLP and CV. There are many researches show that a fined-tuned model can have significant performance improvements. The study in [20] delves into fine-tuning domain-specific LLMs for the financial sector, detailing dataset preparation, model selection, and adaptation strategies. It emphasizes constructing domain-specific vocabularies and ensuring security and regulatory compliance. Through practical applications like stock prediction and sentiment analysis, the research showcases LLMs with model fine-tuning's potential in finance, identifies challenges, and suggests improvements, offering significant insights for NLP in financial services and encouraging proactive LLM adoption across industries.

The findings in [21] conclude that ChatGPT performs well even without labeled data but fine-tuned models (i.e.., a smaller RoBERTa model) generally outperform it. Similarly, via transforming a small portion of supervised financial sentiment analysis data into instruction data and fine-tuning a general-purpose LLM with simple yet effective instruction tuning approach, [22] outperforms state-of-the-art supervised sentiment analysis models. The work [23] shows that smaller LLMs, when fine-tuned, can perform as well as the most advanced fine-tuned LLMs, despite having fewer parameters and a smaller dataset for training. Also, the performance of LLMs in zero-shot and one-shot scenarios is on par with both smaller, fine-tuned LLMs and the best

available results. Moreover, our study reveals that increasing the number of in-context learning examples does not lead to any improvement in performance for sentiment analysis within the finance domain.

However, as the size of the models and the number of tasks increase, the approach of fine-tuning the entire model entails storing a separate model copy for each fine-tuned task, consuming a significant amount of storage and memory space. This becomes especially critical on edge devices, where storage space and network speeds are limited, making the sharing of parameters particularly important. A straightforward method of parameter sharing involves fine-tuning only a subset of parameters or introducing a small number of additional parameters into the pretrained model. For instance, in classification tasks:

- Linear: Only the classifier (a linear layer) is fine-tuned, while the entire backbone network is frozen.
- Partial-k: Only the last k layers of the backbone network are fine-tuned, with the rest of the layers frozen.
- MLP-k: An MLP with k layers is added as the classifier.
- Side-tuning: A "side" network is trained, and its features are combined with the pretrained features before being fed into the classifier.
- Bias: Only the bias parameters of the pretrained network are fine-tuned.
- Adapter: Extra MLP modules are inserted into the Transformer through a residual structure.

These strategies aim to reduce storage requirements by leveraging shared parameters or adding minimal extra parameters, thereby facilitating more efficient model deployment, particularly on devices with limited resources. There are many works discussing the different strategies. Below are some samples.

LoRA (Low-Rank Adaptation) [24] , which integrates low-rank matrices into the model's layers, significantly reducing the number of parameters that need training. This method allows for effective fine-tuning on downstream tasks while minimizing computational demands . **Representation Fine-Tuning (ReFT)** [25] developed by researchers at Stanford, focuses on modifying a minimal fraction of model representations rather than entire weights, which is effective for most downstream tasks and is noted for outperforming other parameter-efficient methods like LoRA .

Other approaches include **distillation and quantization** to build smaller models from LLMs with similar performance. Distillation refers to transferring knowledge from a large, complex model (teacher) to a smaller, more efficient model (student), allowing the student to mimic the teacher's behavior while maintaining performance with reduced computational resources. Quantization involves reducing the precision of model parameters (e.g., from 32-bit to 8-bit) to make the model more efficient in terms of memory usage and speed, while attempting to preserve accuracy in large language models during fine-tuning. For example, [26] presents a distilling approach of LLMs into efficient, specialized models like UniversalNER for open NER (Name Entity Recognition) tasks, using targeted distillation and mission-focused instruction tuning. It is demonstrated that UniversalNER significantly surpasses both its large model counterparts and state-of-the-art systems in NER accuracy across diverse domains, with a considerable reduction in size, through the largest assembled NER benchmark. Additionally, **QLoRA (Quantized LoRA)** [27] further enhances LoRA by reducing memory usage and facilitating the fine-tuning of even larger models with less GPU memory.

Aspect	RAG	Fine-Tuning

Definition	Enhances LLM output by incorporating external knowledge from a knowledge base.	Customizes an existing LLM by training it on specific data for a particular task or domain.
Integration	Integrates an authoritative knowledge base into the pipeline.	Requires retraining the entire LLM with task-specific data.
Benefits	1. Cost-effective and easy implementation. 2. Access to up-to-date information. 3. Greater control and transparency and security. 4. Reduce hallucnations	1. Fine-tuned model is tailored to the specific task. 2. No reliance on external knowledge. 3. Can capture domain-specific nuances.
Limitations	1. Dependency on external knowledge quality. 2. Complexity and maintenance. 3. Potential over-reliance.	1. Requires labeled data for fine-tuning. 2. May overfit to the training data. 3. Resource-intensive process.

Table 8 RAG vs Fine-Tune

Another relevant concept is prompt tuning. To avoid confusion, Figure 2 compares the model tuning and prompt tuning, which are two distinct approaches for adapting LLMs to specific tasks. Model tuning involves adjusting the underlying weights of the entire model or a substantial part of it, which can lead to highly specialized performance on targeted tasks but may require significant computational resources and risk catastrophic forgetting. In contrast, prompt tuning modifies the input prompt or adds trainable tokens without altering the model's core parameters, making it less resource-intensive and preserving the model's general capabilities. Model tuning is suitable for scenarios where deep customization is critical and resources are abundant, while prompt tuning is ideal for applications needing quick adaptation with minimal computational cost and broad model applicability.

Compared with RAG, the traditional fine-tuning serve different purposes in the utilization of LLMs. RAG integrates real-time information retrieval with generation, making it ideal for applications requiring up-to-date, fact-based content by pulling relevant data or documents during the generation process . On the other hand, fine-tuning involves adjusting a pre-trained model's weights on a specific dataset, which enhances its performance on particular tasks or domains but can lead to catastrophic forgetting where the model loses its ability to perform well on tasks outside of the fine-tuned domain . Hence, RAG is best suited for dynamic scenarios needing accurate, context-rich responses, while fine-tuning is preferable for specialized tasks where deep domain knowledge is crucial. See more comparison in Table 7. However, it doesn't mean that they contradict to each other; instead, they can work complementarily. It's often that FIs need a fine-tuned LLM with their application-specific domain knowledge embedded to understand the contents in vector database and generate reasonable output. With the long-contextual capability of recent LLMs, the dependency on external knowledge might be alleviated.

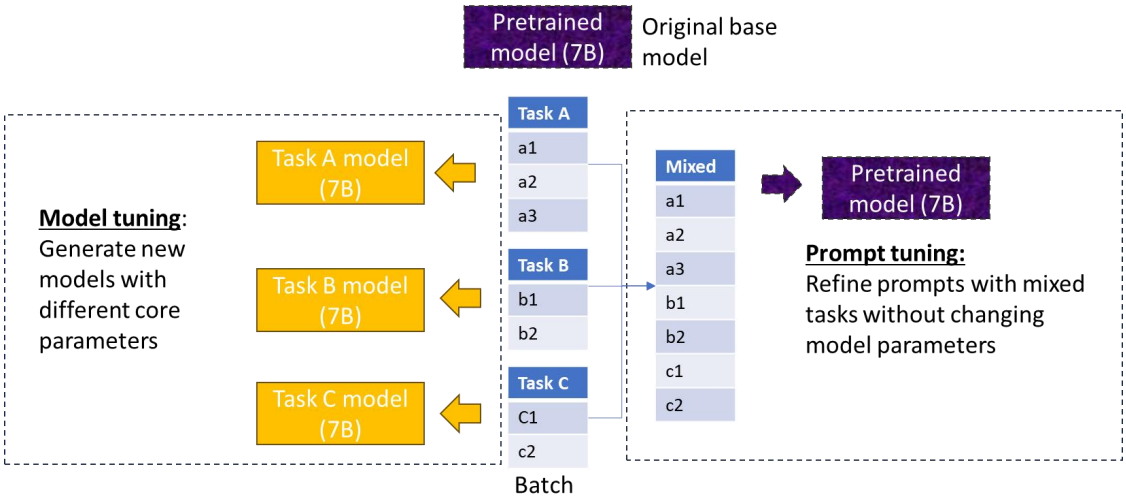


Figure 3 Model tuning vs Prompt tuning

2.1.5 *Agent*

LLM agents are an advanced form of AI that combine the robust capabilities of LLMs with additional functionalities for handling complex tasks and interacting with users in more dynamic ways. Unlike traditional chatbots that mainly respond to queries, LLM agents are equipped with mechanisms to understand, reason, and learn from interactions, which enables them to perform a variety of sophisticated operations.

LLM agents typically consist of several components including memory modules, planning modules, and tools that allow them to execute specific tasks. Memory modules in LLM agents are divided into short-term and long-term types, helping the agent retain and recall information relevant to user interactions and their own internal processing needs. These memory components ensure that agents can maintain context over interactions and refine their responses based on past experiences. The planning module is crucial for decomposing complex queries into manageable tasks, which can involve breaking down a question into several sub-questions that the agent can then address sequentially. This capability is supported by tools—ranging from simple APIs like a weather service to more complex systems like code interpreters or specialized data retrieval systems—which allow the agent to fetch or calculate the necessary information to answer user queries effectively.

Additionally, techniques like ReAct and Self-reflection are used to enhance an LLM agent's reasoning capabilities by iteratively processing thoughts, actions, and observations, which allows the agent to refine its strategies and improve its responses over time. This iterative approach is part of what enables LLM agents to handle complex reasoning and provide more accurate and contextually appropriate responses.

The integration of these components and techniques allows LLM agents to operate across a range of applications from customer service and data analysis to more creative tasks like writing assistance or interactive media. The potential for LLM agents to transform various sectors is vast, given their ability to not only understand and generate human language but also to reason, plan, and learn from their interactions . By combining agent and retrieval augmented generation (RAG) into an agentic RAG framework, the resulting system offers broader applicability, enhanced flexibility, and superior performance compared to using each component individually. Table 8 compares non-agentic RAG and agentic RAG systems in multiple aspects.

Feature/Aspect	Non-agentic RAG	Agentic RAG
Prompt Setup	Dependent on manual crafting of prompts.	Automatically adjusts prompts based on contextual needs.
Contextual Awareness	Static, lacks real-time context integration.	Dynamically integrates ongoing conversation history into decision-making.
Efficiency & Overhead	Prone to inefficiency with potential for high overhead.	Optimizes retrieval to enhance efficiency and reduce unnecessary operations.
Process Complexity	Separate tools and models needed for complex reasoning.	Streamlines complex reasoning and tool integration, removing the need for separate models.
Decision Autonomy	Governed by static rules without real-time adjustments.	High autonomy, making context-aware decisions on information retrieval and processing.
Information Retrieval	Limited to initial queries without follow-up actions.	Proactively seeks additional information as needed during retrieval.
Adaptability & Responsiveness	Restricted to predefined queries and static responses.	Highly adaptable, revising strategies based on immediate feedback and observations.
User Interaction	Frequently requires user input for guidance.	Minimizes user need for interaction by proactively managing tasks.
Error Management	Basic error handling capabilities.	Advanced error correction and robust recovery mechanisms in place.

Table 9 Agentic vs non-agentic RAG

2.2 Decision Workflow

How to design a proper implementable strategy and architecture is a key question, considering the above-mentioned techniques have their advantages and disadvantages: each offer distinct methods to enhance LLMs' capabilities and applicability across various tasks. Table 9 shows a brief comparison.

Figure 3 illustrates the four techniques in x-y axis in terms of knowledge optimization and LLM optimization, which is further explained in Figure 4 with decision paths and the index of a general procedure. Essentially, it outlines a structured decision-making process for applying or augmenting a LLM within various scenarios. Initially, the model is applied with the general prompt engineering (instruct tuning). A key step involves assessing the current performance of the LLM against the desired objectives. If there's a gap in knowledge completeness and freshness, RAG is recommended, which utilizes external data to enhance the model's responses, making it ideal for scenarios requiring up-to-date and context-rich information. If the query is complex and involves multiple steps or pre-defined actions, we may engage the agent-based techniques. If we recognize that existing tools such as search engines, financial database APIs, and financial metric calculators can enhance task performance, it is highly recommended employing the techniques with a focus on utilizing agents with tools through prompt engineering to shape its outputs [15]. Conversely, if the application involves a different domain than the model's initial training, fine-tuning is suggested to adapt the model to the specific characteristics of the new domain, enhancing its accuracy and relevance in specialized contexts.

Note that although these techniques have distinct features and applicable scenarios, it doesn't mean they are exclusive to each other. In fact, they can still merge and

[15] Static workflow with tools might perform better than the LLMs in some cases. For example, to calculate capital asset pricing model (CAPM) or conditional value at risk (cVaR), we can simply use agents to call the corresponding functions, which can provide exact result, instead of using LLM's reasoning capability. Another case might be a SQL to get some commonly used values from a financial database.

integrate to improve the overall performance. For example, modular RAG can use Agents to improve the search and reasoning capabilities. Meanwhile, a fine-tuned smaller LLM might take a specific role for mathematical reasoning inside an agent.

Only if all the four techniques cannot meet the targets and the FIs have sufficient fund and resources, FIs may train their proprietary LLMs. This systematic approach ensures that the LLM can be effectively tailored to meet diverse and specific needs.

Technique	Description	Advantages	Suitable Scenarios
Prompt Engineering	Involves crafting specific input prompts that guide the LLM to generate desired outputs without modifying the underlying model.	- Requires no retraining - Quick to implement - Flexible and creative use of model capabilities	- Rapid prototyping - Tasks requiring quick turnaround and adaptability - Situations where model updates are not feasible
Fine-Tuning	Refers to retraining the LLM on a specific dataset to adapt its responses to particular needs or domains.	- Customizes the model to specific requirements - Improves performance on targeted tasks	- Domain-specific applications - Tasks needing deep customization - When a dedicated dataset for training is available
RAG	Combines the generative capabilities of LLMs with real-time data retrieval to enrich responses with external, factual content.	- Provides up-to-date and factually accurate information - Enhances responses with external data	- Information-intensive tasks requiring accuracy - Dynamic content generation - Fact-checking and knowledge-based applications
Agent Techniques	Employs LLMs within an agent framework to enable complex decision-making, multi-step reasoning, and interaction with other systems or agents.	- Handles complex workflows and tasks - Integrates with multiple systems and APIs - Adaptable to a wide range of environments	- Multi-agent systems - Complex problem-solving environments - Interactive applications requiring ongoing learning and adaptation

Table 10 Comparison of typical LLM extensions

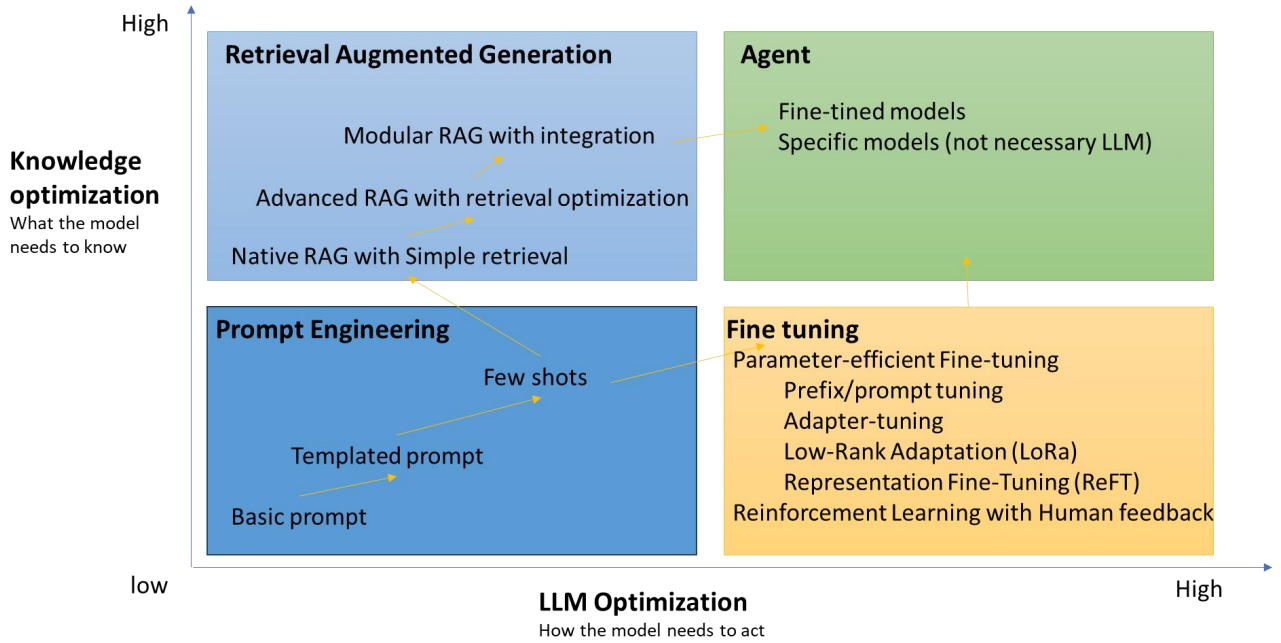


Figure 4 Tech comparison

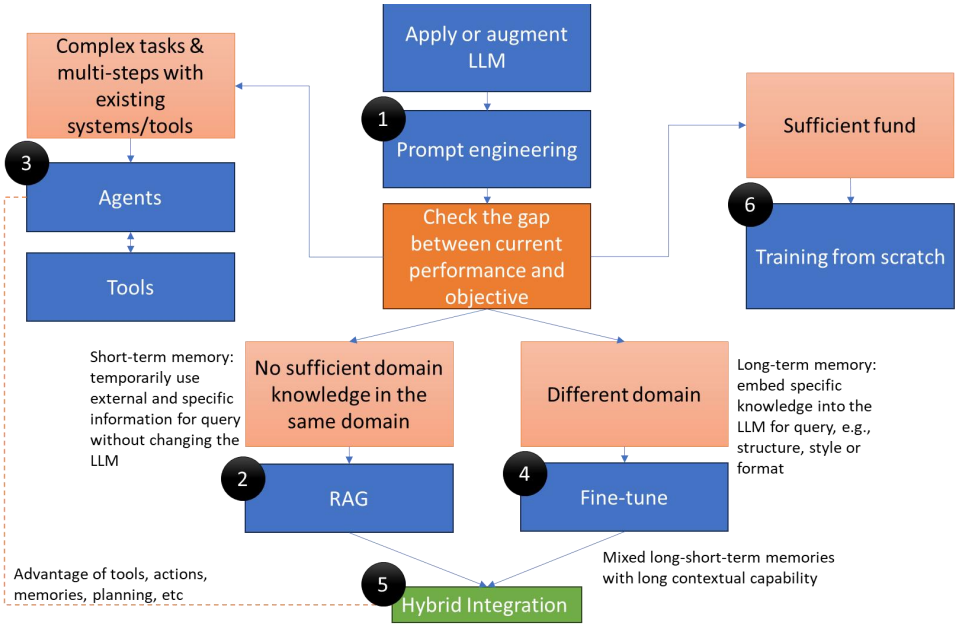


Figure 5 Decision path for suitable techniques

2.3 GenAI lifecycle and LLMOps

The AI and machine learning (ML) lifecycle in FIs is a complex process involving multiple steps, actions and components [1]. Machine Learning Operations (MLOps) is a rapidly growing field that combines the best practices of software development, data science, and DevOps to streamline the lifecycle. MLOps is a set of practices and tools that enable data scientists and machine learning engineers to develop, deploy, and maintain machine learning models in a production-ready environment. It involves the entire ML lifecycle, from data preparation and model development to model deployment and monitoring. MLOps aims to bridge the gap between data science and

software engineering, ensuring that ML models are reliable, reproducible, and scalable. MLOps is crucial in today's data-driven world, where ML models are being used to make critical decisions in various industries.

With the recent emerging LLMs, the operations on LLMs become another interesting topic. This is where Large Language Model Operations (LLMOps) comes into play, a specialized domain focusing on the lifecycle management of applications powered by LLMs. Figure 5 integrates the functionalities and components associated with LLMOps. While not exhaustive, it captures the primary or predominant elements.

Although LLMOps is a subset of MLOPs, LLMOps differs from the general MLOps mainly in handling LLMs, focusing on efficient data usage, specialized experimentation, and distinct evaluation challenges. It incurs higher costs, mainly during inference, and demands robust computational resources for operations like training and fine-tuning. Additionally, LLMOps integrates advanced techniques such as model compression, RLHF, and precise prompt engineering to optimize performance and manage complexities specific to LLMs.

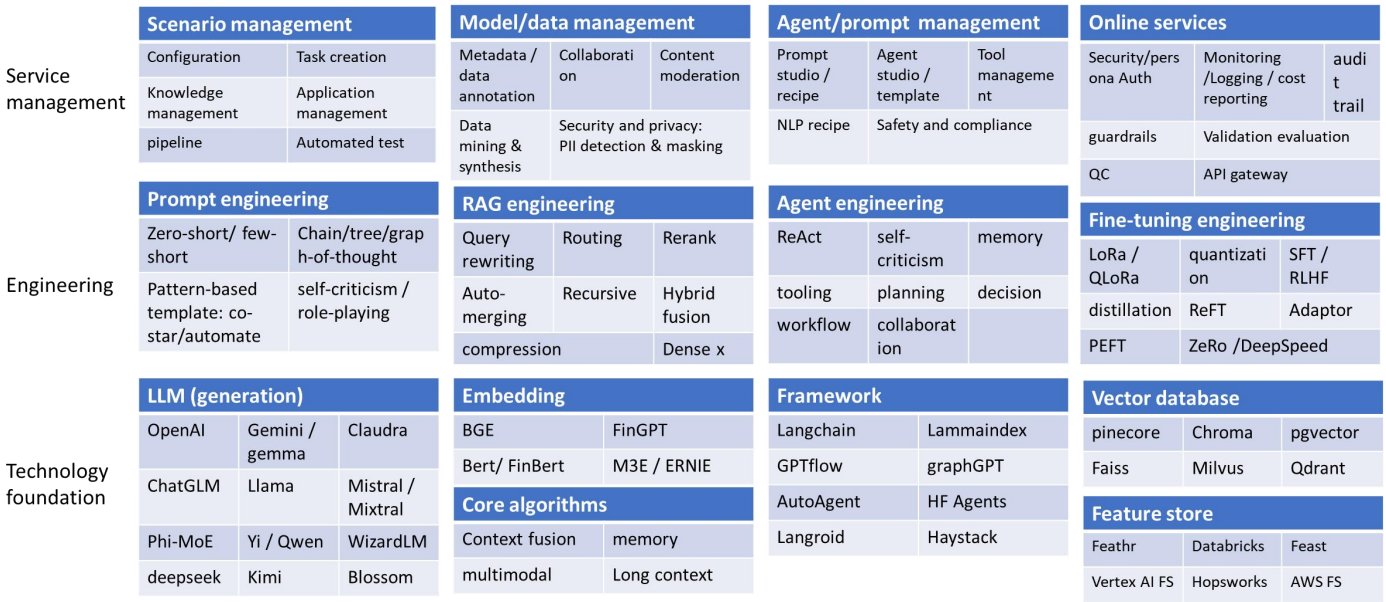


Figure 6 LLM Functionalities

The lifecycle of an LLMOps is comprehensive, starting from problem formulation, data management to model selection, iterative prompt management, testing and evaluation, deployment, and continuous monitoring. Each stage presents its own set of challenges and requires specialized tools and methodologies. Below, the detailed steps and actions of LLMOps are further described.

- Problem Statement and Downstream Definition. We shall have a clear problem description and then properly define the corresponding downstream tasks.
- Data Management: This involves data cleaning, preprocessing, annotation, storage, organization, and version control. Given the large datasets LLMs require, effective data management is crucial for the success of any LLMOps project.

- Model Selection: Choosing the right pre-trained model(s) is a critical step.[16] It involves considering the model's size, performance, and suitability for the task at hand. Once you have chosen your foundation model, you can access the LLM through its API for the adaptation to the downstream tasks. If you are used to working with other APIs, working with LLM APIs will initially feel a little strange because it is not always clear what input will cause what output beforehand. Given any text prompt, the API will return a text completion, attempting to match your pattern. In rare scenarios, we might train the model from scratch.
- (optional) Embedding Storage or Vector Databases: After post-processing, the model may return more than just plain text responses. Advanced applications may require embeddings—high-dimensional vectors that represent semantic content, e.g., RAG, sentimental analysis and recommendation systems. These embeddings can be stored or provided as a service, enabling quick retrieval or comparison of semantic information, thus enriching the ways in which model functionalities are utilized, beyond just text generation.
- Iterative Prompt Management: Crafting the right prompts to guide the LLM's responses is an art. It requires iteration and fine-tuning to ensure the model's outputs are accurate and relevant.
- (Optional) Fine-tuning pre-trained models. This is another way that can help improve your model's performance on your specific task. Although this will increase the training efforts, it can reduce the cost of inference. The cost of LLM APIs is dependent on input and output sequence length. Thus, reducing the number of input-tokens, reduces API costs because you don't have to provide examples in the prompt anymore.
- (Optional) RAG and/or Agent building. The component responsible for the accessing up-to-date data and utilizing known tasks is known as the RAG and Agent-Tool. Here, "Tool" refers to the external connection system, while the "Agent" decides which external tool to use based on the query and activates these external tools accordingly.
- Testing and Evaluation: Assessing the model's performance is complex due to the qualitative nature of language tasks. It involves developing a diverse set of metrics and testing strategies. For Responsible AI (RAI), we shall also satisfy Ethics and Legal Compliance, i.e., ensure that the use of LLMs adheres to ethical standards and complies with legal norms, especially concerning privacy, bias, and fairness. Key measures include selecting cloud storage locations compliant with local data residency laws, securing data transmission through isolated network environments, and employing auditing and monitoring tools like AWS CloudTrail and GCP's Cloud Audit Logs. Additionally, protecting against prompt injection attacks and model inversion attacks is critical, with techniques such as input validation and differential privacy enhancing security. Optionally, the playground component provides an environment where developers can iterate and test AI prompts, ensuring they perform optimally before being embedded into the application. Tools such as OpenAI, nat.dev, and Humanloop offer platforms for fine-tuning and testing LLM prompts.
- Orchestration: This layer coordinates the various components and workflows within the LLM application, abstracting details such as prompt chaining and interfacing with external APIs. These layers are crucial for

[16] Depending on the application and the system architecture, multiple models might be used, e.g., embedding model, generation model, reasoning/schedule/planning model, etc.

maintaining memory across multiple LLM calls and ensuring smooth operation.

- Deployment: Deploying LLMs involves considerations such as API integration, latency, and scalability. Techniques like sampling multiple outputs and ensemble methods can improve the quality of LLM outputs. In this step, we shall also have a proper version control and model management, i.e., managing different versions of language models, including updating models with new data or improvements, caching the model/data and managing the lifecycle of each model version.
- Monitoring: Continuous monitoring is essential to track the model's performance, user satisfaction, and to quickly address any issues that arise. Based on business requirement, we might consider Scalability and Optimization, i.e., expand the infrastructure to support the use of LLMs, and optimize both the model and infrastructure to enhance performance and cost efficiency.

3 LLM Applications in Finance

3.1 Application scope and category

AI, particularly LLMs, is revolutionizing various aspects of the finance domain, driving innovations and enhancing efficiency across multiple areas. This section offers a general overview and classification of LLM applications in finance, illustrating the breadth of their impact and the diversity of their applications. For instance, NLP-related techniques, including sentiment analysis, named entity recognition, summarization, question answering, topic modeling, clustering, classification, information retrieval/extraction, patent and enterprise search, coreference resolution, knowledge graphs with relation extraction, synonym search, and text similarity, are extensively employed [28] [29] [2].

These technologies are pivotal in transforming financial services, from automating customer service inquiries and enhancing the accuracy of investment research to streamlining compliance and regulatory reporting. By leveraging LLMs for these tasks, FIs can not only achieve greater operational efficiencies but also gain deeper insights into market trends, risk factors, and customer needs, thereby delivering more personalized and effective financial products and services. Table 10 highlights some typical applications in different departments of a FI, where the cases in bold font means the higher impact of LLMs.

Customer Service	Financial consulting	Marketing	Risk Control	Operations	Investment Research	Investment Banking	Quantitative Trading	IT
- **Intelligent Outbound Calls (call center)** - Agent Training - **Service Quality Inspection** - **Intelligent RM** - Loyalty program	- **Investment Consulting Assistant** - Investment Consulting Script Recommendations - **Investment Consulting Content Quality Inspection** - **Investment Advice Report Writing**	- **Marketing Content Review** - **Marketing Content Writing** - Market insights / intelligence - SEO optimization - Social media management	- **Public Opinion Analysis** - **Public Opinion Search** - **Event Tagging** - **Event Extraction** - **Regulation Extraction** - **Regulation Search** - KYC/AML	- NL2 Big Screen - NL2 Report - NL2SQL - **Contract Information Extraction** - **Contract Writing** - **Contract Review** - **Form Recognition** - **Comprehensive Search**	- **Research Report Writing** - **Dehydrated Research Report** - **Research Report Retrieval** - **Research Report Tagging**	- **Investment Banking Draft Generation** - Bank Transaction Flow Single Recognition - **Investment management & advisor**	- **Public Opinion Factor** - **Instruction Recognition** - End-to-End Trading - Dialogue (Intent Recognition, Similarity Calculation, NL2SQL)	- **Software development reconstruction** - **Data assets reconstruction** - **Knowledge & doc management**
- Conversional chat (Intelligent Question Answering, intention identification, similarity computing, NL2SQL) - Documentation processing: : information extraction, assistant writing, trend monitoring, index tracking and real-time monitoring for contracts, reports, announcement and regulations - Multimodal processing: multi-format parsing, refined table parsing, conversion between text and image/audio/video								

Table 11 Financial Applications

3.2 Application scenario analysis

3.2.1 Customer Engagement, Service and Support

FIs are using LLMs to transform customer interactions. Chatbots powered by LLMs provide personalized assistance, addressing customer inquiries, providing account information, and even conducting transactions. These models can also analyze customer feedback and sentiment, helping institutions identify areas for improvement in their products and services. AI-powered chatbots are becoming indispensable in the banking sector, providing investment advice and real-time solutions. These models offer multilingual responses and contextual understanding, which are essential for delivering fast and accurate knowledge crucial for decision-making in finance. A type example is the Chatbots for handling customer queries and providing financial advice.

3.2.2 Risk Assessment and Management, Fraud Detection and Prevention

LLMs play a crucial role in risk assessment by analyzing data from various sources such as news articles, social media posts, and financial reports. They can monitor market conditions, regulatory changes, and emerging threats in real-time, providing insights that help financial institutions mitigate risks and detect fraudulent activities. For instance, LLMs can analyze social media sentiment to gauge public perception towards a company or financial instrument, identifying potential risks or opportunities. It can also be used to identify unusual patterns in transactions and spending behaviors to flag potential fraud by analyzing a wide range of complex data. This thorough analysis provides lenders with a more detailed understanding of risk and ensures that borrowers with sparse credit histories receive a fairer evaluation.

3.2.3 Market Analysis and Company Profiling

In the fast-paced world of financial markets, LLMs assist traders and investment professionals by processing and analyzing vast volumes of financial data, including market data, news articles, and social media trends. This information is used to identify market movements, optimize trading strategies, and provide insights into a company's financial health. LLMs can also understand complex documents like quarterly reports and earnings statements, which are invaluable for market analysis. [30] demonstrates how pre-trained and finetuned LLMs can generate company embeddings from SEC filings to identify companies with similar profiles, effectively reproducing GICS classifications and correlating closely with financial performance metrics like return correlation, offering a novel approach for applications in portfolio construction, asset pricing, and risk attribution.

3.2.4 Algorithmic Trading, Investment Opportunities and Portfolio Management

Existing algorithmic trading agents are challenged by several limitations: they often have only short-term memory, which restricts their ability to leverage historical data for long-term decision-making. Their static design fails to adapt to fluctuating market conditions, potentially undermining their effectiveness in complex trading scenarios. Additionally, these agents struggle with processing large and diverse datasets, which can lead to inaccurate or incomplete financial decisions. Furthermore, they lack the intuitive decision-making abilities that human traders apply in unpredictable market environments. These limitations hinder the performance of existing algorithmic trading agents and prevent them from achieving optimal trading outcomes. LLMs support investment services by suggesting portfolio allocations and assessing market trends. They can recommend asset allocation, diversification, and rebalancing strategies, aiding in the optimization of investment portfolios. For example, models like Auto-GPT can optimize portfolios using global equities and bond exchange-traded funds (ETFs), creating comprehensive strategies based on financial information and user-specified

goals. Below are some recent interesting works from simple prompts to complex fine-tuning with model structure changes [5].

Research indicates that while LLMs like ChatGPT can detect profitable trading signals from news sentiment, effectiveness is compromised due to overlap between training and testing periods [31]. Strategies utilizing anonymized headlines have been shown to enhance backtesting and out-of-sample performance, especially for larger companies. Further studies benchmarked various LLMs, including ChatGPT, Erlangshen-RoBERTa, and Chinese FinBERT, for sentiment analysis in Chinese financial texts, establishing a basis for refining LLM-driven trading strategies [32]. Study [33] has shown that GPT-4 outperforms BERT in predicting same-day stock movements for companies like Apple and Tesla, utilizing advanced sentiment analysis from microblogging messages. Further studies [34] using LLMs to create expectation proxies reveal significant deviations similar to traditional surveys, highlighting the need for ongoing research into LLMs' potential in financial expectation formation and challenging existing financial models. Similar conclusions can be found in [35], [36]. The works in [37], [38], [22] show the advantages of model fine-tuning. Additionally, recent innovations in LLMs, such as advanced multi-agent frameworks with hierarchical memory have demonstrated potential to significantly improve automated trading accuracy and financial health assessments, surpassing traditional models in both financial forecasting and sentiment analysis.

Time-series forecasting with LLMs is an emerging field that utilizes the sequential reasoning capabilities of LLMs for temporal data prediction tasks, e.g., converting time-series data into input formats that LLMs can process, such as tokenized embeddings. These adaptations allow LLMs to handle both univariate and multivariate time-series data effectively, capturing subtle trends and seasonal patterns through fine-tuned embeddings and prompt engineering . Furthermore, LLMs' flexibility makes them especially useful in applications where data may be missing or irregular, as their inherent structure allows for interpolation without extensive data preprocessing.

3.2.5 Personalized Financial Planning and Customized Automated Financial Reporting

Personalization is increasingly becoming a defining trend in the financial services sector., e.g.,

- Tailored investment advice or hyper-personalized wealth management based on individual client profiles to drastically reduce the time needed to tailor financial advice to individual client needs
- Personalized customer experiences in banking, which leads to significant increases in customer engagement and sales;
- Integrated personal financial planning platforms help banks analyze customer transactions and behavior to provide real-time financial advice, enhancing the customer's long-term financial health through personalized insights.

The work [39] examines the efficacy of two leading language models, Bard and ChatGPT, in the personal finance sector by evaluating their responses to 13 questions covering a range of banking products (such as bank accounts, credit cards, and certificates of deposit), their interactions, and scenarios involving high-value purchases, payment obligations, and investment advice across various dialects and languages. Despite the models' ability to generate fluent and coherent responses, significant deficiencies were identified in delivering precise and trustworthy financial information through these LLM-based chatbots.

3.2.6 Regulatory Compliance, Audit and Reporting

LLMs assist in ensuring compliance and generating regulatory reports by understanding and interpreting complex regulatory texts. They can automate the extraction of relevant information from legal documents, making it easier for financial institutions to adhere to regulatory requirements and report accurately. For example, it helps to automate the extraction and analysis of regulatory documents. The study [40] explores using LLMs like GPT-4 and Claude for smart contract security audits in Decentralized Finance (DeFi), aiming to streamline the traditionally costly process. By optimizing prompt engineering, the assessment of LLMs' effectiveness on a dataset of 52 compromised DeFi contracts, achieving up to 40% accuracy in vulnerability identification but with a notable false positive rate. Further mutation testing on secure contracts shows a promising 78.7% true positive rate with GPT-4-32k, suggesting LLMs can enhance audit efficiency, though manual oversight remains essential.

3.2.7 Financial Literacy and Education

LLMs are increasingly valuable in the field of financial literacy and education, where they serve a pivotal role in simplifying and democratizing access to financial knowledge. These models are adept at breaking down complex financial concepts into more digestible parts, making it easier for individuals to understand intricate topics such as investments, taxes, retirement planning, and risk management. Furthermore, LLMs can provide interactive tutorials and step-by-step guides that are personalized to the user's financial status and goals, thus enhancing the learning experience.

3.3 Industrial Practices

AI's role in FIs is transformative, offering industries innovative tools to manage and improve their sustainability practices, engage with stakeholders more effectively, and make data-driven decisions that align with their objectives. Many financial institutions have applied AI into the daily works. Table 11 listed some typical industrial examples. For example, Standard Chartered Bank applied the capability of LLM to extract the climate risk information from clients' annual reports and ESG reports and answer the risk questionaries to accelerate the assessment process. Other examples include code assistant in OCBC, BloombergGPT for financial NLP in Bloomberg, etc.

Application Name	Explanation	Financial Institutions	Reference URL
Automated Financial Processes	Uses LLMs to automate tasks like workflow generation, financial document analysis, and report writing	Various	McKinsey & Company
Wealth Management	Leverages LLMs for financial product evaluation, market analysis, investor education, and portfolio management	Morgan Stanley, UBS	Forbes
Insurance Services	Applies LLMs to explain insurance products, create plans, and verify claims	Lemonade, Ping An	Lemonade Blog
Fraud Detection	Analyzes emails and transactions for signs of fraud using LLMs	JPMorgan Chase	Bloomberg
Regulatory Compliance	Uses LLMs to specify information clients must provide to regulators	Wells Fargo	Reuters
Customer Service	Implements LLM-powered chatbots and virtual assistants for improved customer interactions	Bank of America, Capital One	Bank of America Newsroom
Investment	Assists in investment analysis, information	BlackRock, Goldman	BlackRock Blog

Analysis	extraction, and content creation	Sachs	
Risk Management	Enhances risk assessment and management processes	Various	Deloitte Insights
Synthetic Data Creation	Generates synthetic financial data for model training and strategy testing	Various	Synthesis AI
Fundamental Analysis	Refines investment theses and uncovers latent relationships between industries	Point72, Bridgewater Associates	Financial Times
Market Prediction	Explores the use of LLMs for predicting market trends and stock performance	Various hedge funds	MIT Technology Review
Document Processing	Processes and analyzes large volumes of financial documents and reports	KPMG, EY	KPMG Insights

Table 12 Industrial applications

3.4 Case Study: Implementing LLM-based Chatbots for Customer Service

We use a common chatbot to describe the steps to chain your prompts and RAG to build a production-ready financial assistant using LLMs. Table 12 lists the major steps using RAG. Note that for a FAQ (with predefined question and answer pairs) chatbot, the steps can be simplified with prompt and extraction only.

Step	Action	Description
0	Choose Platform and Infrastructure	Select your LLM platform, model, and infrastructure based on application scenarios. Consider data sensitivity, application criticality, and performance-cost tradeoff.
1	Check Query Safety	Use commercial LLM's Moderation API (e.g., OpenAI's) or an internal infrastructure's API to verify if the user's query is safe, e.g., data privacy and content sensitivity. Proceed to Step 2 if safe; otherwise issue warning message to inform users for revision; if the problem persists, go to Step 8.
2	Query Proprietary Data	Embed the user's input/query using an embedding model and query proprietary data stored in a vector DB. Utilize RAG techniques for information retrieval and optionally use text2SQL for database queries.
3	Build the Prompt	Construct the prompt using a predefined template, user's question, extracted financial information, and conversation history or summary.
4	Call the LLM	Use an API to call the LLM with the constructed prompt.
5	Check Answer Safety	Verify the safety of the assistant's answer using OpenAI's Moderation API or internal API. If safe, proceed to Step 6; otherwise, re-generate the answer or issue an alert.
6	Validate Final Answer	Build a validation prompt and use the LLM to confirm the satisfaction of the final answer. If "yes," show the final answer; if "no," indicate insufficient information.
7	Update Conversation History	Add the user's question and the assistant's answer to the history cache to enrich future prompts. Keep only the latest N (question, answer) tuples or a conversation summary to manage context.
8	Human-in-The-Loop Feedback	Review the query manually to find the root cause. If it is human fault, report it as a case to the security or compliance team. If it is a technical issue, refine the model or prompt.

Table 13 Steps to build a chatbot

4 Data Matters

4.1 IT and Data Infrastructure

To meet the increasing demands for LLM adoption and integration, FIs must adapt their information technology (IT) systems and data infrastructure to efficiently collect, aggregate, and distribute a wide range of structured and unstructured financial data. The essential components of modern data architecture include data integration, engineering, quality, observability, and a data catalog for asset management. This also encompasses data governance and privacy, integration of APIs and applications, a data marketplace, various data products, and data mastering. Achieving this integration requires significant changes to IT infrastructure, covering application development, data integration, architecture, and governance. These adaptations are crucial not only for handling specific data types like transactional data, market data, and customer data but also for supporting advanced analytics models that can analyze financial trends, risk assessment, customer behavior analytics, and predictive modeling. To accommodate these changes, FIs need to overhaul their data architecture, devise strategies for data collection, and reform their data governance models to manage and report financial data effectively. By prioritizing strategic investments from the beginning, IT leaders in the financial sector can quickly develop these capabilities into a cohesive financial data platform, thus avoiding the accumulation of technical debt.

Defining solutions for a financial data platform requires a comprehensive approach that emphasizes data integration, accessibility, and modernization. Establishing a centralized data platform that integrates seamlessly with existing finance and risk platforms is vital. This ensures a unified source of truth, improving data accuracy and consistency.

4.2 Data Processing

The confluence of AI and financial markets is potentiated by the vast swathes of unstructured data and the urgency of decision-making. However, the sensitive nature of financial data necessitates robust security measures, stringent compliance with regulatory frameworks, and a commitment to data transparency. The application of NLP and LLMs in this domain underscores the need for real-time or intraday processing capabilities that not only furnish insights but also validate the provenance of data to bolster trust in these insights. Therefore, data progressing and management emerge as Herculean challenges, hindering efficient decision-making. As the demand for real-time financial analytics escalates, so does the imperative for impeccable financial data. Analysts grapple with various hurdles, e.g.,

General issues

- Navigating through unstructured, incomplete, or erroneous data,
- Deciphering the nuances of qualitative and ambiguous financial information,
- Overcoming delays in data transmission,
- Adapting to evolving financial reporting practices,
- Contending with variations in financial reporting structures.

Safety issues

- Some viewpoints of the corpus do not align with the core values of the targeted country
- Some common knowledge do not match the targeted national conditions
- Lack of standardized responses for sensitive issues
- Some crime conditions do not match the actual situation in the targeted country
- Language expression does not conform to industry style

The transition from a historical dearth to a deluge of financial data presents a paradox; this abundance often lacks standardization, compromising data reliability and utility. The quest for high-caliber financial data has spurred advancements that promise to reshape the landscape. To assemble this vast array of data, myriad sources are tapped, ranging from open-source publications to internal and external corporate disclosures. Here, LLMs play a pivotal role, seamlessly integrating diverse datasets to facilitate nuanced financial analysis.

4.2.1 Data Preprocessing Techniques for LLMs

This process is a meticulous orchestration of several steps—tokenization, encoding, quality filtering, and data deduplication. Tokenization meticulously segments text into discrete units known as tokens, enhancing the model's ability to process data efficiently. Positional encoding infuses these tokens with a sense of order, crucial for the model to comprehend the narrative flow within sequences. The sanctity of data is preserved through rigorous quality filtering and deduplication, removing the cacophony of noisy or redundant data. This cleansing ritual ensures that the data is pristine, structured, and optimally primed for training, significantly boosting the model's performance, reliability, and accuracy. Moreover, sophisticated processing techniques tackle the inherent complexities of training, enabling these computational behemoths to thrive even on constrained hardware resources through strategies like optimizer parallelism.

- **Quality Control**: Vital for curating high-quality datasets, techniques such as classifier-based predictions and heuristic rules (involving language metrics, statistics, and keyword analyses) refine data, enhancing the model's efficacy in tasks requiring coding acumen and commonsense reasoning. In addition, we pay an attention to the data diversity (e.g., format, contents) and freshness (e.g., data temporal shift and misalignment). In terms of instruction used for model fine-tuning, the importance of the quality and diversity is even higher than that used for unsupervised phase [41].
- **Data Deduplication**: This technique is crucial for excising redundancy at various data levels—sentences, documents, and datasets—thereby optimizing training efficiency and model accuracy through mechanisms like N-gram similarity assessments using MinHash.
- **Tokenization and Encoding**: Tokenization dissects text into fundamental units, tokens, which could be characters, subwords, or symbols, depending on the tokenization technique employed. Encoding, particularly in the domain of transformers, involves supplementing token embeddings with positional encodings to imbue them with sequential intelligence, crucial for effective model training.
- **Alignment Filtering**: Toxicity control refers to the filtering of the text content which is "rude, disrespectful, or unreasonable language that is likely to make someone leave a discussion". Meanwhile, we must consider the social bias, e.g., the marginalization of minority groups caused by data detoxifying, representational harms as well as excluded voices and identities in large web text corpora.

The landscape of LLM development is a complex tapestry woven from advanced data processing techniques and the relentless pursuit of data purity and structure. As these models continue to evolve, they hold the promise of transforming vast data troves into actionable, reliable insights across various domains, including the rapidly advancing field of financial analytics.

4.3 Data Governance Model

Building a robust data governance model for AI, particularly LLM, applications necessitates a comprehensive and coordinated approach, incorporating several key actions:

- Defining governance goals and principles aligned with responsible AI (e.g., such as fairness, transparency, accountability, privacy, and robustness) and business objectives. An ethical guidance framework shall be established at the beginning with the legal and compliance considerations.
- Developing a comprehensive data taxonomy for the central data catalog to standardize data classification and ensure consistency in data collection, storage, and analysis. In other words, a data catalog is an essential step for data governance. Some organizations also called it as Master Data management (MDM) [42].
- Building a proper data quality control and ethical measure mechanism. It is suggested to regularly audit/monitor data sources to evaluate their origin, licensing terms, and sensitivity levels, to satisfy ethical measures mainly on data security and privacy and fairness. For example, tools and processes would be in place to detect and mitigate bias in training data and model outputs. Meanwhile, robust preprocessing techniques shall be implemented to cleanse the data by removing noise, redundancy, and potential biases. In addition, data validation procedures shall be also implemented to detect and correct errors or inconsistencies. For LLMs, this often includes advanced techniques like tokenization, vectorization, and metadata enrichment.
- Assigning central ownership and responsibility within the organization. This can be achieved by appointing a dedicated data officer, who acts as the central point of contact and ensures cohesive data management across the organization.
- Establishing a cross-functional steering committee for data governance. This committee should include leaders from business, technology, data, risk, compliance and finance departments, fostering joint accountability and streamlined decision-making processes.
- Setting a dynamic framework to adapt changes in market demand and region-specific regulatory requirements. For instance, it should be capable of accommodating investments in emerging sectors like offshore wind turbines and aligning with local regulations, such as the prospective bans on investments in combustion engines in specific countries. LLMs also create additional compliance challenges, requiring specific measures like automated compliance checks and the ability to "forget" data.
- Implementing data controls to maintain compliance with various regulatory frameworks. This involves setting up mechanisms to verify and track compliance markers, such as the assignment of certificates to investments. Role-Based Access Control (RBAC) is one of the techniques. Conducting regular audits and reviews of the data governance framework is required to ensure its effectiveness, relevance, and compliance with evolving LLM governance standards and best practices.
- Integrating continuous monitoring and advanced analytics and reporting tools to enhance the analysis, interpretation, and communication of data, facilitating better decision-making and stakeholder reporting. LLM data governance must prioritize explainability and bias mitigation due to the black-box nature of these models and their potential to amplify biases present in training data. Continuous monitoring is also necessary due to the potential for unintended exposure of sensitive information or harmful contents from both input and output. Incident management and response plan is also required to handle data breaches or other security incidents.

By following these steps, we can build a robust data governance model for LLM applications that ensures compliance, ethical use, and stakeholder trust.

5 Model Matters

Many types of LLM models have been trained, fine-tuned and applied in the industry [3] [43]. There are many considerations, e.g., difference between traditional AI solutions and LLM solution, in-house build vs off-the-shelf solution, open-source vs closed proprietary LLMs, application-specific vs general-purpose LLMs, responsible AI, sustainable AI (see Section 4.2), MLOps/LLMOps platform [44], and etc.

For example, whether choosing an in-house development or an off-the-shelf solution is not just a puzzle for LLMs, even though the LLMs require even much deeper knowledges and more experiences. A completed ROI shall be conducted [17]. However, the key matter is still the qualified resources. Without a long-term sustainable team (e.g., data scientist, data engineer, MLOps engineer) and a scalable infrastructure, it might be a waste of time and money to choose open-source for in-house LLM platform development, although some simple applications are still possible. In terms of application-specific LLMs, many researches show that they can significantly improve the overall performance, and they also require solid skills on prompt engineering and fine-tuning techniques. Table 13 lists some recent LLMs for finance, e.g., [45] introduce BioFinBERT, a finetuned LLM designed for financial sentiment analysis in the biotech sector, leveraging BioBERT's biomedical expertise with financial textual databases. BioFinBERT effectively analyzes press releases impacting biotech stock prices, demonstrating the model's capability in discerning sentiments tied to crucial clinical outcomes or regulatory approvals.

Financial LLM		**Description (e.g., Datasets, Features and Applications)**
BloombergGPT [46]	Closed	a 50-billion parameter large language model trained on a wide range of financial data by Bloomberg using a mix of its proprietary finance data and general-purpose data to address NLP tasks specific to the financial industry. It also released a research paper to detail the development of BloombergGPT. Its popular use cases are financial analysis, research, automated financial reporting, financial data processing, and financial sentiment analysis.
Ant's financial LLM	Closed	Trained based on hundreds of billions of token datasets containing Chinese financial documents and over 1,000 billion tokens from general corpus datasets. It also draws on over 600,000 instructions from more than financial industry cases 300 and was fine-tuned on Ant's self-developed, general-purpose LLM. It has 28 financial-specific task categories; outperforms general-purpose LLMs across five areas: cognition, generation, domain knowledge, professional thinking, and compliance, according to Fin-Eval
FinBERT	Open source	A pre-trained open-source NLP model to analyze the sentiment of the financial text. It is built by further training the BERT language model in the finance domain, using a large financial corpus and thereby fine-tuning it for financial sentiment classification. It is

[17] A typical ROI can be calculated as follows:
ROI = (Total Development Costs + Total Infrastructure Costs + Total Maintenance Costs+Opportunity CostsTotal Revenue Generation+Total Cost Savings+Value of Intangible Benefits−Total Development Costs−Total Infrastructure Costs−Total Maintenance Costs−Opportunity Costs)×100. This detailed breakdown provides a comprehensive view of the financial impact of LLMs, capturing both the tangible and intangible aspects of the investment. Meanwhile, we have to consider the short-term and long-term ROIs. For example, the current commercial LLM API is rather cheaper than self-hosted API of the open source LLMs; however, we cannot simply predict the long-term ones will still maintain the same charge mode.

		based on the FinBert research paper for the development.
Fin-LLaMA	Open source [18]	Utilizes the LLaMA-33B as its base model and undergoes instruction fine-tuning with 16,900 data samples. Known for its high performance, Fin-LLaMA excels in various financial tasks.
FinGPT [47]		Takes a data-centric approach, providing researchers and practitioners with accessible and transparent resources to develop their Financial LLMs (FinLLMs). The team also published a FinGPT paper to be relied on for detailed development. FinGPT could be potentially used for Rob-advisor, financial sentiment analysis, credit scoring, fraud detection, portfolio optimization, risk management, and quantitative trading, to name a few. Provided financial data from 34 diverse sources and a simple yet effective strategy for fine-tuning approach. showcase several FinGPT applications, including robo-advisor, sentiment analysis for algorithmic trading, and low-code development.
FinVis-GP [48]	Open	A novel multimodal LLM specifically designed for financial chart analysis interpreting financial charts and providing valuable analysis
ConFIRM [49]	Open [19]	LLM-based conversational financial information retrieval model tailored for query intent classification and knowledge base labeling. 1) a method to synthesize finance domain-specific question-answer pairs, and 2) evaluation of parameter efficient fine-tuning approaches for the query classification task.
BioFinBERT [45]	Closed	a finetuned LLM to perform financial sentiment analysis of public text associated with stocks of companies in the biotechnology sector.

Table 14 Dedicated financial LLM

4.1 Responsible LLM

Responsible AI forms the bedrock of advancements within LLMs, serving as a crucial element in aligning technological progress with regulatory requirements, ethical standards, and sustainable principles. The integration of responsible AI practices in the development and deployment of LLMs ensures enhanced decision-making capabilities, optimized resource utilization, and innovation that is equitable and environmentally friendly. A central tenet of responsible AI involves ensuring the explainability and transparency of LLM algorithms, which is indispensable for stakeholders to comprehend AI-driven decisions, especially in critical areas such as sustainable investment and risk management.

Moreover, the mitigation of biases is imperative to maintain fairness and prevent discrimination within AI-driven processes, including those related to hiring and lending. Ethical AI development necessitates a steadfast commitment to ethical standards and values throughout all stages, from inception to deployment, emphasizing the importance of user privacy, data security, and the alignment of AI outcomes with broader societal values and Environmental, Social, and Governance (ESG) objectives. Strategies to achieve these goals include actively working to identify and mitigate biases in AI algorithms by diversifying datasets and involving diverse development teams to minimize unintentional discrimination, as well as promoting transparency and

[18] https://github.com/Bavest/fin-llama
[19] https://github.com/WilliamGazeley/ConFIRM

establishing mechanisms for accountability. More considerations and actions can be found in Table 14. Note that due to unique features of LLM different from the traditional ML models, responsible AI criteria might be different too. Table 15 highlights the key differences between LLM-based projects and traditional ML projects.

In the finance sector, the ethical implications of AI extend to the critical areas of data privacy and security, given the vast amounts of sensitive data managed by FIs. Protecting this data from unauthorized access and cyber threats while responsibly managing and utilizing the data in accordance with privacy laws is paramount. Additionally, AI models in finance, including LLMs, face the challenge of inadvertently learning and perpetuating existing biases in training data, potentially leading to unfair practices in loan approvals, risk assessments, or investment advice. Addressing these biases demands a thorough examination and adjustment of training data and algorithms, ensuring diversity and representativeness in training datasets, and implementing measures for algorithmic fairness.

Table 15 Responsible AI considerations and actions

Category	Consideration/Action	Detail/Example
Data/Model Privacy, Reliability and Security	Implement model quality and robustness check	Use advanced cryptographic techniques to protect data at rest and in transit.
	Ensure compliance with data protection regulations	Adhere to GDPR of Europe, CCPA of US, PDPA of Singapore and other privacy laws for data handling and processing for the models.
	Use robust encryption	Use advanced cryptographic techniques to protect data at rest and in transit.
	Conduct regular security audits and monitoring	Perform vulnerability assessments and penetration testing to identify security gaps. Online (continuous) monitoring is an added-value.
	Adopt a privacy-by-design approach	Integrate data privacy into the design of AI systems, ensuring minimal data exposure.
	Enhance authorization and authentication	Utilize access control mechanisms to prevent unauthorized data breaches.
Accountability and Governance	Establish clear AI ethics guidelines	Create a code of conduct for AI development and usage reflecting business values.
	Assign responsibility for AI outcomes	Designate AI ethics officers to oversee responsible implementation and maintenance.
	Document AI decision-making processes	Maintain records of the logic, algorithms, and data used by AI systems for review and accountability.
	Develop transparent AI policies	Publicly share the organization's principles and standards for AI use in business contexts.
	Create a system for AI-related grievances	Set up channels through which stakeholders can report concerns or adverse AI impacts.
Model/Data Fairness and Human-Centerity	Address and mitigate data biases	Use diverse datasets and algorithmic fairness techniques to prevent discriminatory AI outcomes.
	Perform impact assessments	Evaluate the potential consequences of AI deployment on various stakeholders. Determine the accepable thresholds and deviations
	Engage with relevant	Include feedback from those affected by AI systems to guide equitable

	stakeholders and impacted communities	AI development.
	Prioritize accessibility and inclusivity	Ensure AI tools and platforms are accessible to a wide range of users with different abilities.
	Monitor for unintended consequences	Define clear metrics to measure the fairness and biases Continuously review AI systems post-deployment to identify and rectify unforeseen issues.
Transparency and Explainability	Develop explainable/interpretable AI models with clear data lineage and traceability	Use techniques and tools that allow for the interpretation of AI decision-making processes. The data shall be clearly traceable.
	Provide clear and understandable AI documentation and reporting mechanism	Make AI system documentation available and comprehensible for non-technical stakeholders.
	Facilitate third-party AI audits	Allow external evaluations of AI systems to verify their adherence to responsible AI practices.
	Implement a (human) feedback loop	Establish a mechanism for stakeholders to contribute to the continuous improvement of AI systems.
	Offer AI training and educational resources	Educate employees and stakeholders about how AI works and its role in the organization's ESG efforts.

From a regulatory perspective, AI applications in finance are required to adhere to a complex array of financial regulations, including those related to financial transactions, data privacy, and consumer protection, such as the GDPR in Europe and PDPA in Singapore. FIs must remain agile to keep pace with regulatory changes, as the rapid evolution of AI technologies often outstrips existing regulatory frameworks. This necessitates compliance with current regulations and preparedness to adapt to new laws. Furthermore, regulators are increasingly demanding transparency in AI decision-making processes, particularly in decisions that impact customers, underscoring the need for financial institutions to ensure that their AI systems are not only compliant but also capable of providing explainable and accountable decision-making trails.

Table 16 Criteria difference for RAI

Criteria	LLM-Based Projects	Traditional ML Projects
Complexity and Autonomy	More complex and capable of generating autonomous outputs, raising accountability concerns.	Generally, less autonomous with clearer parameters, allowing simpler accountability mechanisms.
Bias and Fairness	Higher risk of amplifying biases from diverse training data; requires rigorous bias detection and mitigation.	Bias is a concern, but typically focused on specific features or outcomes rather than broader implications.
Transparency and Explainability	Opacity in decision-making necessitates strong explainability mechanisms for understanding output generation.	Clearer decision pathways make it easier to explain outputs, focusing on feature importance.
Safety and Security	High potential for misuse (e.g., generating misleading information); requires robust safety protocols.	Safety is important but may not face the same level of risk related to content generation.
User Interaction and Feedback	Interactive nature leads to dynamic learning from user inputs; requires	User interaction may be less dynamic, focusing on performance metrics rather

Loops	monitoring for ethical use.	than ongoing ethical considerations.
Regulatory Compliance	Subject to emerging regulations addressing societal impacts of genAI; requires alignment with these regulations.	Regulatory compliance typically focuses on data handling and model fairness without the same level of adaptation needed for LLMs.

4.1.1 Explainability

Since LLMs are notoriously complex "black-box" systems, their inner working mechanisms are opaque, and the high complexity makes model interpretation much more challenging. This lack of model transparency can lead to the generation of harmful content or hallucinations in some cases. Improving the Explainability of LLMs is crucial for two key reasons:

- End users are able to understand the capabilities, limitations, and potential flaws of LLMs.
- For researchers and developers, explaining model behaviors provides insight to identify unintended biases, risks, and areas for performance improvements.

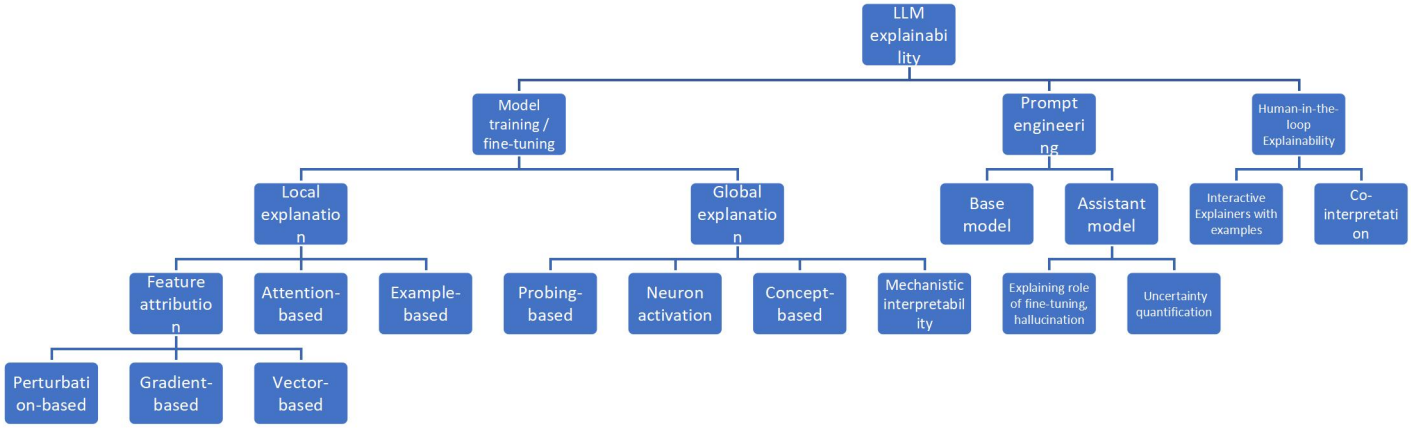


Figure 7 Taxonomy of LLM explainability

The best explainability technique to use will depend on the specific model and the task that it is being used for. The selection of the optimal explainability technique is contingent upon the specific model and the task for which it is deployed. This discussion delineates various paradigms and methodologies for enhancing the explainability of LLMs. We categorize LLM Explainability techniques into three major paradigms [50] [51] [20]:

Paradigm 1: Model Training or Fine-Tuning Paradigm - This paradigm involves the model training from scratch or the refinement of a pre-trained model on specific tasks or domains, thereby enhancing its performance through adaptation to specialized tasks or data sets.

- **Local Explanations**: Techniques under this category aim to elucidate the rationale behind a specific prediction made by an LLM.

Feature Attribution Methods:

- Perturbation-Based: Involves altering input features (e.g., removing words) to see how these changes affect the model's output, helping to identify which parts of the input are most important.

[20] https://github.com/JShollaj/awesome-llm-interpretability

- Gradient-Based: Uses the derivatives of the output with respect to the input features to determine their importance, highlighting which features significantly influence the model's predictions.
- Surrogate Models: Train simpler models to approximate the decision-making of the more complex model, providing insights into the model's behavior by using interpretable models like decision trees.

Attention-Based Explanation:

- Focuses on analyzing the attention weights that the model assigns to different parts of the input, under the assumption that higher weights indicate greater importance for the prediction.

Example-Based Explanation:

- Utilizes specific instances to demonstrate how input changes affect outputs, such as through adversarial examples or counterfactuals, highlighting vulnerabilities or how different inputs could lead to different outcomes.

Natural Language Explanation:

- Involves generating textual explanations for decisions made by the model, often using additional models trained to translate model behaviors into human-understandable text.

- **Global Explanations**: These methods seek to explicate the overarching behavior of an LLM.

- Probing-Based Explanation: Analyzes model representations and parameters to understand the knowledge captured during pre-training.
- Neuron Activation Explanation: Examines individual neurons to understand their role in model performance.
- Concept-Based Explanation: Maps inputs to concepts and measures the importance of these concepts in model predictions.
- Mechanistic Interpretability: Investigates the inner workings of models, especially the connections between neurons.
- Model Distillation: Aims to simplify the LLM to facilitate easier comprehension. For example, utilize a decision tree structure to dissect the decision-making process of the LLM.

Paradigm 2: Prompting Engineering with Model Introspection - As language models scale, prompting-based models display new capabilities that challenge the effectiveness of traditional explainability methods designed for fine-tuning paradigms. The complexity and vast scale of these models make localized explanations inadequate and computationally intensive techniques impractical. This paradigm emphasizes enhancing the interpretability and transparency of LLMs, providing deeper insights into their response and decision-generation processes.

- **Base Model Explanation:** Base models, which grow increasingly large, exhibit abilities such as few-shot learning and complex reasoning through techniques like CoT prompting. These models leverage in-context learning, where the model generates outputs based on examples within the prompt itself, necessitating novel explanation methods to understand how these models adapt and respond to minimal training data.

- Assistant Model Explanation: Assistant models undergo extensive pre-training followed by alignment fine-tuning to tailor them to specific tasks and user preferences. The explainability of these models focuses on the contributions of pre-training versus fine-tuning stages, analyzing how these stages affect the model's ability to follow instructions, avoid hallucinations, and perform complex reasoning, thereby elucidating the inner mechanisms that guide their outputs.

Paradigm 3: Human-in-the-loop (HITL) Explainability - This approach champions the development of mechanisms that enable human interaction with LLMs, facilitating a better understanding of their predictions and behaviors.

- Interactive Explainers with or without Debugging: Tools that allow users to interrogate LLMs, exploring the underpinnings of predictions and explanations. This procedure is usually embedded with iterative refinement process during debug, e.g., overfitting or underperforming.

- Co-interpretation: This entails a collaborative effort between humans and LLMs to unravel predictions, where humans can pose clarifying questions or impart context, aiding the LLM in generating more detailed explanations. It may also contain iterative correction and annotation where necessary.

- User-Centric Evaluation of Explanations: Explanations are evaluated based on how well users can understand and use them to make decisions. This feedback is used to iteratively improve the explanation mechanisms, ensuring that they meet the user's needs in practical scenarios.

Each of these paradigms and associated techniques contributes to the broader endeavor of demystifying the operation of LLMs, rendering them more transparent and accessible to users, thereby fostering trust and enhancing their utility in complex applications.[21]

4.1.2 Fairness and Bias Mitigation

Despite LLM broad capabilities, LLMs can unintentionally learn and perpetuate biases present in their training data, leading to unfair or harmful outputs against certain social groups. This can manifest as stereotyping, derogatory language, or skewed sentiment, reflecting historical and structural inequities. Addressing bias and ensuring fairness in LLMs is crucial for creating equitable AI systems that do not amplify existing social injustices but instead offer balanced and fair perspectives. Typical social biases into two main types: representational harms and allocational harms.

- **Representational Harms**: This type of bias stems from stereotypes and is reflected in negative generalizations involving gender, race, religion, and other social groups. It may manifest as inaccurate or unfair descriptions of specific social groups or as differences in system performance across various social groups. This includes the use of derogatory language and toxicity, the reinforcement of stereotypes, exclusionary norms, and the erasure or underrepresentation of certain groups. These biases can manifest in ways that denigrate or subordinate specific groups through the language produced by models.
- **Allocational Harms**: These biases relate to the distribution of resources or opportunities facilitated by biased model outputs such as in the distribution of credit or job opportunities. Examples include direct discrimination, where decisions made by the model explicitly disadvantage certain groups, and indirect discrimination, where neutral-appearing decisions disproportionately harm specific groups due to underlying biases in data or model processing.

[21] Ala Eddine Ayadi, Advanced Techniques in Explainable AI (XAI) for Responsible Large Language Models (LLM) https://medium.com/@alaeddineayadi/advanced-techniques-in-explainable-ai-xai-for-a-responsible-large-language-models-4c472fde996e, Feb 2024, accessed at 10/08/2024

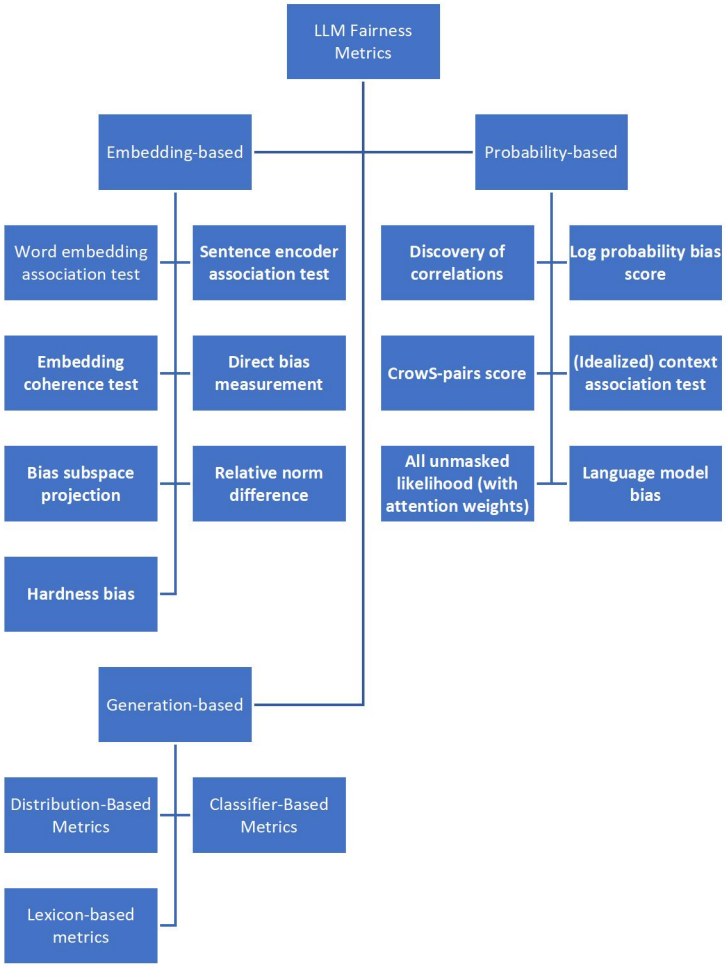


Figure 8 Typical fairness metrics for LLM

Bias evaluation metrics for LLMs are categorized based on the level at which they assess bias: embedding-based, probability-based, and generated text-based [52] . Embedding-based metrics evaluate biases in the vector space representations of words or sentences, revealing how closely certain concepts are associated within the model. Probability-based metrics focus on the likelihoods assigned by the model to different tokens or sequences, often used to measure biases in predictive text completions. Generated text-based metrics examine the actual outputs generated by LLMs under various prompts to assess biases in more dynamic and contextual settings. Each category provides unique insights into the presence and nature of bias, enabling targeted evaluations across different facets of model operation; see Figure 7 for some typical metrics.

Bias mitigation techniques in LLMs are organized based on the stage of the model's lifecycle at which they are applied [1] [52]: pre-processing, in-training, intra-processing, and post-processing. Pre-processing techniques involve modifying the training data to reduce biases before they are learned by the model, such as by augmenting data or adjusting data sampling methods. In-training techniques modify the learning algorithm itself, including adjustments to the loss function or model architecture to counteract biased learning patterns. Intra-processing methods focus on altering the model's behavior during inference, such as by adjusting decision thresholds or re-ranking outputs. Finally, post-processing techniques involve modifying the outputs after generation, typically through rules or additional models that adjust or filter the text to remove or reduce biased content. Each category of techniques offers different strategies to reduce bias, reflecting varying degrees of integration with the model architecture and training process.

4.1.3 Generative Content Detection

The field of generative content detection, particularly in distinguishing AI-generated text from human-generated text, has become increasingly pertinent as language models like ChatGPT evolve and integrate more deeply into various digital communications. Current research, as reviewed in [53] , categorizes the primary detection methods into three types: simple classifiers, zero-shot detection techniques, and fine-tuning based detection. Each method aims to tackle the challenge of identifying text generated by AI, leveraging different approaches based on the complexity of the task and the specific requirements of the detection system. These methods reflect a growing need to understand and mitigate potential issues arising from the misuse of generative models in creating misleading or deceptive content. Watermark, as one popular technique, involve embedding subtle, unique patterns or signals into the content that are imperceptible to users but can be detected algorithmically to verify the origin and authenticity of the material.

The effectiveness of various detection tools highlights the ongoing challenges in the field. Tools such as ZeroGPT, GPTZero, the OpenAI Text Classifier, and Hugging Face's detection tool have been developed to specifically identify content generated by models like ChatGPT. However, as noted, none of these tools have achieved a high success rate in distinguishing between AI-generated and human-generated responses consistently. This indicates a significant gap in the current capabilities of detection technologies, necessitating further research and development to enhance the accuracy and reliability of these tools in various real-world applications.

Improvements in detection technology are crucial not only for maintaining the integrity of information online but also for supporting the responsible use of AI in content creation. The ongoing development and refinement of detection methods will likely focus on enhancing machine learning models to better understand the nuances of human vs. AI-generated text. This might include more advanced training datasets, innovative algorithmic approaches, or hybrid models that combine several detection methods to improve accuracy. As AI technologies continue to evolve, so too will the strategies to detect and manage their outputs, ensuring they are used ethically and effectively in all spheres of digital communication.

4.2 Sustainable LLM

Sustainable AI can be viewed as an encompassing extension of the responsible AI paradigm, which traditionally focuses on the social implications of AI. This broader concept of Sustainable AI takes into account not only the societal impacts but also the environmental footprints of AI technologies with governance requirements. When FIs devise their Sustainable AI strategies, they must navigate a dual-focused pathway: one that leverages AI to advance sustainability goals, and another that scrutinizes the sustainability of AI systems themselves [1].

The facet of "AI for sustainability" involves deploying AI technologies as tools to further the ambitions of global sustainability efforts, such as the United Nations SDGs. Examples include using AI to optimize energy distribution networks, thereby reducing waste and promoting the use of renewable resources, or employing ML algorithms to analyze vast datasets for trends that can inform climate action strategies: initiatives often encapsulated under banners like AI4Good or AI4climate. These applications of AI have the potential to drive significant progress in addressing some of the world's most pressing environmental challenges.

On the flip side, "the sustainability of AI" calls for a critical examination of the energy consumption and resource use throughout the lifecycle of AI technologies—from the data centers that power computation-intensive tasks to the end-of-life disposal of hardware. FIs must adopt practices that increase the energy efficiency of AI operations, such as

- utilizing more energy-efficient algorithms, e.g., use LLM only when necessary if the traditional ML algorithms cannot work well. Some new architectures other than transformers have been proposed, e.g., KAN, TTS.
- leveraging and tuning existing models instead of training new models from scratch
- optimizing resource usage through DevOps automation (MLOps), system auto scale, scheduling algorithm, and selection of cloud providers and data centers.
- investing in green data centers powered by renewable energy sources.

Moreover, the governance of AI applications demands stringent oversight to protect various facets of ethical AI usage as discussed in the previous sectors, including but not limited to ensuring privacy through secure data practices, maintaining transparency in AI decision-making, enforcing accountability for AI-driven outcomes, upholding fairness to prevent algorithmic bias, and safeguarding justice by allowing recourse in the event of AI-induced harm. In essence, sustainable AI strategy within FIs is a comprehensive framework that aims not only to exploit AI's capabilities for promoting ecological and societal wellbeing but also to do so in a way that is itself sustainable and ethically sound [1] [54] [55] [56] [57].

4.3 LLM Evaluation

Evaluating LLMs is a multifaceted endeavor that encompasses a spectrum of metrics to ensure these models are aligned with human values and exhibit safe, reliable, and beneficial behaviors. The assessment of LLMs involves a rigorous examination of their efficiency and performance capabilities, which are critical for their application across various domains.

The efficiency metrics heavily rely on the model architecture and parameter size. Models with more parameters generally exhibit greater linguistic capabilities and contextual understanding, allowing them to generate more accurate and coherent responses across diverse applications. However, larger models also demand significantly more computational resources, which can impede scalability and increase operational costs [22]. The architecture of an LLM plays a crucial role in its efficiency as well; more sophisticated architectures can improve processing speed and reduce latency, even with a large number of parameters. Thus, optimizing an LLM involves a delicate balance between enhancing its capabilities and managing the computational burden, making the choice of model architecture and parameter size critical for maximizing efficiency without compromising performance.

The performance evaluation involves multiple considerations. Truthfulness is a pivotal aspect, with metrics assessing the factual consistency and reliability of LLM-generated text. Relevance (e.g., similarity, BLEU score) evaluates how well the response aligns with the context and the user's needs, providing useful and pertinent information. Adherence (e.g., constraint violation rate or policy adherence scores) checks whether the model follows specific guidelines or constraints, such as staying within a given topic or format. Correctness (e.g., accuracy, F1 score) refers to the factual accuracy and precision of the output, ensuring that the information provided is reliable and error-free.

[22] A simplified formulation to calculate the requirement of VRAM size is as follows: parameter size * value bit size /8 * 1.2. For example, a Llama 7 billion parameter model with float (F16) values may require 7*16/8*1.2=16.8 GB memory.

Moreover, the evaluation extends to alignment with ethical standards, bias detection, and toxicity analysis to guarantee that LLMs produce outputs that are fair, unbiased, and free from harmful content [23]. Furthermore, robustness and risk evaluations (e.g., adversarial robustness scores, out-of-distribution detection, and stress testing under diverse inputs) are crucial for understanding how LLMs perform under adversarial conditions or when faced with complex, real-world tasks. These evaluations are essential for identifying potential weaknesses and guiding the development of LLMs that are not only high-performing but also trustworthy and responsible AI agents. Some examples are shown in Table 16.

Evaluation Task	Evaluation Metric	Description
Classification	Precision	The proportion of model predictions that are positive and actually positive.
	Recall	The proportion of actual positive samples that are correctly predicted by the model.
	F1 Score	A comprehensive measure of the model's precision and recall output.
Language Modeling	Perplexity	Measures the model's probability of modeling the reference text.
Text Generation / QA Task/ Summarization / Machine Translation	BLEU	Measures the overlap between machine translation and reference translation.
	ROUGE	Measures the coverage of machine summaries against reference summaries. variants: ROUGE-N, ROUGE-L, ROUGE-W, and ROUGE-S
	METEOR	Evaluates translation quality by considering word-to-word matches and semantic equivalence, which provides a balance of precision and recall
	NIST	a variant of BLEU includes weighting of n-grams based on their information content, giving importance to rarer n-grams
	SPICE	Focuses on semantic accuracy rather than grammatical correctness, using scene-graphs to compare propositions in the hypothesis and reference texts
Execution Tasks	Success Rate	Measures the proportion of tasks successfully completed by the model.
	Pass @ k	Estimates the probability that at least one of the k solutions generated by the model will pass.
Preference Ranking	Elo Rating	Measures the relative level of the model among candidates.

Table 17 General evaluation metrics

When no sufficient internal data is available for your application evaluation, you may choose some public Evaluation Datasets. For example, diverse datasets like LRA, SCROLLS, and InfiniteBench serve as benchmarks to gauge the long-contextual capabilities of LLMs, providing a comprehensive overview of language types, task modalities, and data quality.

- **Long-Range Arena (LRA)** [24]: a dataset designed for evaluating efficient transformer models, focusing on long-context scenarios. It includes a suite of

[23] https://github.com/junxu-ai/LLM_fairness

[24] https://paperswithcode.com/dataset/lra

tasks with sequences ranging from 1K to 16K tokens, covering various data types and modalities.

- **SCROLLS** [25]: a dataset focused on evaluating the performance of language models on long-context tasks. It emphasizes the importance of understanding how models handle longer sequences. More information can be found in the associated research papers or repositories, but specific links were not provided in the search results.
- **InfiniteBench** [26]: a dataset aimed at benchmarking the capabilities of language models over extended contexts. Similar to SCROLLS, it focuses on long-context evaluation. Specific details and access links were not found in the search results.

4.4 The Human Workforce in the Age of GenAI

In the age of GenAI, the human workforce is encountering transformative shifts, leading to many hot topics, such as productivity efficiency, resource optimization, job fairness, and job rebuilding.

Efficiency and resource optimization are critical aspects where GenAI is making a significant impact. GenAI tools can handle tasks that typically consume substantial human hours with greater speed and accuracy, such as data analysis, content generation, and routine administrative tasks. This shift allows businesses to optimize their resource allocation, focusing human labor on more complex, creative, and strategic roles where emotional intelligence and human judgment are irreplaceable. Moreover, this transition supports a more efficient workflow, reducing burnout and enabling employees to engage in more meaningful and satisfying work.

Fairness in the workforce has become a pivotal concern as GenAI integrates into various industries. The fear that AI might replace human jobs is tempered by initiatives that focus on fair AI integration, ensuring that technology complements human skills rather than replacing them entirely. This involves creating guidelines and policies that prevent bias in AI operations and promote equitable opportunities for workers to transition into new roles that AI technology may introduce. Lastly, job rebuilding in the context of GenAI involves reshaping existing roles and creating new ones that can coexist with advanced technologies. In the current stage, we may expect more HITL tasks with the active learning [27] techniques are assigned due to the regulation constraints. The emergence of roles like AI supervisors and content reviewers or auditors, who ensure the outputs of AI systems are accurate and appropriate, and AI integration specialists, who help merge AI technologies with existing business processes, are becoming increasingly common. Other roles include AI personality designers, custom AI solution developers, human-AI interaction designer and AI ethicist. This evolution requires a rethinking of educational and training programs to equip the workforce with the necessary skills to thrive in a technology-driven environment. It also involves a strategic overhaul of job descriptions and company structures to support a hybrid human-AI workforce effectively.

---

[25] https://www.scrolls-benchmark.com/

[26] https://github.com/OpenBMB/InfiniteBench

[27] Active learning is a semi-supervised machine learning approach where the model selectively queries a human to label the most informative data points, improving learning efficiency and model performance with fewer training examples [1]. In the context of LLMs with a human-in-the-loop, this interaction allows the model to refine its responses based on direct human feedback, enhancing its accuracy and relevance in real-world financial applications.

5 Concluding Remarks and A Strategic Approach for LLM Integration

The adoption of GenAI and LLMs offers significant opportunities as well as challenges. LLMs can substantially enhance financial practices by optimizing resources, automating operations, and improving trading processes. However, their integration into a FI is a comprehensive and demanding process, requiring substantial investment akin to other top strategic priorities. This transformation necessitates incorporating LLM considerations into all facets of corporate functions, including operations, financial planning, business strategy, product development, pricing, credit assessment, facilities management, corporate venture investing, and treasury operations. FIs must assess their organizational readiness, such as current infrastructure capabilities and talent resources, before the strategical planning and implementation. The implementation plan must be managed carefully to address critical issues such as data privacy, potential biases, ethical concerns, and overcoming accessibility barriers. This balanced approach ensures that while leveraging their benefits, the risks associated with LLM technologies are effectively mitigated. The following outlines the steps and strategies necessary for successfully integrating GenAI and LLM into a FI's processes.

1. Consolidate compliance and risk management frameworks

To effectively LLMs into banking operations, existing corporate governance structures and compliance frameworks must be thoroughly reevaluated for privacy, fairness, explainability, accountability/transparency and integrity. It is imperative to update governance frameworks to adequately cover ethical issues, data privacy, and regulatory compliance, including the establishment of oversight committees to monitor LLM usage. These committees should ensure adherence to stringent data privacy regulations like the GDPR and CCPA, particularly as they pertain to the financial sector. Additionally, banks need to reassess their risk management practices to identify and address potential risks related to the development and deployment of LLMs. This involves defining new controls in line with overarching AI principles and implementing mitigation strategies such as robust data encryption and stringent access controls to protect sensitive information and uphold regulatory standards [28]. Banks should also establish regular review mechanisms and reporting protocols to ensure ongoing compliance with evolving standards. Ideally, a special task force with decision makers from various departments shall be organized.

Moreover, enhancing data and model governance is critical to the success of LLM applications in banking. Strong data governance policies must be implemented to preserve data integrity and security, such as enforcing rigorous data handling protocols specifically tailored for AI applications to safeguard customer information. The effectiveness of LLMs hinges significantly on the quality of the training data utilized. There is a notable deficiency in high-quality, diverse, and representative financial datasets, which are essential for training robust and reliable models. Addressing this gap is crucial for ensuring the effectiveness and reliability of LLMs in the financial services sector.

[28] It is also not necessary to overestimate the risks of LLMs, considering the fundamentals of deep neural networks are not changed. Few additional controls might be the checks of prompt input and generated contents.

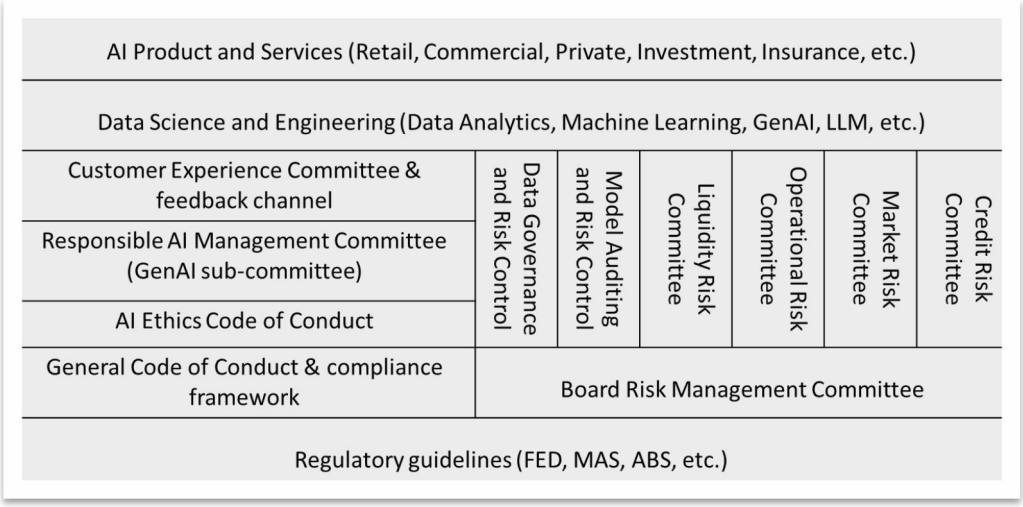


Figure 9 General Risk Governance Framework

2. Upgrade IT infrastructure

IT infrastructure and application architecture need enhancement. For example, the computational power required to train LLMs, especially considering their size and complexity, is substantial. There is a general inadequacy in the current computational support available for training these models. Meanwhile, the microservice (API)-based architecture is critical for LLM gateways. Figure 9 shows a reference logic architecture with the major functional layers [29].

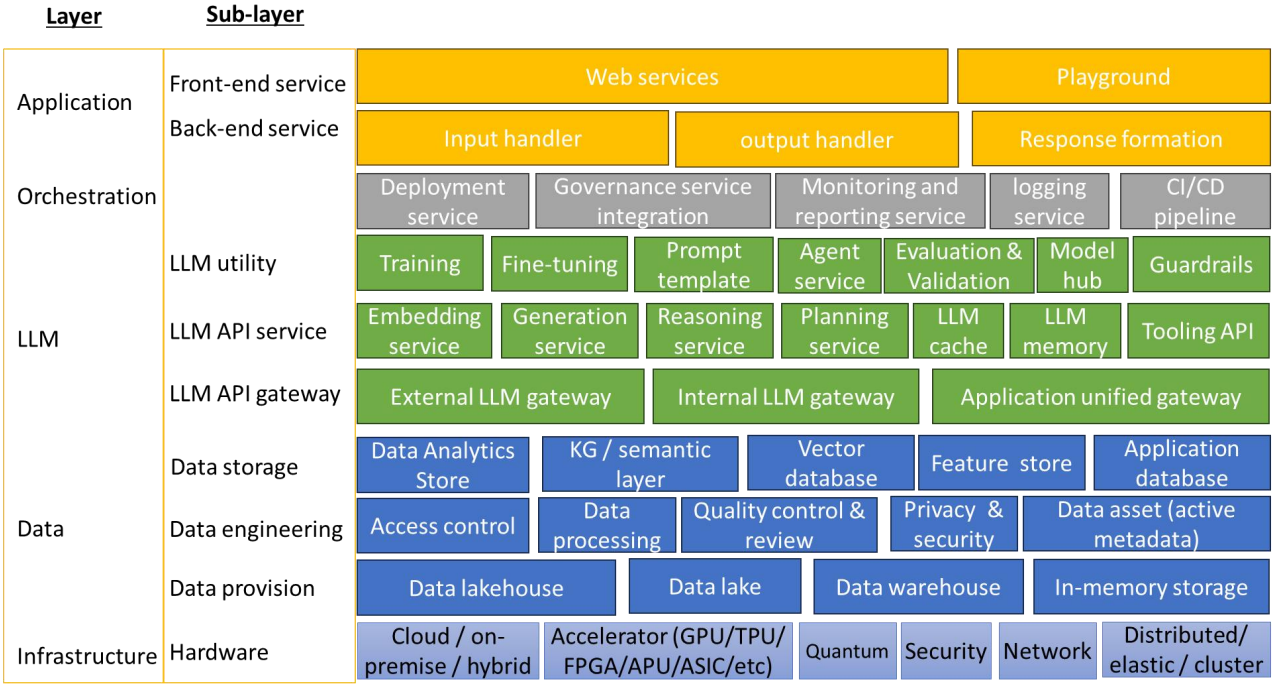


Figure 10 The LLM Stack

[29] I referred the concepts from Andreessen Horwitz, https://a16z.com/emerging-architectures-for-llm-applications/, https://github.com/a16z-infra/llm-app-stack, Jun 2023.

- The Application Layer in LLMOps architectures can be effectively segmented into two main sub-layers: the frontend serves as the primary user interface where users submit their queries and interact with the application; The backend is where the core processing happens.
- The Orchestration Layer involves managing complex workflows that interact with multiple LLMs to automate and optimize various tasks. Unlike traditional operations that might call an LLM once for a single output, orchestration in LLMOps is about creating dynamic and intelligent systems that can loop inputs and outputs through different models or the same model multiple times. Another important component is the governance service integration, which interacts with the existing or new governance frameworks for risk and compliance to accelerate the adoption speed. For example, the model artifacts can be automatically extracted and evaluated by the RAI tool.
- The LLM Layer is a crucial component of the LLMOps architecture, functioning as the central hub for processing and managing language model requests. This layer is specifically engineered to address the intricacies and demands associated with handling large language models, thereby underpinning the core functionalities of the LLMOps.
- The Data Layer is a fundamental component designed to bolster "enterprise-specific" intelligence through advanced data management and processing techniques. This layer utilizes a vector store to house documents broken down into manageable chunks along with their embeddings and relevant metadata.
- The Infrastructure Layer serves as the foundational backbone that supports the deployment, operation, and scaling of LLMs. This layer is typically engineered to handle the high computational demands of LLMs, facilitating efficient data processing, storage, and network management [30]. Meanwhile, the sophistication of LLM applications relies on their harmonious integration with other financial technologies (FinTech) like blockchain, cloud computing, big data, and the Internet of Things (IoT). Standardization protocols across data, modeling, monitoring, governance, and application programming interfaces (APIs) ensure consistency and reliability, with effective governance workflows regulating access control and documentation. This integration bridges theoretical frameworks with practical financial scenarios, advancing financial services while aligning with broader financial objectives.

3. Adjust project management for production

Given the challenges associated with deploying LLM applications in production, it is advisable for FIs to begin with internal, non-critical applications rather than immediately launching external, client-facing ones. Starting with internal projects, such as a coding chatbot or climate risk assessment tools using public data, allows FIs to

[30] When deciding between on-premises vs. cloud solutions (or local LLMs vs. cloud APIs), several key factors must be considered, including organizational needs (e.g., data sensitivity, regulatory compliance, budget constraints, and technical expertise), cost-benefit analysis, and future growth plans. For example, while cloud solutions offer flexibility and ease of use, on-premises deployments provide greater control. However, the long-term costs of on-premises solutions may be difficult to estimate, as LLMs might require specialized hardware for acceleration beyond general-purpose GPUs, which could be challenging for FIs without a dedicated support team. Additionally, organizations with rapid growth expectations may prefer cloud solutions for scalability, whereas those with stable workloads might find on-premises setups more cost-effective. Another interesting trend is the specific "small" LLMs (typically less than 10B), which are dedicated for some purposes and use much less resources. A phased approach is recommended, with an initial reliance on cloud APIs (e.g., 95% cloud API vs. 5% local LLMs) that gradually transitions to a more balanced split (e.g., 70% cloud API vs. 30% local LLMs) over a five-year period, following a gradual adaptation curve.

develop and refine their capabilities in a controlled environment. These internal applications have a limited impact on daily operations, and any failures will not disrupt business-as-usual (BAU) activities or incur significant reputational or financial costs. Meanwhile, commercial secured APIs are usually sufficient with non-sensitive data, considering their price, general availability and rich functionalities (e.g., fine-tune, prompt template and guardrails). A simple logging with basic auditing and monitoring functionalities shall be applied to all API calls. FIs may start to build a research and development team for open-source implementation for the sensitive information processing inside the LLM Layer.

During this initial phase without a centralized platform, FIs are encouraged to run pilot projects for more valuable or external client-facing applications in a Proof of Concept (PoC) or internal Minimum Viable Product (MVP) in a controlled sandbox environment. This approach provides a safe space to experiment, learn, and improve without exposing the organization to undue risk. It is suggested to applied existing centralized authorization and authentication mechanisms for MVPs. Once these internal applications have been successfully executed for a period of time, the centralized platform might be ready to use. FIs can then consider deploying LLMs in more critical scenarios, having gained the necessary experience and confidence in the technology's reliability and effectiveness.

Inside the (LLMOps) platform, it shall have a centralized model (API) management for enhancing cost efficiency and simplifying administration. By pooling computational resources and guardrail control, organizations can reduce redundant usage across departments and optimize the significant processing power required by LLMs. This approach lowers operational costs and streamlines updates and maintenance, ensuring all users access the most current and secure version of the model. Maintaining consistency across the organization minimizes discrepancies that could arise from utilizing multiple model versions.
The development style in LLM projects is evolving to empower business personas directly in the prompt design process. Unlike traditional development methods that require technical expertise for changes, LLMs enable non-technical users to craft and modify prompts themselves. This democratization accelerates the development cycle and aligns model outputs more closely with business objectives. Business users, with their deep understanding of domain-specific requirements and customer needs, can ensure that the LLM's outputs meet client expectations. This shift reduces dependency on developers for minor adjustments, alleviating bottlenecks, and allowing technical teams to focus on more complex tasks. The more advanced playgrounds or sandboxes shall still have their roles to facilitate effective collaboration between technical and business teams. These shared testing platforms provide an environment where users can experiment with the LLM, creating rapid PoCs or MPV to assess potential performance for specific tasks.

Controlled access ensures that only authorized personnel can utilize or modify the model, preventing unauthorized use and potential security breaches. Centralized systems facilitate compliance and auditability by making it easier to track who accessed the model and when, which is vital for adhering to data protection regulations. Efficient user management is also achieved, allowing for swift onboarding, offboarding, and permission updates as organizational roles evolve.
While LLM deployment and monitoring are centralized to ensure optimal performance and security, web application development via API connections remains flexible and unrestricted. Developers have the freedom to build and deploy applications that interact with the LLM, fostering innovation and expanding the range of services available to users.

Close monitoring and the implementation of guardrails are imperative to ensure the LLM is used appropriately. Active monitoring of API calls helps detect and prevent misuse, such as overuse, security breaches, or attempts to generate prohibited content. Establishing guardrails with alarm systems enables immediate notification to the development team if predefined boundaries are approached or breached. Regular audits and reviews of API usage data not only optimize performance but also ensure compliance with organizational policies and regulatory requirements. By adopting these measures, organizations can mitigate risks, uphold ethical standards, and maintain the integrity of the LLM deployment.

When there are more and more projects will be handled, we have to define a proper project priority. Table 17 defines such a decision matrix. Essentially, this matrix provides a structured way to prioritize GenAI/LLM projects based on a deep understanding of technology, product, and business impact. By applying weighted scores, it helps balance immediate value with long-term strategy, technical challenges, and implementation feasibility. The weights assigned reflect a balanced approach, emphasizing both immediate and strategic benefits while considering feasibility and risks. Organizations can adjust these weights based on their specific priorities and strategic focus.

Attribute	Explanation	Weight
Direct Value	Measures the immediate benefits such as profit increase, cost reduction, and efficiency improvements directly attributable to the project. Usually, we shall define an reasonable objective.	**20**
Strategic Value	Assesses how well the project aligns with the organization's long-term strategy, goals, and competitive positioning in the market. Usually, we shall evaluate based on different scenarios.	**20**
Technical Feasibility	Evaluates the likelihood of successful technical implementation based on available technology, data quality, and in-house expertise.	**15**
Implementation Feasibility	Considers the ease of implementing the project given the existing infrastructure and skill sets, including whether to build in-house or purchase solutions.	**10**
ROI (Long-term vs. Short-term)	Estimates the potential return on investment and balances projects that offer immediate benefits against those that provide long-term value.	**15**
Risk tolerance and mangement	Identifies the potential risks involved, including technical challenges, regulatory compliance issues, and operational obstacles.	**10**
Market Adoption	Gauges the potential for market acceptance or internal adoption, including user readiness and competitive dynamics.	**10**
Total		**100**

Table 18 Priority decision matrix

4. Conduct organizational communication, training programs and change management

An organization-wide communication of LLM mandates is essential, achieved through a strategic change management approach that brings all employees on board. This involves not just disseminating information but also engaging all employees through AI-driven communication tools. These tools can tailor and personalize messages for different employee groups, ensuring a clearer understanding and alignment with LLM directives. Such an approach ensures that every member of the organization understands and aligns with the new LLM directives. Specialized LLM roles or teams may be created to monitor and guide LLM integration within investment processes. Developing a clear plan for integrating new LLM policies is crucial for seamless adoption. AI-powered project management and workflow automation tools can facilitate

this integration, providing practical guidelines and tracking the implementation of new measures. This includes adding new certifications to investments, ensuring that these integrations are seamless and effective. LLM training programs, augmented by LLM-driven learning platforms, should be established for employees, particularly those involved in investment decisions. These platforms can offer personalized learning experiences and adaptive content to build a strong foundation in LLM principles and practices while using AI to track learning progress and adapt training materials to individual learning styles.

In addition, to effectively harness LLMs' transformative power in financial industry while minimizing potential drawbacks, a collaborative approach using the Shared Value Model is required [58] . This model recognizes companies as key players in AI development, emphasizing that sustainable practices can enhance profitability over time. This approach should encompass some additional strategic elements besides the abovementioned, each accompanied by specific actions and examples:

- Building Global Cooperation and Industrial Standardization: Establish global cooperation to develop international GenAI standards and best practices under regulatory requirements if any. This approach addresses cross-border AI challenges and ensures uniform governance and ethical considerations, akin to the international collaboration seen in climate change agreements. Some resilient and adaptive regulatory frameworks shall be developed by regulators and industrial associations that can keep pace with AI advancements, ensuring responsible and ethical usage. For instance, creating policies that adapt to new AI technologies in financial markets, ensuring they remain fair and transparent. Ethical guidelines and standards shall be established for GenAI, incorporating transparency, fairness, privacy, and accountability. These guidelines would ensure AI in healthcare respects patient confidentiality and delivers unbiased treatment recommendations.

- Fostering Public-Private Partnerships: Encourage collaboration between governments, private sectors, and academia for a balanced AI development approach. Such partnerships might result in joint ventures for developing AI in sustainable energy, combining academic research with industrial application.

- Encouraging Stakeholder Engagement: Involve a broad spectrum of stakeholders, including underrepresented communities, in GenAI development and deployment. This could include community-led GenAI projects in urban development to ensure technologies meet the real needs of residents.

- Conducting Deep Research and Development: Continuously invest in research to understand AI's societal, environmental, and industrial impacts. This involves initiatives like funding long-term studies on AI’s effects on job markets and developing AI solutions for environmental conservation. There is also a need for enhanced trustworthiness and security of algorithms. The trustworthiness and security of the algorithms underpinning LLMs in finance need to be improved. This includes ensuring the reliability and ethical integrity of the models, as well as safeguarding against potential security vulnerabilities . As we contemplate the future trajectory of LLMs and GenAI in finance, their roles will expand significantly, driven by ongoing research and development aimed at enhancing their capabilities and applications. Integration with other emerging technologies will amplify their impact, creating more efficient and secure financial systems. Future scenarios envision AI-driven monetary systems that are adaptive and responsive to global economic shifts, potentially transforming traditional financial landscapes. Education and training for financial professionals will be crucial to leverage these advanced tools effectively. The long-term implications for financial markets and economies are profound, with AI poised to drive unprecedented levels of innovation, efficiency, and sustainability in the financial sector.

In general, integrating LLM into bank processes is a complex but necessary transformation that aligns FIs with sustainability goals and modern technological advancements. By collaborating across departments and institutions, identifying IT infrastructure gap, and leveraging AI-driven decision-making and communication tools, banks can effectively embed LLM considerations into their operations. This strategic approach ensures that banks not only meet global financial targets but also drive growth and innovation in FinTech domain, reinforcing their commitment to a sustainable future.

Acknowledgments: The author gratefully acknowledges the valuable discussions with colleagues and friends from various financial institutions and tech companies.

Conflicts of Interest: This paper only represents the options from the author and doesn't represent any other organizations.

6 References

[1] J. Xu and et al, The Future and FinTech: ABCDI and Beyond, J. Xu, Ed., World Scientific Press, 2022.

[2] L. Cao, "AI in Finance: Challenges, Techniques and Opportunities," *ACM Computing Surveys,* vol. 12, no. 2, pp. 1-38, 2022.

[3] Y. Li, S. Wang, H. Ding and H. Chen, "Large Language Models in Finance: A Survey," in *4th ACM International Conference on AI in Finance (ICAIF-23)* , 2023.

[4] Y. Nie and et al., "A Survey of Large Language Models for Financial Applications: Progress, Prospects and Challenges," arXiv:2406.11903, 2024.

[5] h. Ding, Y. Li , H. Wang and h. Chen, "Large Language Model Agent in Financial Trading: A Survey," arXiv:2408.06361, 2024.

[6] H. Zhao and et al., "Revolutionizing Finance with LLMs: An Overview of Applications and Insights," arXiv:2401.11641, 2024.

[7] A. W. Lo and J. Ross, "Generative AI from Theory to Practice: A Case Study of Financial Advice," An MIT Exploration of Generative AI, 2024.

[8] Y. Ding and et al., "A Survey on RAG Meets LLMs: Towards Retrieval-Augmented Large Language Models,," arxiv:2405.06211v1, 2024.

[9] Y. Gao and et al., "Retrieval-Augmented Generation for Large Language Models: A Survey," Arxiv:2312.10997, 2023.

[10] C. Maple and A. Sabuncuoglu, "The Impact of Large Language Models in Finance: Towards Trustworthy Adoption," The Alan Turing Institute, 2024.

[11] G. Shabsigh and E. B. Boukherouaa, "Generative Artificial Intelligence in Finance: Risk Considerations," in *Fintech Notes*, vol. 2023, International Monetary Fund, 2023.

[12] E. Callanan, A. Mbakwe, A. Papadimitriou, Y. Pei, M. Sibue, X. Zhu, Z. Ma, X. Liu and S. Shah, *Can GPT models be Financial Analysts? An Evaluation of ChatGPT and GPT-4 on mock CFA Exams,* vol. arXiv:2310.08678, 2023.

[13] F. Xing, *Designing heterogeneous LLM agents for financial sentiment analysis,* vol. arXiv:2401.05799, ArXiv:2401.05799, 2024.

[14] J. Cui, Z. Li, Y. Yan, B. Chen and L. Yuan, *ChatLaw: Open-source legal large language model with integrated external knowledge bases,* vol. arXiv:2306.16092, 2023.

[15] V. Mavi, A. Saparov and C. Zhao, *Retrieval-augmented chain-of-thought in semi-structured domains,* arXiv:2310.14435, 2023.

[16] M. I, S. Saxena, S. Prasad, M. V. S. Prakash, A. Shankar, V. V, V. Vaddina and S. Gopalakrishnan, *Minimizing factual inconsistency and hallucination in large Language Models,* vol. arXiv:2311.13878, 2023.

[17] T. Wu, Y. Qin, E. Zhang, Z. Xu, Y. Gao, K. Li and X. Sun, *Towards robust text retrieval with progressive learning,*

arXiv:2311.11691, 2023.

[18] B. Fatemi, J. Halcrow and B. Perozzi, *Talk like a Graph: Encoding Graphs for Large Language Models,* arXiv:2310.04560, 2023.

[19] P. Sen and S. Sen, *Graph database while computationally efficient filters out quickly the ESG integrated equities in investment management,* arXiv:2401.07483, 2024.

[20] C. Jeong, *Fine-tuning and utilization methods of domain-specific LLMs,* arXiv:2401.02981, 2024.

[21] A. Shah and S. Chava, *Zero is not hero yet: Benchmarking zero-shot performance of LLMs for financial tasks,* arXiv:2305.16633, 2023.

[22] B. Zhang, H. Yang and X.-Y. Liu, "Instruct-FinGPT: Financial sentiment analysis by instruction tuning of general-purpose large language models," in *FinLLM Symposium at IJCAI 2023*, 2023.

[23] S. Fatemi and Y. Hu, *A comparative analysis of fine-tuned LLMs and few-shot learning of LLMs for financial sentiment analysis,* vol. arXiv:2312.08725, 2023.

[24] Y. Hu and et al., LoRA: Low-Rank Adaptation of Large Language Models, arXiv:2106.09685v2, 2021.

[25] Z. Wu and et al., ReFT: Representation Finetuning for Language Models, arXiv:2404.03592v3, 2024.

[26] W. Zhou, S. Zhang, Y. Gu, M. Chen and H. Poon, *UniversalNER: Targeted distillation from large language models for open named entity recognition,* vol. arXiv:2308.03279, 2023.

[27] T. Dettmers, A. Pagnoni, A. Holtzman and L. Zettlemoyer, QLoRA: Efficient Finetuning of Quantized LLMs, arXiv:2305.14314v1 , 2023.

[28] G. Pisoni, B. Molnár and Á. Tarcsi, "Data Science for Finance: Best-Suited Methods and Enterprise Architectures," *Applied System Innovation,* vol. 4, no. 69, p. 20, 2021.

[29] L. Cao, "AI in Finance: A Review," *SSRN Electronic Journal,* p. 36, 2020.

[30] D. Vamvourellis, M. Toth, S. Bhagat, D. Desai, D. Mehta and S. Pasquali, *Company Similarity using Large Language Models,* vol. 2308.08031, 2023.

[31] P. Glasserman and C. Lin, *Assessing look-ahead bias in stock return predictions generated by GPT sentiment analysis,* vol. arXiv:2309.17322, 2023.

[32] H. Zhang, F. Hua, C. Xu, J. Guo, H. Kong and R. Zuo, *Unveiling the potential of sentiment: Can Large Language Models predict Chinese stock price movements?,* vol. arXiv:2306.14222, 2023.

[33] R. Steinert and S. Altmann, *Linking microblogging sentiments to stock price movement: An application of GPT-4,* vol. 2308.16771, 2023.

[34] L. Bybee, *Surveying generative AI's economic expectations,* vol. arXiv:2305.02823, 2023.

[35] X. Yu, Z. Chen, Y. Ling, S. Dong, Z. Liu and Y. Lu, *Temporal data meets LLM - explainable financial time series forecasting,* vol. 2306.11025, 2023.

[36] U. Gupta, *GPT-InvestAR: Enhancing stock investment strategies through annual report analysis with Large Language Models,* vol. 2309.03079, 2023.

[37] Y. Yang, Y. Tang and K. Y. Tam, *InvestLM: A large language model for investment using financial domain instruction tuning,* vol. arXiv:2309.13064, 2023.

[38] Y. Yu, H. Li, Z. Chen, Y. Jiang, Y. Li, D. Zhang, R. Liu, J. W. Suchow and K. Khashanah, *FinMe: A performance-enhanced large language model trading agent with layered memory and character design,* vol. 2311.13743, 2023.

[39] K. Lakkaraju, S. K. R. Vuruma, V. Pallagani, B. Muppasani and B. Srivastava, *Can LLMs be good financial advisors?: An initial study in personal decision making for optimized outcomes,* arXiv:2307.07422, 2023.

[40] I. David, L. Zhou, K. Qin, D. Song, L. Cavallaro and A. Gervais, *Do you still need a manual smart contract audit?,* vol. 2306.12338, 2023.

[41] Z. Wang and et al. , "Data Management For Large Language Models: A Survey," arXiv:2312.01700v2, 2023.

[42] R. Ghous, "How a Modern Data Architecture Brings AI to Life: Data Mastering for AI," Informatica, 19 Octorber 2023. [Online]. Available: https://www.informatica.com/blogs/how-a-modern-data-architecture-brings-ai-to-life-

data-mastering-for-ai.html. [Accessed 19 December 2023].

[43] H. Zhao and et al., "Revolutionizing Finance with LLMs: An Overview of Applications and Insights," arxiv, Janurary 2024. [Online]. Available: https://arxiv.org/pdf/2401.11641.pdf. [Accessed 28 Janurary 2024].

[44] J. Papenbrock, A. John and S. Peter, "Accelerated Data Science, AI and GeoAI for Sustainable Finance in Central Banking and Supervision," in *International Conference on "Statistics for Sustainable Finance"*, Paris, France, 2021.

[45] V. Aparicio, D. Gordon, S. G. Huayamares and Y. Luo, *BioFinBERT: Finetuning large language models (LLMs) to analyze sentiment of press releases and financial text around inflection points of biotech stocks,* arXiv:2401.11011, 2024.

[46] S. Wu, O. Irsoy, S. Lu, V. Dabravolski, M. Dredze, S. Gehrmann, P. Kambadur, D. Rosenberg and G. Mann, *BloombergGPT: A Large Language Model for Finance,* Arxiv:2303.17564, 2023.

[47] X.-Y. Liu, G. Wang, H. Yang and D. Zha, *FinGPT: Democratizing Internet-scale data for financial large language models,* arXiv:2307.10485, 2023.

[48] Z. Wang, Y. Li, J. Wu, J. Soon and X. Zhang, "FinVis-GPT: A Multimodal Large Language Model for Financial Chart Analysis," 2023. [Online]. Available: https://arxiv.org/abs/2308.01430.

[49] S. Choi, W. Gazeley, S. H. Wong and T. Li, *Conversational financial information retrieval model (ConFIRM),* arXiv:2310.13001, 2023.

[50] H. ZHAO and et al., "Explainability for Large Language Models: A Survey," *ACM Trans. Intell. Syst. Technol,* vol. 15, no. 2, p. 20, 2024.

[51] H. Luo and L. Specia, "From Understanding to Utilization: A Survey on Explainability for Large Language Models," arXiv:2401.12874v1, 2024.

[52] I. O. Gallegos and et al., "Bias and Fairness in Large Language Models: A Survey," arXiv:2309.00770v3, 2024.

[53] A. Pegoraro, K. Kumari, H. Fereidooni and A.-R. Sadeghi, *To ChatGPT, or not to ChatGPT: That is the question!,* arXiv:2304.01487, 2023.

[54] A. van Wynsberghe, "Sustainable AI: AI for sustainability and the sustainability of AI," *AI and Ethics,* vol. 1, p. 213–218, 2021.

[55] V. Galaz and et al, "Artificial intelligence, systemic risks, and sustainability," *Technology in Society,* vol. 67, no. 101741, 2021.

[56] C. Isensee, K.-M. Griese and F. Teuteberg, "Sustainable artificial intelligence: A corporate culture perspective," *NachhaltigkeitsManagementForum,* vol. 29, p. 217–230, 2021.

[57] I. S. Banipal, S. Asthana and S. Mazumder , "Sustainable AI - Standards, Current Practices and Recommendations," in *Proceedings of the Future Technologies Conference*, 2023.

[58] M. E. Porter and M. R. Kramer, "Creating shared value: How to reinvent capitalism—and unleash a wave of innovation and growth," *Harvard Business Review,* vol. 89, no. 1-2, p. 62, 2011.

[59] "ML system design: 300 case studies to learn from," Evidently AI, 2 December 2023. [Online]. Available: https://www.evidentlyai.com/ml-system-design?utm_source=talkingdev.uwl.me. [Accessed 19 December 2023].

[60] L. Kelly, "What is ESG Data? Uses, Types & Dataset Examples," Datarade, 13 December 2023. [Online]. Available: https://datarade.ai/data-categories/esg-data. [Accessed 19 December 2023].

[61] "How AI Can Bolster Sustainable Investing," morgan stanley, 31 July 2023. [Online]. Available: https://www.morganstanley.com/ideas/ai-sustainable-investing-use-potential. [Accessed 19 December 2023].

[62] E. Burnaev, E. Mironov, A. Shpilman, M. Mironenko and D. Katalevsky, "Practical AI Cases for Solving ESG Challenges," *sustainability,* vol. 15, no. 12731, p. 15, 2023.

[63] S. Mazumder, S. Dhar and A. Asthana, "A Framework for Trustworthy AI in Credit Risk Management: Perspectives and Practices," *Computer ,* vol. 56, no. 5, pp. 28 - 40, 2023.

[64] "Facilitated Emissions: Global GHG Accounting and Reporting Standard (Part B)," PCAF, 2023.

[65] C. Gerling and S. Lessmann, *Multimodal Document Analytics for Banking Process Automation,* vol. arXiv:2307.11845, 2023.

[66] M. T. Hicks, J. Humphries and J. Slater , "ChatGPT is bullshit," *Ethics and Information Technology* , vol. 26, no. 38,

2024.

[67] Z. Liu and et al., "KAN: Kolmogorov-Arnold Networks," arXiv:2404.19756 , 2024.

[68] S. Minaee and et al., Large Language Models: A Survey, arxiv:2402.06196v2, 2024.

[69] A. Gu and T. Dao, "Mamba: Linear-Time Sequence Modeling with Selective State Spaces," Arxiv:2312.00752, 2024.

[70] O. Lieber and et al. , "Jamba: A Hybrid Transformer-Mamba Language Model," Arxiv:2403.19887, 2024.

[71] Y. Sun and et al., "Learning to (Learn at Test Time): RNNs with Expressive Hidden States," Arxiv:2407.04620v1, 2024.

[72] Y. Li, Y. Yu, H. Li, Z. Chen and K. Khashanah, *TradingGPT: Multi-agent system with layered memory and distinct characters for enhanced financial trading performance,* arXiv:2309.03736, 2023.